Probabilistic Machine Learning

Sayan Mukherjee

Probabilistic Machine Learning

Sayan Mukherjee

[1] Departments of Statistical Science, Computer Science, and Mathematics, Duke University, Durham, 27708.
E-mail address: sayan@stat.duke.edu.

November 19, 2015

LECTURE 1
Course preliminaries

The term machine learning goes back to Arthur Samuels and his computer checker playing algoriths. In 1959 Samuels described machine learning as: ”Field of study that gives computers the ability to learn without being explicitly programmed.”

Machine learning is considered a subfield of artificial intelligence and the idea of a learning machine is given in ”Computing Machinery and Intelligence,” by Alan Turing in 1950 in Mind: A Quarterly Review of Psychology and Philosophy. The question posed in the fist sentence of this paper was “Can machines think ?”.

For this class by ML we are going to consider algorithms and probabilistic methods to ”learn from data.” The material is at the interface of statistics and computer science and one caricature of ML is computer scientists doing statistics. ML is often also associated with the term ”big data” which is often meant to be statistical analysis with very large data sets, here the computational challenge is as serious as the inference problem.

Broadly speaking the methods we will discuss can be placed into two categories:
proceduralists: This will cover both frequentist statistics, as well as algorithmic approaches to ML. This approach is based upon coming up with good procedures to apply to data. What is meant by good is some long run probability of the procedure, for example the long run probability of errors made in classification is small.
Bayesian: A coherent axiomatic approach to inference based on inference of the posterior probability of parameters or models given data. Bayesian inference may not be feasible or practical in certain situations.

1.1. Review

We'll start with a basic review of statistics. We will examine a statistical question using both Bayesian and frequentist analysis. The following formalism will be quantified in both models

$$\begin{aligned} \mathrm{P}(M \mid D) &= \frac{\mathrm{P}(D \mid M)\mathrm{P}(M)}{\mathrm{P}(D)} \\ &\propto \mathrm{P}(D \mid M)\mathrm{P}(M), \end{aligned}$$

where $\mathrm{P}(M \mid D)$ is evidence for model M given data D, $\mathrm{P}(D \mid M)$ is evidence for D given model M, $\mathrm{P}(M)$ is the probability of model M, and $\mathrm{P}(D)$ the probability of data, The standard statistical terms for these objects are

$$\begin{aligned} \mathrm{P}(D \mid M) &\equiv \mathrm{Lik}(D; M), \quad \text{Likelihood of data given model } M \\ \mathrm{P}(M \mid D) &\equiv \mathrm{Post}(D; M), \quad \text{Posterior evidence of model } M \text{ given data} \\ \mathrm{P}(M) &\equiv \pi(M), \quad \text{prior probability (before seeing data) for model } M. \end{aligned}$$

Example 1: Motif estimation

We consider a random variable X that is drawn from a alphabet of $k = 4$ letters $\{A, C, T, G\}$ where we represent $A \equiv 1$, $C \equiv 2$, $T \equiv 3$, and $G \equiv 4$. We set the probability distribution on X as the following multinomial distribution, note we are modeling a draw of four letters

$$\begin{aligned}
\mathrm{P}(n_1, n_2, n_3, n_4 \mid p_1, p_2, p_3, p_4) &\equiv \mathrm{Multi}(p_1, p_2, p_3, p_4) \\
&\propto \prod_{j=1}^{4} p_j^{n_j}, \quad \sum_{j=1}^{4} p_j = 1, p_j \geq 0 \ \forall j = 1, ..., 4,
\end{aligned}$$

where p_i is the probability of observing the i-th letter ($\{A, C, T, G\}$ in the alphabet and n_i states how many times the i-th letter is observed (either 1 or 0). The above is an example of the multinomial distribution.

The random variable X is a string in a sequence and we can think of the random string $Z = (X_1, ..., X_m)$ as a string of length m with each X_i drawn iid from a distribution. This is an example of a string, let us call these strings motifs. The data consists of a series of n strings, $D = \{Z_1, ..., Z_n\}$ with each string Z_i drawn iid (independently and identically distributed).

We first state the likelihood of observing the data D

$$\begin{aligned}
\mathrm{P}(D \mid M) &= \mathrm{Lik}(D \mid p_1, ..., p_4) \\
\mathrm{Lik}(D \mid p_1, ..., p_4) &\propto \prod_{i=1}^{n} \left[\prod_{\ell=1}^{m} \left(\prod_{j=1}^{k} p_j^{n_{i\ell j}} \right) \right] \\
&\propto \prod_{\ell=1}^{m} \left[\prod_{i=1}^{n} \left(\prod_{j=1}^{k} p_j^{n_{i\ell j}} \right) \right] \\
&\propto \prod_{\ell=1}^{m} \left[\prod_{j=1}^{k} p_j^{\tilde{n}_{\ell j}} \right],
\end{aligned}$$

where $n_{i\ell j}$ is the number of observations of letter j at position ℓ in observation i (again this is 0 or 1) and $\tilde{n}_{\ell j} = \sum_i n_{i\ell j}$ is the number of times in the n sequences that letter j is observed at position ℓ.

A classical method for estimating $p_1, .., p_k$ is the maximum likelihood formulation

$$\begin{aligned}
\{\hat{p}_1, ..., \hat{p}_k\} &= \arg \max_{p_1, \cdots, p_k} \left[\mathrm{Lik}(D \mid p_1, ..., p_k) \right], \\
&\quad \text{subject to} \sum_{j=1}^{k} p_j = 1, p_j \geq 0 \ \forall j = 1, ..., k.
\end{aligned}$$

To understand how to do the above optimization learn about the method of lagrange multipliers. This is a very reasonable approach but it has one problem, how does one estimate the uncertainty in the estimate of $\{\hat{p}_1, ..., \hat{p}_k\}$?

We can formally model the uncertainty using Bayes rule

$$\mathrm{P}(M \mid D) \propto \mathrm{P}(D \mid M)\mathrm{P}(M),$$

if we can put a probability distribution on the model space, in this case $(p_1, ..., p_k)$. The space of all points $\mathbf{p} = (p_1, ..., p_k)$ such that $\sum_j p_k = 1$ and $p_k \geq 0$ for all $j = 1, ..., k$ is called the simplex. We now state a classical distribution on the simplex called the Dirichlet distribution

$$\begin{aligned} f(p_1, ..., p_k \mid \alpha_1, ..., \alpha_k) &\equiv \operatorname{Dir}(\alpha_1, ..., \alpha_k) \\ &\propto \prod_{j=1}^{k} p_j^{\alpha_j - 1}, \quad \alpha_j \geq 0 \, \forall j, \, \alpha_j \in \mathbb{N}, \end{aligned}$$

where $\mathbb{N}$ are the natural numbers, it is natural to think of the $\{\alpha_1, ..., \alpha_k\}$ parameters as counts. We can use the Dirichlet distribution as a prior $\pi(M)$ with the uniform prior being $\operatorname{Dir}(\alpha_1 = 1, ..., \alpha_k = 1)$. We now state the posterior

$$\begin{aligned} \mathrm{P}(M \mid D) &\propto \operatorname{Lik}(D \mid p_1, ..., p_4) \times \pi(p_1, ..., p_4) \\ &\propto \prod_{i=1}^{n} \left[\prod_{\ell=1}^{m} \left(\prod_{j=1}^{k} p_j^{n_{i\ell j}} \right) \right] \times \prod_{j=1}^{k} p_j^{\alpha_j - 1} \\ &\propto \prod_{\ell=1}^{m} \left[\prod_{i=1}^{n} \left(\prod_{j=1}^{k} p_j^{n_{i\ell j}} \right) \right] \times \prod_{j=1}^{k} p_j^{\alpha_j - 1} \\ &\propto \prod_{\ell=1}^{m} \left[\prod_{j=1}^{k} p_j^{\tilde{n}_{\ell j}} \right] \times \prod_{j=1}^{k} p_j^{\alpha_j - 1} \\ &\propto \prod_{\ell=1}^{m} \left[\prod_{j=1}^{k} p_j^{\tilde{n}_{\ell j}} \right] \times \prod_{j=1}^{k} p_j^{\alpha_j - 1} \\ &\propto \left[\prod_{j=1}^{k} p_j^{\breve{n}_j} \right] \times \prod_{j=1}^{k} p_j^{\alpha_j - 1} \\ &\propto \left[\prod_{j=1}^{k} p_j^{\breve{n}_j + \alpha_j - 1} \right] \\ &= \operatorname{Dir}(\breve{n}_1 + \alpha_1, ..., \breve{n}_k + \alpha_k), \end{aligned}$$

where $\breve{n}_j = \sum_{i\ell} n_{i\ell j}$. The strength of this estimation procedure is that we end up with not just a point estimate $\{\hat{p}_1, ..., \hat{p}_k\}$ as we did in the MLE approach but we end up with a posterior distribution. We can use the highest probability value for $(p_1, ..., p_k)$ as an estimate or the mean of the posterior distribution. The reason why this example worked so easily is that the multinomial and Dirichlet distributions are conjugate. By this we mean that

$$\operatorname{Multi}(p_1, ..., p_k) \times \operatorname{Dir}(\alpha_1, ..., \alpha_k) = \operatorname{Dir}(p_1 + \alpha_1, ..., p_k + \alpha_k).$$

LECTURE 2
Linear regression the proceduralist approach

2.1. Standard multivariate linear regression

The regression problem is usually stated as

$$Y = f(X) + \varepsilon, \quad \varepsilon \overset{iid}{\sim} \mathrm{N}(0, \sigma^2)$$

the multivariate random variable $X \subseteq \mathbb{R}^p$ are called covariates and the univariate random variable $Y \subseteq \mathbb{R}$ is called the response, the function belongs to a class of functions $f \in \mathcal{F}$. The random variables X, Y have associated with them the following distributions

$$\text{joint: } \rho_{X,Y}(x,y), \quad \text{marginals: } \rho_X(x), \rho_Y(y), \quad \text{conditional: } \rho(y \mid x).$$

For now the class of functions $\mathcal{F}$ will be linear functions

$$f(x) = \beta^T x, \quad \beta \in \mathbb{R}^p.$$

The data we are given consists of n observations $D = \{(x_i, y_i)\}_{i=1}^n \overset{iid}{\sim} \rho(x,y)$, we call this a sample and the sample size is n. We will assume the data we observed is consistent with the following model

$$Y_i = \beta^T X_i + \varepsilon_i, \quad \varepsilon_i \overset{iid}{\sim} \mathrm{N}(0, \sigma^2).$$

Our goal is given the data D solve the following problems:

(1) parameter inference: what is a reasonable estimate for β, we'll call the estimate $\hat{\beta}$
(2) prediction: given a new x_* what is the corresponding y_*, try $y_* = \hat{\beta}^T x_*$.
(3) estimating the conditional distribution: what is $Y \mid X = x$.

An idea to estimate $\hat{\beta}$ that goes back to Gauss is the minimizing the least squares error

$$\arg\min_{\beta \in \mathbb{R}^p} \left[\frac{1}{n} \sum_i (y_i - \beta^T x_i)^2, \right].$$

One can derive the above estimator from the following probabilistic model

$$\mathrm{Lik}(D; \beta) = \prod_{i=1}^n \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\|y_i - \beta^T x_i\|^2}{2\sigma^2} \right),$$

by maximizing the likelihood above with respect to β

$$\arg\max_{\beta} \mathrm{Lik}(D;\beta) \equiv \arg\min_{\beta\in\mathrm{I\!R}^p} \left[\frac{1}{n}\sum_i (y_i - \beta^T x_i)^2, \right].$$

We can state the negative of the log likelihood as

$$L = \sum_{i=1}^n (y_i - x_i^T\beta)^2.$$

We will rewrite the above in matrix notation. In doing this we define a matrix $\mathbf{X}$ which is $n \times p$ and each row of the matrix is a data point x_i. We also define a column vector Y $(p \times 1)$ with y_i as the i-th element of y. Similarly β is a column vector with p rows. We can rewrite the error minimization as

$$\arg\min_{\beta} \left[L = (Y - \mathbf{X}\beta)^T(Y - \mathbf{X}\beta)\right],$$

taking derivatives with respect to θ and setting this equal to zero (taking derivatives with respect to β means taking derivatives with respect to each element in β)

$$\begin{aligned} \frac{dL}{d\beta} &= -2\mathbf{X}^T(Y - \mathbf{X}\beta) = 0 \\ &= \mathbf{X}^T(Y - \mathbf{X}\beta) = 0. \end{aligned}$$

this implies

$$\begin{aligned} \mathbf{X}^T Y &= \mathbf{X}^T\mathbf{X}\beta \\ \hat{\beta} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T Y. \end{aligned}$$

If we look at the above formula carefully there is a serious numerical problem – $(\mathbf{X}^T\mathbf{X})^{-1}$. The matrix $\mathbf{X}^T\mathbf{X}$ is a $p \times p$ matrix of rank n where $p \gg n$. This means that $\mathbf{X}^T\mathbf{X}$ cannot be inverted so we cannot compute the estimate $\hat{\beta}$ by matrix inversion. There are numerical approaches to address this issue but the solution will not be unique or stable. A general rule in estimation problems is that numerical problems in estimating the parameters usually coincide with with statistical errors or variance of the estimate.

2.2. The Stein estimator

The above numerical problem is related to an amazing result that was first observed in 1956 by Charles Stein. The question that Stein asked is if one is given n observations from a multivariate normal

$$(x_i)_{i=1}^n \overset{iid}{\sim} \mathrm{N}(\theta, \sigma^2\mathbf{I}),$$

what is the best estimator of θ. In statistics if there exists an estimator that is better than the one you are using then your estimator is called inadmissible. What Stein found was that for $p \geq 3$ the sample mean

$$\hat{\theta} = \frac{1}{n}\sum_i x_i,$$

is not admissible. A better estimator called the James-Stein estimator is the following

$$\hat{\theta} = \left(1 - \frac{(p-2)\frac{\sigma^2}{n}}{\|\bar{x}\|^2}\right)\bar{x}.$$

The intuition about the above estimator is take the sample mean $\bar{x}$ and move it a bit towards zero, this is called shrinkage or shrinkage towards zero. Now what is an optimal estimator, it is one which minimizes

$$\arg\min_{\beta \in \mathbb{R}^p} \left[\mathbb{E}_{X,Y}\left[(y-\beta^T x)^2\right] = \int_{Y,X} (y-\beta^T x)^2 \rho(x,y)\ \mathrm{d}x\ \mathrm{d}y\right],$$

the idea above is minimizing the error on unseen data, later in the course we will call the above the generalization error

$$I[\hat{f}] = \mathbb{E}_{X,Y}\left[(y-\hat{f}(x))^2\right], \quad \hat{f}(x) = \hat{\beta}^T x,$$

and provide some theory to justify the estimators we will propose.

2.3. Cross-validation

One cannot estimate the generalization error since one does not have access to the generating distribution $\rho(x,y)$. A common proxy for the generalization error is the leave-one-out cross-validation error

$$I[cv] \equiv \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}_{D^{\backslash i}}(x_i))^2,$$

where $D^{\backslash i}$ is the data set with the i-th sample removed and $\hat{f}_{D^{\backslash i}}$ is the function estimated when the i-th sample is removed. The idea is to remove the i-th sample, estimate a function with that sample left out, then test the error made on the i-th sample, and average this over all n observations. Of course one need not leave-out one observation but can leave-out k observations. The leave-one-out estimator is (almost) unbiased.

2.4. Shrinkage models

We can now adapt the idea behind the James-Stein estimator to the linear regression problem. We will minimize the following loss function

$$\arg\min_{\beta \in \mathbb{R}^p} \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2 + \lambda\|\beta\|^2,$$

where $\lambda > 0$ is a parameter and $\|\beta\|^2 = \sum_{i=1}^{p}\beta_i^2$.

$$\begin{aligned} \frac{dL}{d\beta} &= -2\mathbf{X}^T(Y - \mathbf{X}\beta) + 2\lambda n\beta = 0 \\ &= \mathbf{X}^T(\mathbf{X}\beta - Y) + \lambda n\beta. \end{aligned}$$

this implies

$$\begin{aligned} \mathbf{X}^T Y &= \mathbf{X}^T\mathbf{X}\beta + \lambda n\beta \\ \mathbf{X}^T Y &= (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})n\beta \\ \hat{\beta} &= (\mathbf{X}^T\mathbf{X} + \lambda n\mathbf{I})^{-1}\mathbf{X}^T Y, \end{aligned}$$

where $\mathbf{I}$ is the $p \times p$ identity matrix. The matrix $(\mathbf{X}^T\mathbf{X} + \lambda n \mathbf{I})$ is invertible and this penalized loss function approach has had great success in problems with more variables than observations.

LECTURE 3
A Bayesian motivation for the proceduralist approach

There are typically two ways to validate a statistical estimation procedure

(1) Show the procedure is consistent.
(2) Show there is a Bayesian procedure that produces the same results.

3.1. Consistency

What is meant by a consistent estimator in the context of regression is the following:

Definition (Consistency). *In regression an estimator $\hat{f}$ selected from a class of functions $\mathcal{F}$ is consistent if* $\forall \varepsilon > 0$

$$\lim_{n\to\infty} \sup_{\rho} \mathbb{P}_D \left\{ I[\hat{f}] > \inf_{f\in\mathcal{F}} I[f] + \varepsilon \right\} = 0,$$

where ρ is the joint distribution of the data, and $D \equiv \{(X_i, Y_i)\}_{i=1}^n \overset{iid}{\sim} \rho(x,y)$ are data sampled iid from the joint distribution.

Later in the course (in a few lectures) we will discuss why criterion (2) is valid. We will now develop a Bayesian interpretation for the procedure developed in the previous lecture.

3.2. Likelihoods

The first step in any Bayesian formulation is to state the likelihood model for the data. A more formal statement of the previous sentence falls under what is called the Likelihood principle, which can be paraphrased as all the evidence in a sample relevant to model parameters is contained in the likelihood function.

Our model so far has been

$$Y_i = f(X_i) + \varepsilon_i, \quad \varepsilon_i \overset{iid}{\sim} \mathrm{N}(0, \sigma^2),$$

with $f \in \mathcal{F}$ and $\mathcal{F} = \left\{ f \mid f(x) = \beta^T x \right\}$. There are two sets of parameters in this model: the vector β and the variance of the error σ^2 or $\theta = \{\beta, \sigma^2\}$. The likelihood function can be stated as

$$\mathrm{Lik}(D;\theta) = \prod_{i=1}^n \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\|y_i - \beta^T x_i\|^2}{2\sigma^2} \right).$$

An important idea above the above likelihood is the idea of a sufficient statistic $T(x)$. This means that once the sufficient statistic is computed the data can be thrown out with no loss of information. An example is given the likelihood for a univariate normal with known variance σ^2 the sample mean $t(x_1, ...x_n) = n^{-1} \sum_i x_i$ is a sufficient statistic. A sufficient statistic can be thought of as compressing the information in a data set. The standard way to check if a statistic is sufficient is via what is called the Neyman-Fisher Factorization Criteria:

Definition (Neyman-Fisher Factorization)**.** *If a density has the following factorization*

$$f(x_1, ..., x_n; \theta) = g(t(x_1, ..., x_n), \theta)\, h(x_1, ..., x_n),$$

then $t(x_1, ..., x_n)$ is a sufficient statistic. The above suggests we can decouple the sufficient statistic t from the data $x_1, ...x_n$.

There are a class of likelihood functions we will often work with because they have nice properties and they are useful in modeling the error or noise in the data. This class is called the exponential family and the above normal likelihood is one example.

Definition (Exponential family)**.** *A density $f(x \mid \theta)$ belongs to the exponential family if the density function as the following form*

$$f(x \mid \theta) = h(x)\, g(\theta)\, \exp\left(\eta(\theta)^T \cdot T(x) - A(\theta)\right),$$

where $T(x)$ are the sufficient statistics of the data, $\eta(\theta)$ is a function (sometimes the identity of the parameters), $h(x)$ and $g(\theta)$ serve to normalize the density.

A wide variety of likelihood models belong to the exponential family including the multivariate normal, binomial, multinomial, Poisson, and exponential densities.

3.2.1. Univariate normal

We now show that the univariate normal belongs to the exponential family.

$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-(x-\mu)^2/(2\sigma^2)\right)$$

$$\eta = \left(\frac{\mu}{\sigma^2}, -\frac{1}{2\sigma^2}\right)^T, \quad h(x) = \frac{1}{\sqrt{2\pi}}$$

$$T(x) = \left(x, x^2\right)^T, \quad g(\eta) = \frac{\mu^2}{2\sigma^2} + \ln|\sigma|$$

3.2.2. Bernoulli

We now consider the Bernoulli distribution. For a Bernoulli random variable $x \sim \mathrm{Be}(\pi)$ where π is the mean parameter of the random variable X. The exponential family distribution is as follows

$$\begin{aligned} \mathrm{Be}(x \mid \pi) &= \pi^x (1-\pi)^{1-x} \\ &= \exp\left[\log(\pi^x (1-\pi)^{1-x})\right] \\ &= \exp\left[x \log \pi + (1-x)\log(1-\pi)\right] \\ &= \exp\left[x(\log \pi - \log(1-\pi)) + \log(1-\pi))\right] \\ &= \exp\left[x \log\left(\frac{\pi}{1-\pi}\right) + \log(1-\pi)\right] \end{aligned}$$

On comparing the above formula with the exponential family form, we have

$$\begin{aligned} h(x) &= 1 \\ T(x) &= x \\ \eta &= \log\left(\frac{\pi}{1-\pi}\right) \\ g(\eta) &= \log\left(\frac{1}{1-\pi}\right) \\ &= \log(1+\exp(\eta)) \end{aligned}$$

3.3. Maximum a posteriori estimation

From the derivations in the previous section if we assume the standard linear regresion model with know variance we can specify the likelihood as

$$\text{Lik}(D;\beta) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\|y_i - \beta^T x_i\|^2}{2\sigma^2}\right).$$

We can also specify a prior on the effect size parameters β motivated by the James-Stein or shrinkage model

$$\pi(\beta) = \frac{1}{(2\pi)^{p/2}\gamma^{1/2}} \exp\left(-\frac{1}{2}\beta^T(\tau_0^2\mathbf{I}_p)^{-1}\beta\right),$$

By Bayes' rule the posterior probability on β is

$$\text{Post}(\beta \mid D) \propto \left[\prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\|y_i - \beta^T x_i\|^2}{2\sigma^2}\right)\right] \times \frac{1}{(2\pi)^{p/2}\gamma^{1/2}} \exp\left(-\frac{1}{2}\beta^T(\tau_0^2\mathbf{I}_p)^{-1}\beta\right).$$

We can write the negative of the log of the posterior as

$$L = (2\sigma^2)^{-1}\sum_{i=1}^{n} \|y_i - \beta^T x_i\|^2 + \frac{1}{2\tau_0^2}\beta^T\beta,$$

which can be rewritten as

$$\begin{aligned} L &= \frac{1}{n}\sum_{i=1}^{n} \|y_i - \beta^T x_i\|^2 + \frac{\sigma^2}{n\tau_0^2}\beta^T\beta \\ &= \frac{1}{n}\sum_{i=1}^{n} \|y_i - \beta^T x_i\|^2 + \lambda_n\beta^T\beta, \end{aligned}$$

where the regularization parameter λ_n is now a function of the sample size n and has an interpretation of the ration of the variance of the noise over the variance of the prior. We can minimize L to obtain what is called the maximum a posteriori (MAP) estimator

$$\hat{\beta} = \arg\min_{\beta}\left[\frac{1}{n}\sum_{i=1}^{n} \|y_i - \beta^T x_i\|^2 + \lambda_n\beta^T\beta\right],$$

which is nothing but the shrinkage estimator of the previous section and

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X} + \lambda_n n\mathbf{I})^{-1}\mathbf{X}^T Y.$$

3.4. Conjugate priors

When the prior and the posterior have the same form, we say that the prior is a conjugate prior for the corresponding likelihood. For all distributions in the exponential family, we can derive a conjugate prior. Let the prior be $p(\eta \mid \tau)$, where τ denotes the hyper-parameters. The posterior can be written as:

$$p(\eta \mid X) \propto p(X \mid \eta)\, p(\eta \mid \tau)$$

The likelihood of the exponential family is:

$$p(X \mid \eta) = \left(\prod_{i=1}^{n} h(x_i)\right) \exp\left(\eta^T \sum_{i=1}^{n} T(x_i) - ng(\eta)\right)$$

Assume the prior has the form

$$p(\eta \mid \tau) \propto \exp\left\{\eta^T \tau - \tau_0 g(\eta)\right\},$$

where we observe an inner product structure between the hyper-parameters and the parameter η. Then the posterior can be written as:

$$\begin{aligned} p(\eta \mid X) &\propto p(X \mid \eta)\ p(\eta \mid \tau) \\ &\propto \exp\left(\eta^T \sum_{i=1}^{n} T(x_i) - ng(\eta)\right) \exp\left(\eta^T \tau - \tau_0 g(\eta)\right) \\ &= \exp\left\{\eta^T \left(\sum_{i=1}^{n} T(x_i) + \tau\right) - (n + \tau_0) g(\eta)\right\} \end{aligned}$$

The posterior has the same exponential family form as the prior, and the posterior hyper-parameters are adding the sum of the sufficient statistics to hyper-parameters of the conjugate prior. The exponential family is the only family of distributions for which the conjugate priors exist. This is a convenient property of the exponential family because conjugate priors simplify computation of the posterior. We can do algebra instead of calculus, integration.

3.4.0.1. *Algebraic perspective.* A family of priors is conjugate if it is closed under sampling, This means that a family $\mathcal{F}$ of distributions over $\eta \in \Theta$ is closed under sampling with respect to a sampling distribution $p(x \mid \eta)$ if and only if for any sample $p(\eta) \in \mathcal{F}$ it holds that $p(\eta \mid x) \in \mathcal{F}$.

An interesting observation is under mild conditions conjugate priors can be characterized by the following posterior linearity condition

$$\mathbb{E}\Big[\mathbb{E}(X \mid \eta) \mid X = x\Big] = ax + b.$$

USEFUL PROPERTIES OF THE MULTIVARIATE NORMAL*

3.1. Conditionals and marginals

For Bayesian analysis it is very useful to understand how to write joint, marginal, and conditional distributions for the multivariate normal.

Given a vector $x \in \mathbb{R}^p$ the multivariate normal density is

$$f(x) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right).$$

Now split the vector into two parts

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \quad \text{of size} \begin{bmatrix} q \times 1 \\ (p-q) \times 1 \end{bmatrix},$$

and

$$\Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}, \quad \text{of size} \begin{bmatrix} q \times q & q \times (p-q) \\ (p-q) \times q & (p-q) \times (p-q) \end{bmatrix}.$$

We now state the joint and marginal distributions

$$x_1 \sim \mathrm{N}(\mu_1, \Sigma_{11}), \quad x_2 \sim \mathrm{N}(\mu_2, \Sigma_{22}), \quad x \sim \mathrm{N}(\mu, \Sigma),$$

and the conditional density

$$x_1 \mid x_2 \sim \mathrm{N}\left(\mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}\right).$$

The same idea holds for other sizes of partitions.

3.2. Conjugate priors

3.2.1. Univariate normals

3.2.1.1. *Fixed variance, random mean.* We consider the parameter σ^2 fixed so we are interested in the conjugate prior for μ:

$$\pi(\mu \mid \mu_0, \sigma^2) \propto \frac{1}{\sigma_0} \exp\left(-\frac{1}{2\sigma_0^2}(\mu - \mu_0)^2\right),$$

where μ_0 and σ^2 are hyper-parameters for the prior distribution (when we don't have informative prior knowledge we typically consider $\mu_0 = 0$ and σ^2 large).

The posterior distribution for $x_1, ..., x_n$ with a univariate normal likelihood and the above prior will be

$$\text{Post}(\mu \mid x_1, ..., x_n) \sim \text{N}\left(\frac{\sigma_0^2}{\frac{\sigma^2}{n} + \sigma_0^2}\bar{x} + \frac{\sigma^2}{\frac{\sigma^2}{n} + \sigma_0^2}\mu_0, \Big(\frac{1}{\sigma_0^2} + \frac{n}{\sigma^2}\Big)^{-1} \right).$$

3.2.1.2. *Fixed mean, random variance.* We will formulate this setting with two parameterizations of the scale parameter: (1) the variance σ^2, (2) the precision $\tau = \frac{1}{\sigma^2}$.

The two conjugate distributions are the Gamma and the inverse Gamma (really they are the same distribution, just reparameterized)

$$\text{IG}(\alpha, \beta) : f(\sigma^2) = \frac{\beta^\alpha}{\Gamma(\alpha)}(\sigma^2)^{-\alpha-1}\exp(-\beta(\sigma^2)^{-1}), \quad \text{Ga}(\alpha, \beta) : f(\tau) = \frac{\beta^\alpha}{\Gamma(\alpha)}\tau^{\alpha-1}\exp(-\beta\tau).$$

The posterior distribution of σ^2 is

$$\sigma^2 \mid x_1, ..., x_n \sim \text{IG}\Big(\alpha + \frac{n}{2}, \beta + \frac{1}{2}\sum(x_i - \mu)^2\Big).$$

The posterior distribution of τ is not surprisingly

$$\tau \mid x_1, ..., x_n \sim \text{Ga}\Big(\alpha + \frac{n}{2}, \beta + \frac{1}{2}\sum(x_i - \mu)^2\Big).$$

3.2.1.3. *Random mean, random variance.* We now put the previous priors together in what is called a Bayesian hierarchical model:

$$\begin{aligned} x_i \mid \mu, \tau &\overset{iid}{\sim} \text{N}(\mu, (\tau)^{-1}) \\ \mu \mid \tau &\sim \text{N}(\mu_0, (\kappa_0\tau)^{-1}) \\ \tau &\sim \text{Ga}(\alpha, \beta). \end{aligned}$$

For the above likelihood and priors the posterior distribution for the mean and precision is

$$\begin{aligned} \mu \mid \tau, x_1, ..., x_n &\sim \text{N}\Big(\frac{\mu_0\kappa_0 + n\bar{x}}{n + \kappa_0}, (\tau(n + \kappa_0))^{-1}\Big) \\ \tau \mid x_1, ..., x_n &\sim \text{Ga}\Big(\alpha + \frac{n}{2}, \beta + \frac{1}{2}\sum(x_i - \bar{x})^2 + \frac{n}{n+1}\frac{(\bar{x} - x_i)^2}{2}\Big). \end{aligned}$$

3.2.2. Multivariate normal

Given a vector $x \in \mathbb{R}^p$ the multivariate normal density is

$$f(x) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}}\exp\left(-\frac{1}{2}(x - \mu)^T\Sigma^{-1}(x - \mu)\right).$$

We will work with the precision matrix instead of the covariance and we will consider the following Bayesian hierarchical model:

$$\begin{aligned} x_i \mid \mu, \Lambda &\overset{iid}{\sim} \text{N}(\mu, (\Lambda)^{-1}) \\ \mu \mid \Lambda &\sim \text{N}(\mu_0, (\kappa_0\Lambda)^{-1}) \\ \Lambda &\sim \text{Wi}(\Lambda_0, n_0), \end{aligned}$$

the precision matrix is modeled using the Wishart distribution

$$f(\Lambda; V, n) = \frac{|\Lambda|^{(n-d-1)/2}\exp(-.5\text{tr}(\Lambda V^{-1}))}{2^{nd/2}|V|^{n/2}\Gamma_d(n/2)}.$$

For the above likelihood and priors the posterior distribution for the mean and precision is

$$\begin{aligned}
\mu \mid \Lambda, x_1, ..., x_n &\sim \mathrm{N}\Big(\frac{\mu_0\kappa_0 + n\bar{x}}{n+\kappa_0}, (\Lambda(n+\kappa_0))^{-1}\Big) \\
\Lambda \mid x_1, ..., x_n &\sim \mathrm{Wi}\bigg(n_0 + \frac{n}{2}, \Lambda_0 + \frac{1}{2}\left[\bar{\Sigma} + \frac{\kappa_0}{\kappa_0 + n}(\bar{x} - \mu_0)(\bar{x} - \mu_0)^T\right]\bigg).
\end{aligned}$$

LECTURE 4
A Bayesian approach to linear regression

The main motivations behind a Bayesian formalism for inference are a coherent approach to modeling uncertainty as well as an axiomatic framework for inference. We will reformulate multivariate linear regression from a Bayesian formulation in this section.

Bayesian inference involves thinking in terms of probability distributions and conditional distributions. One important idea is that of a conjugate prior. Another tool we will use extensively in this class is the multivariate normal distribution and its properties.

4.1. Conjugate priors

Given a likelihood function $p(x \mid \theta)$ and a prior $\pi(\theta)$ on can write the posterior as

$$p(\theta \mid x) = \frac{p(x \mid \theta)\pi(\theta)}{\int_{\theta'} p(x \mid \theta')\pi(\theta')\,\mathrm{d}\theta'} = \frac{p(x,\theta)}{p(x)},$$

where $p(x)$ is the marginal density for the data, $p(x,\theta)$ is the joint density of the data and the parameter θ.

The idea of a prior and likelihood being conjugate is that the prior and the posterior densities belong to the same family. We now state some examples to illustrate this idea.

Beta, Binomial: Consider the Binomial likelihood with n (the number of trials) fixed

$$f(x \mid p, n) = \binom{n}{x} p^x (1-p)^{n-x},$$

the parameter of interest (the probability of a success) is $p \in [0,1]$. A natural prior distribution for p is the Beta distribution which has density

$$\pi(p;\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}, \quad p \in (0,1) \text{ and } \alpha,\beta > 0,$$

where $\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$ is a normalization constant. Given the prior and the likelihood densities the posterior density modulo normalizing constants will take the form

$$\begin{aligned} f(p \mid x) &\propto \left[\binom{n}{x} \times \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\right] p^x(1-p)^{n-x} \times p^{\alpha-1}(1-p)^{\beta-1}, \\ &\propto p^{x+\alpha-1}(1-p)^{n-x+\beta-1}, \end{aligned}$$

which means that the posterior distribution of p is also a Beta with

$$p \mid x \sim \text{Beta}(\alpha+x, \beta+n-x).$$

Normal, Normal: Given a normal distribution with unknown mean the density for the likelihood is

$$f(x \mid \theta, \sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2}(x-\theta)^2\right),$$

and one can specify a normal prior

$$\pi(\theta; \theta_0, \tau_0^2) \propto \exp\left(-\frac{1}{2\tau_0^2}(\theta-\theta_0)^2\right),$$

with hyper-parameters θ_0 and τ_0. The resulting posterior distribution will have the following density function

$$f(\theta \mid x) \propto \exp\left(-\frac{1}{2\sigma^2}(x-\theta)^2\right) \times \exp\left(-\frac{1}{2\tau_0^2}(\theta-\theta_0)^2\right),$$

which after completing squares and reordering can be written as

$$\theta \mid x \sim \text{N}(\theta_1, \tau_1^2), \quad \theta_1 = \frac{\frac{\theta_0}{\tau_0^2} + \frac{x}{\sigma^2}}{\frac{1}{\tau_0^2} + \frac{1}{\sigma^2}}, \quad \tau_1^2 = \frac{1}{\frac{1}{\tau_0^2} + \frac{1}{\sigma^2}}.$$

4.2. Bayesian linear regression

We start with the likelihood as

$$f(Y \mid \mathbf{X}, \beta, \sigma^2) = \prod_{i=1}^n \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\|y_i - \beta^T x_i\|^2}{2\sigma^2}\right).$$

and the prior as

$$\pi(\beta) \propto \exp\left(-\frac{1}{2\tau_0^2}\beta^T\beta\right).$$

The density of the posterior is

$$\text{Post}(\beta \mid D) \propto \left[\prod_{i=1}^n \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\|y_i - \beta^T x_i\|^2}{2\sigma^2}\right)\right] \times \frac{1}{(2\pi)^{p/2}\gamma^{1/2}} \exp\left(-\frac{1}{2\tau_0^2}\beta^T\beta\right).$$

With a good bit of manipulation the above can be rewritten as a multivariate normal distribution

$$\beta \mid Y, \mathbf{X}, \sigma^2 \sim \text{N}_p(\mu_1, \Sigma_1)$$

with

$$\Sigma_1 = (\tau_0^{-2}\mathbf{I}_p + \sigma^{-2}\mathbf{X}^T\mathbf{X})^{-1}, \quad \mu_1 = \sigma^{-2}\Sigma_1 \ \mathbf{X}^T Y.$$

Note the similarities of the above distribution to the MAP estimator. Relate the mean of the above estimator to the MAP estimator.

Predictive distribution: Given data $D = \{(x_i, y_i)\}_{i=1^n}$ and and a new value x_* one would like to estimate y_*. This can be done using the posterior and is called the posterior predictive distribution

$$f(y_* \mid D, x_*, \sigma^2, \tau_0^2) = \int_{\mathbb{R}^p} f(y_* \mid x_*, \beta, \sigma^2) f(\beta \mid Y, \mathbf{X}, \sigma^2, \tau_0^2) \, \mathrm{d}\beta,$$

where with some manipulation

$$y_* \mid D, x_*, \sigma^2, \tau_0^2 \sim \mathrm{N}(\mu_*, \sigma_*^2),$$

where

$$\mu_* = \frac{1}{\sigma^2} \Sigma_1 \mathbf{X}^T Y x_*, \quad \sigma_*^2 = \sigma^2 + x_*^T \Sigma_1 x_*.$$

REVIEW OF FUNCTIONAL ANALYSIS*

A function space is a space of functions where each function can be thought of as a point in Euclidean space. Functional analysis is loosely speaking a mathematical understanding of function spaces. In the next lecture we will study a very useful function space called a Reproducing kernel Hilbert space (riches) which is used extensively in non-linear regression.

4.1. Hilbert Spaces

Examples. *The following are three examples of function spaces defined on a subset of the real line. In these examples the subset of the real line we consider is $x \in [a, b]$ where for example $a = 0$ and $b = 10$.*

1. *$C[a, b]$ is the set of all real-valued continuous functions on $x \in [a, b]$.* *$y = x^3$ is in $C[a, b]$ while $y = \lceil x \rceil$ is not in $C[a, b]$.*
2. *$L_2[a, b]$ is the set of all square integrable functions on $x \in [a, b]$. If $(\int_a^b |f(x)|^2 \, dx)^{1/2} < \infty$ then $f \in L_2[a, b]$.* *$y = x^3$ is in $L_2[a, b]$ and so is $y = x^3 + \delta(x - c)$ where $a < c < b$, however the second function is not defined at $x = c$.*
3. *$L_1[a, b]$ is the set of all functions whose absolute value is integrable on $x \in [a, b]$.* *$y = x^3$ is in $L_1[a, b]$ and so is $y = x^3 + \delta(x - c)$ where $a < c < b$, however the second function is not defined at $x = c$.*

Definition. *A normed vector space is a space $\mathcal{F}$ in which a norm is defined. A function $\| \cdot \|$ is a norm iff for all $f, g \in \mathcal{F}$*

1. $\|f\| \geq 0$ *and* $\|f\| = 0$ iff $f = 0$
2. $\|f + g\| \leq \|f\| + \|g\|$
3. $\|\alpha f\| = |\alpha| \; \|f\|$.

Note, if all conditions are satisfied except $\|f\| = 0$ iff $f = 0$ then the space has a seminorm instead of a norm.

Definition. *An inner product space is a linear vector space $\mathcal{E}$ in which an inner product is defined. A real valued function $\langle \cdot, \cdot \rangle$ is an inner product iff $\forall f, g, h \in \mathcal{E}$ and $\alpha \in \mathbb{R}$*

1. $\langle f, g \rangle = \langle g, f \rangle$
2. $\langle f + g, h \rangle = \langle f, h \rangle + \langle g, h \rangle$ *and* $\langle \alpha f, g \rangle = \alpha \langle f, g \rangle$
3. $\langle f, f \rangle \geq 0$ *and* $\langle f, f \rangle = 0$ *iff* $f = 0$.

Given an inner product space the norm is defined as $\|f\| = \sqrt{\langle f, f \rangle}$ and an angle between vectors can be defined.

Definition. *For a normed space $\mathcal{A}$ a subspace $\mathcal{B} \subset \mathcal{A}$ is dense in $\mathcal{A}$ iff $\mathcal{A} = \bar{\mathcal{B}}$. Where $\bar{\mathcal{B}}$ is the closure of the set $\mathcal{B}$.*

Definition. *A normed space $\mathcal{F}$ is separable iff $\mathcal{F}$ has a countable dense subset.*

Example. *The set of all rational points is dense in the real line and therefore the real line is separable. Note, the set of rational points is countable.*

Counterexample. *The space of right continuous functions on $[0, 1]$ with the sup norm is not separable. For example, the step function*

$$f(x) = U(x - a) \;\; \forall a \in [0, 1]$$

cannot be approximated by a countable family of functions in the sup norm since the jump must occur at a and the set of all a is uncountable.

Definition. *A sequence $\{x_n\}$ in a normed space $\mathcal{F}$ is called a Cauchy sequence if $\lim_{n\to\infty} \sup_{m\geq n} \|x_n - x_m\| = 0$.*

Definition. *A normed space $\mathcal{F}$ is called complete iff every Cauchy sequence in it converges.*

Definition. *A Hilbert space, $\mathcal{H}$ is an inner product space that is complete, separable, and generally infinite dimensional.*
A Hilbert space has a countable basis.

Examples. *The following are examples of Hilbert spaces.*

(1) *$\mathbb{R}^n$ is the textbook example of a Hilbert space. Each point in the space $x \in \mathbb{R}^n$ can be represented as a vector $x = \{x_1, ..., x_n\}$ and the metric in this space is $\|x\| = \sqrt{\sum_{i=1}^n |x_i|^2}$. The space has a very natural basis composed of the n basis functions $e_1 = \{1, 0, ..., 0\}$, $e_2 = \{0, 1, ..., 0\}$,..., $e_n = \{0, 0, ..., 1\}$. The inner product between a vector x and a basis vector e_i is simply the projection of x onto the ith coordinate $x_i = \langle x, e_i \rangle$.*
Note, this is not an infinite dimensional Hilbert space.

(2) *L_2 is also a Hilbert space. This Hilbert space is infinite dimensional.*

4.2. Functionals and operators

Definition. *A linear functional on a Hilbert space $\mathcal{H}$ is a linear transformation $T : V \to \mathbb{R}$ from $\mathcal{H}$ into $\mathbb{R}$.*

A linear functional takes an element in a Hilbert space and outputs a real number, integration is an example of a linear functional.

Theorem (Riesz representation theorem)**.** *Let V be a finite-dimensional inner product space and let $T : V \to \mathbb{R}$ be a linear functional. Then there is a vector $w \in V$ such that $Tv = \langle v, w \rangle$ for all $v \in V$.*

An integral transformation is one example of an operator (in the rest of the course all examples of operators will be integral transforms). An operator T maps one vector space into another.

Definition. *An integral transform T maps one function into another function as follows*

$$g(u) = (Tf)(u) := \int_{t_1}^{t_2} K(t,u)f(t)\,\mathrm{d}t.$$

LECTURE 5
Reproducing kernel Hilbert spaces

Reproducing Kernel Hilbert Spaces (rkhs) are hypothesis spaces with some very nice properties. The main property of these spaces is the reproducing property which relates norms in the Hilbert space to linear algebra. This class of functions also has a nice interpretation in the context of Gaussian processes. Thus, they are important for computational, statistical, and functional reasons.

5.1. Reproducing Kernel Hilbert Spaces (rkhs)

We will use two formulations to describe rkhs. The first is less general and more constructive. The second is more general and abstract. The key idea in both formulations is that there is a kernel function $K : X \times X \to \mathbb{R}$ and this kernel function has associated to it a Hilbert space $\mathcal{H}_K$ that has wondrous properties for optimization and inference.

The algorithm we will study in detail in the next lecture is the following

$$\widehat{f} := \arg \min_{f \in \mathcal{H}_k} \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|^2_{\mathcal{H}_K},$$

where $\mathcal{H}_K$ is a rkhs and $\|f\|_{\mathcal{H}_K}$ is specific norm as defined by the reproducing kernel K. The beauty of the rkhs is the optimization problem in the above infinite dimensional function space can be rewritten as a quadratic programming problem which involves only vectors and matrices.

For the remainder of this lecture we constrain the Hilbert spaces to a compact domain X.

5.1.1. Constructive formulation

The development of rkhs in this subsection is seen in most formulations of Support Vector Machines (SVMs) and Kernel Machines. It is less general in that it relies on the reproducing kernel being a Mercer Kernel. It however requires less knowledge of functional analysis and is more intuitive for most people.

We start by defining the kernel or reproducing kernel function.

Definition. *The reproducing kernel (rk), $K(\cdot, \cdot)$ is a symmetric real valued function of two variables $s, t \in X$*

$$K(s,t) : X \times X \to \mathbb{R}.$$

In addition $K(s,t)$ must be positive definite, that is for all real $a_1, ..., a_n$ and $t_1, ..., t_n \in X$

$$\sum_{i,j=1}^{n} a_i a_j K(t_i, t_j) \geq 0.$$

If the above inequality is strict then $K(s,t)$ is strictly positive definite.

In this formulation we consider continuous reproducing kernels $K : X \times X \to \mathbb{R}$. We define an integral operator $L_K : L_2[X] \to C[X]$ by the following integral transform

$$L_K f := \int_X K(s,t) f(t) dt = g(t). \tag{5.1}$$

If K is positive definite then L_K is positive definite (the converse is also true) and therefore the eigenvalues of (5.1) are nonnegative.

We denote the eigenvalues and eigenvectors of (5.1) as $\{\lambda_1, ..., \lambda_k\}$ and $\{\phi_1, ..., \phi_k\}$ respectively, where

$$\int_X K(s,t)\phi_k(t) dt = \lambda_k \phi_k(t) \ \ \forall k.$$

We now state Mercer's theorem.

Theorem. *Given the eigenfunctions and eigenvalues of the integral equation defined by a symmetric positive definite kernel K*

$$\int_X K(s,t)\phi(s) ds = \lambda \, \phi(t).$$

The kernel has the expansion

$$K(s,t) = \sum_j \lambda_j \phi_j(s) \phi_j(t),$$

where convergence is in the $L_2[X]$ norm.

We can define the rkhs as the space of functions spanned by the eigenfunctions of the integral operator defined by the kernel

$$\mathcal{H}_K = \{ \, f \, | f(s) = \sum_k c_k \phi_k(s) \ \text{ and } \ \|f\|_{\mathcal{H}_K} < \infty \},$$

where the rkhs norm $\| \cdot \|_{\mathcal{H}_K}$ is defined as follows

$$\|f(s)\|^2_{\mathcal{H}_K} = \left\langle \sum_j c_j \phi_j(s), \sum_j c_j \phi_j(s) \right\rangle^2_{\mathcal{H}_K} := \sum_j \frac{c_j^2}{\lambda_j}.$$

Similarly the inner product is defined as follows

$$\langle f, g \rangle = \left\langle \sum_j c_j \phi_j(s), \sum_j d_j \phi_j(s) \right\rangle_{\mathcal{H}_K} := \sum_j \frac{d_j c_j}{\lambda_j}.$$

Part of a homework problem will be to prove the representer property

$$\langle f(\cdot, K(\cdot, x) \rangle_{\mathcal{H}_k} = f(x),$$

using Mercer's theorem and the above definition of the rkhs norm.

5.1.2. Kernels and feature space

The rkhs concept has been utilized in the SVM and kernel machines literature in what is unfortunately called the kernel trick.

Points in the domains $x \in X \subset \mathrm{I\!R}^d$ are mapped into a higher dimensional space by the eigenvalues and eigenfunctions of the reproducing kernel (the space is of the dimensionality of the number of nonzero eigenvalues of the integral operator defined by the kernel)

$$x \to \Phi(x) := \{\sqrt{\lambda_1}\phi_1(x), ..., \sqrt{\lambda_k}\phi_k(x)\}.$$

A standard L_2 inner product of two points mapped into the feature space can be evaluated by a kernel due to Mercer's theorem

$$K(s,t) = \langle \Phi(s), \Phi(t) \rangle_{L_2}.$$

5.1.3. Examples of kernel functions

Any (semi) positive definite function can be used as a kernel function. Examples include

(1) Linear kernel: $k(u,v) = \langle u, v \rangle$
(2) Polynomial kernel: $k(u,v) = (\langle u, v \rangle + b)^p$
(3) Gaussian kernel: $k(u,v) = \exp(-\kappa \|u - v\|^2)$
(4) Double exponential kernel: $k(u,v) = \exp(-\kappa \|u - v\|)$

5.2. Abstract formulation

Proposition. *A linear evaluation function L_t evaluates each function in a Hilbert space $f \in \mathcal{H}$ at a point t. It associates $f \in \mathcal{H}$ to a number $f(t) \in \mathrm{I\!R}$, $L_t[f] = f(t)$.*

(1) $L_t[f + g] = f(t) + g(t)$
(2) $L_t[af] = af(t)$.

Example. *The delta function $\delta(x - t)$ is a linear evaluation function for $C[a,b]$*

$$f(t) = \int_a^b f(x)\delta(x - t)dx.$$

Proposition. *A linear evaluation function is bounded if there exists an M such that for all functions in the Hilbert space $f \in \mathcal{H}$*

$$|L_t[f]| = |f(t)| \leq M\|f\|,$$

where $\|f\|$ is the Hilbert space norm.

Example. *For the Hilbert space $C[a,b]$ with the sup norm there exists a bounded linear evaluation function since $|f(x)| \leq M$ for all functions in $C[a,b]$. This is due to continuity and compactness of the domain. The evaluation function is simply $L_t[f] : t \to f(t)$ and $M = 1$.*

Counterexample. *For the Hibert space $L_2[a,b]$ there exists no bounded linear evaluation function. The following function is in $L_2[a,b]$*

$$y = x^3 + \delta(x - c) \quad \textit{where } a < c < b.$$

At the point $x = c$ there is no M such that $|f(c)| \leq M$ since the function is evaluated as "∞". This is an example of a function in the space that is not even defined pointwise.

Definition. *If a Hilbert space has a bounded linear evaluation function, L_t, then it is a Reproducing Kernel Hilbert Space (rkhs), $\mathcal{H}_K$.*

The following property of a rkhs is very important and is a result of the Riesz representation theorem.

Proposition. *If $\mathcal{H}_K$ is a rkhs then there exists an element in the space K_t with the property such that for all $f \in \mathcal{H}_K$*

$$L_t[f] = \langle K_t, f \rangle = f(t).$$

The inner product is in the rkhs norm and the element K_t is called the representer of evaluation of t.

Remark. The above property is somewhat amazing in that it says if a Hilbert space has a bounded linear evaluation function then there is an element in this space that evaluates all functions in the space by an inner product.
In the space $L_2[a, b]$ we say that the delta function evaluates all functions in $L_2[a, b]$

$$L_t[f] = \int_a^b f(x)\delta(x - t)dx.$$

However, the delta function is not in $L_2[a, b]$.

There is a deep relation between a rkhs and its reproducing kernel. This is characterized by the following theorem.

Theorem. *For every Reproducing Kernel Hilbert Space (rkhs) there exists a unique reproducing kernel and conversely given a positive definite function K on $X \times X$ we can construct a unique rkhs of real valued functions on X with K as its reproducing kernel (rk).*

Proof.
If $\mathcal{H}_K$ is a rkhs then there exists an element in the rkhs that is the representer evaluation by the Reisz representer theorem. We define the rk

$$K(s, t) := \langle K_s, K_t \rangle$$

where K_s and K_t are the representers of evaluation at s and t. The following hold by the properties of Hilbert spaces and the representer property

$$\begin{aligned}
\left\| \sum_j a_j K_{t_j} \right\|^2 &\geq 0 \\
\left\| \sum_j a_j K_{t_j} \right\|^2 &= \sum_{i,j} a_i a_j \langle K_{t_i}, K_{t_j} \rangle \\
\sum_{i,j} a_i a_j K(t_i, t_j) &= \sum_{i,j} a_i a_j \langle K_{t_i}, K_{t_j} \rangle.
\end{aligned}$$

Therefore $K(s, t)$ is positive definite.

We now prove the converse. Given a rk $K(\cdot, \cdot)$ we construct $\mathcal{H}_K$. For each $t \in X$ we define the real valued function

$$K_t(\cdot) = K(t, \cdot).$$

We can show that the rkhs is simply the completion of the space of functions spanned by the the set K_{t_i}

$$\mathcal{H} = \{f \,|\, f = \sum_i a_i K_{t_i} \quad \text{where} \quad a_i \in \mathbb{R},\;\; t_i \in X,\; \text{ and } \quad i \in \mathbb{N}\}$$

with the following inner product

$$\left\langle \sum_i a_i K_{t_i}, \sum_i a_i K_{t_i} \right\rangle = \sum_{i,j} a_i a_j \langle K_{t_i}, K_{t_j} \rangle = \sum_{i,j} a_i a_j K(t_i, t_j).$$

Since $K(\cdot,\cdot)$ is positive definite the above inner product is well defined. For any $f \in \mathcal{H}_K$ we can check that

$$\langle K_t, f \rangle = f(t)$$

because for any function in the above linear space norm convergence implies pointwise convergence

$$|f_n(t) - f(t)| = |\langle f_n - f, K_t \rangle| =\leq \|f_n - f\| \|K_t\|,$$

the last step is due to Cauchy-Schwartz. Therefore every Cauchy sequence in this space converges and it is complete. □

LECTURE 6
Non-linear regression

The algorithm we will study in detail in the next lecture is the following

$$\widehat{f} := \arg \min_{f \in \mathcal{H}_k} \frac{1}{n} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|^2_{\mathcal{H}_K}.$$

We will see the above minimizer as a particular form well suited for optimization due to the representer theorem.

6.1. A result of the representer theorem

The following are the three standard regularization methods:

(1) Tikhonov regularization: indirectly constrain the hypothesis space by adding a penalty term.

$$\min_{f \in \mathcal{H}} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) + \lambda \Omega(f) \right].$$

(2) Ivanov regularization: directly constrain the hypothesis space

$$\min_{f \in \mathcal{H}} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right] \quad \text{subject to} \quad \Omega(f) \leq \tau.$$

(3) Phillips regularization: directly constrain the hypothesis space

$$\min_{f \in \mathcal{H}} \Omega(f) \quad \text{subject to} \quad \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right] \leq \kappa.$$

Consider the rkhs norm will be as the regularization functional

$$\Omega(f) = \|f\|^2_{\mathcal{H}_K}.$$

This defines the following optimization problems:

$$(P1) \qquad \min_{f \in \mathcal{H}} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) + \lambda \|f\|^2_{\mathcal{H}_K} \right],$$

$$(P2) \qquad \min_{f \in \mathcal{H}} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right] \quad \text{subject to} \quad \|f\|^2_{\mathcal{H}_K} \leq \tau,$$

$$(P3) \qquad \min_{f \in \mathcal{H}} \|f\|^2_{\mathcal{H}_K} \quad \text{subject to} \quad \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right] \leq \kappa.$$

All the above optimization problems above are over spaces of functions that contain an infinite number of functions. Using the formulation in section 5.1.1 we can write any function in the rhks as

$$\mathcal{H}_K = \left\{ f \,|\, f(x) = \sum_k c_k \phi_k(x) \right\},$$

so the optimization procedure is over the coefficients c_k. The number of nonzero coefficients in the expansion defines the dimensionality of the rkhs and this can be infinite, for example the Gaussian kernel.

One of the amazing aspects of the all above optimization problems is that a minimizer satisfies the form

$$\widehat{f}(x) = \sum_{i=1}^{n} c_i K(x, x_i).$$

So the optimization procedure is over n real variables. This is formalized in the following "Representer Theorem."

Theorem. *Given a set of points $\{(x_i, y_i)\}_{i=1}^n$ a function of the form*

$$\widehat{f}(x) = \sum_{i=1}^{n} c_i K(x, x_i),$$

is a minimizer of the following optimization procedure

$$c\left((f(x_1), y_1), ..., (f(x_n), y_n)\right) + \lambda g(\|f\|_{\mathcal{H}_K}),$$

where $\|f\|_{\mathcal{H}_K}$ is a rkhs norm, $g(\cdot)$ is monotonically increasing, and c is an arbitrary cost function.

Procedure (P1) is special case of the optimization procedure stated in the above theorem.

Proof. For ease of notation all norms and inner products in the proof are rkhs norms and inner products.

Assume that the function f has the following form

$$f = \sum_{i=1}^{n} b_i \phi_i(x_i) + v,$$

where

$$\langle \phi_i(x_i), v \rangle = 0 \quad \forall i = 1, .., n.$$

The orthogonality condition simple ensures that v is not in the span of $\{\phi_i(x_i)\}_{i=1}^n$.

So for any point x_j $(j = 1, ..., n)$

$$f(x_j) = \left\langle \sum_{i=1}^n b_i \phi(x_i) + v, \phi(x_j) \right\rangle = \sum_{i=1}^n b_i \langle \phi(x_i), \phi(x_j) \rangle,$$

so v has no effect on the cost function

$$c\left((f(x_1), y_1), ..., (f(x_n), y_n)\right).$$

We now look at the rkhs norm

$$g(\|f\|) = g\left(\left\|\sum_{i=1}^n b_i \phi_i(x_i) + v\right\|\right) = g\left(\sqrt{\left\|\sum_{i=1}^n b_i \phi_i(x_i)\right\|^2 + \|v\|^2}\right) \geq g\left(\sqrt{\left\|\sum_{i=1}^n b_i \phi_i(x_i)\right\|^2}\right).$$

So the extra factor v increases the rkhs norm and has effect on the cost functional and therefore must be zero and the function has the form

$$\widehat{f} = \sum_{i=1}^n b_i \phi_i(x_i),$$

and by the reproducing property

$$\widehat{f}(x) = \sum_{i=1}^n a_i K(x, x_i). \ \square$$

Homework: proving a representer theorem for the other two regularization formulations.

6.2. Kernel ridge-regression

The Kernel ridge-regression (KRR) algorithm has been invented and reinvented many times and has been called a variety of names such as Regularization networks, Least Square Support Vector Machine (LSSVM), Regularized Least Square Classification (RLSC).

We start with Tikhonov regularization

$$\min_{f \in \mathcal{H}_K} \left[n^{-1} \sum_{i=1}^n V(f, z_i) + \lambda \Omega(f) \right]$$

and then set the regularization functional to a RKHS norm

$$\Omega(f) = \|f\|^2_{\mathcal{H}_K}$$

and use the square loss functional

$$n^{-1} \sum_{i=1}^n V(f, z_i) = n^{-1} \sum_{i=1}^n (f(x_i) - y_i)^2.$$

The resulting optimization problem is

$$\min_{f \in \mathcal{H}_K} \left[n^{-1} \sum_{i=1}^n (f(x_i) - y_i)^2 + \lambda \|f\|^2_{\mathcal{H}_K} \right], \tag{6.1}$$

the minimizer of which we know by the Representer theorem has the form

$$\widehat{f}(x) = \sum_{i=1}^n c_i K(x, x_i).$$

This implies that we only need to solve the optimization problem for the c_i. This turns the problem of optimizing over functions which maybe infinite-dimensional into a problem of optimizing over n real numbers.

Using the representer theorem we derive the optimization problem actually solved for Kernel ridge-regression.

We first define some notation. We will use the symbol K to refer to either the kernel function K or the $n \times n$ matrix K where

$$K_{ij} \equiv K(x_i, x_j).$$

Using this definition the function $f(x)$ evaluated at a training point x_j can be written in matrix notation as

$$\begin{aligned} f(x_j) &= \sum_{i=1}^n K(x_i, x_j) c_i \\ &= [Kc]_j, \end{aligned}$$

where $[Kc]_j$, is the jth element of the vector obtained in multiplying the kernel matrix K with the vector c. In this notation we can rewrite equation (6.1) as

$$\min_{f \in \mathcal{H}_K} \frac{1}{n} (Kc - y)^2 + \lambda \|f\|_K^2,$$

where y is the vector of y values. Also by the representer theorem the RKHS norm can be evaluated using linear algebra

$$||f||_K^2 = c^T K c,$$

where c^T is the transpose of the vector c. Substituting the above norm into equation (6.1) results in an optimization problem on the vector c

$$\arg \min_{c \in \mathbb{R}^n} \left[g(c) := \frac{1}{\ell}(Kc - y)^2 + \lambda c^T K c. \right]$$

This is a convex, differentiable function of c, so we can minimize it simply by taking the derivative with respect to c, then setting this derivative to 0.

$$\frac{\partial g(c)}{\partial c} = \frac{2}{\ell} K(Kc - y) + 2\lambda K c = 0.$$

We show that the solution of the above equation is the following linear system

$$c = (K + \lambda \ell I)^{-1} y,$$

where I is the identity matrix:

$$\begin{array}{cl} \text{differentiation} & 0 = \frac{2}{\ell} K(Kc - y) + 2\lambda K c \\ \text{multiplication} & K(Kc) + \lambda \ell K c = Ky \\ \text{“left multiplication by } K^{-1}\text{”} & (K + \lambda \ell I)c = y \\ \text{inversion} & c = (K + \lambda \ell I)^{-1} y. \end{array}$$

The matrix $K + \lambda \ell I$ is positive definite and will be well-conditioned if λ is not too small.

A few properties of the linear system are:

(1) The matrix $(K + \lambda \ell I)$ is guaranteed to be invertible if $\lambda > 0$. As $\lambda \to 0$, the regularized least-squares solution goes to the standard Gaussian least-squares solution which minimizes the empirical loss. As $\lambda \to \infty$, the solution goes to $f(\mathbf{x}) = 0$.
(2) In practice, we don't actually invert $(K + \lambda \ell I)$, but instead use an algorithm for solving linear systems.
(3) In order to use this approach, we need to compute and store the entire kernel matrix K. This makes it impractical for use with very large training sets.

Lastly, there is nothing to stop us for using the above algorithm for classification. By doing so, we are essentially treating our classification problem as a regression problem with y values of 1 or -1.

6.2.1. Solving for c

The conjugate gradient (CG) algorithm is a popular algorithm for solving positive definite linear systems. For the purposes of this class, we need to know that CG is an iterative algorithm. The major operation in CG is multiplying a vector v by the matrix A. Note that matrix A need not always be supplied explicitly, we just need some way to form a product Av.

For ordinary positive semidefinite systems, CG will be competitive with direct methods. CG can be much faster if there is a way to multiply by A quickly.

Example. *Suppose our kernel K is linear:*

$$K(x, y) = \langle x, y \rangle.$$

Then our solution x can be written as

$$\begin{aligned} f(x) &= \sum c_i \langle x_i, x \rangle \\ &= \left\langle \left(\sum c_i x_i \right), x \right\rangle \\ &:= \langle w, x \rangle, \end{aligned}$$

and we can apply our function to new examples in time d rather than time nd.

This is a general property of Tikhonov regularization with a linear kernel, not related to the use of the square loss.

We can use the CG algorithm to get a huge savings for solving regularized least-squares regression with a linear kernel ($K(\mathbf{x_1}, \mathbf{x_2}) = \mathbf{x_1} \cdot \mathbf{x_2}$). With an arbitrary kernel, we must form a product Kv explicitly — we multiply a vector by K. With the linear kernel, we note that $K = AA^T$, where A is a matrix with the data points as row vectors. Using this:

$$\begin{aligned} (K + \lambda n I)v &= (AA^T + \lambda n I)v \\ &= A(A^T v) + \lambda n I v. \end{aligned}$$

Suppose we have n points in d dimensions. Forming the kernel matrix K explicitly takes $n^2 d$ time, and multiplying a vector by K takes n^2 time.

If we use the linear representation, we pay nothing to form the kernel matrix, and multiplying a vector by K takes $2dn$ time.

If $d \ll n$, we save approximately a factor of $\frac{n}{2d}$ per iteration. The memory savings are even more important, as we cannot store the kernel matrix at all for large training sets, and if were to recompute the entries of the kernel matrix as needed, each iteration would cost $n^2 d$ time.

Also note that if the training data are sparse (they consist of a large number of dimensions, but the majority of dimensions for each point are zero), the cost of multiplying a vector by K can be written as $2\bar{d}n$, where $\bar{d}$ is the average number of nonzero entries per data point.

This is often the case for applications relating to text, where the dimensions will correspond to the words in a "dictionary". There may be tens of thousands of words, but only a few hundred will appear in any given document.

6.3. Equivalence of the three forms

The three forms of regularization have a certain equivalence. The equivalence is that given a set of points $\{(x_i, y_i)\}_{i=1}^n$ the parameters $\lambda, \tau,$ and κ can be set such that the same function $f(x)$ minimizes (P1), (P2), and (P3). Given this equivalence and and the representer theorem for (P1) it is clear that a representer theorem holds for (P2) and (P3).

Proposition. *Given a convex loss function function the following optimization procedures are equivalent*

$$(P1) \qquad \min_{f \in \mathcal{H}_K} \left[n^{-1} \sum_{i=1}^n V(f, z_i) + \lambda \|f\|_{\mathcal{H}_K}^2 \right],$$

$$(P2) \qquad \min_{f \in \mathcal{H}_K} \left[n^{-1} \sum_{i=1}^n V(f, z_i) \right] \quad \textit{subject to} \quad \|f\|_{\mathcal{H}_K}^2 \leq \tau,$$

$$(P3) \qquad \min_{f \in \mathcal{H}_K} \|f\|_{\mathcal{H}_K}^2 \quad \textit{subject to} \quad \left[n^{-1} \sum_{i=1}^n V(f, z_i) \right] \leq \kappa.$$

By equivalent we mean that if $f_0(x)$ is a solution of one of the problems then there exist parameters τ, κ, λ for which $f_0(x)$ is a of the others.

Proof.

Let f_0 be the solution of (P2) for a fixed τ and assume that the constraint under the optimization is tight ($\|f_0\|_{\mathcal{H}_K}^2 = \tau$). Let $\left[n^{-1} \sum_{i=1}^n V(f_0, z_i) \right] = b$. By inspection the solution of (P3) with $\kappa = b$ will be f_0.

Let f_0 be the solution of (P3) for a fixed κ and assume that the constraint under the optimization is tight ($[n^{-1} \sum_{i=1}^n V(f_0, z_i)] = \kappa$). Let $\|f_0\|_{\mathcal{H}_K}^2 = b$. By inspection the solution of (P2) with $\tau = b$ will be f_0.

For both (P2) and (P3) the above argument can be adjusted for the case where the constraints are not tight but the solution f_0 is not necessarily unique.

Let f_0 be the solution of (P2) for a fixed τ. Using Lagrange multipliers we can rewrite (P2) as

$$\min_{f \in \mathcal{H}_K, \alpha} \left[n^{-1} \sum_{i=1}^n V(f, z_i) \right] + \alpha \left(\|f\|_{\mathcal{H}_K}^2 - \tau \right), \tag{6.2}$$

where $\alpha \geq 0$ the optimal $\alpha = \alpha_0$. By the Karush-Kuhn-Tucker (KKT) conditions (complimentary slackness) at optimality

$$\alpha_0 \left(\|f_0\|_{\mathcal{H}_K}^2 - \tau \right) = 0.$$

If $\alpha_0 = 0$ then $\|f\|^2_{\mathcal{H}_K} < \tau$ and we can rewrite equation (6.2) as

$$\min_{f \in \mathcal{H}_K} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right],$$

which corresponds to (P1) with $\lambda = 0$ and the minima is f_0. If $\alpha_0 > 0$ then $\|f\|^2_{\mathcal{H}_K} = \tau$ and we can rewrite equation (6.2) as the following equivalent optimization procedures

$$(P2) \qquad \min_{f \in \mathcal{H}_K} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right] + \alpha_0 \left(\|f\|^2_{\mathcal{H}_K} - \tau \right),$$

$$(P2) \qquad \min_{f \in \mathcal{H}_K} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right] + \alpha_0 \|f\|^2_{\mathcal{H}_K},$$

which corresponds to (P1) with $\lambda = \alpha_0$ and the minima is f_0.

Let f_0 be the solution of (P3) for a fixed κ. Using Lagrange multipliers we can rewrite (P3) as

$$(6.3) \qquad \min_{f \in \mathcal{H}_K, \alpha} \|f\|^2_{\mathcal{H}_K} + \alpha \left(\left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right] - \kappa \right),$$

where $\alpha \geq 0$ with the optimal $\alpha = \alpha_0$. By the KKT conditions at optimality

$$\alpha_0 \left(\left[n^{-1} \sum_{i=1}^{n} V(f_0, z_i) \right] - \kappa \right) = 0.$$

If $\alpha_0 = 0$ then $\left[n^{-1} \sum_{i=1}^{n} V(f_0, z_i) \right] < \kappa$ and we can rewrite equation (6.3) as

$$\min_{f \in \mathcal{H}_K} \|f\|^2_{\mathcal{H}_K},$$

which corresponds to (P1) with $\lambda = \infty$ and the minima is f_0. If $\alpha_0 > 0$ then $\left[n^{-1} \sum_{i=1}^{n} V(f_0, z_i) \right] = \kappa$ and we can rewrite equation (6.3) as the following equivalent optimization procedures

$$(P3) \qquad \min_{f \in \mathcal{H}_K} \|f\|^2_{\mathcal{H}_K} + \alpha_0 \left(\left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right] - \kappa \right),$$

$$(P3) \qquad \min_{f \in \mathcal{H}_K} \|f\|^2_{\mathcal{H}_K} + \alpha_0 \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right],$$

$$(P3) \qquad \min_{f \in \mathcal{H}_K} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right] + \frac{1}{\alpha_0} \|f\|^2_{\mathcal{H}_K},$$

which corresponds to (P1) with $\lambda = 1/\alpha_0$ and the minima is f_0. □

REVIEW OF CONVEX OPTIMIZATION*

Concepts from convex optimization such as Karush-Kuhn-Tucker (KKT) conditions were used in the previous sections of this lecture. In this section we give a brief introduction and derivation of these conditions.

Definition. *A set $\mathcal{X} \in \mathbb{R}^n$ is convex if*

$$\forall x_1, x_2 \in \mathcal{X},\ \forall \lambda \in [0,1],\ \lambda x_1 + (1-\lambda)x_2 \in \mathcal{X}.$$

A set is convex if, given any two points in the set, the line segment connecting them lies entirely inside the set.

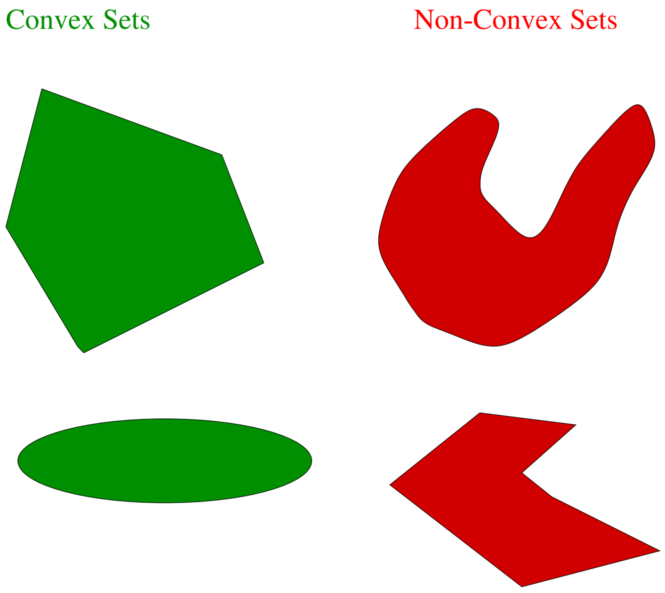


Figure 1. Examples of convex and nonconvex sets in $\mathbb{R}^2$.

Definition. *A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if:*

(1) *For any x_1 and x_2 in the domain of f, for any $\lambda \in [0,1]$,*

$$f(\lambda x_1 + (1-\lambda)x_2) \le \lambda f(x_1) + (1-\lambda) f(x_2).$$

(2) *The line segment connecting two points $f(x_1)$ and $f(x_2)$ lies entirely on or above the function f.*

(3) *The set of points lying on or above the function f is convex.*

A function is strictly convex if we replace "on or above" with "above", or replace "$\leq$" with $<$.

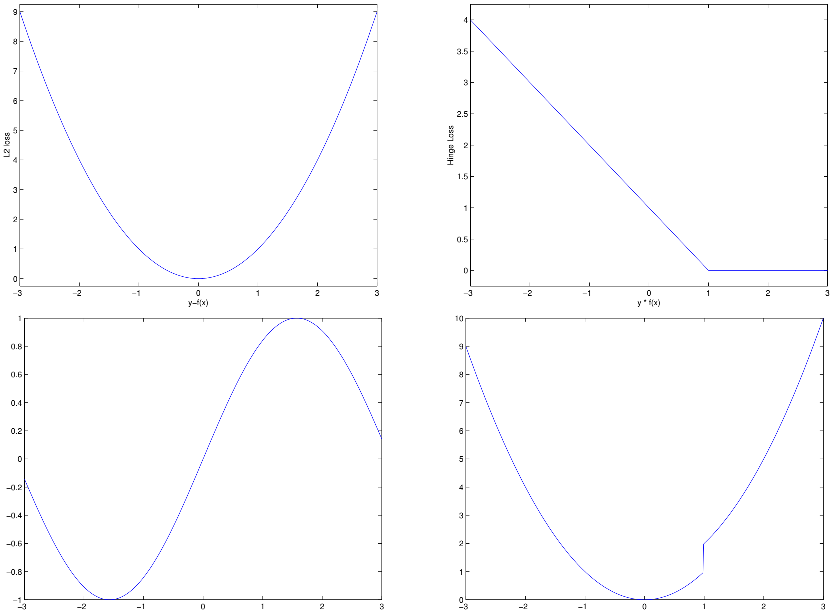


Figure 2. The top two figures are convex functions. The first function is strictly convex. Bottom figures are nonconvex functions.

Definition. *A point $\mathbf{x}^*$ is called a local minimum of f if there exists $\varepsilon > 0$ such that $f(\mathbf{x}^*) < f(\mathbf{x})$ for all $\|\mathbf{x} - \mathbf{x}^*\| \leq \varepsilon$.*

Definition. *A point $\mathbf{x}^*$ is called a global minimum of f if $f(\mathbf{x}^*) < f(\mathbf{x})$ for all feasible $\mathbf{x}$.*

Unconstrained convex functions (convex functions where the domain is all of $\mathbb{R}^n$) are easy to minimize. Convex functions are differentiable almost everywhere. Directional derivatives always exist. If we cannot improve our solution by moving locally, we are at the optimum. If we cannot find a direction that improves our solution, we are at the optimum.

Convex functions over convex sets (a convex domain) are also easy to minimize. If the set and the functions are both convex, if we cannot find a direction which we are able to move in which decreases the function, we are done. Local optima are global optima.

Example. *Linear programming is always a convex problem*

$$\begin{aligned} \min_{c} \quad & \langle c, x \rangle \\ \text{subject to}: \quad & Ax = b \\ & Cx \leq d. \end{aligned}$$

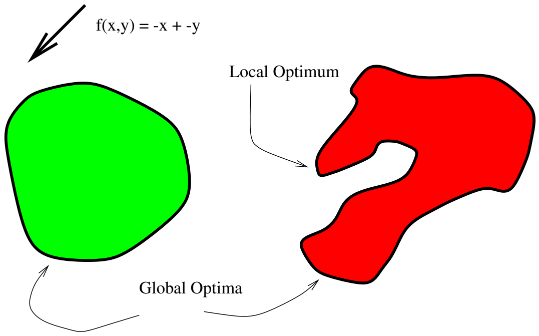


Figure 3. Optimizing a convex function of convex and nonconvex sets. In the example on the left the set is convex and the function is convex so a local minima corresponds to a global minima. In the example on the right the set is nonconvex and the function is convex one can find local minima that are not global minima.

Example. *Quadratic programming is a convex problem iff the matrix Q is positive semidefinite*

$$\begin{aligned} \min \quad & x'Qx + \langle c, x\rangle \\ \text{subject to}: \quad & Ax = b \\ & Cx \leq d. \end{aligned}$$

Definition. *The following constrained optimization problem P will be called the primal problem*

$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ \text{subject to}: \quad & g_i(\mathbf{x}) \geq 0 \quad i = 1, \ldots, m \\ & h_i(\mathbf{x}) = 0 \quad i = 1, \ldots, n \\ & x \in \mathcal{X}. \end{aligned}$$

Here, f is our objective function, the g_i are inequality constraints, the h_i are equality constraints, and $\mathcal{X}$ is some set.

Definition. *We define a Lagrangian dual problem D:*

$$\begin{aligned} \max \quad & \Theta(\mathbf{u}, \mathbf{v}) \\ \text{subject to}: \quad & \mathbf{u} \geq 0 \end{aligned}$$

where $\Theta(\mathbf{u}, \mathbf{v}) := \inf \left\{ f(\mathbf{x}) - \sum_{i=1}^{m} u_i g_i(\mathbf{x}) - \sum_{j=1}^{n} v_i h_i(\mathbf{x}) : x \in \mathcal{X} \right\}$.

Theorem (Weak Duality)**.** *Suppose x is a feasible solution of P. Then $x \in \mathcal{X}, g_i(\mathbf{x}) \leq 0 \;\forall i, h(\mathbf{x}) = 0 \;\forall i$. Suppose $\mathbf{u}, \mathbf{v}$ are a feasible solution of D. Then for all $\mathbf{u} \geq 0$*

$$f(\mathbf{x}) \geq \Theta(\mathbf{u}, \mathbf{v}).$$

Proof.

$$\begin{aligned}
\Theta(\mathbf{u},\mathbf{v}) &= \inf\left\{ f(\mathbf{y}) - \sum_{i=1}^{m} u_i g_i(\mathbf{y}) - \sum_{j=1}^{n} v_i h_i(\mathbf{y}) : y \in \mathcal{X} \right\} \\
&\leq f(\mathbf{x}) - \sum_{i=1}^{m} u_i g_i(\mathbf{x}) - \sum_{i=1}^{n} v_i h_i(\mathbf{x}) \\
&\leq f(\mathbf{x}).
\end{aligned}$$

Weak duality says that every feasible solution to P is at least as expensive as every feasible solution to D. It is a very general property of duality, and we did not rely on any convexity assumptions to show it.

Definition. *Strong duality holds when the optima of the primal and dual problems are equivalent $Opt(P) = Opt(D)$.*

If strong duality, does not hold, we have the possibility of a duality gap. Strong duality is very useful, because it usually means that we may solve whichever of the dual or primal is more convenient computationally, and we can usually obtain the solution of one from the solution of the other.

Proposition. *If the objective function f is convex, and the feasible region is convex, under mild technical we have strong duality.*

We now look at a what are called saddle points of the Lagrangian function. We defined the Lagrangian function as the dual problem

$$L(\mathbf{x},\mathbf{u},\mathbf{v}) = f(\mathbf{x}) - \sum_{i=1}^{m} \mathbf{u}_i \mathbf{g}_i(\mathbf{x}) - \sum_{j=1}^{n} \mathbf{v}_j \mathbf{h}_j(\mathbf{x}).$$

A set $(\mathbf{x}^*, \mathbf{u}^*, \mathbf{v}^*)$ of feasible solutions to P and D is called a saddle point of the Lagrangian if

$$L(\mathbf{x}^*,\mathbf{u},\mathbf{v}) \leq L(\mathbf{x},\mathbf{u},\mathbf{v}) \leq L(\mathbf{x},\mathbf{u}^*,\mathbf{v}^*) \; \forall x \in \mathcal{X}, \quad \forall \mathbf{u} \geq 0$$

$\mathbf{x}^*$ minimizes L if $\mathbf{u}$ and $\mathbf{v}$ are fixed at $\mathbf{u}^*$ and $\mathbf{v}^*$, and $\mathbf{u}^*$ and $\mathbf{v}^*$ maximize L if $\mathbf{x}$ is fixed at $\mathbf{x}^*$.

Definition. *The points $(\mathbf{x}^*, \mathbf{u}^*, \mathbf{v}^*)$ satisfy the Karush Kuhn Tucker (KKT) conditions or are KKT points if they are feasible to P and D and*

$$\begin{aligned}
\nabla f(\mathbf{x}^*) - \nabla \mathbf{g}(\mathbf{x}^*)'\mathbf{u}^* - \nabla \mathbf{h}(\mathbf{x}^*)'\mathbf{v} &= 0 \\
\mathbf{u}^* \mathbf{g}(\mathbf{x}^*) &= 0.
\end{aligned}$$

In a convex, differentiable problem, with some minor technical conditions, points that satisfy the KKT conditions are equivalent to saddle points of the Lagrangian.

LECTURE 7
Support vector machines

SVMs have been used in a multitude of applications and are one of the most popular machine learning algorithms. We will derive the SVM algorithm from two perspectives: Tikhonov regularization, and the more common geometric perspective.

7.1. SVMs from Tikhonov regularization

We start with Tikhonov regularization

$$\min_{f \in \mathcal{H}} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) + \lambda \Omega(f) \right]$$

and then set the regularization functional to a RKHS norm

$$\Omega(f) = \|f\|^2_{\mathcal{H}_K}$$

and use the hinge loss functional

$$n^{-1} \sum_{i=1}^{n} V(f, z_i) := n^{-1} \sum_{i=1}^{n} (1 - y_i f(x_i))_+,$$

where $(k)_+ := \max(k, 0)$.

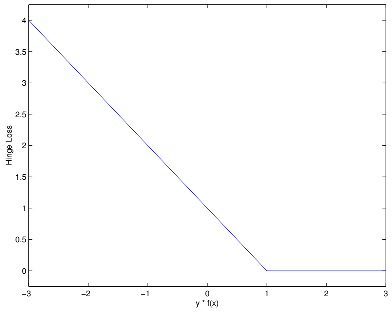


Figure 1. Hinge loss.

The resulting optimization problem is

$$\min_{f\in\mathcal{H}_K}\left[n^{-1}\sum_{i=1}^{n}(1-y_if(x_i))_+ + \lambda\|f\|^2_{\mathcal{H}_K}\right], \tag{7.1}$$

which is non-differentiable at $(1-y_if(x_i))=0$ so we introduce slack variables and write the following constrained optimization problem:

$$\begin{array}{lll} \displaystyle\min_{f\in\mathcal{H}_K} & n^{-1}\sum_{i=1}^n \xi_i + \lambda\|f\|_K^2 & \\ \text{subject to}: & y_if(x_i)\geq 1-\xi_i & i=1,\dots,n \\ & \xi_i \geq 0 & i=1,\dots,n. \end{array}$$

By the Representer theorem we can rewrite the above constrained optimization problem as a constrained quadratic programming problem

$$\begin{array}{lll} \displaystyle\min_{c\in\mathbb{R}^n} & n^{-1}\sum_{i=1}^n \xi_i + \lambda c^TKc & \\ \text{subject to}: & y_i\sum_{j=1}^n c_jK(x_i,x_j)\geq 1-\xi_i & i=1,\dots,n \\ & \xi_i \geq 0 & i=1,\dots,n. \end{array}$$

The SVM contains an unregularized bias term b so the Representer theorem results in a function

$$f(x)=\sum_{i=1}^n c_iK(x,x_i)+b.$$

Plugging this form into the above constrained quadratic problem results in the "primal" SVM

$$\begin{array}{lll} \displaystyle\min_{\mathbf{c}\in\mathbb{R}^n,\xi\in\mathbb{R}^n} & n^{-1}\sum_{i=1}^n \xi_i + \lambda \mathbf{c}^TK\mathbf{c} & \\ \text{subject to}: & y_i\left(\sum_{j=1}^\ell c_jK(x_i,x_j)+b\right)\geq 1-\xi_i & i=1,\dots,n \\ & \xi_i \geq 0 & i=1,\dots,n. \end{array}$$

We now derive the Wolfe dual quadratic program using Lagrange multiplier techniques:

$$\begin{aligned} L(\mathbf{c},\xi,b,\alpha,\zeta) &= n^{-1}\sum_{i=1}^n\xi_i+\lambda\mathbf{c}^TK\mathbf{c} \\ &\quad -\sum_{i=1}^n\alpha_i\left(y_i\left\{\sum_{j=1}^n c_jK(x_i,x_j)+b\right\}-1+\xi_i\right) \\ &\quad -\sum_{i=1}^n\zeta_i\xi_i. \end{aligned}$$

We want to minimize L with respect to $\mathbf{c}$, b, and ξ, and maximize L with respect to α and ζ, subject to the constraints of the primal problem and nonnegativity constraints on α and ζ. We first eliminate b and ξ by taking partial derivatives:

$$\begin{aligned} \frac{\partial L}{\partial b}=0 &\implies \sum_{i=1}^n\alpha_iy_i=0 \\ \frac{\partial L}{\partial \xi_i}=0 &\implies \frac{1}{n}-\alpha_i-\zeta_i=0\implies 0\leq\alpha_i\leq\frac{1}{n}. \end{aligned}$$

The above two conditions will be constraints that will have to be satisfied at optimality. This results in a reduced Lagrangian:

$$L^R(\mathbf{c}, \alpha) = \lambda \mathbf{c}^T K \mathbf{c} - \sum_{i=1}^n \alpha_i \left(y_i \sum_{j=1}^n c_j K(x_i, x_j) - 1 \right).$$

We now eliminate $\mathbf{c}$

$$\frac{\partial L^R}{\partial \mathbf{c}} = 0 \implies 2\lambda K\mathbf{c} - KY\alpha = 0 \Longrightarrow c_i = \frac{\alpha_i y_i}{2\lambda},$$

where Y is a diagonal matrix whose i'th diagonal element is y_i; $Y\alpha$ is a vector whose ith element is $\alpha_i y_i$. Substituting in our expression for $\mathbf{c}$, we are left with the following "dual" program:

$$\begin{array}{rll} \max\limits_{\alpha \in \mathrm{I\!R}^\mathbf{n}} & \sum_{i=1}^n \alpha_i - \frac{1}{4\lambda}\alpha^T Q \alpha & \\ \text{subject to :} & \sum_{i=1}^n y_i \alpha_i = 0 & \\ & 0 \leq \alpha_i \leq \frac{1}{n} & i = 1, \ldots, n, \end{array}$$

where Q is the matrix defined by

$$Q = \mathbf{y}K\mathbf{y^T} \Longleftrightarrow Q_{ij} = y_i y_j K(x_i, x_j).$$

In most of the SVM literature, instead of the regularization parameter λ, regularization is controlled via a parameter C, defined using the relationship

$$C = \frac{1}{2\lambda n}.$$

Using this definition (after multiplying our objective function by the constant $\frac{1}{2n}$, the basic regularization problem becomes

$$\min_{f \in \mathcal{H}_K} C \sum_{i=1}^n V(y_i, f(\mathbf{x}_i)) + \frac{1}{2}||f||_K^2.$$

Like λ, the parameter C also controls the trade-off between classification accuracy and the norm of the function. The primal and dual problems become respectively:

$$\begin{array}{rll} \min\limits_{\mathbf{c} \in \mathrm{I\!R}^n, \xi \in \mathrm{I\!R}^n} & C\sum_{i=1}^n \xi_i + \frac{1}{2}\mathbf{c}^T K \mathbf{c} & \\ \text{subject to :} & y_i \left(\sum_{j=1}^n c_j K(x_i, x_j) + b \right) \geq 1 - \xi_i & i = 1, \ldots, n \\ & \xi_i \geq 0 & i = 1, \ldots, n \end{array}$$

$$\begin{array}{rll} \max\limits_{\alpha \in \mathrm{I\!R}^\mathbf{n}} & \sum_{i=1}^n \alpha_i - \frac{1}{2}\alpha^T Q \alpha & \\ \text{subject to :} & \sum_{i=1}^n y_i \alpha_i = 0 & \\ & 0 \leq \alpha_i \leq C & i = 1, \ldots, n. \end{array}$$

7.2. SVMs from a geometric perspective

The "traditional" approach to developing the mathematics of SVM is to start with the concepts of separating hyperplanes and margin. The theory is usually developed in a linear space, beginning with the idea of a perceptron, a linear hyperplane that separates the positive and the negative examples. Defining the margin as the distance from the hyperplane to the nearest example, the basic observation is that intuitively, we expect a hyperplane with larger margin to generalize better than one with smaller margin.

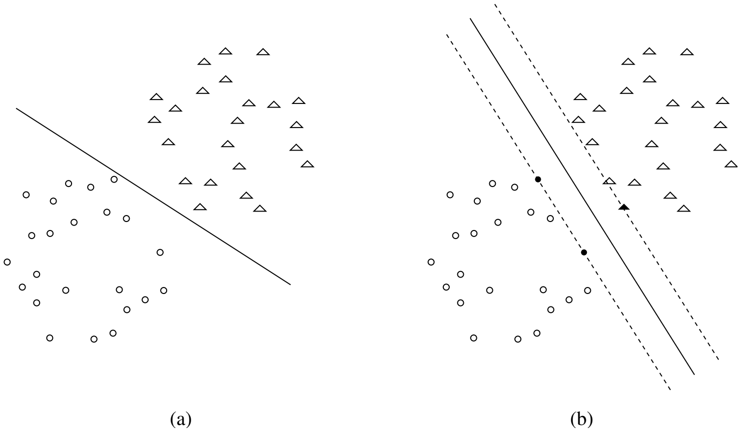


Figure 2. Two hyperplanes (a) and (b) perfectly separate the data. However, hyperplane (b) has a larger margin and intuitively would be expected to be more accurate on new observations.

We denote our hyperplane by $\mathbf{w}$, and we will classify a new point $\mathbf{x}$ via the function

$$f(x) = \text{sign}\left[\langle \mathbf{w}, \mathbf{x} \rangle\right]. \tag{7.2}$$

Given a separating hyperplane $\mathbf{w}$ we let $\mathbf{x}$ be a datapoint closest to $\mathbf{w}$, and we let $\mathbf{x}^{\mathbf{w}}$ be the unique point on $\mathbf{w}$ that is closest to x. Obviously, finding a maximum margin $\mathbf{w}$ is equivalent to maximizing $||\mathbf{x} - \mathbf{x}^{\mathbf{w}}||$. So for some k (assume $k > 0$ for convenience),

$$\begin{gathered}
\langle \mathbf{w}, \mathbf{x} \rangle = k \\
\langle \mathbf{w}, \mathbf{x}^{\mathbf{w}} \rangle = 0 \\
\langle \mathbf{w}, (\mathbf{x} - \mathbf{x}^{\mathbf{w}}) \rangle = k.
\end{gathered}$$

Noting that the vector $\mathbf{x} - \mathbf{x}^{\mathbf{w}}$ is parallel to the normal vector w,

$$\begin{aligned}
\langle \mathbf{w}, (\mathbf{x} - \mathbf{x}^{\mathbf{w}}) \rangle &= \left\langle \mathbf{w}, \left(\frac{||\mathbf{x} - \mathbf{x}^{\mathbf{w}}||}{||\mathbf{w}||} \mathbf{w} \right) \right\rangle \\
&= ||\mathbf{w}||^2 \frac{||\mathbf{x} - \mathbf{x}^{\mathbf{w}}||}{||\mathbf{w}||} \\
&= ||\mathbf{w}|| \; ||\mathbf{x} - \mathbf{x}^{\mathbf{w}}|| \\
k &= ||\mathbf{w}|| \; ||(\mathbf{x} - \mathbf{x}^{\mathbf{w}})|| \\
\frac{k}{||\mathbf{w}||} &= ||\mathbf{x} - \mathbf{x}^{\mathbf{w}}||.
\end{aligned}$$

k is a "nuisance parameter" and without any loss of generality, we fix k to 1, and see that maximizing $||\mathbf{x} - \mathbf{x}^{\mathbf{w}}||$ is equivalent to maximizing $\frac{1}{||w||}$, which in turn is equivalent to minimizing $||w||$ or $||w||^2$. We can now define the margin as the distance between the hyperplanes $\langle \mathbf{w}, \mathbf{x} \rangle = 0$ and $\langle \mathbf{w}, \mathbf{x} \rangle = 1$.

So if the data is linear separable case and the hyperplanes run through the origin the maximum margin hyperplane is the one for which

$$\begin{array}{rl} \min\limits_{\mathbf{w}\in\mathbb{R}^n} & ||\mathbf{w}||^2 \\ \text{subject to}: & y_i\langle \mathbf{w}, \mathbf{x}_i\rangle \geq 1 \quad i = 1, \ldots, n. \end{array}$$

The SVM introduced by Vapnik includes an unregularized bias term b, leading to classification via a function of the form:

$$f(x) = \text{sign}[\langle \mathbf{w}, \mathbf{x}\rangle + b].$$

In addition, we need to work with datasets that are not linearly separable, so we introduce slack variables ξ_i, just as before. We can still define the margin as the distance between the hyperplanes $\langle \mathbf{w}, \mathbf{x}\rangle = 0$ and $\langle \mathbf{w}, \mathbf{x}\rangle = 1$, but the geometric intuition is no longer as clear or compelling.

With the bias term and slack variables the primal SVM problem becomes

$$\begin{array}{rll} \min\limits_{\mathbf{w}\in\mathbb{R}^n, b\in\mathbb{R}} & C\sum_{i=1}^n \xi_i + \frac{1}{2}||\mathbf{w}||^2 & \\ \text{subject to}: & y_i\left(\langle \mathbf{w}, \mathbf{x}\rangle + b\right) \geq 1 - \xi_i & i = 1, \ldots, n \\ & \xi_i \geq 0 & i = 1, \ldots, n. \end{array}$$

Using Lagrange multipliers we can derive the same dual from in the previous section.

Historically, most developments begin with the geometric form, derived a dual program which was identical to the dual we derived above, and only then observed that the dual program required only dot products and that these dot products could be replaced with a kernel function. In the linearly separable case, we can also derive the separating hyperplane as a vector parallel to the vector connecting the closest two points in the positive and negative classes, passing through the perpendicular bisector of this vector. This was the "Method of Portraits", derived by Vapnik in the 1970's, and recently rediscovered (with non-separable extensions) by Keerthi.

7.3. Optimality conditions

The primal and the dual are both feasible convex quadratic programs. Therefore, they both have optimal solutions, and optimal solutions to the primal and the dual have the same objective value.

We derived the dual from the primal using the (now reparameterized) Lagrangian:

$$\begin{aligned} L(\mathbf{c}, \xi, b, \alpha, \zeta) &= C\sum_{i=1}^n \xi_i + \mathbf{c}^T K\mathbf{c} \\ &\quad - \sum_{i=1}^n \alpha_i \left(y_i \left\{ \sum_{j=1}^n c_j K(x_i, x_j) + b \right\} - 1 + \xi_i \right) \\ &\quad - \sum_{i=1}^n \zeta_i \xi_i. \end{aligned}$$

We now consider the dual variables associated with the primal constraints:

$$\begin{aligned} \alpha_i &\implies y_i \left\{ \sum_{j=1}^n c_j K(x_i, x_j) + b \right\} - 1 + \xi_i \\ \zeta_i &\implies \xi_i \geq 0. \end{aligned}$$

Complementary slackness tells us that at optimality, either the primal inequality is satisfied at equality or the dual variable is zero. In other words, if $\mathbf{c}$, ξ, b, α and ζ are optimal solutions to the primal and dual, then

$$\begin{aligned} \alpha_i \left(y_i \left\{ \sum_{j=1}^n c_j K(x_i, x_j) + b \right\} - 1 + \xi_i \right) &= 0 \\ \zeta_i \xi_i &= 0 \end{aligned}$$

All optimal solutions must satisfy:

$$\begin{aligned} \sum_{j=1}^n c_j K(x_i, x_j) - \sum_{j=1}^n y_i \alpha_j K(x_i, x_j) = 0 & \qquad i = 1, \ldots, n \\ \sum_{i=1}^n \alpha_i y_i = 0 & \\ C - \alpha_i - \zeta_i = 0 & \qquad i = 1, \ldots, n \\ y_i \left(\sum_{j=1}^n y_j \alpha_j K(x_i, x_j) + b \right) - 1 + \xi_i \geq 0 & \qquad i = 1, \ldots, n \\ \alpha_i \left[y_i \left(\sum_{j=1}^n y_j \alpha_j K(x_i, x_j) + b \right) - 1 + \xi_i \right] = 0 & \qquad i = 1, \ldots, n \\ \zeta_i \xi_i = 0 & \qquad i = 1, \ldots, n \\ \xi_i, \alpha_i, \zeta_i \geq 0 & \qquad i = 1, \ldots, n \end{aligned}$$

The above optimality conditions are both necessary and sufficient. If we have $\mathbf{c}$, ξ, b, α and ζ satisfying the above conditions, we know that they represent optimal solutions to the primal and dual problems. These optimality conditions are also known as the Karush-Kuhn-Tucker (KKT) conditions.

Suppose we have the optimal α_i's. Also suppose (this "always" happens in practice") that there exists an i satisfying $0 < \alpha_i < C$. Then

$$\begin{aligned} \alpha_i < C &\implies \zeta_i > 0 \\ &\implies \xi_i = 0 \\ &\implies y_i \left(\sum_{j=1}^n y_j \alpha_j K(x_i, x_j) + b \right) - 1 = 0 \\ &\implies b = y_i - \sum_{j=1}^n y_j \alpha_j K(x_i, x_j) \end{aligned}$$

So if we know the optimal α's, we can determine b.

Defining our classification function $f(x)$ as

$$f(x) = \sum_{i=1}^{\ell} y_i \alpha_i K(x, x_i) + b,$$

we can derive "reduced" optimality conditions. For example, consider an i such that $y_i f(x_i) < 1$:

$$\begin{aligned} y_i f(x_i) < 1 \quad &\Longrightarrow \quad \xi_i > 0 \\ &\Longrightarrow \quad \zeta_i = 0 \\ &\Longrightarrow \quad \alpha_i = C. \end{aligned}$$

Conversely, suppose $\alpha_i = C$:

$$\begin{aligned} \alpha_i = C \quad &\Longrightarrow \quad y_i f(x_i) - 1 + \xi_i = 0 \\ &\Longrightarrow \quad y_i f(x_i) \leq 1. \end{aligned}$$

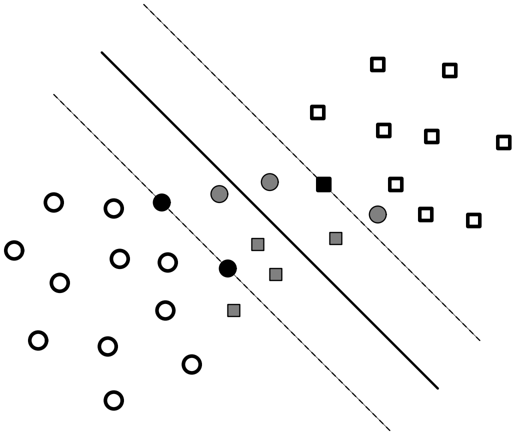


Figure 3. A geometric interpretation of the reduced optimality conditions. The open squares and circles correspond to cases where $\alpha_i = 0$. The dark circles and squares correspond to cases where $y_i f(x_i) = 1$ and $\alpha_i \leq C$, these are samples at the margin. The grey circles and squares correspond to cases where $y_i f(x_i) < 1$ and $\alpha_i = C$.

7.4. Solving the SVM optimization problem

Our plan will be to solve the dual problem to find the α's, and use that to find b and our function f. The dual problem is easier to solve the primal problem. It has simple box constraints and a single inequality constraint, even better, we will see that the problem can be decomposed into a sequence of smaller problems.

We can solve QPs using standard software. Many codes are available. Main problem — the Q matrix is dense, and is $n \times n$, so we cannot write it down. Standard QP software requires the Q matrix, so is not suitable for large problems.

To get around this memory issue we partition the dataset into a working set W and the remaining points R. We can rewrite the dual problem as:

$$\begin{array}{cc}
\max\limits_{\alpha_W \in \mathbb{R}^{|W|},\ \alpha_R \in \mathbb{R}^{|R|}} & \sum_{\substack{i=1\\ i\in W}}^{n} \alpha_i + \sum_{\substack{i=1\\ i\in R}} \alpha_i \\
 & -\frac{1}{2}[\alpha_{\mathbf{W}}\ \alpha_{\mathbf{R}}] \left[\begin{array}{cc} Q_{WW} & Q_{WR} \\ Q_{RW} & Q_{RR} \end{array} \right] \left[\begin{array}{c} \alpha_{\mathbf{W}} \\ \alpha_{\mathbf{R}} \end{array} \right] \\
\text{subject to :} & \sum_{i\in W} y_i\alpha_i + \sum_{i\in R} y_i\alpha_i = 0 \\
 & 0 \le \alpha_i \le C,\ \forall i.
\end{array}$$

Suppose we have a feasible solution α. We can get a better solution by treating the $\alpha_{\mathbf{W}}$ as variable and the $\alpha_{\mathbf{R}}$ as constant. We can solve the reduced dual problem:

$$\begin{array}{cc}
\max\limits_{\alpha_W \in \mathbb{R}^{|W|}} & (\mathbf{1} - Q_{WR}\alpha_{\mathbf{R}})\alpha_W - \frac{1}{2}\alpha_{\mathbf{W}} Q_{WW} \alpha_{\mathbf{W}} \\
\text{subject to:} & \sum_{i\in W} y_i\alpha_i = -\sum_{i\in R} y_i\alpha_i \\
 & 0 \le \alpha_i \le C,\ \forall i \in W.
\end{array}$$

The reduced problems are fixed size, and can be solved using a standard QP code. Convergence proofs are difficult, but this approach seems to always converge to an optimal solution in practice.

An important issue in the decomposition is selecting the working set. There are many different approaches. The basic idea is to examine points not in the working set, find points which violate the reduced optimality conditions, and add them to the working set. Remove points which are in the working set but are far from violating the optimality conditions.

LECTURE 8
Regularized logistics regression

One drawback with the SVM is that the method does not explicitly output a probability or likelihood of the labels, instead the output is a real value the magnitude of which should be monotonic with respect to the probability

$$P(y = \pm 1 | x) \propto y\, f(x).$$

This issue can be addressed by using a loss function based upon logistic or binary regression. The main idea behind logistic regression is that we are trying to model the log likelihood ratio by the function $f(x)$

$$f(x) = \log \left(\frac{P(y = 1|x)}{P(y = -1|x)} \right).$$

Since $P(y = 1|x)$ is a Bernoulli random variable we can rewrite the above equation as

$$\begin{aligned} f(x) &= \log \left(\frac{P(y = 1|x)}{P(y = -1|x)} \right) \\ &= \log \left(\frac{P(y = 1|x)}{1 - P(y = 1|x)} \right) \end{aligned}$$

which implies

$$\begin{aligned} P(y = 1|x) &= \frac{1}{1 + \exp(f(x))} \\ P(y = -1|x) &= \frac{1}{1 + \exp(-f(x))} \\ P(y = \pm 1|x) &= \frac{1}{1 + \exp(y\, f(x))}. \end{aligned}$$

Given a data set $D = \{(x_i, y_i)\}_{i=1}^n$ and a class of functions $f \in \mathcal{H}$ the maximum likelihood estimator (MLE) is the function that maximizes the likelihood of observing the data set D

$$f^*_{\text{MLE}} := \arg \max_{f \in \mathcal{H}} \left[P(D|f) \right] = \arg \max_{f \in \mathcal{H}} \left[\prod_{i=1}^n \frac{1}{1 + \exp(y_i\, f(x_i))} \right].$$

As in the case of Empirical risk minimization the MLE estimate may overfit the data since there is no smoothness or regularization term. A classical way of imposing smoothness in this context is by placing a prior on the functions $f \in \mathcal{H}$

$$P(f) \propto e^{-\|f\|^2_{\mathcal{H}_K}}.$$

Given a prior and a likelihood we can use Bayes rule to compute the posterior distribution $P(f|D)$

$$P(f|D) = \frac{P(D|f)\,P(f)}{P(D)}.$$

If we plug the prior and likelihood into Bayes rule we can compute the maximum a posteriori (MAP) estimator

$$\begin{aligned} f^*_{\text{MAP}} &:= \arg\max_{f\in\mathcal{H}} \left[\frac{P(D|f)P(f)}{P(D)}\right] \\ &= \arg\max_{f\in\mathcal{H}} \left[\frac{\prod_{i=1}^n \frac{1}{1+\exp(y_i\, f(x_i))} \; e^{-\|f\|^2_{\mathcal{H}_K}}}{P(D)}\right] \\ &= \arg\max_{f\in\mathcal{H}} \left[\sum_{i=1}^n \log\left(\frac{1}{1+\exp(y_i\, f(x_i))}\right) - \|f\|^2_{\mathcal{H}_K}\right]. \end{aligned}$$

With some simple algebra the above MAP estimator can be rewritten in the form of Tikhonov regularization

$$f^*_{\text{MAP}} = \arg\min_{f\in\mathcal{H}_K} \left[n^{-1}\sum_{i=1}^n \log(1+\exp(-y_i\, f(x_i))) + \lambda\|f\|^2_{\mathcal{H}_K}\right],$$

where λ is the regularization parameter. By the representer theorem the above equation has a solution of the form

$$f^*(x) = \sum_{i=1}^n c_i K(x, x_i).$$

Given the above representer theorem we can solve for the variables c_i by the following optimization problem

$$\min_{\mathbf{c}\in\mathbb{R}^n} \left[n^{-1}\sum_{i=1}^n \log(1+\exp(-y_i\,(\mathbf{c}^T\mathbf{K})_i)) + \lambda\mathbf{c}^T\mathbf{K}\mathbf{c}\right],$$

where $(\mathbf{c}^T\mathbf{K})_i$ is the ith element of the vector $\mathbf{c}^T\mathbf{K}$. This optimization problem is convex and differentiable so a classical approach for solving for $\mathbf{c}$ is using the Newton-Raphson method.

8.1. Newton-Raphson

The Newton-Raphson method was initially used to solve for roots of polynomials and the application to optimization problems is fairly starightforward. We first describe the Newton-Raphson method for the case of a scalar, the optimization is in terms of one variable. We then describe the multivariate form and apply this to the optimization problem in logistic regression.

- Newton's method for finding roots: Newton's method is primarily a method for finding roots of polynomials. It was proposed by Newton around 1669 and Raphson improved on the method in 1690, therefore the Newton-Raphson method. Given a polynomial $f(x)$ the Taylor series expansion of $f(x)$ around the point $x = x_0 + \varepsilon$ is given by

$$f(x_0 + \varepsilon) = f(x_0) + f'(x_0)\varepsilon + \frac{1}{2}f''(x_0)\varepsilon^2 + ...$$

truncating the expansion after first order terms results in

$$f(x_0 + \varepsilon) \approx f(x_0) + f'(x_0)\varepsilon.$$

From the above expression we can estimate the offset ε needed to get closer to the root $(x : f(x) = 0)$ starting from the intial guess x_0. This is done by setting $f(x_0 + \varepsilon) = 0$ and solving for ε.

$$\begin{aligned} 0 &= f(x_0 + \varepsilon) \\ 0 &\approx f(x_0) + f'(x_0)\varepsilon \\ -f(x_0) &\approx f'(x_0)\varepsilon \\ \epsilon_0 &\approx -\frac{f(x_0)}{f'(x_0)}. \end{aligned}$$

This is the first order or linear adjustment to the root's position. This can be turned into an iterative procedure by setting $x_1 = x_0 + \varepsilon_0$, calculating a new ε_1 and then iterating until converegence:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

- The Newton-Raphson method as an optimization method for scalars: We are given a convex minimization problem

$$\min_{x \in [a,b]} g(x),$$

where $g(x)$ is a convex function. The extrema of $g(x)$ will occur at a value of x_m such that $g'(x_m) = 0$ and since the function is convex this extrema will be a minima. If $g(x)$ is a polynomial then $g'(x)$ is also a poynomial and we can apply Newton's method for root finding to $g'(x)$. If $g(x)$ is not a polynomial then we apply the root finding method to a polynomial approximation of $g(x)$. We now describe the steps involved.

(1) Taylor expand $g(x)$: A truncated Taylor expansion of $g(x)$ results in a second order polynomial approximation of $g(x)$

$$g(x) \approx g(x_0) + g'(x_0)(x - x_0) + \frac{1}{2}(x - x_0)^2 g''(x_0).$$

(2) Set derivative to zero: Take the derivative of the Taylor expansion and set it equal to zero

$$\frac{dg}{dx} = f(x) = g'(x_0) + g''(x_0)(x - x_0) = 0.$$

This leaves us with with a root finding problem, find the root of $f(x)$ for which we we use Newton's method for finding roots.

(3) Update rule: The update rule reduces to

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} = x_n - \frac{g'(x_n)}{g''(x_n)}.$$

A key point in the above procedure is the convexity of $g(x)$. To be sure that the procedure converges the second derivative $g''(x)$ must be positive in the domain of optimization, the interval $[a, b]$. Convexity of $g(x)$ ensures this.

- The Newton-Raphson method as an optimization method for vectors: We are given a convex minimization problem

$$\min_{\mathbf{x} \in \mathcal{X}} g(\mathbf{x}),$$

where $\mathcal{X} \subseteq \mathbb{R}^n$ is convex and $g(\mathbf{x})$ is a convex function. We follow the logic of the scalar case except using vector calculus.

(1) Taylor expand $g(\mathbf{x})$: A truncated Taylor expansion of $g(\mathbf{x})$ results in a second order polynomial approximation of $g(\mathbf{x})$

$$g(\mathbf{x}) \approx g(\mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0)^T \cdot \nabla g(\mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T \cdot \mathbf{H}(x_0) \cdot (\mathbf{x} - \mathbf{x}_0),$$

where $\mathbf{x}$ is a column vector of length n, $\nabla g(\mathbf{x}_0)$ is the gradient of g evaluated at $\mathbf{x}_0$ and is also a column vector of length n, $\mathbf{H}(x_0)$ is the Hessian matrix evaluated at $\mathbf{x}_0$

$$\mathbf{H}_{i,j}(x_0) = \frac{\partial^2 g(\mathbf{x})}{\partial \mathbf{x}^i \partial \mathbf{x}^j}\bigg|_{\mathbf{x}_0}, \quad i, j = 1, ..., n.$$

(2) Set derivative to zero: Take the derivative of the Taylor expansion and set it equal to zero

$$\nabla g(\mathbf{x}) = \nabla g(\mathbf{x}_0) + \frac{1}{2}\mathbf{H}(\mathbf{x}_0) \cdot (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T \cdot \mathbf{H}(\mathbf{x}_0) = 0,$$

the Hessian matrix is symmetric and twice differentiable (due to convexity) so we can reduce the above to

$$\nabla g(\mathbf{x}) = \nabla g(\mathbf{x}_0) + \mathbf{H}(\mathbf{x}_0) \cdot (\mathbf{x} - \mathbf{x}_0) = 0.$$

This implies that at a minima $\mathbf{x}^*$, the gradient is zero

$$0 = \mathbf{H}(\mathbf{x}_0) \cdot (\mathbf{x}^* - \mathbf{x}_0) + \nabla g(\mathbf{x}_0).$$

(3) Update rule: Solving the above linear system of equations for $\mathbf{x}^*$ leads to the following update rule

$$\mathbf{x}_{n+1} = \mathbf{x}_n - \mathbf{H}^{-1}(\mathbf{x}_n) \cdot \nabla g(\mathbf{x}_n),$$

where $-\mathbf{H}^{-1}(\mathbf{x}_n) \cdot \nabla g(\mathbf{x}_n)$ is called the Newton direction.

For the above procedure to converge to a minima the Newton direction must be a direction of descent

$$\nabla g^T(\mathbf{x}_n) \cdot (\mathbf{x}_{n+1} - \mathbf{x}_n) < 0.$$

If the Hessian matrix is positive definite then the Newton direction will be a direction of descent, this is the matrix analog of a positive second derivative. Convexity of $g(\mathbf{x})$ in the domain $\mathcal{X}$ ensures that the Hessian is positvie definite. If the function $g(\mathbf{x})$ is quadratic the procedure will converge in one iteration.

- The Newton-Raphson method for regularized logistic regression: The optimization problem for regularized logistic regression is

$$f^*_{\text{MAP}} = \arg \min_{f \in \mathcal{H}_K} \left[n^{-1} \sum_{i=1}^{n} \log(1 + \exp(-y_i\, f(x_i))) + \lambda \|f\|^2_{\mathcal{H}_K} \right],$$

by the representer theorem

$$f^*(x) = \sum_{i=1}^{n} c_i K(x, x_i) + b,$$

$\|f\|_{\mathcal{H}_K}$ is a seminorm that does not penalize constants, like the SVM case. The optimization problem can be rewritten as

$$\min_{\mathbf{c} \in \mathbb{R}^n, b \in \mathbb{R}} \left[L[\mathbf{c}, b] = n^{-1} \sum_{i=1}^{n} \log(1 + \exp(-y_i\, ((\mathbf{c}^T \mathbf{K})_i + b))) + \lambda \mathbf{c}^T \mathbf{K} \mathbf{c} \right],$$

where $(\mathbf{c}^T\mathbf{K})_i$ is the ith element of the vector $\mathbf{c}^T\mathbf{K}$.

LECTURE 9
Gaussian process regression

The idea behind a Gaussian process regression is to place a distribution over a space of functions say $\mathcal{H}$. Consider for example an rkhs $\mathcal{H}_K$ over which we want to do Bayesian inference. Assume a regression model with the standard noise assumption

$$Y_i = f(X_i) + \varepsilon_i, \quad \varepsilon_i \overset{iid}{\sim} \mathrm{N}(0, \sigma^2), \quad f \in \mathcal{H}_K.$$

If we knew how to place a prior over the function space we in theory could do Bayesian inference.

9.1. Gaussian process

A Gaussian process is a specification of probability distributions over functions $f(x)$, $f \in \mathcal{H}$ and $x \in \mathcal{X}$ parameterized ny a mean function μ and a covariance function $K(\cdot, \cdot)$. The idea can be informally stated as

$$p(f) \propto \exp\Big(-\frac{1}{2}\|f\|^2_{\mathcal{H}_K}\Big), \quad p(f) \geq 0 \,\forall\, f \in \mathcal{H}, \int_{f \in \mathcal{H}} p(f)\,\mathrm{d}f = 1,$$

where we use the term informal because $\mathrm{d}f$ is not well defined, it is not clear what the normalization constant is for $p(f)$ and what the space of functions $\mathcal{H}$ is not clear not is the relation of $\mathcal{H}$ to $\mathcal{H}_K$ stated clearly. Instead of making all the points clear we will develop Gaussian processes from an alternative perspective. There are many ways to define and think about a Gaussian process. A standard formulation is that a Gaussian process is an infinite version of a multivariate Gaussian distribution and has two parameters: a mean function μ corresponding to the mean vector and a positive definite covariance or kernel function K corresponding to a positive definite covariance matrix.

A common approach in defining an infinite dimensional object is by defining it's finite dimensional projections. This is the approach we will take with a Gaussian process. Consider $x_1, ..., x_n$ as a finite collection of points in $\mathcal{X}$. For a Gaussian process over functions $f \in \mathcal{H}$ the probability density of $\mathbf{f} = \{f(x_1), ..., f(x_n)\}^T$ is a multivariate normal with $\boldsymbol{\mu} = \{\mu(x_1), ..., \mu(x_n)\}$ and covariance $\boldsymbol{\Sigma}_{ij} = K(x_i, x_j)$

$$\mathrm{f} \sim \mathrm{N}\Big(\boldsymbol{\mu}, \boldsymbol{\Sigma}\Big),$$

where $\mu(x) = \mathbb{E}f(x)$ and $K(x_i, x_j) = \mathbb{E}\big[(f(x_i) - \mu(x_i))(f(x_j) - \mu(x_j))\big]$ and

$$f \sim \mathcal{GP}(\mu(\cdot), K(\cdot,\cdot)).$$

Definition. *A stochastic process over domain $\mathcal{X}$ with mean function μ and covariance kernel K is a Gaussian process if and only if for any $\{x_1, ..., x_n\} \in \mathcal{X}$ and $n \in \mathbb{N}$ the distribution of $\mathbf{f} = \{f(x_1), ..., f(x_n)\}^T$ is*

$$\mathbf{f} = \left(\begin{array}{c} f(x_1) \\ \vdots \\ f(x_n) \end{array} \right) \sim N\left(\left[\begin{array}{c} \mu(x_1) \\ \vdots \\ \mu(x_n) \end{array} \right], \left[\begin{array}{ccc} K(x_1,x_1) & \cdots & K(x_1,x_n) \\ \vdots & \ddots & \vdots \\ K(x_1,x_n) & \cdots & K(x_n,x_n) \end{array} \right] \right).$$

9.2. Gaussian process regression

Consider data $D = \{(x_i, y_i)\}_{i=1}^n$ drawn from the model

$$Y_i = f(x_i) + \varepsilon_i, \quad \varepsilon_i \overset{iid}{\sim} \mathrm{N}(0, \sigma^2),$$

we will place a prior on the space of functions using a Gaussian process

$$f \sim \mathcal{GP}(\mu(\cdot), K(\cdot,\cdot)).$$

We are also given some new variables or test data $T = \{x_i^*\}_{i=1}^m$ each of which would have a corresponding y_i^*.

We now provide some notation

$$\mathbf{X} = \left[\begin{array}{c} -x_1- \\ \vdots \\ -x_n- \end{array} \right], \quad \mathbf{X}^* = \left[\begin{array}{c} -x_1^*- \\ \vdots \\ -x_m^*- \end{array} \right], \quad \mathbf{Y} = \left[\begin{array}{c} y_1 \\ \vdots \\ y_n \end{array} \right], \quad \mathbf{Y}^* = \left[\begin{array}{c} y_1^* \\ \vdots \\ y_m^* \end{array} \right],$$

$$\boldsymbol{\varepsilon} = \left[\begin{array}{c} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{array} \right], \quad \boldsymbol{\varepsilon}^* = \left[\begin{array}{c} \varepsilon_1^* \\ \vdots \\ \varepsilon_m^* \end{array} \right], \quad \mathbf{f} = \left[\begin{array}{c} f(x_1) \\ \vdots \\ f(x_n) \end{array} \right], \quad \mathbf{f}^* = \left[\begin{array}{c} f(x_1^*) \\ \vdots \\ f(x_m^*) \end{array} \right].$$

Our ultimate objective will be to specify the predictive distribution on $\mathbf{Y}^*$ which we know will be multivariate normal

$$\mathbf{Y}^* \mid \mathbf{X}^*, \mathbf{X} \sim \mathrm{N}(\mu^*, \Sigma^*).$$

Now first observe

$$\left[\begin{array}{c} \mathbf{Y} \\ \mathbf{Y}^* \end{array} \right] \Big| \mathbf{X}^*, \mathbf{X} = \left[\begin{array}{c} \mathbf{f} \\ \mathbf{f}^* \end{array} \right] + \left[\begin{array}{c} \boldsymbol{\varepsilon} \\ \boldsymbol{\varepsilon}^* \end{array} \right] \sim \mathrm{N}\left(\mathbf{0}, \left[\begin{array}{cc} K(\mathbf{X},\mathbf{X}) + \sigma^2\mathbf{I} & K(\mathbf{X},\mathbf{X}^*) \\ K(\mathbf{X}^*,\mathbf{X}) & K(\mathbf{X}^*,\mathbf{X}^*) + \sigma^2\mathbf{I} \end{array} \right] \right),$$

where $K(\mathbf{X},\mathbf{X})$ is the $n \times n$ matrix with $\mathbf{K}_{ij} = K(x_i, x_j)$ and $K(\mathbf{X}^*,\mathbf{X}^*)$ is the $m \times m$ matrix with $\mathbf{K}_{ij}^* = K(x_i^*, x_j^*)$.

To get to the predictive distribution on $\mathbf{Y}^*$ we write the conditional $\mathbf{Y}^* \mid \mathbf{X}^*, \mathbf{X}$. Given the above multivariate normal distribution we simply condition on all the other variables to get the mean and covariance for the normal distribution for the posterior predictive density:

$$\begin{aligned} \mu^* &= K(\mathbf{X}^*,\mathbf{X})(K(\mathbf{X},\mathbf{X}) + \sigma^2\mathbf{I})^{-1}\,\mathbf{Y} \\ \Sigma^* &= K(\mathbf{X}^*,\mathbf{X}^*) + \sigma^2\mathbf{I} - K(\mathbf{X}^*,\mathbf{X})(K(\mathbf{X},\mathbf{X}) + \sigma^2\mathbf{I})^{-1}K(\mathbf{X},\mathbf{X}^*). \end{aligned}$$

The beauty of Gaussian process regression is that we can place priors over functions using a kernel and evaluating the variance of the function values at a finite number of points, all just based on properties of the multivariate normal distribution. This is a very powerful non-linear prediction tool. There is a strong relation between the kernels, rkhs and Gaussian processes. There are also some subtle differences. The main difference comes from what is called the Kalianpur $0-1$ law

Theorem (Kallianpur 1970)**.** *If $Z \sim \mathcal{G}P(\mu, K)$ is a Gaussian process with covariance kernel K and mean $\mu \in \mathcal{H}_K$ and $\mathcal{H}_K$ is infinite dimensional then*

$$\mathbf{P}(Z \in \mathcal{H}_K) = 0.$$

The point of the above theorem is that if we specify a kernel K and ensure the mean of the Gaussian process is in the rkhs $\mathcal{H}_K$ corresponding to the kernel K, draws from this Gaussian process will not be in the rkhs. What one can formally show is that if one takes any of the random functions, call them g then the following is true for all g

$$\int_{\mathcal{X}} g(u) K(x, u) \, \mathrm{d}u \in \mathcal{H}_K.$$

LECTURE 10
Sparse regression

We have seen previously that for the case that $p \gg n$ the following ridge regression model allows us stable inference

$$\arg\min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^T \beta)^2 + \lambda \|\beta\|^2, \quad \lambda > 0.$$

Often we don't just want a good predictive model but we also want to know which variables are relevant to the prediction. The problem of simultaneously inferring a good regression model as well as selecting variable is called simultaneous regression and variable selection. In this lecture we will state some standard methods for simultaneous regression and variable selection.

We fist state the standard model

$$Y_i = (\beta^*)^T x_i + \varepsilon_i,$$

however we now assume that the regression coefficients are zero for the majority coordinates ($i = 1, .., p$). The subset of non-zero coordinates for the true model $\mathcal{A}_* = \{j : |\beta_*^{(j)}| \neq 0\}$ and the number of non-zero coefficients is denoted as $|\mathcal{A}_*|$. Our objective is given data $D = \{(x_i, y_i)_{i=1}^n\}$ to infer $\widehat{\beta}$ such that

(1) Selection consistency: The non-zero subset of $\widehat{\beta}$ is denoted as $\widehat{\mathcal{A}} = \{j : |\widehat{\beta}^{(j)}| \neq 0\}$. We would like the two subsets $\mathcal{A}_*$ and $\widehat{\mathcal{A}}$ to be close for any finite n and identical as $n \to \infty$.

(2) Estimation consistency: How well do the coefficients in the selected set converge:

$$\lim_{n \to \infty} \widehat{\beta}_{\mathcal{A}_*} = \beta^*_{\mathcal{A}_*}$$

The approach we will use for simultaneous regression and variable selection is the following minimization problem

$$\arg\min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^T \beta)^2 + \lambda \|\beta\|_q^q, \quad \lambda > 0,$$

where $\|\beta\|_q^q$ is a penalization by the q-norm. We've already seen the result of minimizing the 2-norm leads to ridge regression. We will now explore two other norms: the 1-norm and the 0-norm.

We start with the zero norm

$$\widehat{\beta} := \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^n (y_i - x_i^T \beta)^2 + \lambda \|\beta\|_0^0, \quad \lambda > 0,$$

This is equivalent to the following minimization problem

$$\widehat{\beta} := \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^n (y_i - x_i^T \beta)^2 + \lambda \sum_{j=1}^p I(\beta_j \neq 0), \quad \lambda > 0,$$

which is suggesting minimizing the square error using the fewest variables possible with λ acting as the tradeoff between the number of variables and the error. The above minimization problem is NP-hard as it reduces to to exact cover by three sets. This means we can't practically implement the above optimization problem with any efficiency, even $p = 10$ requires a search over a massive space.

10.1. LASSO: Least Absolute Selection and Shrinkage Operator

The idea behind the lasso procedure is to minimize

$$\widehat{\beta} := \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^n (y_i - x_i^T \beta)^2 + \lambda \|\beta\|_1, \quad \lambda > 0,$$

where $\|\beta\|_1 = \sum_{j=1}^p |\beta_j|$. For reasons we will discuss minimizing the above penalized loss function results in variable selection and regression, results in regression coefficients that are exactly zero. An argument has been made that minimizing the 1-norm regularized problem is a good approximation of the 0-norm minimization problem. We will explore both why this minimization problem approximates the 0-norm as well procedures to minimize the 1-norm.

10.1.1. The geometry of polytopes

Recall that there is an equivalence between

$$\arg\min_{\beta \in \mathbb{R}^p} \tfrac{1}{n} \textstyle\sum_{i=1}^n (y_i - x_i^T \beta)^2 + \lambda \|\beta\|_1,$$
$$\arg\min_{\beta \in \mathbb{R}^p} \tfrac{1}{n} \textstyle\sum_{i=1}^n (y_i - x_i^T \beta)^2 \text{ subject to } \|\beta\|_1 \leq \tau.$$

We will contrast the following two minimization problems

$$\arg\min_{\beta \in \mathbb{R}^p} \tfrac{1}{n} \textstyle\sum_{i=1}^n (y_i - x_i^T \beta)^2 \text{ subject to } \|\beta\|_1 \leq \tau.$$
$$\arg\min_{\beta \in \mathbb{R}^p} \tfrac{1}{n} \textstyle\sum_{i=1}^n (y_i - x_i^T \beta)^2 \text{ subject to } \|\beta\|_2^2 \leq \tau.$$

Solutions to the upper problem is constrained to a 1-norm ball around the origin and the solutions to the lower problem is constrained to a 2-norm around the origin. Consider the true β vector to be β_*, the geometry of the square loss has ellipses as contours of equal loss. The minimizer is the smallest loss value that intersects the boundary of the p-norm ball.In the figure below we show this for two variables.

A minimizer with a sparse solution will touch/intersect the contours of the error ellipses on the axes that is sparse faces of the p-dimensional polytope. For example, when the constraint is the 2-norm ball around the origin it is very unlikely that the intersecting point will be concentrated on the axes. The geometry of the 1-norm ball especially in high dimensions intersects the ellipse at a few points. For example, the 0-norm is a star or spike that is on the axes so it will always be sparse.

Although we have considered the constrained optimization 1-norm problem the same results hold for lasso.

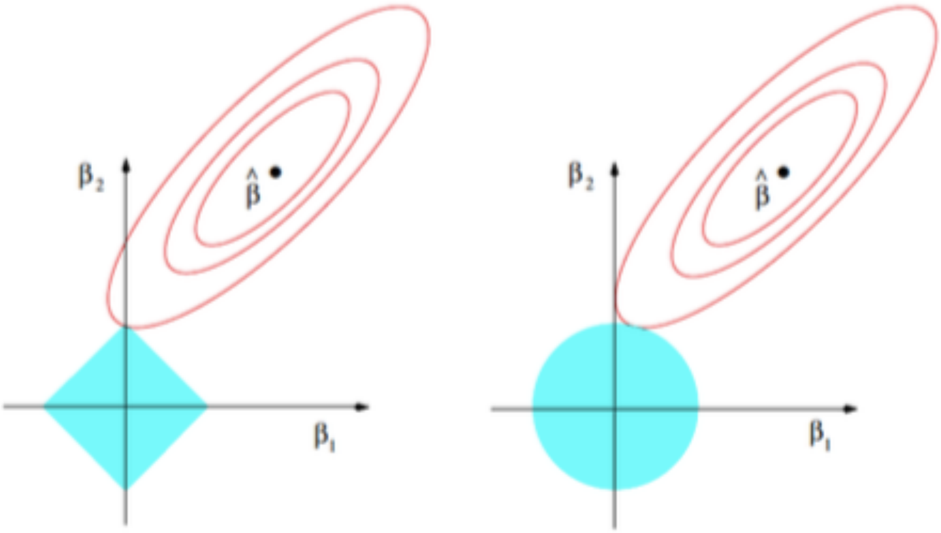


Figure 1. The 1-norm minimization for two variables is on the left and the 2-norm minimization is on the right.

10.1.2. The regularization path

Recall the optimization problem

$$\widehat{\beta} := \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^T \beta)^2 + \lambda \|\beta\|_1, \quad \lambda > 0,$$

It is known that for $\lambda = 0$ the solution $\widehat{\beta}_{\text{LASSO}} = \widehat{\beta}_{\text{OLS}}$ and for $\lambda = \infty$ the solution is $\widehat{\beta}_{\text{LASSO}} = 0$. The regression coefficients β_λ for the lasso with regularization parameter λ is a p-dimensional vector with many of its values set to zero for larger values of λ. The idea of the regularization path is to examine how the β's change with λ the picture one should consider is λ as the x-axis and the β's on the y-axis. It is a mathematical fact that the graph of the β's will be piecewise continuous and approach zero at some point.

The idea behind the regularization path is to help select how many variables to keep in the model. In the ridge model it is hard to interpret a regularization parameter as coefficients are not sent to zero and the changes are slow. This is somewhat mitigated in the lasso model.

In the figure below we consider two regression analyses, one using ridge and the other with lasso, the dataset is a prostate cancer related problem. The response variable is PSA, a biomarker marker for prostate cancer, the covariates are several other biomarkers.

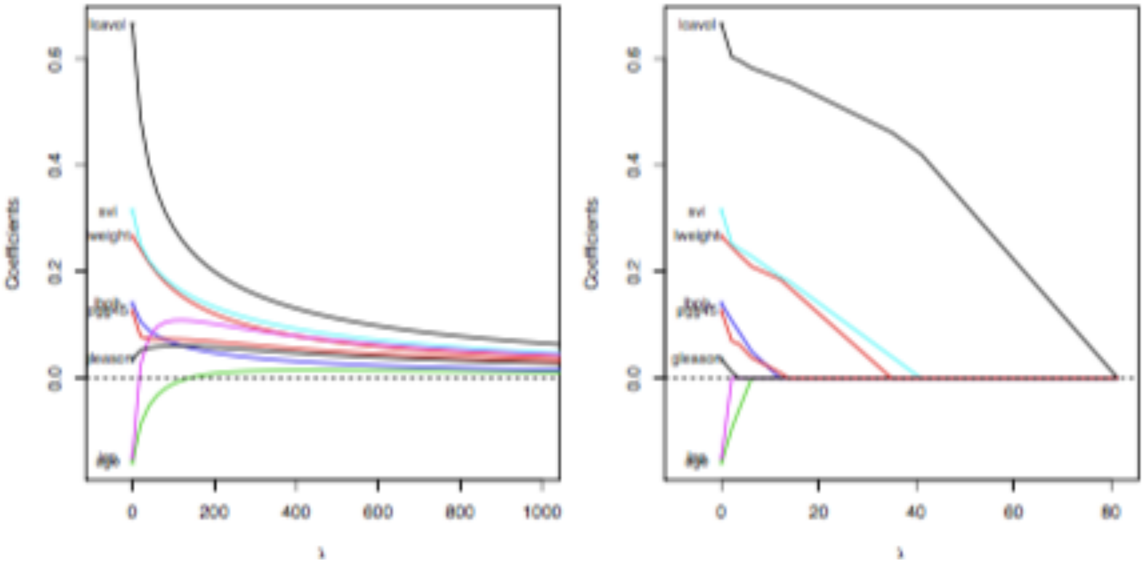


Figure 2. The left figure is the regularization path for ridge regression, the x-axis is the λ parameter and the y-axis is plotting the coefficients. The right figure is the same plot but for lasso. The response variable is PSA, a biomarker marker for prostate cancer, the covariates are a several biomarkers.

LECTURE 11
The boosting hypothesis and Adaboost

Voting algorithms or algorithms where the final classification or regression function is a weighted combination of "simpler" or "weaker" classifiers have been used extensively in a variety of applications.

We will study two examples of voting algorithms in greater depth: Bootstrap AGGregatING (BAGGING) and boosting.

11.1. Boosting

Boosting algorithms especially AdaBoost (adaptive boosting) have had a significant impact on a variety of practical algorithms and also have been the focus of theoretical investigation for a variety of fields. The formal term boosting and the first boosting algorithm came out of the field of computational complexity in theoretical Computer Science. In particular, learning as formulated by boosting came from the concept of Probably Approximatley Correct (PAC) learning.

11.2. PAC learning

The idea of Probably Approximatley Correct (PAC) learning was formulated in 1984 by Leslie Valiant as an attempt to characterize what is learnable. Let $\mathcal{X}$ be a set. This set contains encodings of all objects of interest in the learning problem. The goal of the learning algorithm is to infer some unknown subset of $\mathcal{X}$, called a concept, from a known class of concepts, $\mathcal{C}$. Unlike the previous statistical formulations of the learning problem, the issue of representation arises in this formulation due to computational issues.

- Concept classes A representation class over $\mathcal{X}$ is a pair $(\sigma, \mathcal{C})$, where $\mathcal{C} \subseteq \{0,1\}^*$ and $\sigma : \mathcal{C} \to 2^{\mathcal{X}}$. For $c \in \mathcal{C}$, $\sigma(c)$ is a concept over $\mathcal{X}$ and the image space $\sigma(\mathcal{C})$ is the concept class represented by $(\sigma, \mathcal{C})$. For $c \in \mathcal{C}$ the positive examples are $\text{pos}(c) = \sigma(c)$ and the negative examples are $\text{neg}(c) = \mathcal{X} - \sigma(c)$. The notations $c(x) = 1$ is equivalent to $x \in \text{pos}(c)$ and $c(x) = 0$ is equivalent to $x \in \text{neg}(c)$. We assume that domain points $x \in \mathcal{X}$ and representations $c \in \mathcal{C}$ are efficiently encoded with by codings of length $|x|$ and $|c|$ respectively.

- Parameterized representation We will study representation classes parameterized by an index n resulting in the domain $\mathcal{X} = \cup_{n \geq 1} \mathcal{X}_n$ and representation class $\mathcal{C} = \cup_{n \geq 1} \mathcal{C}_n$. The index n serves as a measure of the complexity of concepts in $\mathcal{C}$. For example, $\mathcal{X}$ may be the set $\{0, 1\}^n$ and $\mathcal{C}$ the set of all Boolean formulae over n variables.
- Efficient evaluation of representations If $\mathcal{C}$ is a representation class over $\mathcal{X}$, then $\mathcal{C}$ is polynomially evaluatable if there is a (probabilistic) polynomial-time evaluation algorithm $\mathcal{A}$ that given a representation $c \in \mathcal{C}$ and domain point $x \in \mathcal{X}$ outputs $c(x)$.
- Samples A labeled example from a domain $\mathcal{X}$ is a pair $< x, b >$ where $x \in \mathcal{X}$ and $b \in \{0, 1\}$. A sample $S = (< x_1, b_1 >, ..., < x_m, b_m >)$ is a finite sequence of labeled examples. A labeled example of $c \in \mathcal{C}$ has the form $< x, c(x) >$. A representation h and an example $< x, b >$ agree if $h(x) = b$. A representation h and a sample S are consistent if h agrees with each example in S.
- Distributions on examplesA learning algorithm for a representation class $\mathcal{C}$ will receive examples from a single representation $c \in \mathcal{C}$ whichw e call the target representation. Examples of the target representation are generated probabilistically: D_c is a fixed but arbitrary distribution over pos(c) and neg(c). This is the target distributions. The learning algorithm will be give access to an oracle EX which returns in unit time an example of the target representation drawn according to the target distribution D_c.
- Measure of error Given a target representation $c \in \mathcal{C}$ and a target distribution D the error of a representation $h \in \mathcal{H}$ is

$$e_c(h) = D(h(x) \neq c(x)).$$

In the above formulation $\mathcal{C}$ is the target class. In the above formulation $\mathcal{H}$ is the hypothesis class. The algorithm $\mathcal{A}$ is a learning algorithm for $\mathcal{C}$ and the output $h_{\mathcal{A}} \in \mathcal{H}$ is the hypothesis of $\mathcal{A}$.

We can now define learnability:

Definition (Strong learning). *Let $\mathcal{C}$ and $\mathcal{H}$ be representation classes over $\mathcal{X}$ that are polynomially evaluatable. Then $\mathcal{C}$ is polynomially learnable by $\mathcal{H}$ if there is a (probabilistic) algorithm $\mathcal{A}$ with access to EX, taking inputs ε, δ with the property that for any target representation $c \in \mathcal{C}$, for any target distribution D, and for any input values $0 < \varepsilon, \delta < 1$, algorithm $\mathcal{A}$ halts in time polynomial in $\frac{1}{\varepsilon}, \frac{1}{\delta}, |c|, n$ and outputs a representation $h_{\mathcal{A}} \in \mathcal{H}$ that with probability greater than $1 - \delta$ satsifies $e_c(h) < \varepsilon$.*

The parameter ε is the accuracy parameter and the parameter δ is the confidence parameter. These two parameters characterize the name Probably (δ) Approximatley (ε) Correct ($e_c(h)$). The above definition is sometimes called distribution free learning since the property holds ofver an target represenation and target distribution.

Considerable research in PAC learning has focused on which representation classes $\mathcal{C}$ are polynomially learnable.

So far we have defined learning as approximating aribirarily close the target concept. Another model of learning called weak learning considers the case where the learning algorithm is required to perform slightly better than chance.

Definition (Weak learning)**.** *Let $\mathcal{C}$ and $\mathcal{H}$ be representation classes over $\mathcal{X}$ that are polynomially evaluatable. Then $\mathcal{C}$ is polynomially weak learnable by $\mathcal{H}$ if there is a (probabilistic) algorithm $\mathcal{A}$ with access to EX, taking inputs ε, δ with the property that for any target representation $c \in \mathcal{C}$, for any target distribution D, and for any input values $0 < \varepsilon, \delta < 1$, algorithm $\mathcal{A}$ halts in time polynomial in $\frac{1}{\varepsilon}, \frac{1}{\delta}, |c|, n$ and outputs a representation $h_{\mathcal{A}} \in \mathcal{H}$ that with probability greater than $1 - \delta$ satsifies $e_c(h) < \frac{1}{2} - \frac{1}{p(|c|)}$, where p is a polynomial.*

Definition (Sample complexity)**.** *Given a learning algorithm $\mathcal{A}$ for a representation class $\mathcal{C}$. The number of calls $s_{\mathcal{A}}(\varepsilon, \delta)$ to the oracle EX made by $\mathcal{A}$ on inputs ε, δ for the worst-case measure over all target representations $c \in \mathcal{C}$ and target distributions D is the sample complexity of the algorithm $\mathcal{A}$.*

We now state some Boolean classes whose learnability we will state as positive or negative examples of learnability.

- The class M_n consist of monomials over the Boolean variables $x_i, ..., x_n$.
- For a constant k, the class $kCNF_n$ (conjunctive normal forms) consists of Boolean formulae of the form $C_1 \wedge \cdots \wedge C_l$ where each C_i is a disjunction of at most k monomials over the Boolean variables $x_i, ..., x_n$.
- For a constant k, the class $kDNF_n$ (disjunctive normal forms) consists of Boolean formulae of the form $T_1 \vee \cdots \vee T_l$ where each T_i is a conjunction of at most k monomials over the Boolean variables $x_i, ..., x_n$.
- Boolean threshold functions $I(\sum_{i=1}^n w_i x_i > t)$ where $w_i \in \{0, 1\}$ and I is the indicator function.

Definition (Empirical risk minimization, ERM)**.** *A consistent algorithm $\mathcal{A}$ is one that outputs hypotheses h that are consistent with the sample S and the range over possible hypotheses for $\mathcal{A}$ is $h \in \mathcal{C}$.*

The above algorithm is ERM in the case of zero error with the target concept in the hypothesis space.

Theorem. *If the hypothesis class is finite then $\mathcal{C}$ is learnable by the consistent algorithm $\mathcal{A}$.*

Theorem. *Boolean threshold functions are not learnable.*

Theorem. *$\{f \vee g : f \in kCNF, g \in kDNF\}$ is learnable.*

Theorem. *$\{f \wedge g : f \in kDNF, g \in kCNF\}$ is learnable.*

Theorem. *Let $\mathcal{C}$ be a concept class with finite VC dimension $VC(\mathcal{C}) = d < \infty$. Then $\mathcal{C}$ is learnable by the consistent algorithm $\mathcal{A}$.*

11.3. The hypothesis boosting problem

An important question both theoretically and practically in the late 1980's was whether strong learnablity and weak learnability were equivalent. This was the hypothesis boosting problem:

Conjecture. *A concept class $\mathcal{C}$ is weakly learnable if and only if it is strongly learnable.*

The above conjecture was proven true in 1990 by Robert Schapire.

Theorem. *A concept class $\mathcal{C}$ is weakly learnable if and only if it is strongly learnable.*

The proof of the above theorem was based upon a particular algorithm. The following algorithm takes as input a weaklearner, an error paramter ε, a confidence parameter δ, an oracle EX, and outputs a strong learner. At each iteration of the algorithm a weaklearner with error rate ε gets boosted so that its error rate decreases to $3\varepsilon^2 - 2\varepsilon^3$.

Algorithm 1: Learn(ε, δ, EX)

```
input  : error parameter $\varepsilon$, confidence parameter $\delta$, examples oracle $EX$
return: $h$ that is $\varepsilon$ close to the target concept $c$ with probability $\geq 1 - \delta$

if $\varepsilon \geq 1/2 - 1/p(n,s)$  then return WeakLearn($\delta, EX$);
$\alpha \leftarrow g^{-1}(\varepsilon)$ :  where $g(x) = 3x^2 - 2x^3$;

$EX_1 \leftarrow EX$;
$h_1 \leftarrow$ Learn($\alpha, \delta/5, EX_1$);
$\tau_1 \leftarrow \varepsilon/3$;
let $\hat{a}_1$ be an estimate of $a_1 = \Pr_{x \sim D}[h_1(x) \neq c(x)]$
  choose a sample sufficiently large that
  $|a_1 - \hat{a}_1| \leq \tau_1$ with probability $\geq 1 - \delta/5$
if $\hat{a}_1 \leq \varepsilon - \tau_1$  then  return $h_1$;

defun $EX_2$()
   {flip coin
  if heads  then  return first $x : h_1(x) = c(x)$;
  else tails  return first $x : h_1(x) \neq c(x)$}
$h_2 \leftarrow$ Learn($\alpha, \delta/5, EX_2$);
$\tau_2 \leftarrow (1 - 2\alpha)\varepsilon/9$;
let $\hat{e}$ be an estimate of $e = \Pr_{x \sim D}[h_2(x) \neq c(x)]$
  choose a sample sufficiently large that
  $|e - \hat{e}| \leq \tau_2$ with probability $\geq 1 - \delta/5$
if $\hat{e} \leq \varepsilon - \tau_2$  then  return $h_2$;

defun $EX_3$()
   {return first $x : h_1(x) \neq h_2(x)$ };
$h_3 \leftarrow$ Learn($\alpha, \delta/5, EX_3$);

defun $h(x)$
  { $b_1 \leftarrow h_1(x)$,  $b_2 \leftarrow h_2(x)$
  if $b_1 = b_2$  then  return $b_1$
  else  return $h_3(x)$}
return $h$
```

The above algorithm can be summarized as follows:

(1) Learn an initial classifier h_1 on the first N training points

(2) Learn h_2 on a new sample of N points, half of which are misclassifief by h_1
(3) Learn h_3 on N points for which h_1 and h_2 disagree
(4) The boosted classifier $h = \text{Majority vote}(h_1, h_2, h_3)$.

The basic result is that if the individual classifiers h_1, h_2, and h_3 have error ε the boosted classifier has error $2\varepsilon^2 - 3\varepsilon^3$.

To prove the theorem one needs to show that the algorithm is correct in the sense following sense.

Theorem. *For $0 < \varepsilon < 1/2$ and for $0 < \delta <\leq 1$, the hypothesis returned by calling Learn$(\varepsilon, \delta, EX)$ is ε close to the target concept with probability at least $1 - \delta$.*

We first define a few quantities

$$\begin{aligned} p_i &= \Pr_{x\sim D}[h_i(x) = c(x)] \\ q &= \Pr_{x\sim D}[h_1(x) \neq h_2(x)] \\ w &= \Pr_{x\sim D}[h_2(x) \neq h_1(x) = c(x)] \\ v &= \Pr_{x\sim D}[h_1(x) = h_2(x) = c(x)] \\ y &= \Pr_{x\sim D}[h_1(x) \neq h_2(x) = c(x)] \\ z &= \Pr_{x\sim D}[h_1(x) \neq h_2(x) \neq c(x)]. \end{aligned}$$

Given the above quantities

$$w + v = \Pr_{x\sim D}[h_1(x) = c(x)] = 1 - a_1 \tag{11.1}$$

$$y + z = \Pr_{x\sim D}[h_1(x) \neq c(x)] = a_1. \tag{11.2}$$

We can explicitly express the chance that EX_i returns an instance x in terms of the above variable:

$$\begin{aligned} D_1(x) &= D(x) \\ D_2(x) &= \frac{D(x)}{2}\left(\frac{p_1(x)}{a_1} + \frac{1 - p_(x)}{1 - a_1}\right) \\ D_3(x) &= \frac{D(x)q(x)}{w + y}. \end{aligned} \tag{11.3}$$

From equation (11.3) we have

$$\begin{aligned} 1 - a_2 &= \sum_{x\in\mathcal{X}_n} D_2(x)(1 - p_2(x)) \\ &= \frac{1}{2a_1}\sum_{x\in\mathcal{X}_n} D(x)p_1(x)(1 - p_2(x)) + \frac{1}{2(1 - a_1)}\sum_{x\in\mathcal{X}_n} D(x)(1 - p_1(x))(1 - p_2(x)) \\ &= \frac{y}{2a_1} + \frac{z}{2(1 - a_1)}. \end{aligned}$$

Combining the above equation with equations (11.1) and (11.2) we can solve for w and z in terms of y, a_1, a_2.

$$w = (2a_2 - 1)(1 - a_1) + \frac{y(1 - a_1)}{a_1}$$

$$z = a_1 - y.$$

We now control the quantity

$$\begin{aligned}
\Pr_{x\sim D}[h(x) \neq c(x)] &= \Pr_{x\sim D}[(h_1(x) = h_2(x)) \vee (h_1(x) \neq h_2(x) \wedge h_3(x) \neq c(x))] \\
&= z + \sum_{x\in\mathcal{X}_n} D(x)q(x)p_3(x) \\
&= z + \sum_{x\in\mathcal{X}_n} (w+y)D_3(x)p_3(x) \\
&= z + a_3(w+y) \\
&\leq z + \alpha(w+y) \\
&= \alpha(2a_2 - 1)(1-a_1) + a_1 + \frac{y(\alpha - a_1)}{a_1} \\
&\leq \alpha(2a_2 - 1)(1-a_1) + \alpha \\
&\leq \alpha(2\alpha - 1)(1-\alpha) + \alpha = 3\alpha^2 - 2\alpha^3 = \varepsilon,
\end{aligned}$$

the inequalities follow from the fact that $a_i \leq \alpha < 1/2$ and $y \leq a_1$. □

One also needs to show that the algorithm runs in polynomial time. The following lemma implies this. The proof of the lemma is beyond the scope of the lecture notes.

Lemma. *On a good run the expected execution time of Learn*$(\varepsilon, \delta/2, EX)$ *is polynomial in* $m, 1/\delta, 1/\epsilon$.

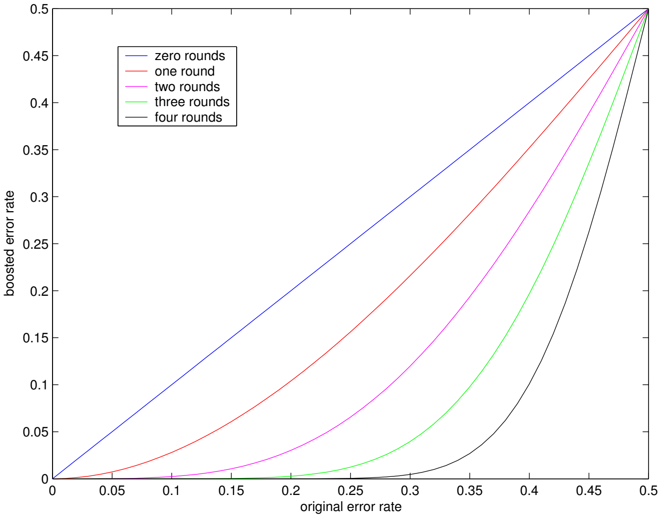


Figure 1. A plot of the boosted error rate as a function of the initial error for different numbers of boosting rounds.

11.4. ADAptive BOOSTing (AdaBoost)

We will call the above formulation of boosting the boost-by-majority algorithm. The formulation of boosting by majority by Schapire involved boosting by filtering since one weak learner served as a filter for the other. Another formulation of boost by majority was developed by Yoav Fruend also based upon filtering. Both of these algorithms were later adjusted so that sampling weights could be instead of filtering. However, all of these algorithms had the problem that the strength $1/2 - \gamma$ of the weak learner had to be known a priori.

Freund and Schapire developed the following adaptive boosting algorithm, AdaBoost, to address these issues.

Algorithm 2: AdaBoost

```
input : samples $S = (x_i, y_i)_{i=1}^N$, weak learner, number of iterations $T$
return: $h(x) = \text{sign}\left[\sum_{i=1}^T \alpha_i h_i(x)\right]$

for i=1 to N do $w_i^0 = 1/N$;

for t=1 to T do
    $h_t \leftarrow$ Call WeakLearn with weights $w^t$;
    $\varepsilon_t = \sum_{j=1}^N w_j^t I_{\{y_j \neq h_t(x_j)\}}$;
    $\alpha_t = \log((1-\varepsilon_t)/\varepsilon_t)$;
    for j=1 to N do $w_j^{t+1} = w_j^t \exp(-\alpha_t y_j h_t(x_j))$;
    $Z_t = \sum_{j=1}^N w_j^{t+1}$;
    for j=1 to N do $w_j^{t+1} = w_j^{t+1}/Z_t$;
```

For the above algorithm we can prove that the training error will decrease over boosting iterations. The advantage of the above algorithm is we don't need a uniform γ over all rounds. All we need is for each boosting round there exists a $\gamma_t > 0$.

Theorem. *Suppose WeakLearn when called by AdaBoost generates hypotheses with errors $\varepsilon_1, ..., \varepsilon_T$. Assume each $\varepsilon_i \leq 1/2$ and let $\gamma_i = 1/2 - \varepsilon_i$ then the following upper bound holds on the hypothesis h*

$$\frac{|j : h(x_j) \neq y_j|}{N} \leq \prod_{i=1}^T \sqrt{1 - 4\gamma_i^2} \leq \exp\left(-2\sum_{i=1}^T \gamma_i^2\right).$$

Proof.

If $y_i \neq h(x_i)$ then $y_i h(x_i) \leq 0$ and $e^{-y_i h(x_i)} \geq 1$. Therefore

$$\begin{aligned} \frac{|j : h(x_j) \neq y_j|}{N} &\leq \frac{1}{N}\sum_{i=1}^N e^{-y_i h(x_i)}, \\ &= \sum_{i=1}^N w_i^{T+1} \prod_{t=1}^T Z_t = \prod_{t=1}^T Z_t. \end{aligned}$$

In addition, since $\alpha_t = \log((1-\varepsilon_t)/\varepsilon_t)$ and $1 + x \leq e^x$

$$Z_t = 2\sqrt{\varepsilon_t(1-\varepsilon_t)} = \sqrt{1 - 4\gamma_t^2} \leq e^{-2\gamma_t}. \quad \square$$

11.5. A statistical interpretation of Adaboost

In this section we will reinterpret boosting as a greedy algorithm to fit an additive model. We first define our weak learners as a paramterized class of functions $h_\theta(x) = h(x;\theta)$ where $\theta \in \theta$. If we think of each weak learner as a basis function then the boosted hypothesis $h(x)$ can be thought of as a linear combination the weak learners

$$h(x) = \sum_{i=1}^{T} \alpha_i h_{\theta_i}(x),$$

where the $h_{\theta_i}(x)$ is the ith weak learner parameterized by θ_i. One approach to setting the parameters θ_i and weights α_i is called forward stagewise modelling. In this approach we sequentially add new basis functions or weak learners without adjusting the paramters and coefficients of the current solution. The following agorithm implements forward stagewise additive modeling.

```
Algorithm 3: Forward stagewise additive modeling
  input : samples $S = (x_i, y_i)_{i=1}^N$, weak learner, number of iterations $T$, loss
          function $L$
  return: $h(x) = \left[\sum_{i=1}^T \alpha_i h_{\theta_i}(x)\right]$

  $h_0(x) = 0$;
  for $i$=1 to $T$ do
      $(\alpha_t, \theta_t) = \arg\min_{\alpha \in \mathbb{R}^+, \theta \in \Theta} \sum_{i=1}^N L(y_i, h_{t-1}(x_i) + \alpha h_\theta(x))$;
      $h_t(x) = h_{t-1}(x) + \alpha_t h_{\theta_i}(x)$;
```

We will now show that the above algorithm with exponential loss

$$L(y, f(x)) = e^{-yf(x)}$$

is equivalent to AdaBoost.

At each iteration the following minimization is performed

$$\begin{aligned}
(\alpha_t, \theta_t) &= \arg \min_{\alpha \in \mathbb{R}^+, \theta \in \Theta} \sum_{i=1}^{N} \exp[-y_i(h_{t-1}(x_i) + \alpha h_\theta(x))], \\
(\alpha_t, \theta_t) &= \arg \min_{\alpha \in \mathbb{R}^+, \theta \in \Theta} \sum_{i=1}^{N} \exp[-y_i h_{t-1}(x_i)] \exp[-y_i \alpha h_\theta(x)], \\
(\alpha_t, \theta_t) &= \arg \min_{\alpha \in \mathbb{R}^+, \theta \in \Theta} \sum_{i=1}^{N} w_i^t \exp[-y_i \alpha h_\theta(x)], \qquad (11.4)
\end{aligned}$$

where $w_i^t = \exp[-y_i h_{t-1}(x_i)]$ does not effect the optimization functional. For any $\alpha > 0$ the objective function in equation (11.4) can be rewritten as

$$\begin{aligned}
\theta_t &= \arg\min_{\theta \in \Theta} \left[e^{-\alpha} \sum_{y_i = h_\theta(x_i)} w_i^t + e^{\alpha} \sum_{y_i \neq h_\theta(x_i)} w_i^t \right], \\
\theta_t &= \arg\min_{\theta \in \Theta} \left[(e^{-\alpha} - e^{\alpha}) \sum_{i=1}^N w_i^t I_{\{y_i \neq h_\theta(x_i)\}} + e^{\alpha} \sum_{i=1}^N w_i^t \right], \\
\theta_t &= \arg\min_{\theta \in \Theta} \sum_{i=1}^N w_i^t I_{\{y_i \neq h_\theta(x_i)\}}.
\end{aligned}$$

Therefore the weak learner that minimizes equation (11.4) will minimize the weighted error rate which if we plug back into equation (11.4) we can solve for α_t which is

$$\alpha_t = \frac{1}{2} \log \frac{1 - \varepsilon_t}{\varepsilon_t},$$

where

$$\varepsilon_t = \sum_{i=1}^N w_i^t I_{\{y_i \neq h_t(x_i)\}}.$$

The last thing to show is the updating of the linear model

$$h_t(x) = h_{t-1}(x) + \alpha_t h_t(x),$$

is equivalent to the reweighting used in AdaBoost. Due to the exponential loss function and the additive updating at each iteration the above sum can be rewritten as

$$w_i^{t+1} = w_i^t e^{-\alpha y_i h_t(x_i)}.$$

So AdaBoost can be interpreted as an algorithm that minimizes the exponential loss criterion via forward stagewise additive modeling.

We now give some motivation for why the exponential loss is a reasonable loss function in the classification problem. The first argument is that like the hinge loss for SVM classification the exponential loss serves as an aupper bound on the missclassification loss (see figure 2).

Another simple motivation for using the exponential loss is the minimizer of the expected loss with respect to some function class $\mathcal{H}$

$$f^*(x) = \arg\min_{f \in \mathcal{H}} \mathbb{E}_{Y|x} \left[e^{-Yf(x)} \right] = \frac{1}{2} \log \frac{\Pr(Y = 1|x)}{\Pr(Y = -1|x)},$$

estimates one-half the log-odds ratio

$$\Pr(Y = 1|x) = \frac{1}{1 + e^{-2f^*(x)}}.$$

11.6. A margin interpretation of Adaboost

We developed a geometric formulation of support vector machines in the seperable case via maximizing the margin. We will formulate AdaBoost as a margin maximization problem.

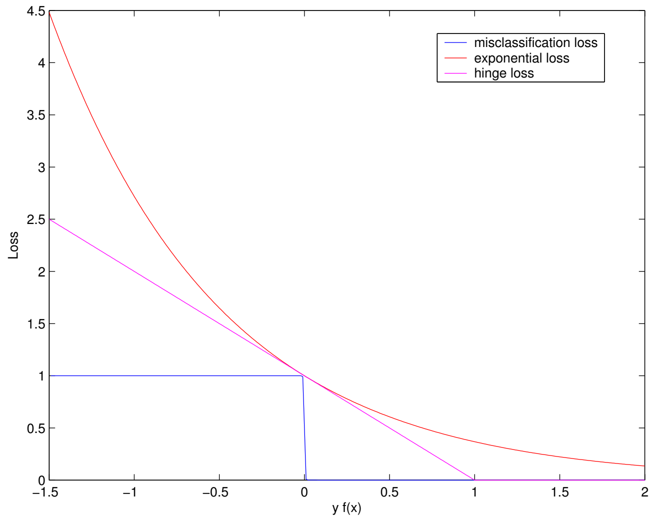


Figure 2. A comparison of loss functions for classification.

Recall that for the linear seperable SVM with points in $\mathbb{R}^d$ given a dataset S the following optimization problem characterizes the maximal margin classifier

$$\hat{w} = \arg \max_{w \in \mathbb{R}^d} \min_{x_i \in S} \frac{y_i \langle w, x_i \rangle}{||w||_{L_2}}.$$

In the case of AdaBoost we can construct a coordinate space with as many dimensions as weak classifiers, T, $u \in \mathbb{R}^T$, where the elements of $\{u_1, ..., u_T\}$ correspond to the outputs of the weak classifiers $\{u_1 = f_1(x), ..., u_T = f_T(x)\}$. We can show that AdaBoost is an iterative way to solve the following mini-max problem

$$\hat{w} = \arg \max_{w \in \mathbb{R}^T} \min_{u_i \in S} \frac{y_i \langle w, u_i \rangle}{||w||_{L_1}},$$

where $u_i = \{f_1(x_i), ..., f_T(x_i)\}$ and the final classifier has the form

$$h_T(x) = \sum_{i=1}^{T} \hat{w}_i^T f_i(x).$$

This follows immediately from the forward additive stagewise modeling interpretation since under seperabality the addition at each iteration of a weak classifier to the linear expansion will result in a boosted hypothesis h_t that as a function of t will be nondecreasing in $y_i h_t(x_i)$ $\forall i$ with the L_1 norm on w^t constrained, $||w||_{L_1} = 1$, following from the fact that the weights at each iteration must satisfy the distribution requirement.

An interesting geometry arises from the two different norms on the weights w in the two different optimization problems. The main idea is that we want to relate the norm on w to properties of norms on points in either $\mathbb{R}^d$ in the SVM case or $\mathbb{R}^T$ in the boosting case. By Hölder's inequality for the dual norms $||x||_{L_q}$ and $||w||_{L_p}$ with $\frac{1}{p} + \frac{1}{q} = 1$ and $p, q \in [1, \infty]$ the following holds

$$|\langle x, w \rangle| \leq ||x||_{L_q} ||w||_{L_p}.$$

The above inequality implies that minimizing the L_2 norm on w is equivalent to maximizing the L_2 distance between the hyperplane and the data. Similarly, minimizing the L_1 norm on w is equivalent to maximizing the L_∞ norm between the hyperplane and the data.

ONE DIMENSIONAL CONCENTRATION INEQUALITIES*

11.1. Law of Large Numbers

In this lecture, we will look at concentration inequalities or law of large numbers for a fixed function. Let $(\Omega, \mathcal{L}, \mu)$ be a probability space. Let $x_1, ..., x_n$ be real random variables on Ω. A sequence of random variables y_n converges almost surely to a random variable Y iff $\mathbb{P}(y_n \rightarrow Y) = 1$. A sequence of random variables y_n converges in probability to a random variable Y iff for every $\epsilon > 0$, $\lim_{n\rightarrow\infty} \mathbb{P}(|y_n - Y| > \epsilon) = 0$. Let $\hat{\mu}_n := n^{-1} \sum_{i=1}^n x_n$. The sequence $x_1, ..., x_n$ satisfies the strong law of large numbers if for some constant c, $\hat{\mu}_n$ converges to c almost surely. The sequence $x_1, ..., x_n$ satisfies the weak law of large numbers iff for some constant c, $\hat{\mu}_n$ converges to c in probability. In general the constant c will be the expectation of the random variable $\mathbb{E}x$.

A given function $f(x)$ of random variables x concentrates if the deviation between its empirical average, $n^{-1} \sum_{i=1}^n f(x_i)$ and expectation, $\mathbb{E}f(x)$, goes to zero as n goes to infinity. That is $f(x)$ satisfies the law of large numbers.

11.2. Polynomial inequalities

Theorem (Jensen). *If ϕ is a convex function then $\phi(\mathbb{E}x) \leq \mathbb{E}\phi(x)$.*

Theorem (Bienaymé-Chebyshev). *For any random variable x, $\epsilon > 0$*

$$\mathbb{P}(|x| \geq \epsilon) \leq \frac{\mathbb{E}x^2}{\epsilon^2}.$$

Proof.

$$\mathbb{E}x^2 \geq E(x^2 I_{\{|x|\geq\epsilon\}}) \geq \epsilon^2 \mathbb{P}(|x| > \epsilon). \ \square$$

Theorem (Markov). *For any random variable x, $\epsilon > 0$*

$$\mathbb{P}(|x| \geq \epsilon) \leq \frac{\mathbb{E}e^{\lambda x}}{e^{\lambda\epsilon}}$$

and

$$\mathbb{P}(|x| \geq \epsilon) \leq \inf_{\lambda<0} e^{-\lambda\epsilon}\mathbb{E}e^{\lambda x}.$$

Proof.

$$\mathbb{P}(x > \epsilon) = \mathbb{P}(e^{\lambda x} > e^{\lambda \epsilon}) \leq \frac{\mathbb{E}e^{\lambda x}}{e^{\lambda \epsilon}}. \ \square$$

11.3. Exponential inequalities

For the sums or averages of independent random variables the above bounds can be improved from polynomial in $1/\epsilon$ to exponential in ϵ.

Theorem (Bennet)**.** *Let $x_1, ..., x_n$ be independent random variables with $\mathbb{E}x = 0$, $\mathbb{E}x^2 = \sigma^2$, and $|x_i| \leq M$. For $\epsilon > 0$*

$$\mathbb{P}\left(|\sum_{i=1}^n x_i| > \epsilon\right) \leq 2e^{\frac{-n\sigma^2}{M^2}\phi\left(\frac{\epsilon M}{n\sigma^2}\right)},$$

where

$$\phi(z) = (1+z)\log(1+z) - z.$$

Proof. We will prove a bound on one-side of the above theorem

$$\mathbb{P}\left(\sum_{i=1}^n x_i > \epsilon\right).$$

$$\begin{aligned}\mathbb{P}\left(\sum_{i=1}^n x_i > \epsilon\right) \leq e^{-\lambda\epsilon}\mathbb{E}e^{\lambda\sum x_i} &= e^{-\lambda\epsilon}\Pi_{i=1}^n \mathbb{E}e^{\lambda x_i} \\ &= e^{-\lambda\epsilon}(\mathbb{E}e^{\lambda x})^n.\end{aligned}$$

$$\begin{aligned}\mathbb{E}e^{\lambda x} = \mathbb{E}\sum_{k=0}^\infty \frac{(\lambda x)^k}{k!} &= \sum_{k=0}^\infty \lambda^k \frac{\mathbb{E}x^k}{k!} \\ = 1 + \sum_{k=2}^\infty \frac{\lambda^k}{k!}\mathbb{E}x^2 x^{k-2} &\leq 1 + \sum_{k=2}^\infty \frac{\lambda^k}{k!} M^{k-2}\sigma^2 \\ = 1 + \frac{\sigma^2}{M^2}\sum_{k=2}^\infty \frac{\lambda^k M^k}{k!} &= 1 + \frac{\sigma^2}{M^2}(e^{\lambda M} - 1 - \lambda M) \\ &\leq e^{\frac{\sigma^2}{M^2}(e^{\lambda M} - \lambda M - 1)}.\end{aligned}$$

The last line holds since $1 + x \leq e^x$.

Therefore,

$$\mathbb{P}\left(\sum_{i=1}^n x_i > \epsilon\right) \leq e^{-\lambda\epsilon} e^{\frac{\sigma^2}{M^2}(e^{\lambda M} - \lambda M - 1)}. \tag{11.5}$$

We now optimize with respect to λ by taking the derivative with respect to λ

$$\begin{aligned} 0 &= -\epsilon + \frac{n\sigma^2}{M^2}(Me^{\lambda M} - M), \\ e^{\lambda M} &= \frac{\epsilon M}{n\sigma^2} + 1, \\ \lambda &= \frac{1}{M}\log\left(1 + \frac{\epsilon M}{n\sigma^2}\right).\end{aligned}$$

The theorem is proven by substituting λ into equation (11.5). □

The problem with Bennet's inequality is that it is hard to get a simple expression for ϵ as a function of the probability of the sum exceeding ϵ.

Theorem (Bernstein). *Let $x_1, ..., x_n$ be independent random variables with $\mathbb{E}x = 0$, $\mathbb{E}x^2 = \sigma^2$, and $|x_i| \leq M$. For $\epsilon > 0$*

$$\mathbb{P}\left(|\sum_{i=1}^n x_i| > \epsilon \right) \leq 2e^{-\frac{\epsilon^2}{2n\sigma^2 + \frac{2}{3}\epsilon M}}.$$

Proof.
Take the proof of Bennet's inequality and notice

$$\phi(z) \geq \frac{z^2}{2 + \frac{2}{3}z}. \ \square$$

Remark. With Bernstein's inequality a simple expression for ϵ as a function of the probability of the sum exceeding ϵ can be computed

$$\sum_{i=1}^n x_i \leq \frac{2}{3}uM + \sqrt{2n\sigma^2 u}.$$

Outline.

$$\mathbb{P}\left(\sum_{i=1}^n x_i > \epsilon \right) \leq 2e^{-\frac{\epsilon^2}{2n\sigma^2 + \frac{2}{3}\epsilon M}} = e^{-u},$$

where

$$u = \frac{\epsilon^2}{2n\sigma^2 + \frac{2}{3}\epsilon M}.$$

we now solve for ϵ

$$\epsilon^2 - \frac{2}{3}\epsilon M - 2n\sigma^2 \epsilon = 0$$

and

$$\epsilon = \frac{1}{3}uM + \sqrt{\frac{u^2M^2}{9} + 2n\sigma^2 u}.$$

Since $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$

$$\epsilon = \frac{2}{3}uM + \sqrt{2n\sigma^2 u}.$$

So with large probability

$$\sum_{i=1}^n x_i \leq \frac{2}{3}uM + \sqrt{2n\sigma^2 u}. \ \triangle$$

If we want to bound

$$|n^{-1}\sum_{i=1}^n f(x_i) - \mathbb{E}f(x)|$$

we consider

$$|f(x_i) - \mathbb{E}f(x)| \leq 2M.$$

Therefore

$$\sum_{i=1}^n (f(x_i) - \mathbb{E}f(x)) \leq \frac{4}{3}uM + \sqrt{2n\sigma^2 u}$$

and

$$n^{-1}\sum_{i=1}^{n} f(x_i) - \mathbb{E}f(x) \leq \frac{4}{3}\frac{uM}{n} + \sqrt{\frac{2\sigma^2 u}{n}}.$$

Similarly,

$$\mathbb{E}f(x) - n^{-1}\sum_{i=1}^{n} f(x_i) \geq \frac{4}{3}\frac{uM}{n} + \sqrt{\frac{2\sigma^2 u}{n}}.$$

In the above bound

$$\sqrt{\frac{2\sigma^2 u}{n}} \geq \frac{4uM}{n}$$

which implies $u \leq \frac{n\sigma^2}{8M^2}$ and therefore

$$|n^{-1}\sum_{i=1}^{n} f(x_i) - \mathbb{E}f(x)| \lesssim \sqrt{\frac{2\sigma^2 u}{n}} \text{ for } u \lesssim n\sigma^2,$$

which corresponds to the tail probability for a Gaussian random variable and is predicted by the Central Limit Theorem (CLT) Condition that $\lim_{n\to\infty} n\sigma^2 \to \infty$. If $\lim_{n\to\infty} n\sigma^2 = C$, where C is a fixed constant, then

$$|n^{-1}\sum_{i=1}^{n} f(x_i) - \mathbb{E}f(x)| \lesssim \frac{C}{n}$$

which corresponds to the tail probability for a Poisson random variable.

We now look at an even simpler exponential inequality where we do not need information on the variance.

Theorem (Hoeffding)**.** *Let $x_1, ..., x_n$ be independent random variables with $\mathbb{E}x = 0$ and $|x_i| \leq M_i$. For $\epsilon > 0$*

$$\mathbb{P}\left(|\sum_{i=1}^{n} x_i| > \epsilon\right) \leq 2e^{-\frac{2\epsilon^2}{\sum_{i=1}^{n} M_i^2}}.$$

Proof.

$$\mathbb{P}\left(\sum_{i=1}^{n} x_i > \epsilon\right) \leq e^{-\lambda\epsilon}\mathbb{E}e^{\lambda\sum_{i=1}^{n} x_i} = e^{-\lambda\epsilon}\Pi_{i=1}^{n}\mathbb{E}e^{\lambda x_i}.$$

It can be shown (Homework problem)

$$\mathbb{E}(e^{\lambda x_i}) \leq e^{\frac{\lambda^2 M_i^2}{8}}.$$

The bound is proven by optimizing the following with respect to λ

$$e^{-\lambda\epsilon}\Pi_{i=1}^{n}e^{\frac{\lambda^2 M_i^2}{8}}. \;\square$$

Applying Hoeffding's inequality to

$$n^{-1}\sum_{i=1}^{n} f(x_i) - \mathbb{E}f(x)$$

we can state that with probability $1 - e^{-u}$

$$n^{-1}\sum_{i=1}^{n} f(x_i) - \mathbb{E}f(x) \leq \sqrt{\frac{2Mu}{n}},$$

which is a sub-Gaussian as in the CLT but without the variance information we can never achieve the $\frac{1}{n}$ rate we achieved when the random variable has a Poisson tail distribution.

We will use the following version of Hoeffding's inequality in later lectures on Kolmogorov chaining and the Dudley's entropy integral.

Theorem (Hoeffding)**.** *Let $x_1, ..., x_n$ be independent random variables with $\mathbb{P}(x_i = M_i) = 1/2$ and $\mathbb{P}(x_i = -M_i) = 1/2$. For $\epsilon > 0$*

$$\mathbb{P}\left(|\sum_{i=1}^{n} x_i| > \epsilon\right) \leq 2e^{-\frac{2\epsilon^2}{\sum_{i=1}^n M_i^2}}.$$

Proof.

$$\mathbb{P}\left(\sum_{i=1}^{n} x_i > \epsilon\right) \leq e^{-\lambda\epsilon}\mathbb{E}e^{\lambda\sum_{i=1}^n x_i} = e^{-\lambda\epsilon}\Pi_{i=1}^n \mathbb{E}e^{\lambda x_i}.$$

$$\begin{aligned} \mathbb{E}(e^{\lambda x_i}) &= \frac{1}{2}e^{\lambda M_i} + \frac{1}{2}e^{-\lambda M_i}, \\ \frac{1}{2}e^{\lambda M_i} + \frac{1}{2}e^{-\lambda M_i} &= \sum_{k=0}^{\infty} \frac{(M_i\lambda)^{2k}}{(2k)!} \leq e^{\frac{\lambda^2 M_i^2}{2}}. \end{aligned}$$

Optimize the following with respect to λ

$$e^{-\lambda\epsilon}\Pi_{i=1}^n e^{\frac{\lambda^2 M_i^2}{2}}. \quad \square$$

11.4. Martingale inequalities

In the previous section we stated some concentration inequalities for sums of independent random variables. We now look at more complicated functions of independent random variables and introduce a particular Martingale inequality to prove concentration.

Let $(\Omega, \mathcal{L}, \mu)$ be a probability space. Let $x_1, ..., x_n$ be real random variables on Ω. Let the function $Z(x_1, ..., x_n) : \Omega^n \to \mathbb{R}$ be a map from the random variables to a real number.

The function Z concentrates if the deviation between $Z(x_1, ..., x_n)$ and $\mathbb{E}_{x_1,..,x_n} Z(x_1, .., x_n)$ goes to zero as n goes to infinity.

Theorem (McDiarmid)**.** *Let $x_1, ..., x_n$ be independent random variables let $Z(x_1, ..., x_n) : \Omega^n \to \mathbb{R}$ such that*

$$\forall x_1, ..., x_n, x_1', ..., x_n' \quad |Z(x_1, .., x_n) - Z(x_1, ..., x_{i-1}, x_i', x_{i+1}, x_n)| \leq c_i,$$

then

$$\mathbb{P}\left(Z - \mathbb{E}Z > \epsilon\right) \leq e^{-\frac{\epsilon^2}{2\sum_{i=1}^n c_i^2}}.$$

Proof.

$$\mathbb{P}(Z - \mathbb{E}Z > \epsilon) = \mathbb{P}(e^{\lambda(Z-\mathbb{E}Z)} > e^{\lambda\epsilon}) \leq e^{-\lambda\epsilon}\mathbb{E}e^{\lambda(Z-\mathbb{E}Z)}.$$

We will use the following very useful decomposition

$$\begin{aligned}
Z(x_1,...,x_n) - \mathbb{E}_{x_1',..,x_n'} Z(x_1',..,x_n') &= [Z(x_1,...,x_n) - E_{x_1'} Z(x_1', x_2, ..., x_n)] \\
&+ [E_{x_1'} Z(x_1', x_2, ..., x_n) - E_{x_1', x_2'} Z(x_1', x_2', x_3, ..., x_n)] \\
&+ ... \\
&+ [E_{x_1',...,x_{n-1}'} Z(x_1', x_2', ... x_{n-1}', x_n) - E_{x_1',...,x_n'} Z(x_1', ..., x_n')].
\end{aligned}$$

We denote the random variable

$$z_i(x_i,...,x_n) := \mathbb{E}_{x_1',...,x_{i-1}'} Z(x_1',...,x_{i-1}', x_i, ..., x_n) - \mathbb{E}_{x_1',...,x_i'} Z(x_1',...,x_i', x_{i+1},...,x_n),$$

and

$$Z(x_1,...,x_n) - \mathbb{E}_{x_1',..,x_n'} Z(x_1',..,x_n') = z_1 + ... + z_n.$$

The following inequality is true (see the following Lemma for a proof)

$$\mathbb{E}_{x_i} e^{\lambda z_i} \leq e^{\lambda^2 c_i^2/2} \;\; \forall \lambda \in \mathbb{R}.$$

$$\begin{aligned}
\mathbb{E} e^{\lambda(Z - \mathbb{E}Z)} &= \mathbb{E} e^{\lambda(z_1 + ...,+z_n)} \\
\mathbb{E}\mathbb{E}_{x_1} e^{\lambda(z_1+...,+z_n)} &= \mathbb{E} e^{\lambda(z_2+...,+z_n)} \mathbb{E}_{x_1} e^{\lambda z_1} \\
&\leq \mathbb{E} e^{\lambda(z_2+...,+z_n)} e^{\lambda c_1^2/2},
\end{aligned}$$

by induction

$$\mathbb{E} e^{\lambda(Z-\mathbb{E}Z)} \leq e^{\lambda^2 \sum_{i=1}^n c_i^2/2}.$$

To derive the bound we optimize with respect to λ

$$e^{-\lambda\epsilon + \lambda^2 \sum_{i=1}^n c_i^2/2}. \;\square$$

Lemma. *For all* $\lambda \in \mathbb{R}$

$$\mathbb{E}_{x_i} e^{\lambda z_i} \leq e^{\lambda^2 c_i^2/2}.$$

Proof.
For any $t \in [-1,1]$ the function $e^{\lambda t}$ is convex with respect to λ.

$$\begin{aligned}
e^{\lambda t} &= e^{\lambda(\frac{1+t}{2}) - \lambda(\frac{1-t}{2})} \\
&\leq \frac{1+t}{2} e^{\lambda} + \frac{1-t}{2} e^{-\lambda} \\
&= \frac{e^{\lambda} + e^{-\lambda}}{2} + t\frac{e^{\lambda} - e^{-\lambda}}{2} \\
&\leq e^{\lambda^2/2} + t\,\mathrm{sh}(\lambda).
\end{aligned}$$

Set $t = \frac{z_i}{c_i}$ and notice that $\frac{z_i}{c_i} \in [-1,1]$ so,

$$e^{\lambda z_i} = e^{\lambda c_i \frac{z_i}{c_i}} \leq e^{\lambda^2 c_i^2/2} + \frac{z_i}{c_i}\mathrm{sh}(\lambda c_i),$$

and

$$\mathbb{E}_{x_i} e^{\lambda z_i} \leq e^{\lambda^2 c_i^2/2}. \;\square$$

Example. *We can use McDiarmid's inequality to prove the concentration of the empirical minima. Given a dataset* $\{v_1 = (x_1, y_1), ..., v_n = (x_n, y_n)\}$ *the empirical minima is*

$$Z(v_1,...,v_n) = \min_{f \in \mathcal{H}_K} n^{-1} \sum_{i=1}^n V(f(x_i), y_i).$$

If the loss function is bounded one can show that for all $(v_1, ..., v_n, v_i')$

$$|Z(v_1, ..., v_n) - Z(v_1, ..., v_{i-1}, v_i', ...v_n)| \leq \frac{k}{n}.$$

Therefore with probability $1 - e^{-u}$

$$|Z - \mathbb{E}Z| \leq \sqrt{\frac{2ku}{n}}.$$

So the empirical minima concentrates.

LECTURE 12
Vapnik-Červonenkis theory

12.1. Uniform law of large numbers

In the previous lecture we considered law of large numbers for a single or fixed function. We termed this as one dimensional concentration inequalities. We now look at uniform law of large numbers, that is a law of large numbers that holds uniformly over a class of functions.

The point of these uniform limit theorems is that if the law of large numbers holds for all functions in a hypothesis space then it holds for the empirical minimizer.

The reason this chapter is called Vapnik-Červonenkis theory is that they provided some of the basic tools to study these classes.

12.2. Generalization bound for one function

Before looking at uniform results we prove generalization results when the hypothesis space $\mathcal{H}$ consists of one function, f_1.

In this case the empirical risk minimizer is f_1

$$f_1 = f_S := \arg\min_{f \in \mathcal{H}} \left[n^{-1} \sum_{i=1}^{n} V(f, z_i) \right].$$

Theorem. *Given $0 \leq V(f_1, z) \leq M$ for all z and $S = \{z_i\}_{i=1}^n$ drawn i.i.d. then with probability at least $1 - e^{-t}$ $(t > 0)$*

$$\mathbb{E}_z V(f_1, z) \leq n^{-1} \sum_{i=1}^{n} V(f_1, z_i) + \sqrt{\frac{M^2 t}{n}}.$$

Proof.

By Hoeffding's inequality

$$\mathbb{P}\left(\mathbb{E}_z V(f_1, z) - n^{-1} \sum_{i=1}^{n} V(f_1, z_i) > \varepsilon \right) \leq e^{-n\varepsilon^2/M^2}$$

so

$$\mathbb{P}\left(\mathbb{E}_z V(f_1, z) - n^{-1} \sum_{i=1}^{n} V(f_1, z_i) \leq \varepsilon \right) > 1 - e^{-n\varepsilon^2/M^2}.$$

Set $t = n\varepsilon^2/M^2$ and the result follows. □

12.3. Generalization bound for a finite number of functions

We now look at the case of ERM on a hypothesis space $\mathcal{H}$ with a finite number of functions, $k = |\mathcal{H}|$. In this case, the empirical minimizer will be one of the k functions.

Theorem. *Given $0 \leq V(f_j, z) \leq M$ for all $f_j \in \mathcal{H}, z$ and $S = \{z_i\}_{i=1}^n$ drawn i.i.d. then with probability at least $1 - e^{-t}$ $(t > 0)$ for the empirical minimizer, f_S,*

$$\mathbb{E}_z V(f_S, z) < n^{-1} \sum_{i=1}^n V(f_S, z_i) + \sqrt{\frac{M^2(\log K + t)}{n}}.$$

Proof.

The follow implication of events holds

$$\left\{ \max_{f_j \in \mathcal{H}} \ \mathbb{E}_z V(f_j, z) - n^{-1} \sum_{i=1}^n V(f_j, z_i) < \varepsilon \right\} \Rightarrow \left\{ \mathbb{E}_z V(f_S, z) - n^{-1} \sum_{i=1}^n V(f_S, z_i) < \varepsilon \right\}.$$

$$\begin{aligned} & \mathbb{P}\left(\max_{f_j \in \mathcal{H}} \ \mathbb{E}_z V(f_j, z) - n^{-1} \sum_{i=1}^n V(f_j, z_i) \geq \varepsilon \right) \\ = \ & \mathbb{P}\left(\bigcup_{f \in \mathcal{H}} \left\{ \mathbb{E}_z V(f, z) - n^{-1} \sum_{i=1}^n V(f, z_i) \geq \varepsilon \right\} \right) \\ \leq \ & \sum_{f_j \in \mathcal{H}} \mathbb{P}\left(\mathbb{E}_z V(f_j, z) - n^{-1} \sum_{i=1}^n V(f_j, z_i) \geq \varepsilon \right) \\ \leq \ & k e^{-n\varepsilon^2/M^2}, \end{aligned}$$

the last step comes from our single function result. Set $e^{-t} = ke^{-n\varepsilon^2/M^2}$ and the result follows. □

12.4. Generalization bound for compact hypothesis spaces

We now prove a sufficient condition for the generalization of hypothesis spaces with an infinite number of functions and then give some examples of such spaces.

We first assume that our hypothesis space is a subset of the space of continuous functions, $\mathcal{H} \subset \mathcal{C}(\mathcal{X})$.

Definition. *A metric space is compact if and only if it is totally bounded and complete.*

Definition. *Let R be a metric space and ϵ any positive number. Then a set $A \subset R$ is said to be an ϵ-net for a set $M \subset R$ if for every $x \in M$, there is at least one point $a \in A$ such that $\rho(x, a) < \epsilon$. Here $\rho(\cdot, \cdot)$ is a norm.*

Definition. *Given a metric space R and a subset $M \subset R$ suppose M has a finite ϵ-net for every $\epsilon > 0$. Then M is said to be totally bounded.*

Proposition. *A compact space has a finite ϵ-net for all $\epsilon > 0$.*

For the remainder of this section we will use the supnorm,

$$\rho(a, b) = \sup_{x \in \mathcal{X}} |a(x) - b(x)|.$$

Definition. *Given a hypothesis space $\mathcal{H}$ and the supnorm, the covering number $\mathcal{N}(\mathcal{H}, \epsilon)$ is the minimal number $\ell \in \mathbb{N}$ such that for every $f \in \mathcal{H}$ there exists functions $\{g_i\}_{i=1}^{\ell}$ such that*

$$\sup_{x \in \mathcal{X}} |f(x) - g_i(x)| \leq \epsilon \textit{ for some } i.$$

We now state a generalization bound for this case. In the bound we assume $V(f, z) = (f(x) - y)^2$ but the result can be easily extended for any Lipschitz loss

$$|V(f_1, z) - V(f_2, z)| \leq C||f_1(x) - f_2(x)||_\infty \ \forall z.$$

Theorem. *Let $\mathcal{H}$ be a compact subset of $\mathcal{C}(\mathcal{X})$. Given $0 \leq |f(x) - y| \leq M$ for all $f \in \mathcal{H}, z$ and $S = \{z_i\}_{i=1}^n$ drawn i.i.d. then with probability at least $1 - e^{-t}$ ($t > 0$) for the empirical minimizer, f_S,*

$$\mathbb{E}_{x,y}(f_S(x) - y)^2 < n^{-1} \sum_{i=1}^n (f_S(x_i) - y_i)^2 + \sqrt{\frac{M^2(\log \mathcal{N}(\mathcal{H}, \varepsilon/8M) + t)}{n}}.$$

We first prove two useful lemmas. Define

$$D(f, S) := \mathbb{E}_{x,y}(f(x) - y)^2 - n^{-1} \sum_{i=1}^n (f(x_i) - y_i)^2.$$

Lemma. *If $|f_j(x) - y| \leq M$ for $j = 1, 2$ then*

$$|D(f_1, S) - D(f_2, S)| \leq 4M||f_1 - f_2||_\infty.$$

Proof. Note that

$$(f_1(x) - y)^2 - (f_2(x) - y)^2 = (f_1(x) - f_2(x))(f_1(x) + f_2(x) - 2y)$$

so

$$\begin{aligned} |\mathbb{E}_{x,y}(f_1(x) - y)^2 - \mathbb{E}_{x,y}(f_2(x) - y)^2| &= \left| \int (f_1(x) - f_2(x))(f_1(x) + f_2(x) - 2y) d\mu(x, y) \right| \\ &\leq ||f_1 - f_2||_\infty \int |f_1(x) - y + f_2(x) - y| du(x, y) \\ &\leq 2M||f_1 - f_2||_\infty, \end{aligned}$$

and

$$\begin{aligned} |n^{-1} \sum_{i=1}^n [(f_1(x_i) - y_i)^2 - (f_2(x_i) - y_i)^2]| &= n^{-1} \left| \sum_{i=1}^n (f_1(x_i) - f_2(x_i))(f_1(x_i) + f_2(x_i) - 2y_i) \right| \\ &\leq ||f_1 - f_2||_\infty \frac{1}{n} \sum_{i=1}^n |f_1(x_i) - y_i + f_2(x_i) - y_i| \\ &\leq 2M||f_1 - f_2||_\infty. \end{aligned}$$

The result follows from the above inequalities. □

Lemma. *Let $\mathcal{H} = B_1 \bigcup ... \bigcup B_\ell$ and $\varepsilon > 0$. Then*

$$\mathbb{P}\left(\sup_{f \in \mathcal{H}} D(f, S)\right) \leq \sum_{j=1}^{\ell} \mathbb{P}\left(\sup_{f \in B_j} D(f, S)\right).$$

Proof.

The result follows from the following equivalence and the union bound

$$\sup_{f \in \mathcal{H}} D(f,S) \geq \varepsilon \Longleftrightarrow \exists j \leq \ell \text{ s.t. } \sup_{f \in B_j} D(f,S) \geq \varepsilon. \ \square$$

We now prove Theorem 12.4.

Let $\ell = \mathcal{N}\left(\mathcal{H}, \frac{\varepsilon}{4M}\right)$ and the functions $\{g_j\}_{j=1}^{\ell}$ have the property that the disks B_j centered at f_j with radius $\frac{\varepsilon}{4M}$ cover $\mathcal{H}$. By the first lemma for all $f \in B_j$

$$|D(f,S) - D(f_j,S)| \leq 4M||f - f_j||_\infty \leq 4M\frac{\varepsilon}{4M} = \varepsilon,$$

this implies that for all $f \in B_j$

$$\sup_{f \in B_j} |D(f,S)| \geq 2\varepsilon \Rightarrow |D(f_j,S)| \geq \varepsilon.$$

So

$$\mathbb{P}\left(\sup_{f \in B_j} |D(f,S)| \geq 2\varepsilon\right) \leq \mathbb{P}\left(|D(f_j,S)| \geq \varepsilon\right) \leq 2e^{-\varepsilon^2 n/M^2}.$$

This combined with the second lemma implies

$$\mathbb{P}\left(\sup_{f \in \mathcal{H}} D(f,S)\right) \leq \mathcal{N}\left(\mathcal{H}, \frac{\varepsilon}{8M}\right) e^{-\varepsilon^2 n/M^2}.$$

Since the following implication of events holds

$$\left\{\sup_{f \in \mathcal{H}} \ \mathbb{E}_z V(f_j,z) - n^{-1}\sum_{i=1}^{n} V(f_j,z_i) < \varepsilon\right\} \Rightarrow \left\{\mathbb{E}_z V(f_S,z) - n^{-1}\sum_{i=1}^{n} V(f_S,z_i) < \varepsilon\right\}$$

the result is obtained by setting $e^{-t} = \mathcal{N}\left(\mathcal{H}, \frac{\varepsilon}{8M}\right) e^{-n\varepsilon^2/M^2}$. □

A result of the above theorem is the following sufficient condition for uniform convergence and consistency of ERM.

Corollary. *For a Lipschitz loss function ERM is consistent if for all* $\varepsilon > 0$

$$\lim_{n \to \infty} \frac{\log \mathcal{N}\left(\mathcal{H}, \varepsilon\right)}{n} = 0.$$

Proof.

This follows directly from the statement

$$\mathbb{P}\left(\sup_{f \in \mathcal{H}} D(f,S)\right) \leq \mathcal{N}\left(\mathcal{H}, \frac{\varepsilon}{8M}\right) e^{-\varepsilon^2 n/M^2}. \ \square$$

We now compute covering numbers for a few types of hypothesis spaces.

We also need the definition of packing numbers.

Definition. *Given a hypothesis space* $\mathcal{H}$ *and the supnorm,* ℓ *functions* $\{g_i\}_{i=1}^{\ell}$ *are* ϵ*-separated if*

$$\sup_{x \in \mathcal{X}} |g_j(x) - g_i(x)| > \epsilon \ \ \forall i \neq j.$$

The packing number $\mathcal{P}(\mathcal{H}, \epsilon)$ *is the maximum cardinality of* ϵ*-separated sets.*

The following relationship between packing and covering numbers is very useful.

Lemma. *Given a metric space (A, ρ). Then for all $\epsilon > 0$ and for every $W \subset A$, the covering numbers and packing numbers satisfy*

$$\mathcal{P}(W, 2\epsilon, \rho) \le \mathcal{N}(W, \epsilon, \rho) \le \mathcal{P}(W, \epsilon, \rho).$$

Proof.

For the second inequality suppose P is an ϵ-packing of maximal cardinality, $\mathcal{P}(W, \epsilon, d)$. Then for any $w \in W$ there must be a $u \in P$ with $\rho(u, w) < \epsilon$, otherwise w is not an element of P and $P \cup w$ is an ϵ-packing. This contradicts the assumption that P is a maximal ϵ-packing. So any maximal ϵ packing is an ϵ-cover.

For the first inequality suppose C is an ϵ-cover for W and and that P is a 2ϵ-packing of W with maximum cardinality $\mathcal{P}(W, \epsilon, d)$. We will show that $|P| \le |C|$. Assume that $|C| > |P|$. Then for two points $w_1, w_2 \in P$ and one point $u \in C$ the following will hold

$$\rho(w_1, u) \le \epsilon \text{ and } \rho(w_2, u) \le \epsilon \Longrightarrow \rho(w_1, w_2) \le 2\epsilon.$$

This contradicts the fact that the points in P are 2ϵ-separated. □

In general we will compute packing numbers for hypothesis spaces and use the above lemma to obtain the covering number.

The following proposition will be useful.

Proposition. *Given $x \in \mathbb{R}^d$, the restriction the space to the unit ball $B = \{x : ||x|| \le M\}$, and the standard Euclidean metric $\rho(x, y) = ||x - y||$, then for $\epsilon \le M$*

$$\mathcal{P}(B, \epsilon, \rho) \le \left(\frac{3M}{\epsilon}\right)^d.$$

Proof.

The ℓ points $w_1, .., w_\ell$ form an optimal ϵ-packing so

$$\begin{aligned}
\text{Vol}\left(M + \frac{\epsilon}{2}\right) &= C_d \left(M + \frac{\epsilon}{2}\right)^d \\
\text{Vol}\left(\frac{\epsilon}{2}\right) &= C_d \left(\frac{\epsilon}{2}\right)^d \\
\ell[C_d \left(\frac{\epsilon}{2}\right)^d] &= C_d \left(M + \frac{\epsilon}{2}\right)^d \\
\ell &\le \left(\frac{2M + \epsilon}{\epsilon}\right)^d \\
&\le \left(\frac{3M}{\epsilon}\right)^d \text{ for all } \epsilon \le M. \ \square
\end{aligned}$$

Example. *Covering numbers for a finite dimensional RKHS.*

For a finite dimensional bounded RKHS

$$\mathcal{H}_K = \left\{ f : f(x) = \sum_{p=1}^{m} c_p \phi_p(x) \right\},$$

with $\|f\|_K^2 \le M$.

By the reproducing property and Cauchy-Schwartz inequality, the supnorm can be bound by the RKHS norm:

$$\begin{aligned} ||f(\mathbf{x})||_\infty &= ||\langle K(\mathbf{x},\cdot), f(\cdot)\rangle_K||_\infty \\ &\leq ||K(\mathbf{x},\cdot)||_K ||f||_K \\ &= \sqrt{\langle K(x,\cdot), K(x,\cdot)\rangle} ||f||_K \\ &= \sqrt{K(\mathbf{x},\mathbf{x})} ||f||_K \\ &\leq \kappa ||f||_K \end{aligned}$$

This means that if we can cover with the RKHS norm we can cover with the supnorm.

Each function in our cover, $\{g_i\}_{i=1}^{\ell}$ can be written as

$$g_i(x) = \sum_{p=1}^{m} d_{ip}\phi_p(x)$$

So if we find ℓ vectors d_i for which for all $c : \sum_{p=1}^{m} \frac{c_p^2}{\lambda_p} \leq M$ there exists a d_i such that

$$\sum_{p=1}^{m} \frac{(c_p - d_{ip})^2}{\lambda_p} < \epsilon^2,$$

we have a cover at scale ϵ. The above is simply a weighted Euclidean norm and can be reduced to the problem of covering a ball of radius M in $\mathbb{R}^m$ using the Euclidean metric. Using proposition 12.4 we can bound the packing number with the RKHS norm and the supnorm

$$\begin{aligned} \mathcal{P}(\mathcal{H}, \epsilon, ||\cdot||_{\mathcal{H}_k}) &\leq \left(\frac{3M}{\epsilon}\right)^m, \\ \mathcal{P}(\mathcal{H}, \epsilon, ||\cdot||_\infty) &\leq \left(\frac{3M}{\kappa\epsilon}\right)^m. \end{aligned}$$

Using lemma 12.4 we can get a bound on the covering number

$$\mathcal{N}(\mathcal{H}, \epsilon, ||\cdot||_\infty) \leq \left(\frac{3M}{\kappa\epsilon}\right)^m.$$

We have shown that for $\mathcal{H} \subset \mathcal{C}(\mathcal{X})$ that is compact with respect to the supnorm we have uniform convergence. This requirement is to strict to determine necessary conditions. A large class of functions that these conditions do not apply to are indicator functions $f(x) \in \{0, 1\}$.

12.5. Generalization bound for hypothesis spaces of indicator functions

In this section we derive necessary and sufficient conditions for uniform convergence of indicator functions and as a result generalization bounds for indicator functions, $f(x) \in \{0, 1\}$.

As in the case of compact functions we will take a class of indicator functions $\mathcal{H}$ and reduce this to some finite set of functions. In the case of indicator functions this is done via the notion of a growth function which we now define.

Definition. *Given a set of n points $\{x_i\}_{i=1}^n$ and a class of indicator functions $\mathcal{H}$ we say that a function $f \in \mathcal{H}$ picks out a certain subset of $\{x_i\}_{i=1}^n$ if this set can be formed by the operation $f \cap \{x_i\}_{i=1}^n$. The cardinality of the number of subsets that can be picked out is called the growth function:*

$$\triangle_n(\mathcal{H}, \{x_i\}_{i=1}^n) = \#\left\{f \cap \{x_i\}_{i=1}^n : f \in \mathcal{H}\right\}.$$

We will now state a lemma which will look very much like the generalization results for the compact or finite dimensional case.

Lemma. *Let $\mathcal{H}$ be a class of indicator functions and $S = \{z_i\}_{i=1}^n$ drawn i.i.d. then with probability at least $1 - e^{-t/8}$ $(t > 0)$ for the empirical minimizer, f_S,*

$$\mathbb{E}_{x,y} I_{\{f_S(x) \neq y\}} < n^{-1} \sum_{i=1}^n I_{\{f_S(x_i) \neq y_i\}} + \sqrt{\frac{(8 \log 8 \triangle_n(\mathcal{H}, S) + t)}{n}},$$

where $\triangle_n(\mathcal{H}, S)$ is the growth function given S observations.

Note: the above result depends upon a particular draw of $2n$ samples through the growth function. We will remove this dependence soon.

We first prove two useful lemmas. Define

$$D(f, S) := \mathbb{E}_{x,y} I_{\{f(x) \neq y\}} - n^{-1} \sum_{i=1}^n I_{\{f(x_i) \neq y_i\}}.$$

The first lemma is based upon the idea of symmetrization and replaces the deviation between the empirical and expected error to the difference between two empirical errors.

Lemma. *Given two independent copies of the data $S = \{z_i\}_{i=1}^n$ and $S = \{z_i'\}_{i=1}^n$ then for any fixed $f \in \mathcal{H}$ if $n \geq 2/\epsilon^2$*

$$\mathbb{P}\left(|D(f, S)| > \epsilon\right) \leq 2\, \mathbb{P}\left(|D(f, S) - D(f, S')| > \epsilon/2\right),$$

where

$$|D(f, S) - D(f, S')| = n^{-1} \sum_{i=1}^n I_{\{f(x_i) \neq y_i\}} - n^{-1} \sum_{i=1}^n I_{\{f(x_i') \neq y_i'\}}.$$

Proof. We first assume that

$$\mathbb{P}(|D(f, S')| \leq \epsilon/2 \,|\, S) \geq 1/2,$$

where we have conditioned on S. Since S and S' are independent we can integrate out

$$1/2\, \mathbb{P}(|D(f, S')| > \epsilon) \leq \mathbb{P}(|D(f, S')| \leq \epsilon/2, |D(f, S')| > \epsilon).$$

By the triangle inequality $|D(f, S)| > \epsilon$ and $|D(f, S')| \leq \epsilon/2$ implies

$$|D(f, S) - D(f, S')| \geq \epsilon/2,$$

so

$$\mathbb{P}(|D(f, S')| \leq \epsilon/2, |D(f, S')| > \epsilon) \leq \mathbb{P}(|D(f, S) - D(f, S')| \geq \epsilon/2).$$

To complete the proof we need to show our initial assumption holds. Since $\mathcal{H}$ is a class of indicator functions the elements in the sum are binomial random variables and the variance of n of them will be at most $1/4n$. So by the Bienaymé-Chebyshev inequality

$$\mathbb{P}(|D(f, S')| > \epsilon/2) \geq 1/4n\epsilon^2,$$

which implies the initial assumption when $n \geq 2/\epsilon^2$. □

By symmetrizing we now have a term that depends only on samples. The problem is that it depends on the samples we have but also an independent compy. This nuisance is removed by a second step of symmetrization.

Lemma. *Let σ_i be a Rademacher random variable ($\mathbb{P}(\sigma_i = \pm 1) = 1/2$) then*

$$\mathbb{P}\left(|D(f,S) - D(f,S')| > \epsilon/2\right) \leq 2\mathbb{P}\left(\left|n^{-1}\sum_{i=1}^n \sigma_i I_{\{f(x_i)\neq y_i\}}\right| > \epsilon/4\right).$$

Proof.

$$\begin{aligned}\mathbb{P}\left(|D(f,S) - D(f,S')| > \epsilon/2\right) &= \mathbb{P}\left(\left|n^{-1}\sum_{i=1}^n \sigma_i I_{\{f(x_i)\neq y_i\}} - n^{-1}\sum_{i=1}^n \sigma_i I_{\{f(x_i')\neq y_i'\}}\right| > \epsilon/2\right) \\ &\leq \mathbb{P}\left(\left|n^{-1}\sum_{i=1}^n \sigma_i I_{\{f(x_i)\neq y_i\}}\right| > \epsilon/4\right) + \\ &\quad\ \mathbb{P}\left(\left|n^{-1}\sum_{i=1}^n \sigma_i I_{\{f(x_i')\neq y_i'\}}\right| > \epsilon/4\right) \\ &\leq 2\mathbb{P}\left(\left|n^{-1}\sum_{i=1}^n \sigma_i I_{\{f(x_i)\neq y_i\}}\right| > \epsilon/4\right). \ \square\end{aligned}$$

The second symmetrization step allows is to bound the deviation between the empirical and expected errors based upon a quantity computed just on the empirical data.

We now prove Lemma 12.5.

By the symmetrization lemmas for $n \geq 8/\epsilon^2$

$$\mathbb{P}(|D(f,S)| > \epsilon) \leq 4\,\mathbb{P}\left(\left|n^{-1}\sum_{i=1}^n \sigma_i I_{\{f(x_i)\neq y_i\}}\right| > \epsilon/4\right).$$

By the Rademacher version of Hoeffdings inequality

$$\mathbb{P}\left(\left|n^{-1}\sum_{i=1}^n \sigma_i I_{\{f(x_i)\neq y_i\}}\right| > \epsilon\right) \leq 2e^{-2\epsilon^2}.$$

Combining the above for a single function

$$\mathbb{P}(|D(f,S)| > \epsilon) \leq 8e^{-\epsilon^2/8}.$$

Given data S the growth function characterizes the cardinality of subsets that can be "picked out" which is a bound on the number of possible labellings or realizable functions, $\ell = \triangle_n(\mathcal{H}, S)$. We index the possible labelings by f_j where $j = 1, ..., \ell$.

We now proceed as in the case of a finite number of functions

$$\begin{aligned}\mathbb{P}\left(\sup_{f\in\mathcal{H}}|D(f,S)|\geq\epsilon\right) &= \mathbb{P}\left(\bigcup_{f\in\mathcal{H}}|D(f,S)|\geq\epsilon\right)\\ &\leq \sum_{i=1}^{\ell}\mathbb{P}\left(|D(f_j,S)|\geq\varepsilon\right)\\ &\leq 8\triangle_n(\mathcal{H},S)e^{-n\epsilon^2/8}.\end{aligned}$$

Setting $e^{-t/8}=8\triangle_n(\mathcal{H},S)e^{-n\epsilon^2/8}$ completes the proof. □

This bound is not uniform since the growth function depends on the data S. We can make the bound uniform by defining a uniform notion of the growth function.

Definition. *The uniform growth function is*

$$\triangle_n(\mathcal{H})=\max_{x_1,\ldots,x_n\in\mathcal{X}}\triangle_n(\mathcal{H},\{x_i\}_{i=1}^n).$$

Corollary. *Let $\mathcal{H}$ be a class of indicator functions and $S=\{z_i\}_{i=1}^n$ drawn i.i.d. then with probability at least $1-e^{-t/8}$ $(t>0)$ for the empirical minimizer, f_S,*

$$\mathbb{E}_{x,y}I_{\{f_S(x)\neq y\}}<n^{-1}\sum_{i=1}^n I_{\{f_S(x_i)\neq y_i\}}+\sqrt{\frac{(\log 8\triangle_n(\mathcal{H})+t)}{n}},$$

where $\triangle_n(\mathcal{H})$ is the uniform growth function.

Corollary. *For a class of indicator functions ERM is consistent if and only if for all $\varepsilon>0$*

$$\lim_{n\to\infty}\frac{8\log\triangle_n(\mathcal{H})}{n}=0.$$

We now characterize conditions under which the uniform growth function grows polynomially. To do this we need a few definitions.

Definition. *A hypothesis space, $\mathcal{H}$, shatters a set $\{x_1,...,x_n\}$ if each of its 2^n subsets can be "picked out" by $\mathcal{H}$. The Vapnik-Červonenkis (VC) dimension, $v(\mathcal{H})$, of a hypothesis space is the largest n for which all sets of size n are shattered by $\mathcal{H}$*

$$v(\mathcal{H})=\sup\left\{n:\triangle_n(\mathcal{H})=2^n\right\},$$

if the exists no such n then the VC dimension is infinite.

Definition. *A hypothesis space of indicator functions $\mathcal{H}$ is a VC class if and only if it has finite VC dimension.*

Examples.

The VC dimension controls the growth function via the following lemma.

Lemma. *For a hypothesis space $\mathcal{H}$ with VC dimension d and $n>d$*

$$\triangle_n(\mathcal{H})\leq\sum_{i=1}^d\binom{n}{i}.$$

Proof.

The proof will be by induction on $n+d$. We define $\binom{n}{i} := 0$ if $i < 0$ or $i > n$. In addition one can check

$$\binom{n}{i} = \binom{n-1}{i-1} + \binom{n-1}{i}.$$

When $d = 0$ $|\mathcal{H}| = 1$ since no points can be shattered so for all n

$$\triangle_n(\mathcal{H}) = 1 = \binom{n}{0} = \Phi_0(n).$$

When $n = 0$ there is only one way to label 0 examples so

$$\triangle_0(\mathcal{H}) = 1 = \sum_{i=1}^{d} \binom{0}{i} = \Phi_d(0).$$

Assume the lemma to hold for n', d' such that $n' + d' < n + d$.

Given $S = \{x_1, ..., x_n\}$ and $S_n = \{x_1, ..., x_{n-1}\}$. We now define three hypothesis spaces $\mathcal{H}_0$, $\mathcal{H}_1$, and $\mathcal{H}_2$:

$$\begin{aligned}
\mathcal{H}_0 &:= \{f_i : i = 1, ..., \triangle_n(\mathcal{H}, S)\} \\
\mathcal{H}_1 &:= \{f_i : i = 1, ..., \triangle_{n-1}(\mathcal{H}, S_n)\} \\
\mathcal{H}_2 &:= \mathcal{H}_0 - \mathcal{H}_1,
\end{aligned}$$

where each f_i in set $\mathcal{H}_0$ is a possible labeling of S by $\mathcal{H}$, each f_i in set $\mathcal{H}_1$ is a possible labeling of S_n by $\mathcal{H}$.

For the set $\mathcal{H}_1$ over S_n: $n_1 = n-1$ since there is one fewer sample and $v(\mathcal{H}_1) \leq d$ since reducing the number of hypotheses cannot increase the VC dimension.

For the set $\mathcal{H}_2$ over S_n: $n_1 = n-1$ since there is one fewer sample and $v(\mathcal{H}_2) \leq d-1$. If $S' \subseteq S_n$ is shattered by $\mathcal{H}_2$ then all labellings of S' must occur both in $\mathcal{H}_1$ and $\mathcal{H}_2$ but with different labels on x_n. So $S' \cup \{x_n\}$ which has cardinality $|S'|+1$ is shattered by $\mathcal{H}$ and so $|S'|$ cannot be more than $d-1$.

By induction $\triangle_{n-1}(\mathcal{H}_1, S_n) \leq \Phi_d(m-1)$ and $\triangle_{n-1}(\mathcal{H}_2, S_n) \leq \Phi_{d-1}(m-1)$.

By construction

$$\begin{aligned}
\triangle_n(\mathcal{H}, S) &= |\mathcal{H}_1| + |\mathcal{H}_2| = \triangle_{n-1}(\mathcal{H}_1, S_n) + \triangle_{n-1}(\mathcal{H}_2, S_n) \\
&\leq \Phi_d(n-1) + \Phi_{d-1}(n-1) \\
&= \sum_{i=0}^{d} \binom{n-1}{i} + \sum_{i=0}^{d-1} \binom{n-1}{i} \\
&= \sum_{i=0}^{d} \binom{n-1}{i} + \sum_{i=0}^{d} \binom{n-1}{i-1} \\
&= \sum_{i=0}^{d} \left[\binom{n-1}{i} + \binom{n-1}{i-1} \right] \\
&= \sum_{i=0}^{d} \binom{n}{i}. \quad \square
\end{aligned}$$

Lemma. *For $n \geq d \geq 1$*

$$\sum_{i=1}^{d} \binom{n}{i} < \left(\frac{en}{d}\right)^d.$$

Proof.

For $0 \leq i \leq d$ and $n \geq d$

$$(m/d)^d (d/m)^i \geq 1,$$

so

$$\sum_{i=1}^{d} \binom{n}{i} \leq (n/d)^d \sum_{i=1}^{d} \binom{n}{i} (d/n)^i \leq (n/d)^d (1 + d/n)^n < (ne/d)^d. \quad \square$$

This now lets state the generalization bound in terms of VC dimension.

Theorem. *Let $\mathcal{H}$ be a class of indicator functions with VC dimension d and $S = \{z_i\}_{i=1}^n$ drawn i.i.d. then with probability at least $1 - e^{-t/8}$ ($t > 0$) for the empirical minimizer, f_S,*

$$\mathbb{E}_{x,y} I_{\{f_S(x) \neq y\}} < n^{-1} \sum_{i=1}^{n} I_{\{f_S(x_i) \neq y_i\}} + 2\sqrt{\frac{(8d \log(8en/d) + t)}{n}}.$$

Proof. From the proof of lemma we have

$$\mathbb{P}\left(\sup_{f \in \mathcal{H}} |D(f, S)| \geq \epsilon \right) \leq 8 \triangle_n(\mathcal{H}, S) e^{-n\epsilon^2/8},$$

therefore since $\triangle_n(\mathcal{H}, S) \leq \left(\frac{en}{d}\right)^d$, we have

$$\mathbb{P}\left(\sup_{f \in \mathcal{H}} |D(f, S)| \geq \epsilon \right) \leq 8 \left(\frac{en}{d}\right)^d e^{-n\epsilon^2/8},$$

and setting $e^{-t/8} = 8\left(\frac{en}{d}\right)^d e^{-n\epsilon^2/8}$ gives us

$$\mathbb{E}_{x,y} I_{\{f_S(x) \neq y\}} < n^{-1} \sum_{i=1}^{n} I_{\{f_S(x_i) \neq y_i\}} + \sqrt{\frac{(8d \log(8en/d) + t + 8 \log 8)}{n}},$$

for $n > 2$ and $d > 1$ $8 \log 8 < 8d \log(en/d)$ so

$$\sqrt{\frac{(8d \log(8en/d) + t + 8 \log 8)}{n}} < 2\sqrt{\frac{(8d \log(8en/d) + t)}{n}},$$

which proves the theorem. $\square$

Theorem. *For a class of indicator functions ERM the following are equivalent*

(1) *ERM is consistent*

(2) *for all $\varepsilon > 0$*

$$\lim_{n \to \infty} \frac{8 \log \triangle_n(\mathcal{H})}{n} = 0.$$

(3) *the VC dimension $v(\mathcal{H})$ is finite.*

12.6. Kolmogorov chaining

In this section we introduce Kolmogorov chaining which is a much more efficient way of constructing a cover. In the process we derive Dudley's entropy integral.

We first define an empirical norm.

Definition. *Given $S = \{x_1, ..., x_n\}$ the empirical ℓ_2 norm is*

$$\rho_S(f,g) = \left(n^{-1} \sum_{i=1}^n (f(x_i) - g(x_i))^2 \right)^{1/2}.$$

We can define a cover given the empirical norm

Definition. *Given a hypothesis space $\mathcal{H}$ and the above norm, the covering number $\mathcal{N}(\mathcal{H}, \epsilon, \rho_S)$ is the minimal number $\ell \in \mathbb{N}$ such that for every $f \in \mathcal{H}$ there exists functions $\{g_i\}_{i=1}^\ell$ such that*

$$\rho_S(f, g_i) \leq \epsilon \text{ for some } i.$$

The proof of the following theorem is identical to the proof of lemma 12.5.

Theorem. *Given the square loss and $\mathcal{H}$ be a functions such that $-1 \leq f(x) \leq 1$, $y \in [-1,1]$ and $S = \{z_i\}_{i=1}^n$ drawn i.i.d. then with probability at least $1 - e^{-t/8}$ $(t > 0)$ for the empirical minimizer, f_S,*

$$\mathbb{E}_{x,y}(f_S(x) - y)^2 < n^{-1} \sum_{i=1}^n (f_S(x_i) - y_i)^2 + \sqrt{\frac{(8 \log \mathcal{N}(\mathcal{H}, \varepsilon/8M, \rho_S) + t)}{n}},$$

where $\mathcal{N}(\mathcal{H}, \varepsilon/8M, \rho_S)$ is the empirical cover.

The key idea in the proof of both lemma 12.5 and the above theorem is that

$$\mathbb{P}(|D(f,S)| > \epsilon) \leq 4\, \mathbb{P}\left(\left| n^{-1} \sum_{i=1}^n \sigma_i f(x_i) \right| > \epsilon/4 \right),$$

where

$$D(f,S) := \mathbb{E}_{x,y}(f(x) - y)^2 - n^{-1} \sum_{i=1}^n (f(x_i) - y_i)^2,$$

and σ_i is a Rademacher random variable.

We now prove the chaining theorem.

Theorem. *Given a hypothesis space $\mathcal{H}$ where for all $f \in \mathcal{H}$ $-1 \leq f(x) \leq 1$ if we define*

$$R(f) = n^{-1} \sum_{i=1}^n \sigma_i f(x_i),$$

then

$$\mathbb{P}\left(\forall f \in \mathcal{H},\ R(f) \leq \frac{2^{9/2}}{\sqrt{n}} \int_0^{d(0,f)} \log^{1/2} \mathcal{P}(\mathcal{H}, \varepsilon, \rho_S) d\varepsilon + 2^{7/2} d(0,f) \sqrt{\frac{u}{n}} \right) \geq 1 - e^{-u},$$

where $\mathcal{P}(\mathcal{H}, \varepsilon, \rho_S)$ is the empirical packing number and

$$\int_0^{d(0,f)} \log^{1/2} \mathcal{P}(\mathcal{H}, \varepsilon, \rho_S) d\varepsilon$$

is Dudley's entropy interal.

Proof.

Without loss of generality we will assume that the zero function $\{0\}$ is in $\mathcal{H}$. We will construct a nested sets of functions

$$\{0\} = \mathcal{H}_0 \subseteq \mathcal{H}_1 \subseteq \mathcal{H}_2 ... \subseteq \mathcal{H}_j \subseteq ... \mathcal{H}.$$

These subsets will have the following properties

(1) $\forall f, g \in \mathcal{H}_j \;\; \rho_S(f,g) > 2^{-j}$
(2) $\forall f \in \mathcal{H} \;\; \exists \;\; f \in \mathcal{H}_j$ such that $\rho_S(f,g) \leq 2^{-j}$.

Given a set $\mathcal{H}_j$ we can construct $\mathcal{H}_{j+1}$ via the following procedure:

(1) $\mathcal{H}_{j+1} := \mathcal{H}_j$
(2) Find all $f \in \mathcal{H}$ such that for all $g \in \mathcal{H}_{j+1} \;\; \rho_S(f,g) > 2^{-(j+1)}$
(3) Add the above f to $\mathcal{H}_{j+1}$.

We now define a projection operation $\pi_j : \mathcal{H} \to \mathcal{H}_j$ where given $f \in \mathcal{H}$

$$\pi_j(f) := g \text{ where } g \in \mathcal{H}_j \text{ such that } \rho_S(g,f) \leq 2^{-j}.$$

For all $f \in \mathcal{H}$ the following chaining holds

$$\begin{aligned} f &= \pi_0(f) + (\pi_1(f) - \pi_0(f)) + (\pi_2(f) - \pi_1(f)) + ... \\ &= \sum_{j=1}^{\infty} (\pi_j(f) - \pi_{j-1}(f)), \end{aligned}$$

and

$$\begin{aligned} \rho_S(\pi_{j-1}(f), \pi_j(f)) &\leq \rho(\pi_{j-1}(f), f) + \rho_S(\pi_j(f), f) \\ &\leq 2^{-(j-1)} + 2^{-j} = 3 \cdot 2^{-j} \leq 2^{-j+2}. \end{aligned}$$

$R(f)$ is a linear function, so

$$R(f) = \sum_{j=1}^{\infty} (\pi_j(f) - \pi_{j-1}(f)).$$

The set of links in the chain between two levels are defined as follows

$$L_{j-1,j} := \{f - g : f \in \mathcal{H}_j, \, g \in \mathcal{H}_{j-1} \text{ and } \rho_S(f,g) \leq 2^{-j+2}\}.$$

For a fixed link $\ell \in L_{j-1,j}$

$$R(l) = n^{-1} \sum_{i=1}^{n} \sigma_i \ell(x_i),$$

and $|\ell(x_i)| \leq 2^{-j+2}$ so by Hoeffding's inequality

$$\begin{aligned} \mathbb{P}(R(\ell) \geq t) &\leq e^{-nt^2/(\frac{2}{n}\sum_{i=1}^n \ell^2(x_i))} \\ &\leq e^{-nt^2/(2 \cdot 2^{-2j+4})}. \end{aligned}$$

The cardinality of the set of links is

$$|L_{j-1,j}| \leq |\mathcal{H}_j| \cdot |\mathcal{H}_{j-1}| \leq (|\mathcal{H}_j|)^2.$$

So

$$\mathbb{P}\left(\forall \ell \in L_{j-1,j}, \; R(\ell) \leq t\right) \geq 1 - (|\mathcal{H}_j|)^2 \, e^{-nt^2/2^{-2j+5}},$$

setting

$$t = \sqrt{\frac{2^{-2j+5}}{n}(4 \log |\mathcal{H}_j| + u)} \leq \sqrt{\frac{2^{-2j+5}}{n} 4 \log |\mathcal{H}_j|} + \sqrt{\frac{2^{-2j+5} u}{n}},$$

gives us

$$\mathbb{P}\left(\forall \ell \in L_{j-1,j}, \; R(\ell) \leq \frac{2^{7/2} 2^{-j} \log^{1/2} |\mathcal{H}_j|}{\sqrt{n}} + 2^{5/2} 2^{-j} \sqrt{\frac{u}{n}}\right) \geq 1 - \frac{1}{|\mathcal{H}_j|} e^{-u}.$$

If $\mathcal{H}_{j-1} = \mathcal{H}_j$ then

$$\pi_{j-1}(f) = \pi_j(f) \text{ and } L_{j-1,j} = \{0\}.$$

So over all levels and links with probability at least $1 - \sum_{j=1}^{\infty} \frac{1}{|\mathcal{H}_j|} e^{-u}$

$$\forall j \geq 1,\ \forall \ell \in L_{j-1,j},\ R(\ell) \leq \frac{2^{7/2} 2^{-j} \log^{1/2} |\mathcal{H}_j|}{\sqrt{n}} + 2^{5/2} 2^{-j} \sqrt{\frac{u}{n}},$$

and

$$1 - \sum_{j=1}^{\infty} \frac{1}{|\mathcal{H}_j|} e^{-u} \geq 1 - \sum_{j=1}^{\infty} \frac{1}{j^2} e^{-u} = 1 - \left(\frac{\pi^2}{6} - 1\right) e^{-u} \geq 1 - e^{-u}.$$

For some level k

$$2^{-(k+1)} \leq d(0,f) \leq 2^{-k}$$

and

$$0 = \pi_0(f) = \pi_1(f) = \cdots = \pi_k(f).$$

So

$$\begin{aligned} R(f) &= \sum_{j=k+1}^{\infty} R(\pi_j(f) - \pi_{j-1}(f)) \\ &\leq \sum_{j=k+1}^{\infty} \left(\frac{2^{7/2} 2^{-j}}{\sqrt{n}} \log^{1/2} |\mathcal{H}_j| + 2^{5/2} 2^{-j} \sqrt{\frac{u}{n}} \right) \\ &\leq \sum_{j=k+1}^{\infty} \left(\frac{2^{7/2} 2^{-j}}{\sqrt{n}} \log^{1/2} \mathcal{P}(\mathcal{H}, 2^{-j}, \rho_S) \right) + 2^{5/2} 2^{-k} \sqrt{\frac{u}{n}}. \end{aligned}$$

Since $2^{-k} < 2d(f,0)$ we get the second term in the theorem

$$2^{7/2} d(0,f) \sqrt{\frac{u}{n}}.$$

For the first term

$$\begin{aligned} \sum_{j=k+1}^{\infty} \frac{2^{7/2} 2^{-j}}{\sqrt{n}} \log^{1/2} \mathcal{P}(\mathcal{H}, 2^{-j}, \rho_S) &\leq \frac{2^{9/2}}{\sqrt{n}} \sum_{j=k+1}^{\infty} 2^{-(j+1)} \log^{1/2} \mathcal{P}(\mathcal{H}, 2^{-j}, \rho_S) \\ &\leq \frac{2^{9/2}}{\sqrt{n}} \int_0^{2^{-(k+1)}} \log^{1/2} \mathcal{P}(\mathcal{H}, \varepsilon, \rho_S) d\varepsilon \\ &\leq \frac{2^{9/2}}{\sqrt{n}} \int_0^{d(0,f)} \log^{1/2} \mathcal{P}(\mathcal{H}, \varepsilon, \rho_S) d\varepsilon, \end{aligned}$$

the above quantity is Dudley's entropy integral. □

12.7. Covering numbers and VC dimension

In this section we will show how to bound covering numbers via VC dimension. Covering numbers as we have introduced them have been in general for real-valued functions and not indicator functions.

The notion of VC-dimension and VC classes can be extended to real-valued functions in a variety of mappings. The most standard extension is the notion of VC subgraph classes.

Definition. *A subgraph of function $f(x)$ where $f : \mathcal{X} \to \mathbb{R}$ is the set*

$$\mathcal{F}_f = \{(x,t) \in \mathcal{X} \times \mathbb{R} : 0 \leq t \leq f(x) \text{ or } f(x) \leq t \leq 0\}.$$

Definition. *The subgraph of a class of funtions $\mathcal{H}$ are the sets*

$$\mathcal{F} = \{\mathcal{F}_f : f \in \mathcal{H}\}.$$

Definition. *If $\mathcal{F}$ is a VC class of sets then $\mathcal{H}$ is a VC subgraph class of functions and $v(\mathcal{H}) = v(\mathcal{F})$.*

We now show that we can upper-bound the covering number with the empirical ℓ_1 norm with a function of then VC dimension for a hypothesis spaces with finite VC dimension.

Theorem. *Given a VC subgraph class $\mathcal{H}$ where $-1 \leq f(x) \leq 1 \; \forall f \in \mathcal{H}$ and $x \in \mathcal{X}$ with $v(\mathcal{H}) = d$ and $\rho_S(f,g) = n^{-1} \sum_{i=1}^n |f(x_i) - g(x_i)|$ then*

$$\mathcal{P}(\mathcal{H}, \varepsilon, \rho_s) \leq \left(\frac{8e}{\varepsilon} \log \frac{7}{\varepsilon} \right)^d .$$

The bound in the above theorem can be improved to

$$\mathcal{P}(\mathcal{H}, \varepsilon, \rho_s) \leq \left(\frac{K}{\varepsilon} \right)^d ,$$

however, the proof is more complicated so we prove the weaker statement.
Proof.

Set $m = \mathcal{P}(\mathcal{H}, \varepsilon, \rho_s)$ so $\{f_1, ..., f_m\}$ are ε-separated and each function f_k has its respective subgraph $\mathcal{F}_{f_k}$.

Sample uniformly from $\{x_1, ..., x_n\}$ k elements $\{z_1, ...z_k\}$ and uniformly on $[-1,1]$ k elements $\{t_1, ..., t_k\}$.

We now bound the probability that the subgraphs of two ε-separated functions pick out different subsets of $\{(z_1,t_1), ..., (z_k,t_k)\}$

$$\begin{aligned} & \mathbb{P}\left(\mathcal{F}_{f_k} \text{ and } \mathcal{F}_{f_l} \text{ pick out different subsets of } \{(z_1,t_1), ..., (z_k,t_k)\}\right) \\ = \; & \mathbb{P}\left(\text{at least one } (z_i,t_i) \text{ is picked out by either } \mathcal{F}_{f_k} \text{ or } \mathcal{F}_{f_l} \text{ but not the other}\right) \\ = \; & 1 - \mathbb{P}\left(\text{all } (z_i,t_i) \text{ are picked out by both or none}\right). \end{aligned}$$

The probability that (z_i,t_i) is either picked out by either both $\mathcal{F}_{f_k}, \mathcal{F}_{f_l}$ or by neither

12.8. Symmetrization and Rademacher complexities

In the previous lectures we have considered various complexity measures, such as covering numbers. But what is the right notion of complexity for the learning problem we posed? Consider the covering numbers for a moment. Take a small function class and take its convex hull. The resulting class can be extremely large. Nevertheless, the supremum of the difference of expected and empirical errors will be attained at the vertices, i.e. at the base class. In some sense, the “inside” of the class does not matter. The covering numbers take into account the whole class, and therefore become very large for the convex hull, even though the essential complexity is that of the base class. This suggests that the covering numbers are not the ideal complexity measure. In this lecture we introduce another notion (Rademacher averages), which can be claimed to be the “correct” one. In particular, the Rademacher averages of a convex hull will be equal to those of the base class. This notion of complexity will be shown to have other nice properties.

Instead of jumping right to the definition of Rademacher Averages, we will take a longer route and show how these averages arise. Results on this topic can be found in the Theory of Empirical Processes, and so we will give some definitions from it.

Let $\mathcal{F}$ be a class of functions. Then $(Z_i)_{i \in \mathcal{I}}$ is a random process indexed by $\mathcal{F}$ if $Z_i(f)$ is a random variable for any i.

As before, μ is a probability measure on Ω, and data $x_1, ..., x_n \sim \mu$. Then μ_n is the empirical measure supported on $x_1, ..., x_n$:

$$\mu_n = \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i}.$$

Define $Z_i(\cdot) = (\delta_{x_i} - \mu)(\cdot)$, i.e.

$$Z_i(f) = f(x_i) - \mathbb{E}_\mu(f).$$

Then $Z_1, ..., Z_n$ is an i.i.d. process with 0 mean.

In the previous lectures we looked at the quantity

$$\sup_{f \in \mathcal{F}} \left| \frac{1}{n} \sum_{i=1}^{n} f(x_i) - \mathbb{E} f \right|, \tag{12.1}$$

which can be written as $n \sup_{f \in \mathcal{F}} |\sum_{i=1}^{n} Z_i(f)|$.

Recall that the difficulty with (12.1) is that we do not know μ and therefore cannot calculate $\mathbb{E}f$. The classical approach of covering $\mathcal{F}$ and using the union bound is too loose.

Proposition. *Symmetrization: If $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ is close to $\mathbb{E}f$ for data $x_1, ..., x_n$, then $\frac{1}{n}\sum_{i=1}^{n} f(x_i)$ is close to $\frac{1}{n}\sum_{i=1}^{n} f(x_i')$, the empirical average on $x_1', ..., x_n'$ (an independent copy of $x_1, ..., x_n$). Therefore, if the two empirical averarages are far from each other, then empirical error is far from expected error.*

Now fix one function f. Let $\epsilon_1, ..., \epsilon_n$ be i.i.d. Rademacher random variables (taking on values 0 or 1 with probability 1/2). Then

$$\begin{aligned}
\mathbb{P}\left[\left|\sum_{i=1}^{n}(f(x_i) - f(x_i'))\right| \geq t\right] &= \mathbb{P}\left[\left|\sum_{i=1}^{n}\epsilon_i(f(x_i) - f(x_i'))\right| \geq t\right] \\
&\leq \mathbb{P}\left[\left|\sum_{i=1}^{n}\epsilon_i f(x_i)\right| \geq t/2\right] + \mathbb{P}\left[\left|\sum_{i=1}^{n}\epsilon_i f(x_i')\right| \geq t/2\right] \\
&= 2\mathbb{P}\left[\left|\sum_{i=1}^{n}\epsilon_i f(x_i)\right| \geq t/2\right]
\end{aligned}$$

Together with symmetrization, this suggests that controlling $\mathbb{P}\left(|\sum_{i=1}^{n}\epsilon_i f(x_i)| \geq t/2\right)$ is enough to control $\mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^{n} f(x_i) - \mathbb{E}f\right| \geq t\right)$. Of course, this is a very simple example. Can we do the same with quantities that are uniform over the class?

Definition. *Suprema of an Empirical process:*

$$Z(x_1, ..., x_n) = \sup_{f \in \mathcal{F}} \left[\mathbb{E}f - \frac{1}{n}\sum_{i=1}^{n} f(x_i)\right].$$

Definition. *Suprema of a Rademacher Process:*

$$R(x_1, ..., x_n, \epsilon_1, ..., \epsilon_n) = \sup_{f \in \mathcal{F}} \left[\frac{1}{n} \sum_{i=1}^{n} \epsilon_i f(x_i) \right].$$

Proposition. *The expectation of the Rademacher process bounds the expectation of the empirical process:*

$$\mathbb{E}Z \leq 2\mathbb{E}R^1.$$

Proof.

$$\begin{aligned}
\mathbb{E}Z &= \mathbb{E}_x \sup_{f \in \mathcal{F}} \left[\mathbb{E}f - \frac{1}{n} \sum_{i=1}^{n} f(x_i) \right] \\
&= \mathbb{E}_x \sup_{f \in \mathcal{F}} \left[\mathbb{E}_{x'} \left(\frac{1}{n} \sum_{i=1}^{n} f(x'_i) \right) - \frac{1}{n} \sum_{i=1}^{n} f(x_i) \right] \\
&\leq \mathbb{E}_{x,x'} \sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (f(x'_i) - f(x_i)) \\
&= \mathbb{E}_{x,x',\epsilon} \sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \epsilon_i (f(x'_i) - f(x_i)) \\
&\leq \mathbb{E}_{x,x',\epsilon} \sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \epsilon_i f(x'_i) + \mathbb{E}_{x,x',\epsilon} \sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} (-\epsilon_i) f(x_i) \\
&= 2\mathbb{E}R \quad \square.
\end{aligned}$$

As we discussed previously, we would like to bound the empirical process Z since this will imply "generalization" for any function in $\mathcal{F}$. We will bound Z by the Rademacher average $\mathbb{E}R$ which we will see has some nice properties.

Theorem. *If the functions in $\mathcal{F}$ are uniformly bounded between $[a, b]$ then with probability $1 - e^{-u}$*

$$Z \leq 2\mathbb{E}R + \sqrt{\frac{2u(b-a)}{n}}.$$

Proof. The inequality involves two steps

(1) the concentration of Z around its mean $\mathbb{E}Z$
(2) applying the bound $\mathbb{E}Z \leq 2\mathbb{E}R$

We will use McDiarmid's inequality for the first step. We define the following two variables $Z := Z(x_1, ..., x_i, ..., x_n)$ and $Z^i := Z(x_1, ..., x'_i, ..., x_n)$. Since $a \leq f(x) \leq b$ for all x and $f \in \mathcal{F}$:

$$\begin{aligned}
\left| Z^i - Z \right| &= \left| \sup_{f \in \mathcal{F}} |\mathbb{E}f - n^{-1} \sum_{j=1}^{n} f(x_j) + \left(n^{-1} f(x_i) - n^{-1} f(x'_i) \right) | - \sup_{f \in \mathcal{F}} |\mathbb{E}f - n^{-1} \sum_{j=1}^{n} f(x_j)| \right| \\
&\leq \sup_{f \in \mathcal{F}} \frac{1}{n} |f(x_i) - f(x'_i)| \leq \frac{b-a}{n} = c_i.
\end{aligned}$$

[1] The quantity $\mathbb{E}R$ is called a *Rademacher average*.

This bounds the Martingale difference for the empirical process. Given the difference bound McDiarmid's inequality states

$$\mathbb{P}\left(Z - \mathbb{E}Z > t\right) \leq \exp\left(\frac{-t^2}{2\sum_{i=1}^n \frac{(b-a)^2}{n^2}}\right) = \exp\left(\frac{-nt^2}{2(b-a)^2}\right).$$

Therefore, with probability at least $1 - e^{-u}$,

$$Z - \mathbb{E}Z < \sqrt{\frac{2u(b-a)}{n}}.$$

So as the number of samples, n, grows, Z becomes more and more concentrated around $\mathbb{E}Z$.

Applying symmetrization proves the theorem. With probability at least $1-e^{-u}$.

$$Z \leq \mathbb{E}Z + \sqrt{\frac{2u(b-a)}{n}} \leq 2\mathbb{E}R + \sqrt{\frac{2u(b-a)}{n}}. \ \square$$

McDiamid's inequality does not incorporate a notion of variance so it is possible to obtain a sharper inequality using see Talagrand's inequality for the suprema of empirical processes.

We are now left with bounding the Rademacher average. Implicit in the previous lecture on on Kolmogorov chaining was such a bound. Before we restate that result and give some examples we state some nice and useful properties of Rademacher averages.

Properties. *Let $\mathcal{F}$, $\mathcal{G}$ be classes of real-valued functions. Then for any n,*

1. *If $\mathcal{F} \subseteq \mathcal{G}$, then $\mathbb{E}R(\mathcal{F}) \leq \mathbb{E}R(G)$*
2. *$\mathbb{E}R(\mathcal{F}) = \mathbb{E}R(conv\mathcal{F})$*
3. *$\forall c \in \mathbb{R}$, $\mathbb{E}R(c\mathcal{F}) = |c|\mathbb{E}R(\mathcal{F})$*
4. *If $\phi : \mathbb{R} \to \mathbb{R}$ is L-Lipschitz and $\phi(0) = 0$, then $\mathbb{E}R(\phi(\mathcal{F})) \leq 2\,L\,\mathbb{E}R(\mathcal{F})$*
5. *For RKHS balls, $c\left(\sum_{i=1}^\infty \lambda_i\right)^{1/2} \leq \mathbb{E}R(\mathcal{F}_k) \leq C(\sum_{i=1}^\infty \lambda_i)^{1/2}$, where λ_i's are eigenvalues of the corresponding linear operator in the RKHS.*

Theorem. *The Rademacher average is bounded by Dudley's entropy integral*

$$\mathbb{E}_\epsilon R \leq c\frac{1}{\sqrt{n}} \int_0^D \sqrt{\log \mathcal{N}(\epsilon, \mathcal{F}, L_2(\mu_n))} d\epsilon,$$

where $\mathcal{N}$ denotes the covering number.

Example. *Let $\mathcal{F}$ be a class with finite VC-dimension V. Then*

$$\mathcal{N}(\epsilon, \mathcal{F}, L_2(\mu_n)) \leq \left(\frac{2}{\epsilon}\right)^{kV},$$

for some constant k. The entropy integral above is bounded as

$$\begin{aligned}\int_0^1 \sqrt{\log \mathcal{N}(\epsilon, \mathcal{F}, L_2(\mu_n))} d\epsilon \quad &\leq \quad \int_0^1 \sqrt{kV \log 2/\epsilon}\ d\epsilon \\ &\leq \quad k'\sqrt{V} \int_0^1 \sqrt{\log 2/\epsilon}\ d\epsilon \leq k\sqrt{V}.\end{aligned}$$

Therefore, $\mathbb{E}_\epsilon R \leq k\sqrt{\frac{V}{n}}$ for some constant k.

LECTURE 13
Mixture models and latent space models

We now consider models that have extra unobserved variables. The variables are called latent variables or state variables and the general name for these models are state space models.A classic example from genetics/evolution going back to 1894 is whether the carapace of crabs come from one normal or from a mixture of two normal distributions.

We will start with a common example of a latent space model, mixture models.

13.1. Mixture models

13.1.1. Gaussian Mixture Models (GMM)

Mixture models make use of latent variables to model different parameters for different groups (or **clusters**) of data points. For a point x_i, let the cluster to which that point belongs be labeled z_i; where z_i is latent, or unobserved. In this example (though it can be extended to other likelihoods) we will assume our observable features $\mathbf{x}_i$ to be distributed as a Gaussian, so the mean and variance will be cluster-specific, chosen based on the cluster that point x_i is associated with. However, in practice, almost any distribution can be chosen for the observable features. With d-dimensional Gaussian data x_i, our model is:

$$\begin{aligned} z_i \mid \boldsymbol{\pi} &\sim \text{Mult}(\boldsymbol{\pi}) \\ \mathbf{x}_i \mid z_i = k &\sim \mathcal{N}_D(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \end{aligned}$$

where $\boldsymbol{\pi}$ is a point on a K-dimensional simplex, so $\boldsymbol{\pi} \in \mathbb{R}^K$ obeys the following properties: $\sum_{k=1}^{K} \pi_k = 1$ and $\forall k \in \{1, 2, .., K\} : \pi_k \in [0, 1]$. The variables π are known as the *mixture proportions* and the cluster-specific distributions are called the *mixture components*. Variables $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ are the k^{th} cluster specific mean and covariance parameters, respectively ($k \in \{1, 2, .., K\}$). Figure 1 is an example of a GMM where $d = 1$ and $K = 3$.

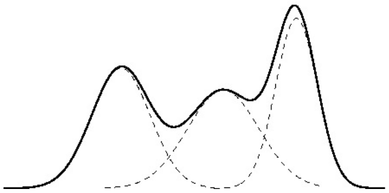


Figure 1. The density of a univariate Gaussian Mixture Model with three Gaussian mixture components, each with their own mean and variance terms ($K = 3$, $d = 1$). [Source: http://prateekvjoshi.com]

The likelihood for $\mathbf{x_i}$ conditioned on parameters is then:

$$\begin{aligned} p(\mathbf{x}_i \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\pi}) &= \sum_{k=1}^{K} p(x_i, z_i = k \mid \pi, \theta) \\ &= \sum_{k=1}^{K} p(z_i = k|\pi_k) p(x_i \mid z_i = k, \theta) \\ &= \sum_{k=1}^{K} \pi_k \mathcal{N}_d(\mathbf{x_i}; \boldsymbol{\mu}_\mathbf{k}, \boldsymbol{\Sigma}_\mathbf{k}), \end{aligned}$$

which is a weighted, linear sum of Gaussians. This gives a nice interpretation of π_k as the probabilistic 'weight' we place on each cluster k in our model.

13.1.2. Mixture Models for Clustering

Mixture models are often used for clustering; this is a *generative model* because we specifically model $p(z)$ and $p(x \mid z)$. For general parameters $\theta = \{\pi, \mu, \Sigma\}$, the posterior probability of assigning point x_i to cluster k is given by (using Bayes rule):

$$r_{ik} \triangleq p(z_i = k \mid \mathbf{x}_i, \theta) \propto p(\mathbf{x}_i \mid z_i = k, \theta)\ p(z_i = k \mid \pi).$$

Calculating the posterior probability of each cluster for a data point x_i is known as *soft clustering. Hard clustering* is assigning the best cluster z_i^* to data point x_i^* such that

$$z_i^* = \arg\max_k (r_{ik}) = \arg\max_k\ \log\left(p\left(\mathbf{x}_i \mid z_i = k, \theta\right)\right) + \log\left(p\left(z_i = k \mid \pi\right)\right).$$

Hard clustering induces a linear boundary between clusters that assigns points to a single cluster, whereas *soft clustering* computes the probability for each point that it was generated from each of the clusters.

13.1.3. Estimation Attempt for GMM

One possible approach to estimate all of our parameters and clusters is through the MLE process given we have observed $\mathbf{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}]$. Our parameters we wish to estimate are $\theta = \{\pi, \mu, \Sigma\}$. For notational simplicity we will write out $z_i = k$ as the vector $\mathbf{z}_i$, where $z_{ij} = \mathbf{1}(j = k)$, meaning that there is a single 1 in a vector of length K of all zeros that indicates the value that the multinomial z_i takes on. This is called the multinomial vector representation. So $\mathbf{z}_i^k$ is defined as the boolean value at the k^{th} position of the $\mathbf{z}_i$ vector. Using this:

$$p\left(\mathbf{z}_i \mid \pi\right) = \prod_{k=1}^{K} \pi_k^{\mathbf{z}_i^k}$$

$$p\left(\mathbf{x}_i \mid \mathbf{z}_i, \theta\right) = \prod_{k=1}^{K} \mathcal{N}\left(\mu_k, \Sigma_k\right)^{\mathbf{z}_i^k}$$

Thus, the log likelihood of the GMM is written as

$$\begin{aligned}
\ell(\theta; \mathbf{D}) &= \sum_{i=1}^{N} \log p(\mathbf{x}_i \mid \theta) = \sum_{i=1}^{N} \log\left[\sum_{\mathbf{z}_i=1}^{K} p(\mathbf{x}_i, \mathbf{z}_i \mid \theta)\right] \\
&= \sum_{i=1}^{N} \log\left[\sum_{\mathbf{z}_i \in Z} \prod_{k=1}^{K} \pi_k^{\mathbf{z_i^k}} \mathcal{N}\left(\mu_k, \Sigma_k\right)^{\mathbf{z}_i^k}\right]
\end{aligned}$$

note that $\sum_{\mathbf{z}_i \in Z}$ indicates the sum through all possible categorical values of $\mathbf{z}_i$

This does not decouple the likelihood because the log cannot be 'pushed' inside the summation; however, if we had observed each $\mathbf{z}_i$, then would this problem decouple? Let's say **we observe** each $\mathbf{z}_i$, so now $\mathbf{D} = \{(\mathbf{x}_1\mathbf{z}_1), \ldots, (\mathbf{x}_N\mathbf{z}_N)]$. The log likelihood now becomes:

$$\begin{aligned}
\ell(\theta; \mathbf{D}) &= \log \prod_{i=1}^{N} p\left(\mathbf{z}_i \mid \pi\right) p\left(\mathbf{x}_i \mid \mathbf{z}_i, \theta\right) \\
&= \sum_{i=1}^{N} \log p\left(\mathbf{z}_i \mid \pi\right) + \log p\left(\mathbf{x}_i \mid \mathbf{z}_i, \theta\right) \\
&= \sum_{i=1}^{N} \sum_{k=1}^{K} \left[\mathbf{z}_i^k \log \pi_k + \mathbf{z}_i^k \log \mathcal{N}\left(\mu_k, \Sigma_k\right)\right]
\end{aligned}$$

Our parameters estimations are now decoupled since we can estimate π_k and μ_k, Σ_k separately. In fact, we have a unimodal posterior distribution $p\left(\theta \mid \mathbf{D}\right)$ with respect to each of the parameters, so we say that we have **identifiable** parameters.

The issue of **unidentifiable** parameters comes up when we do not have a unique MLE or MAP estimates of θ, meaning our posterior has multiple modes. In other words, it is possible that multiple values of θ produce the same likelihood. In the case of unobserved $\mathbf{z}_i$'s for our GMM, we could not compute a unique MLE estimate of our parameters since the posterior depended on the unobserved $\mathbf{z}_i$'s.

Another problem to consider with mixture models is *label switching*: If we order the clusters A, B, C, and then run it again, it is possible we may get the clusters C, B, A, which would have the same likelihood as clusters A, B, C. This just needs to be considered when we compare different models or different runs from a given model.

Without observations of z, there is no ground truth, so we only have the feature distribution to guess the hidden causes from.

13.2. Expectation Maximization (EM)

The **EM Algorithm** is a general method for parameter estimation when the model depends on unobserved or latent variables, published in 1977 by Demster, Laird, and Rubin. The EM algorithm alternates estimating model parameters, starting from some initial guess, with estimating the values of the latent variables. Each iteration consists of an E step (Expectation step) and an M step (Maximization step). Let $X = \{x_1, \ldots, x_n\}$ be a set of observed variables and $Z = \{z_1, \ldots, z_n\}$ be a set of latent variables. Recall the marginal log likelihood:

$$\begin{aligned}\ell\left(\theta; Z, X\right) &= \sum_{i=1}^{N} \log p(x_i \mid \theta) \\ &= \sum_{i=1}^{N} \log\left[\sum_{z_i} p(x_i, z_i \mid \theta)\right].\end{aligned}$$

As discussed in Section 2.3, this is hard to optimize since we have the summation inside the log, which does not allow our latent variables and parameters to separate in this equation, causing multiple possible modes. Instead let's define the **complete data log likelihood**, or the likelihood of the data if we assume that we have complete observations (i.e., latent variables and observed variables both):

$$\ell_C\left(\theta; Z, X\right) = \sum_{i=1}^{N} \log p(x_i, z_i \mid \theta)$$

This is simple to work with since we do not have any summations inside the log, and it decouples nicely. But it still depends on the latent states which are unknown. To this end, we will see that the E step in EM to estimates realizations of the latent states. From the complete log likelihood, we can take the expectation of the latent variables with respect to the current values of our parameters:

$$\begin{aligned}Q\left(\theta^{(t)}\right) &= \mathbb{E}_{\theta^{(t)}}\left[\ell_C\left(\theta; Z, X\right) \mid \mathbf{D}, \theta^{(t-1)}\right] \\ &= E\left[\sum_{i=1}^{n}\sum_{k=1}^{K} z_i^k \log \pi_k + z_i^k \log \mathcal{N}(\mathbf{x_i}; \boldsymbol{\mu}_\mathbf{k}, \boldsymbol{\Sigma}_\mathbf{k})\right] \\ &= \sum_{i=1}^{n}\sum_{k=1}^{K} E[z_i^k] \log \pi_k + E[z_i^k] \log \mathcal{N}(\mathbf{x_i}; \boldsymbol{\mu}_\mathbf{k}, \boldsymbol{\Sigma}_\mathbf{k}).\end{aligned}$$

This equation Q as known as the *expected complete log likelihood* or the *auxiliary function.* This function denotes the *expected sufficient statistics* for our model. To solve $Q\left(\theta^{(t)}\right)$, the EM algorithm is composed of two main steps:

- E step: compute the expected sufficient statistics of $\mathbf{z}_i$:

  $$r_{ik}^t = \mathbb{E}_{\theta^t}\left[\mathbf{z}_i^k\right]$$

  In this Gaussian mixture model, we will compute $p\left(\mathbf{z}_i = k \mid \theta, \mathbf{x}_i\right)$
- M step: update θ^t to θ^{t+1} where

  $$\theta^{t+1} = \operatorname*{argmax}_{\theta}\left(Q(\theta^t)\right)$$

Instead of computing the ML, we could also compute the MAP. Then,

$$\theta^{(t+1)} = \underset{\theta}{\operatorname{argmax}} \Big(Q(\theta^{(t)}) + \log(p(\theta^{(t)})\Big)$$

Because these variables are separated, we know that, for a given set of expected values for the z parameters, the ECLL is concave with respect to each of our parameters. Thus, the M step involves maximizing the ECLL with respect to each of our variables by taking the derivative with respect to each of them, setting to zero, and solving, as in our standard MLE process.

13.2.1. EM for GMM

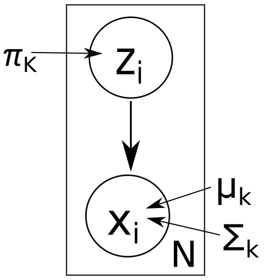


Figure 2. Graphical model of GMM

Here we will show an example of using EM for the GMM we have be discussing (Figure 5 and Section 2.3). Where the parameters we wish to estimate are $\theta = \{\pi_k, \mu_k, \Sigma_k\}$. Our expected complete data log likelihood decouples as:

$$\begin{aligned}
Q(\theta^t) &= \mathbb{E}_{\theta^t}(\ell_C(\theta; Z, X)) \\
&= \mathbb{E}(\sum_{i=1}^{n} \log p(\mathbf{x}_i, \mathbf{z}_i|\theta)) \\
&= \sum_{i=1}^{N} \mathbb{E}_{\theta^t}\left[\log p(z_i|\pi) p(\mathbf{x}_i \mid \mathbf{z}_i, \theta)\right] \\
&= \sum_{i=1}^{N}\sum_{k=1}^{K} \mathbb{E}_{\theta^t}\left[\mathbf{z}_i^k \log(\pi_k p_k(\mathbf{x}_i \mid \mu_k^t, \Sigma_k^t))\right] \\
&= \sum_{i=1}^{N}\sum_{k=1}^{K} \mathbb{E}_{\theta^t}\left[\mathbf{z}_i^k\right] \log(\pi_k p_k(\mathbf{x}_i \mid \mu_k^t, \Sigma_k^t)) \\
&= \sum_{i=1}^{N}\sum_{k=1}^{K} \mathbb{E}_{\theta^t}\left[\mathbf{z}_i^k\right] \log(\pi_k) + \mathbb{E}_{\theta^t}\left[\mathbf{z}_i^k\right] \log(p_k(\mathbf{x}_i \mid \mu_k^t, \Sigma_k^t)) \\
&= \sum_{i=1}^{N}\sum_{k=1}^{K} r_{ik}^t \log(\pi_k) + r_{ik}^t \log(p_k(\mathbf{x}_i \mid \mu_k^t, \Sigma_k^t))
\end{aligned}$$

Now for each iteration, t, we compute the E step and M step as follows.

- E step:

$$r_{ik}^{t+1} = p\left(\mathbf{z}_i = k \mid \theta, \mathbf{x}_i\right) = \frac{\pi_k^t \, \mathcal{N}(\mathbf{x}_i \mid \mu_k^t, \Sigma_k^t)}{\sum_{j=1}^K \pi_j^t \, \mathcal{N}(\mathbf{x}_i \mid \mu_j^t, \Sigma_j^t)}$$

  r_{ik}^{t+1} is the posterior probability of each cluster assignment to k on a specific data point.

- M step: After we have computed r_i^t, the ML updates are the same as in the Naive Bayes, where we update each parameter by the respective partial derivatives of $Q(\theta^t)$.

$$\begin{aligned}
\pi_k^{t+1} &= \frac{\sum_{i=1}^N r_{ik}^t}{N} \\
\mu_k^{t+1} &= \frac{\sum_{i=1}^N r_{ik}^t \mathbf{x}_i}{\sum_{i=1}^N r_{ik}^t} \\
\Sigma_k^{t+1} &= \frac{\sum_{i=1}^N r_{ik}^t \left(\mathbf{x}_i - \mu_k^{t+1}\right)\left(\mathbf{x}_i - \mu_k^{t+1}\right)^T}{\sum_{i=1}^N r_{ik}^t}
\end{aligned}$$

13.3. K-Means

The **K-means algorithm** is a simple clustering method that aims to partition n observations into K clusters using hard clustering, where each observation is assigned to the cluster with the nearest mean/centroid. As we will see, EM assigns probabilities (or weights) of observations belonging to each cluster. K-means, on the other hand, has no underlying probability model, and instead it assigns each observation to a specific cluster in a *hard clustering* manner. This is why we call the cluster means *centroids*: to emphasize that there is no underlying (Gaussian) probability model. The following steps implement the K-means algorithm.

(1) Set K and choose the initial centroids, η_k (often one can choose these from K data points).

(2) Assign each data point to its closest centroid:

$$z_{ik} = \begin{cases} 1, & \text{if } k = \arg\min_k \|x_i - \eta_k\|^2 \\ 0, & \text{otherwise} \end{cases} \tag{13.1}$$

(3) Update each cluster center by computing the mean of all points assigned to it

$$\eta_k = \frac{\sum_{i=1}^N z_{ik} x_i}{\sum_{i=1}^N z_{ik}}$$

(4) Repeat the second and third steps until cluster assignments do not change between iterations.

An inappropriate choice of K may result in poor results, so it is important to vary K and run diagnostic checks. The Euclidian distance (the term in step 2 defining the distance between point x_i and centroid η_k also need not be the metric minimized; other metrics such as the Mahalanobis distance may also be used. The distance metric may be customized for the specific space and feature set you are working with. Figure 4 illustrates provides a visualization of the K-means algorithm. In essence, K-means pretends we observe the latent states z and updates the cluster-specific centroids based on this (pretend) observation. The next question we should ask is how can we adapt K-means in a probabilistic framework? *We will see that the EM algorithm does this.*

As with GLMs, we can use a gradient-based approach to find the local minima in the likelihood (marginalizing out the latent variables z. Instead, we will follow the ideas in K-means in a probabilistic way.

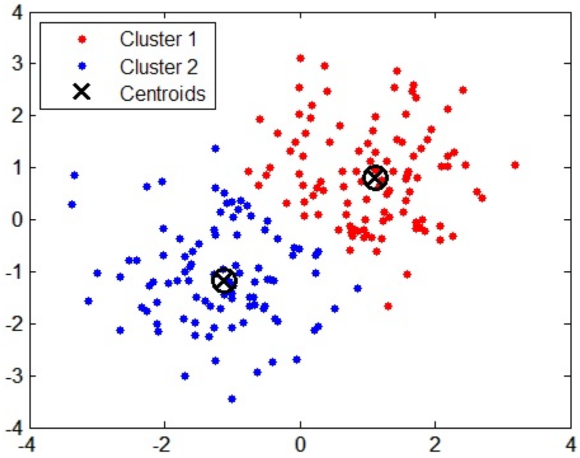


Figure 3. Example of K-means algorithm results. Source: www.mathworks.com.

LECTURE 14
Latent Dirichlet Allocation

The biological problem we look at is inference of population structure from genetic data. This is an important problem in population genetics/biology both to understand the genetic history of of populations as well to control for population structure when examining genome wide association studies (GWAS). It turns out this same model can be used to model documents, and in the text analysis setting is called topic modeling.

There are several statistical ideas used these ideas include mixture models, Gibb's sampling, and conjugate priors. The general form of a mixture model over multinomial data is called Latent Dirichlet Allocation.

Inference of population structure with no admixture

We first look the case where the observed individuals are not admixed. By this we mean that each individual is drawn from an allele distribution coming from one of $k = 1, ..., K$ ancestral populations. We will then look at the case with admixture where each individual's genome can come from a mixture of of the K ancestral populations.

The quantities that define the problem are

(a) $\{X_1, ..., X_n\}$ – The genotypes of the n individuals. These are n observed variables where for each individual we have $x_\ell^{(i,a)} \equiv (x_\ell^{(i,1)}, x_\ell^{(i,2)})$ = the genotype of the i-th individual at the ℓ-th locus where $i = 1, ..., n$ and $\ell = 1, ..., L$.

(b) $\{Z_1, ..., Z_n\}$ – The population of origin of the i-th individual, z^i = the population form which individual i originated where $z^i = \{1, 2, ...K\}$.

(c) $p_{k\ell j}$ = frequency of allele j at locus ℓ in population k where $j = 1, ..., J_\ell$ is the number of possible alleles at locus ℓ and $k = 1, ..., K$. Note that $p_{z(i)\ell j} = \Pr(x_\ell^{(i,a)} = j \mid Z, P)$

The genotypes X are observed.
The population of origin Z is hidden and must be inferred. The frequency variables P must also be inferred.

From the perspective of conditional probabilities we would like to compute the posterior distribution given a likelihood model for the genotypes (and priors on Z and P)

$$\Pr(Z, P \mid X) \propto \Pr(Z) \times \Pr(P) \times \text{Lik}(X; Z, P), \quad \text{Lik}(X; Z, P) \equiv \Pr(X \mid Z, P).$$

We now build up the problem from the simplest setting to the general setting with no admixture. Pretend that there is only one ancestral population $K = 1$ and that at a locus ℓ we have only two possibile alleles $J_\ell = 2$. The likelihood over n individuals at allele ℓ is

$$\text{Lik}(X_\ell^{1,a}, ..., X_\ell^{(n,a)}; p) \propto \prod_{i=1}^n p^{X_\ell^{(i,a)}}(1-p)^{1-X_\ell^{(i,a)}},$$

which is a binomial distribution. The generalization from $J_\ell = 2$ to $J_\ell > 2$ or $X_\ell^{(i,a)} = \{0, 1, 2, ..., J_\ell\}$ corresponds to moving from the binomial distribution to the multinomial distribution

$$\text{Lik}(X_\ell^{(1,a)}, ..., X_\ell^{(n,a)}; \{p_{\ell 1}, ..., p_{\ell J_\ell}\}) \propto \prod_{i=1}^n \left[\prod_{j=1}^{J_\ell} p_{\ell j}^{I(X_\ell^{(i,a)}, j)}\right] = \prod_{j=1}^{J_\ell} p_{\ell j}^{S_{\ell j}}$$

where $p_{\ell j}$ is the probability of the j-th allele at locus ℓ with $\sum_{j=1}^{J_\ell} p_{\ell j} = 1$, $p_{\ell j} \geq 0$, $S_{\ell j} = \#\{X_\ell^{(i,a)} = j\}$ is the number of individuals that have allele j at locus ℓ, and $I(X_\ell^{(i,a)}, j) = 1$ if $X_\ell^{(i,a)}$ is the j-th allele and 0 otherwise (this is called the indicator function).

The parameters $P = \{p_{\ell 1}, ..., p_{\ell J_\ell}\}$ are uncertain. These parameters also can be modeled using a probability distribution. We agin first look at the case where $J_\ell = 2$ the binomial case where we have one parameter p. A natural probability distribution to model p is the beta distribution with parameters $\alpha, \beta > 0$ with

$$f(p; \alpha, \beta) \propto p^{\alpha-1}(1-p)^{\beta-1}, \quad 0 \leq p \leq 1$$

the case of $\alpha = \beta = 1$ returns the uniform distribution. If we use the beta distribution to set our prior on p and use the binomial likelihood we obtain the following posterior distribution for p give our data

$$\begin{aligned}
\Pr(p \mid X_\ell^{(1,a)}, ..., X_\ell^{(n,a)}) &\propto \text{Lik}(X_\ell^{(1,a)}, ..., X_\ell^{(n,a)}; p) \times f(p; \alpha, \beta) \\
&= \left[\prod_{i=1}^n p^{X_\ell^{(i,a)}}(1-p)^{1-X_\ell^{(i,a)}}\right] p^{\alpha-1}(1-p)^{\beta-1} \\
&= \left[p^{S_\ell}(1-p)^{n-S_\ell}\right], \quad S_\ell = \#\{X_\ell^{1,a} = 1\} \\
&= p^{S_\ell+\alpha-1}(1-p)^{n+\beta-S_\ell-1} \\
&= \text{Beta}(S_\ell + \alpha, n - S_\ell + \beta),
\end{aligned}$$

so the posterior distribution is a beta. The beta and binomial are conjugate distributions. In the case of the multinomial the natural distribution on $\{p_{\ell 1}, ..., p_{\ell J_\ell}\}$ is given by a Dirichlet distribution

$$f(\{p_{\ell 1}, ..., p_{\ell J_\ell}\}; \{\alpha_1, ..., \alpha_{J_\ell}\}) \propto \prod_{j=1}^{J_\ell} p_{\ell j}^{\alpha_j - 1}.$$

Using the Dirichlet as the prior we can show that the posterior distribution of the parameters $(\{p_{\ell 1}, ..., p_{\ell J_\ell})$ given the genotype is also Dirichlet

$$\begin{aligned}
\Pr(p \mid X_\ell^{(1,a)}, ..., X_\ell^{(n,a)}) &\propto \text{Lik}(X_\ell^{(1,a)}, ..., X_\ell^{(n,a)}; \{p_{\ell 1}, ..., p_{\ell J_\ell}\}) \times f(\{p_{\ell 1}, ..., p_{\ell J_\ell}\}; \{\alpha_1, ..., \alpha_{J_\ell}\}) \\
&= \left[\prod_{j=1}^{J_\ell} p_{\ell j}^{S_{\ell j}}\right] \left[\prod_{j=1}^{J_\ell} p_{\ell j}^{\alpha_j - 1}\right] \\
&= \prod_{j=1}^{J_\ell} p_{\ell j}^{S_{\ell j} + \alpha_j - 1} \\
&= \text{Dir}(S_{\ell 1} + \alpha_1, S_{\ell 2} + \alpha_2, ..., S_{\ell J_\ell} + \alpha_{J_\ell}).
\end{aligned}$$

So at this point we know how to infer the posterior distribution of allele frequencies if we only had one ancestral population, it is given by the Dirichlet distrbution. Obviously this is not so interesting since this case defeats the point of inferring population structure.

We now extend to the case where $K > 2$ where we have real population structure. We introduce a latent variable $Z^{(i)}$ which assigns to each individual a population of origin. This adding a variable is sometimes called augmentation. If we knew $Z^i = k$ we could write out the posterior distribution of the allele frequencies $p_{k\ell j}$ which are the allele frequencies for alleles $j = 1, ...J_\ell$ at locus ℓ for group k with $S_{k\ell j} = \#\{X^{(i,a)} = j, z^{(i)} = k\}$

$$\begin{aligned}
\Pr(p_{k\ell 1}, ..., p_{k\ell J_\ell} \mid Z^{(i)} = k, X^{(1,a,)}, ..., X^{(n,a)}) &\propto \left[\prod_{j=1}^{J_\ell} p_{k\ell j}^{S_{k\ell j}}\right] \left[\prod_{j=1}^{J_\ell} p_j^{\alpha_j - 1}\right], \\
&= \text{Dir}(\alpha_1 + S_{k\ell 1}, ..., \alpha_{J_\ell} + S_{k\ell J_\ell}).
\end{aligned}$$

This gives us a way to sample from the posterior distribution $\Pr(P \mid Z, X)$.

We will show that we can also sample from $\Pr(Z \mid P, X)$. We can write using Bayes' rule

$$\Pr(Z^{(i)} = k \mid X, P) = \frac{\Pr(X^{(i)} \mid P, z^{(i)} = k)}{\sum_{k'} \Pr(X^{(i,a)} \mid P, z^{(i)} = k')},$$

where

$$\Pr(X^{(i,a)} \mid P, Z^{(i)} = k) = \prod_{\ell=1}^{L} p_{k\ell x^{(i,1)}} p_{k\ell x^{(i,2)}}.$$

At this point we know how to draw $\Pr(Z \mid P, X)$ and $\Pr(P \mid Z, X)$. The problem is we want to draw $\Pr(Z, P \mid X)$. There is a way of doing this in many cases using a procedure called Gibb's sampling. The idea behind Gibb's sampling is if I want to sample from a joint distribution $\Pr(Z, X)$ but can only compute the condiitionals $\Pr(Z \mid P)$ and $\Pr(P \mid Z)$ then I can use the following iterative procedure to sample the joint:

(1) Guess a $Z_{(0)}$
(2) For $t = 1$ to T
  (a) sample $P_{(t)} \mid Z_{(t-1)}$
  (b) sample $Z_{(t)} \mid P_{(t)}$
(3) Remove the first t_0 pairs of $(P_{(t)}, Z_{(t)})$, this is called burn-in
(4) Keep every a-th pair of the remaining $(P_{(t)}, Z_{(t)})$, this is called thinning
(5) We now have a iid draws from $\Pr(Z, P)$

The procedure in STRUCTURE adapts the above algorithm in the following way

(1) For $i = 1$ to n: $Z^{(i)}_{(0)} \overset{iid}{\sim} \text{Uni}(1, ..., K)$

(2) For $t = 1$ to T

  (a) For each k, ℓ

$$P^{(t)}_{k\ell\cdot} \mid X, Z_{(t-1)} \sim \text{Dir}(\lambda_1 + n_{k\ell 1}, ..., \lambda_{J_\ell} + n_{k\ell J_\ell}),$$

  where $n_{k\ell j} = \#\{(i,a) : X^{i,a}_\ell = k, Z^{(i)}_{(t-1)} = k\}$

  (b) For each i

$$\Pr\left(Z^{(i)}_{(t)} = k \mid X, P^{(t)}\right) = \frac{\Pr(X^{(i)} \mid P^{(t)}, z^{(i)} = k)}{\sum_{k'} \Pr(X^{(i,a)} \mid P^{(t)}, z^{(i)} = k')},$$

  where

$$\Pr\left(X^{(i,a)} \mid P^{(t)}, Z^{(i)} = k\right) = \prod_{\ell=1}^{L} p_{k\ell x^{(i,1)}} p_{k\ell x^{(i,2)}}.$$

Inference of population structure with admixture

In the case where there is admixture each individual does not necessarily come from one ancestral population. Their genome comes from a mixture of ancestral populations. This is admixture. To model this we need to introduce a new variable Q and adjust our previous variable Z. Our variable P and X are the same as before. The new or adjusted variables are:

(a) $\{Q_1, ..., Q_n\}$ – Vectors of admixture proportions for each individual with

$q^{(i)}_k$ = proportion of i-th individuals genome that originated in population k

(b) $\{Z\}$ – Allele copy $X^{(i,a)}_\ell$ originated in unknown population $Z^{(i,a)}_\ell$

$z^{(i,a)}_\ell$ = population of origin of allele copy $X^{(i,a)}_\ell$

note previously we only needed one Z for each individual.

We observe that

$$\Pr(X^{(i,a)}_\ell = j \mid Z, P, Q) = p_{z^{(i,a)}_\ell \ell j},$$

and

$$\Pr(z^{(i,a)} = k \mid P, Q) = q^{(i)}_k,$$

and we can place the prior

$$q^{(i)} \sim \text{Dir}(\alpha, ..., \alpha).$$

We will see soon we can write the following conditionals

$$P, Q \mid X, Z, \quad Z \mid X, P, Q.$$

This lets us write out the following Gibbs sampler.

(1) For each i, a: $Z^{(i,a)}_{(0)} \overset{iid}{\sim} \text{Uni}(1, ..., K)$

(2) For $t = 1$ to T

  (a) For each k, ℓ

$$P^{(t)}_{k\ell\cdot} \mid X, Z_{(t-1)} \sim \text{Dir}(\lambda_1 + n_{k\ell 1}, ..., \lambda_{J_\ell} + n_{k\ell J_\ell}),$$

  where $n_{k\ell j} = \#\{(i,a) : X^{i,a}_\ell = k, Z^{(i,a)}_{(t-1)} = k\}$

(b) For each i

$$q_{(t)}^{(i)} \mid X, Z_{(t-1)}^{(i,a)} \sim \mathrm{Dir}(\alpha + m_1^{(i)}, ..., \alpha + m_k^{(i)},$$

where

$$m_k^{(i)} = \#\{(\ell, a) : z_\ell^{(i,a)} = k\}.$$

(c) For each i, a, ℓ

$$\Pr\left(Z_{(t)}^{(i,a)} = k \mid X, P^{(t), Q_{(t)}}\right) = \frac{q_k^{(i)} \Pr(X_\ell^{(i)} \mid P^{(t)}, z^{(i)} = k)}{\sum_{k'} q_k'^{(i)} \Pr(X_\ell^{(i,a)} \mid P^{(t)}, z^{(i)} = k')},$$

where

$$\Pr\left(X_\ell^{(i,a)} \mid P^{(t)}, Z^{(i)} = k\right) = p_{k\ell x^{(i,a)}}.$$

LECTURE 15
Markov chain Monte Carlo

There are many settings when posterior computation is a challenge in that one does not have a closed form expression for the posterior distribution. Markov chain Monte Carlo methods are a general all purpose method for sampling from a posterior distribution. To explain MCMC we will need to present some general Markov chain theory. However, first we first justify Gibbs sampling, this can be done without the use of any Markov chain theory.

The basic problem is we would like to generate samples from

$$\pi(\theta) \equiv f(\theta \mid x) = \frac{f(x \mid \theta) f(\theta)}{f(x)} \equiv \frac{w(\theta)}{Z},$$

here the normalization constant $Z = \int f(x, \theta) d\theta$ is in general intractable. The objective of our MCMC algorithms will be to sample from $\pi(\theta)$ without ever having to compute Z. The computation of $w(\theta)$ is usually tractable since evaluating the likelihood and prior are typically analytic operations.

15.1. Gibbs sampler

The idea behind a Gibbs sampler is that one wants to sample from a joint posterior distribution. We do not have access to a closed form for the joint, however we do have an analytic form form for the conditionals. Consider a joint posterior $f(\theta \mid x)$ where x is the data and $\theta = \{\theta_1, \theta_2\}$. For ease of notation we will write $\pi(\theta_1, \theta_2) \equiv f(\theta_1, \theta_2 \mid x)$. The idea behind the Gibbs sampling algorithm is that the following procedure will provide samples from the joint distribution $\pi(\theta_1, \theta_2)$:

```
1) Set $\theta_2 \sim$ Unif[support of $\theta_2$]
2) Draw $\theta_1' \sim \pi(\theta_1' \mid \theta_2)$
3) Draw $\theta_2' \sim \pi(\theta_2' \mid \theta_1')$
4) Set $\theta_2 := \theta_2'$
5) Goto step 2.
```

We now show why the draws θ_1', θ_2' from the above algorithm are from $\pi(\theta_1, \theta_2)$. The following chain starts with the joint distribution specified by following the above algorithm and proceeds to show it is the joint distribution

$$\begin{aligned}
p(\theta_1', \theta_2') &= \int \pi(\theta_1, \theta_2)\pi(\theta_1' \mid \theta_2)\pi(\theta_2' \mid \theta_1') d\theta_1 d\theta_2 \\
&= \int \pi(\theta_1, \theta_2)\frac{\pi(\theta_1', \theta_2)}{\pi(\theta_2)}\frac{\pi(\theta_1', \theta_2')}{\pi(\theta_1')} d\theta_1 d\theta_2 \\
&= \int \pi(\theta_1 \mid \theta_2)\pi(\theta_2 \mid \theta_1')\pi(\theta_2', \theta_1') d\theta_1 d\theta_2 \\
&= \pi(\theta_1', \theta_2')\left[\int \pi(\theta_1 \mid \theta_2)\pi(\theta_2 \mid \theta_1') d\theta_1 d\theta_2\right] \\
&= \pi(\theta_1', \theta_2')\left[\int \pi(\theta_1, \theta_2 \mid \theta_1') d\theta_1 d\theta_2\right] \\
&= \pi(\theta_1', \theta_2').
\end{aligned}$$

One can also derive the Gibbs sampler from the more general Metropolis-Hastings algorithm. We will leave that as an exercise.

15.2. Markov chains

Before we discuss Markov chain Monte Carlo methods we first have to define some basic properties of Markov chains. The Markov chains we discuss will always be discrete time. The state space or values the chain can take at an time $t = 1, ..., T$ can be either discrete or continuous. We will denote the state space as $\mathcal{S}$ and for almost all examples we will consider a finite discrete state space $\mathcal{S} = \{s_1, ..., s_L\}$, this is so we can use linear algebra rather than operator theory for all our analysis. In the context of Bayesian inference it is natural to think of the state space $\mathcal{S}$ as the space of parameter values Θ and a state s_i corresponding to a parameter value θ_i.

For a discrete state Markov chain we can define a Makov transition matrix $\mathbf{P}$ where each element

$$\mathbf{P}_{s_t \to s_{t+1}} = \Pr(s_t \to s_{t+1}),$$

is the probability of moving from one state to another at any time t. We consider a probability vector ν as a vector of L numbers with $\sum_\ell \nu_\ell = 1$ and $\nu_\ell \geq 0$. We will require our chain to mix and have a unique stationary distribution. This requirement will be captured by two criteria: invariance and irreducibility or ergodicity.

We start with invariance: we would like the chain to have the following property

$$\lim_{T\to\infty} \mathbf{P}^T \nu = \nu^*, \quad \forall \nu$$

the limit ν^* is called the invariant measure or limiting distribution. The existence of a unique invariant distribution piles the following general balance condition

$$\sum_{s'} \mathbf{P}(s' \to s)\, \nu^*(s') = \nu^*(s).$$

There is a simple check for invariance given the transition matrix $\mathbf{P}$ by computing

$$\mathbf{P} = U^T \Lambda U,$$

where if we rank the eigenvalues λ_ℓ from largest to smallest we know the largest eigenvalue $\lambda_1 = 1$. We also know that all the eigenvalues cannot be less than -1.

So we now consider

$$\lim_{N\to\infty}\left[\mathbf{P}^N = \left(\sum_\ell \lambda_\ell^N u_\ell u_\ell^T\right)\right]$$

which will converge as long as no eigenvalues $\lambda_\ell = -1$. In addition, all eigenvalues $\lambda_\ell \in (-1,1)$ will not have an effect on the limit.

Ergodicity or irreducibility of the chain means the following:
There exists an N such that $\mathbf{P}^N(s' \to s) > 0$ for all s' and $\nu^*(s) > 0$.
Another way of stating the above is that the entire state space is reachable from any point in the state space. Again we can check for irreducibility using linear algebra. We first define the generator of the chain $L = \mathbf{P} - I$. We now look at the eigenvalues of L ordered from smallest to largest. We know the smallest eigenvalue $\lambda_1 = 0$ and has corresponding eigenvector $\mathbf{1}$. If the second eigenvalue $\lambda_2 > 0$ then the chain is irreducible and $\lambda_2 - \lambda_1 = \lambda_2$ is called the spectral gap.

For a Makov chain with a unique invariant measure that is ergodic the following mixing rate holds

$$\sup_\nu \|\nu^* - \mathbf{P}\nu\| = O((1-\lambda)^N).$$

We want our chains to mix.

Algorithmically we will design Markov chains that satisfy what is called detailed balance:

$$\mathbf{P}(s' \to s)\nu^*(s') = \mathbf{P}(s \to s')\nu^*(s), \quad \forall s, s'.$$

Detailed balance is easy to check for in an algorithm and detailed balance plus ergodicity implies that the chain mixes. In the next section we see why detailed balance is easy to verify for the most common MCMC algorithm Metropolis-Hastings.

15.3. Metropolis-Hastings algorithm

We begin with some notation we define a Markov transition probability or Markov transition kernel as

$$Q(s'; s) \equiv f(s' \mid s),$$

as a conditional probability of $s' \mid s$, in the case of a finite state space these values are given by a Markov transition matrix. We also have a state probabilities

$$p(s) \equiv \frac{w(s)}{Z},$$

where we can evaluate $w(s)$ using the prior and likelihood. Note that whenever we write $\frac{p(s)}{p(s')}$ we can use the computation $\frac{w(s)}{w(s')}$ as a replacement.

The following is the Metropolis-Hastings algorithm

```
1) $t = 1$
2) $s^{(t)} \sim$ Unif[support of $s$]
3) Draw $s' \sim Q(s'; s^{(t)})$
4) Compute acceptance probability $\alpha$
       $$\alpha = \min\left(1, \frac{p(s')Q(s; s')}{p(s)Q(s'; s)}\right)$$
5) Accept $s'$ with probability $\alpha$: $u \sim$ Unif[0, 1], If $u \leq \alpha$ then $\begin{cases} t = t+1 \\ s^{(t)} = s' \end{cases}$
6) If $t < T$ goto step 3 else stop
```

The Metropolis-Hastings algorithm is designed to generate $(s^{(1)}, ..., s^{(T)})$ samples from the posterior distribution $p(s)$. We will show soon that the algorithm satisfies detailed balance. Before that we will state a properties of the above algorithm. A common proposal $Q(s'; s)$ is a random walk proposal $s' \sim \mathrm{N}(s, \sigma^2)$. If σ^2 is very small then typically the acceptance ratio α will be near 1, however in this case two consecutive draws $s^{(t)}, s^{(t+1)}$ will be conditionally dependent. If σ^2 is very large then the acceptance ratio α will be near 0, however in this case two consecutive draws $s^{(t)}, s^{(t+1)}$ will be independent. There is a trick of how local/global the steps should be and what acceptance ratio α is good, some theory suggests $\alpha = .25$ is optimal. It is also the case that the first T_0 samples are not drawn form the stationary distribution, the stationary distribution has not kicked in yet. For this reason one typically does not include the first T_0 samples, this is called the burn-in period.

We now show detailed balance. First observe that $P(s \to s') = \alpha Q(s'; s)$. We start with

$$\begin{aligned}
P(s \to s')\nu^*(s) &= Q(s'; s) \min\left(1, \frac{\nu^*(s')Q(s; s')}{\nu^*(s)Q(s'; s)}\right) \nu^*(s) \\
&= \min\left(\nu^*(s)Q(s'; s), \nu^*(s')Q(s; s')\right) \\
&= Q(s; s') \min\left(1, \frac{\nu^*(s)Q(s'; s)}{\nu^*(s')Q(s; s')}\right) \nu^*(s') \\
&= P(s' \to s)\nu^*(s').
\end{aligned}$$

LECTURE 16
Hidden Markov Models

The idea of a hidden Markov model (HMM) is an extension of a Markov chain. The basic formalism is that we have two variables $X_1, ..., X_T$ which are observed and $Z_1, ..., Z_T$ which are hidden states and they have the following conditional dependence structure

$$\begin{aligned} x_{t+1} &= f(x_t; \theta_1) \\ z_{t+1} &= g(x_{t+1}; \theta_2), \end{aligned}$$

where we think of t as time and $f(\cdot)$ and $g(\cdot)$ are conditional distributions. In this case we think of time as discrete. Typically in HMMs we consider the hidden states to be discrete, there are more general state space models where both the hidden variables and the observables are continuous. The parameters of the conditional distribution $g(\cdot)$ is often called the transition probabilities and the parameters for observed distribution $g(x_{t+1}; \theta_2)$ are often called the emission probabilities. We will often use the notation $x_{1:t} \equiv x_1, ..., x_t$.

The questions normally asked using a HMM include:

- Filtering: Given the observations $x_1, ..., x_t$ we want to know the hidden states $z_1, ..., z_t$ so we want to infer – $p(z_{1:t} \mid x_{1:t})$.
- Smoothing: Given the observations $x_1, ..., x_T$ we want to know the hidden states $z_1, ..., z_t$ where $t < T$. Here we are using past and future observation to infer hidden states – $p(z_{1:t})$
- Posterior sampling: $z_{1:T} \sim p(z_{1:T} \mid x_{1:T})$

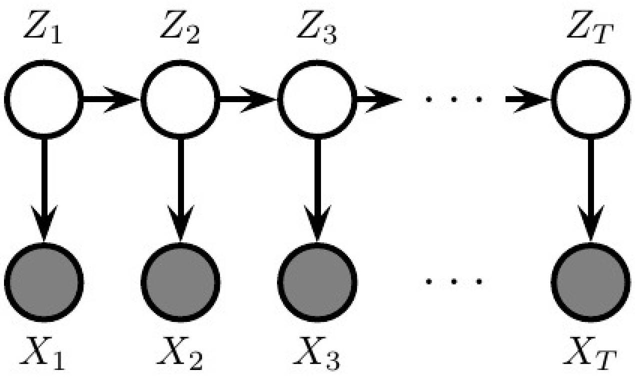
The hidden variables in an HMM are what make inference challenging. We start by writing down the joint (complete) likelihood

$$\text{Lik}(x_1, ..., x_T, z_1, ..., z_T; \theta_1, \theta_2) = \pi(z_1) \prod_{t=2}^{T} f(z_{t+1} \mid z_t, \theta_1) \prod_{t=1}^{T} g(x_t \mid z_t, \theta_2),$$

here $\pi(\cdot)$ is the probability of the initial state. One can obtain the likelihood of the observed data by marginalization

$$\text{Lik}(x_1, ..., x_T; \theta_1, \theta_2) = \sum_{z_1, \ldots, x_T} \left(\pi(z_1) \prod_{t=2}^{T} f(z_{t+1} \mid z_t, \theta_1) \prod_{t=1}^{T} g(x_t \mid z_t, \theta_2) \right).$$

Naively the above sum is brutal since it consists of all possible hidden trajectories. If we assume N hidden states then we would have N^T possible trajectories. We will see that the Markov structure will buy us a great deal in terms of reducing computations.



16.1. EM algorithm

We start with the complete log likelihood

$$\begin{aligned}
\ell_c(z,x;\theta) &= \log[\mathrm{Lik}(z,x \mid \theta)] \\
&= \log\left\{p(z_1)\left[\prod_{t=1}^{T} p(z_t \mid z_{t-1})\right]\left[\prod_{t=1}^{T} p(x_t \mid z_t)\right]\right\} \\
&= \log \pi(z_1) + \sum_{t=1}^{T-1} \log a_{z_t,z_{t+1}} + \sum_{t=1}^{T} \log p(x_t \mid z_t, \theta_2).
\end{aligned}$$

We then write the expected complete log likelihood

$$\begin{aligned}
\mathbb{E}\ell_c(z,x;\theta) &= \mathbb{E}\log[\mathrm{Lik}(z,x \mid \theta)] \\
&= \mathbb{E}\log\left\{p(z_1)\left[\prod_{t=1}^{T} p(z_t \mid z_{t-1})\right]\left[\prod_{t=1}^{T} p(x_t \mid z_t)\right]\right\} \\
&= \sum_{k=1}^{N} \mathbb{E}[z_1^k]\log \pi_k + \log \pi(z_1) + \sum_{t=1}^{T-1}\sum_{j,k=1}^{K} \mathbb{E}[z_t^j z_{t+1}^k]\log a_{jk} \\
&\quad + \sum_{t=1}^{T} \mathbb{E}[\log p(X_t \mid Z_t, \theta_2)],
\end{aligned}$$

where z_t^k indicates that at time t one is in the k-th state.

For the E step of theEM algorithm we will need to compute

$$\mathbb{E}[Z_1^k] = \mathbb{E}[Z_1^k \mid X_{1:T}, \theta] = p(Z_1^k = 1 \mid X_{1:T}, \theta)$$

This is what we expect since Z_1 follows a Multinomial distribution, so its expectation is simply the vector of posterior probabilities. We will also need to compute

$$\mathbb{E}[Z_t^j, Z_{t+1}^k] = \mathbb{E}[Z_t^j, Z_{t+1}^k \mid X_{1:T}, \theta] = \sum_{t=1}^{T-1} p(Z_t^j Z_{t+1}^k \mid X_{1:T}, \theta)$$

Note that intuitively, $\mathbb{E}[Z_t^j, Z_{t+1}^k]$ counts how often we see transition pairs.

We now state the forward-backward algorithm which is an efficient way of computing the expectations above. We would like to compute $p(z_1 \mid x_{1:T})$ so we start by writing

$$\begin{aligned}
p(z_t \mid x_{1:T}) &= \frac{p(z_t, x_{1:T})}{p(y_{1:T})} \\
p(z_t, x_{1:T}) &= p(x_{1:T} \mid z_t)p(z_t) \\
&= p(x_{1:t}, z_t)p(x_{t+1:T} \mid z_t) \\
&= \alpha(z_t)\beta(z_t),
\end{aligned}$$

where $\alpha(z_t)$ looks back and $\beta(z_t)$ looks forward. Both can be computed recursively.

For α:

$$\begin{aligned}
\alpha(z_t) &= p(x_{1:t}, z_t) \\
&= \sum_{z_{t-1}} p(x_{1:t}, z_t, z_{t-1}) \\
&= \sum_{z_{t-1}} p(x_{1:t-1}, z_{t-1})p(x_t, z_t \mid x_{1:t-1}, z_{t-1}) \\
&= \sum_{z_{t-1}} p(x_{1:t-1}, z_{t-1})p(z_t \mid z_{t-1})p(x_t \mid z_t) \\
&= \sum_{z_{t-1}} \alpha(z_{t-1})p(z_t \mid z_{t-1})p(x_t \mid z_t),
\end{aligned}$$

note that given parameter models the above is easy to compute since $p(x_t \mid z_t)$ is the emission probability and $p(z_t \mid z_{t-1})$ is the state transition probability. Note that we can initialize α as $\alpha(z_1) = p(x_1, z_1) = p(z_1)p(x_1 \mid z_1)$.

For β:

$$\begin{aligned}
\beta(z_t) &= p(x_{t+1:T} \mid z_t) \\
&= \sum_{z_{t+1}} p(x_{t+1:T}, z_{t+1} \mid z_t) \\
&= \sum_{z_{t+1}} p(x_{t+1:T} \mid z_{t+1}, z_t)p(z_{t+1} \mid z_t) \\
&= \sum_{z_{t+1}} p(x_{t+2:T} \mid z_{t+1})p(x_{t+1} \mid y_{t+1})p(z_{t+1} \mid z_t) \\
&= \sum_{z_{t+1}} \beta(z_{t+1})p(x_{t+1} \mid z_{t+1})p(z_{t+1} \mid z_t)
\end{aligned}$$

note that given parameter models the above is easy to compute since $p(x_{t+1} \mid z_{t+1})$ is the emission probability and $p(z_{t+1} \mid z_t)$ is the state transition probability. Note that we can initialize β as $\beta(z_{T-1}) = p(x_T \mid z_{T-1}) = \sum_{z_T} p(x_T \mid z_T)p(z_T \mid z_{T-1})$.

This results in an algorithm with two phases
forward phase: $\alpha(z_t) = p(x_t \mid z_t) \sum_{z_{t-1}} p(z_t \mid z_{t-1})\alpha(z_{t-1})$
backward phase: $\beta(z_t) = \sum_{z_{t+1}} p(x_{t+1} \mid z_{t+1})p(z_{t+1} \mid z_t)\beta(z_{t+1})$.
Also we observe

$$p(z_t \mid x_{1:T}) = \frac{p(z_1 \mid x_{1:T})}{p(x_{1:T})} \propto \alpha(z_t)\beta(z_t).$$

Recall in the E step we need to compute

$$\mathbb{E}[Z_1^k] = p(z_1^k \mid x_{1:T}) \propto \alpha(z_1)\beta(z_1),$$

and

$$\begin{aligned}
\mathbb{E}[Z_t^j Z_{t+1}^k] &= p(z_t^j z_{t+1}^k \mid x_{1:T}) \\
&\propto p(z_t^j z_{t+1}^k, x_{t+1:T}) \\
&\propto p(x_{t+2:T} \mid z_{t+1}^k) p(x_{t+1} \mid z_{t+1}^k) p(z_{t+1}^k \mid z_t^j) p(z_t^j \mid x_{1:t})) \\
&= \beta(z_{t+1}^k)\, p(x_{t+1} \mid z_{t+1}^k)\, p(z_{t+1}^k \mid z_t^j)\, \alpha(z_t^j).
\end{aligned}$$

The above equations provide our estimates of $\mathbb{E}[Z_1^k]$ and $\mathbb{E}[Z_t^j Z_{t+1}^k]$ give current model parameters and the α and β computations.

We now specify the M step. For notation, we set the parameters of the transition probabilities are denoted as $a_{jk} = p(z_t^j \mid z_{t+1}^k)$, the initial probabilities as π_i, the parameters of the emission probabilities which is again a multinomial as $\eta_{jk} = p(x_t^j \mid z_t^k)$. The complete log likelihood with the parameters can be stated as

$$\sum_{i=1}^N E[Z_1^i] \log \pi_i + \sum_{t=1}^T \sum_{i,j=1}^N E[Z_t^i Z_t^j] \log a_{ij} + \sum_{t=1}^T \sum_{i,j=1}^{N,O} \mathbb{E}[Z_t^i X_t^j] \log \eta_{ij},$$

we have assumed N hidden states and O observable states. For ease of notation we define the following terms $\hat{z}_t^i = E[Z_t^i]$, $\widehat{z}_t^{ij} = E[Z_t^i Z_t^j]$. We now write down the sufficient statistics

$$z_1^i, \quad m_{ij} = \sum_{t=1}^T \widehat{z}_t^{ij}, \quad n_{ij} = \sum_{t=1}^T \hat{z}_t^i x_t^j.$$

Given the sufficient statistics and the parameters we minimize the complete log likelihood subject to the constraints

$$\sum_i \pi_i = 1, \quad \sum_{j=1}^N a_{ij} = 1, \quad \sum_{i=1}^O n_{ij} = 1.$$

Using Lagrange multipliers we obtain

$$\begin{aligned}
\hat{\pi}_i &= z_1^i \\
\hat{a}_{ij} &= \frac{m_{ij}}{\sum_{k=1}^N m_{ik}} \\
\hat{\eta}_{ij} &= \frac{n_{ij}}{\sum_{k=1}^O n_{ik}}.
\end{aligned}$$

LECTURE 17
Spectral methods and manifold learning

A key idea in much of high-dimensional data analysis is that the underlying structure of the data is low dimensional so even if the $X \subseteq \mathbb{R}^p$ the underlying degrees-of-freedom or the support of the data is low dimensional say, $d \ll p$. The key questions are how to find this low dimensional structure, how to use the low dimensional structure, and what assumptions are made in extracting this low dimensional structure from data. The key tool we will used to address these questions are spectral methods.

17.1. Spectral methods in general

For the purpose of this lecture by spectral methods we will mean an eigen-decomposition of a positive semi-definite symmetric operator. We will construct different operators from data and these different operators will correspond to different assumptions about the data or different quantities we want to preserve about the data.