Bayesian Statistics

Muchang Bahng

Spring 2025

Contents

1 Probability Densities

1.1 Notation: Probability Densities & Sampling

A d-dimensional random variable X is any stochastic d-vector that can be "generated" or "realized" from an **outcome space** Ω. That is, the random variable X would randomly pick an element in Ω. The uncertainty of these possible values generated by the random variables is specified by some distribution Dist with some parameter θ.

$$X \sim \text{Dist}(\theta) \tag{1}$$

In this case, Dist is called a d-dimensional distribution. The probability density function of the random variable X can be written in many forms and may define different things, depending on context. Generally, if we do not include the parameter θ in the density expression, then we assume that it is fixed.

1. $\text{Dist}(x \mid \theta)$ or $\text{Dist}(x; \theta)$ tells us the probability of the random variable (following distribution Dist) will generate value x, given some fixed θ. Note that this notation allows us to write densities without having to explicitly name a random variable.
2. If we have defined the distribution of the random variable X, we generally treat $p_X(x) = p_X(x; \theta) = \text{Dist}(x \mid \theta)$.
3. Sometimes, we replace the p with an f, and call $f_X(x) = f(x; \theta) = \text{Dist}(x \mid \theta)$.

From a distribution X, we can take n samples, which we will denote

$$\mathbf{x} = \{x^{(i)}\}_{i=1}^n = \{x^{(1)}, x^{(2)}, \dots, x^{(n)}\} \tag{2}$$

with each $x^{(i)} \in \Omega$. This set is often called an **observation**, or **data**. Note that the space Ω can be discrete or continuous, and the density expression accounts for cases.

1. If Ω is discrete, then we can assume that X generates discrete $x^{(i)} \in \Omega \subset \mathbb{N}^d$. In the discrete case, the *sum* of all probabilities equals 1.
$$\sum_{x \in \Omega} x \, p_X(x) = 1 \tag{3}$$
2. If Ω is continuous, then we assume that X generates real-valued $x^{(i)} \in \Omega \subset \mathbb{R}^d$. In the continuous case, the *integral* of all probabilities equals 1.
$$\int_{x \in \Omega} x \, p_X(x) = 1 \tag{4}$$

1.2 Bayes' Rule

We have seen that Bayesian statistics depends on having some initial belief about an event. Upon some observation, we can gain some sort of information about the event, allowing us to *modify* our prior distribution to a new one, called the posterior distribution. This simple property is the reason why Bayesian statistics is so useful for machine learning. The way we do this is through **Bayes' Rule**, which states

$$p(H \mid D) = \frac{p(D \mid H) \, p(H)}{p(D)} \tag{5}$$

Note that:

1. H is the **hypothesis** whose probability may be affected by **data** D, also called **evidence**.
2. $p(H)$ is the **prior distribution**, our initial hypothesis of what the distribution would have been.
3. $p(H \mid D)$ is the **posterior distribution**, which was determined upon observing the event B.

4. $p(D \mid H)$ is the **likelihood**. If you were to assume that A is true, then the likelihood tells you the probability of getting result B.
5. $p(D)$ is the **marginal likelihood**, which is calculated by conditioning on A

$$p(D) = \sum_H p(D \mid H)\, p(H) \text{ or } p(D) \int_H p(D \mid H)\, p(H)\, dH \tag{6}$$

When computing our prior, the outcomes H are the **hypotheses**. We can assume that hypotheses are mutually exclusive and exhaustive (if one of these is true, it can't be some undefined third option). These assumptions are reasonable since it is almost always possible to redefine an arbitrary set of hypotheses into a set of hypotheses that *are* mutually exclusive and exhaustive.

There are multiple ways to write Bayes rule. When attempting to calculate the posterior, we can see that $p(D)$ is really just a normalization constant and therefore does not affect the type of distribution the posterior is. So, we can in effect write the above as

$$p(H \mid D) \propto p(D \mid H)\, p(H) \tag{7}$$

or

$$\text{Posterior} \;\propto\; \text{Prior} \;\times\; \text{Likelihood} \tag{8}$$

where the $\propto$ symbol means "proportional to." We use this notation more often when calculating posteriors since the normalizing constant isn't as important as finding the shape of the posterior density.

2 Inference: Parameter Estimation

Descriptive statistics is a summary statistic that quantitatively describes or summarizes features from a collection of samples $\{x^{(i)}\}$. It is extremely useful, but quite boring. However, inferential statistics is a different story. Given a set of samples $\{x^{(i)}\}_{i=1}^n$, we may have to try to predict/infer either which distribution X these samples came from, or if we know the distribution, what its parameters θ are. This is called an *inference problem*, and we approach it by constructing and refining a **statistical model** that we assume the data has been generated from. Assuming that we know what distribution (but not the parameter θ) the $x^{(i)}$'s come from, we can do 2 things:

- **Frequentist inference** tells us to find the likelihood function

$$L(\theta) = p(\mathbf{x} \mid \theta) \tag{9}$$

  which is a function of θ. The function L tells us that given that we know θ, what the probability of sampling $\mathbf{x}$ is. Clearly the value of θ that maximizes L represents the best statistical model.

- **Bayesian inference** tells us to find the desired posterior distribution $p(\theta \mid \mathbf{x})$ by assuming a reasonable prior, determining the likelihood, and multiplying them together using Bayes rule.

$$p(\theta \mid \mathbf{x}) \propto p(\theta)\, p(\mathbf{x} \mid \theta) = f(\theta) \tag{10}$$

  Finding the maximum of this function $f(\theta)$ that is proportional to $p(\theta \mid \mathbf{x})$ with respect to θ is the best statistical model. But unlike the frequentist approach, we have an entire distribution to work with. It tells us that given this data $\mathbf{x}$, what is the probability that the parameter value of the statistical model is θ, for all θ.

Throughout this section, we will show how parameter estimation problems are approached, often comparing both the frequentist and Bayesian approach.

2.1 Computing Posteriors with Beta Prior and Binomial Likelihood

The motivation behind the Beta distribution is that it satisfies **conjugacy** with a binomial likelihood. That is, assume that we have some data $\mathbf{x}$ of N observations containing m successes and $N-m$ failures (note that this observation $\mathbf{x}$ was in a way "reduced" to the information of only the number of successes m). We assume that there is some true success rate θ (between 0 and 1, of course) coming from these samples, and our job is to try and guess the true rate to the best of our abilities.

Before we even observe the data $\mathbf{x}$, our initial guess of θ might be modeled by the prior distribution $\theta \sim \text{Beta}(a,b)$. Furthermore, the likelihood is clearly a binomial (since it represents the probability of getting m successes out of N samples with fixed rate of success θ), so $m \mid \theta \sim \text{Binomial}(N, \theta)$. With these conditions, we claim that the posterior is also a beta, since

$$\begin{aligned} p(\theta \mid m) &\propto p_\theta(\theta)\, p(m \mid \theta) \\ &\propto \theta^{a-1}(1-\theta)^{b-1} \cdot \theta^m (1-\theta)^{N-m} \\ &= \theta^{a+m-1}(1-\theta)^{b+N-m-1} \end{aligned}$$

2.2 Bayesian Inference for Gaussian

The maximum likelihood framework gave point estimates for the parameters μ and Σ. Now we develop a Bayesian treatment by introducing prior distributions over these parameters. Given a set of N D-dimensional observations $\mathbf{X} = \{x_1, \dots, x_n\}$, the likelihood function is given by (the unnormalized function of μ):

$$p(\mathbf{X} \mid \mu, \Sigma) = \prod_{n=1}^{N} p(x_n \mid \mu, \Sigma) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp \sum_{n=1}^{N} \left(-\frac{1}{2}(x_n - \mu)^T \Sigma^{-1} (x_n - \mu) \right) \tag{11}$$

The likelihood function takes the form of the exponential of a quadratic form in μ. Thus, if we choose a prior $p(\mu)$ given by a Gaussian, it will be a conjugate distribution for this likelihood function. Taking our prior distribution to be

$$p(\mu, \Sigma) = \mathcal{N}(\mu, \Sigma \mid \mu_0, \Sigma_0) \tag{12}$$

The similarity of the symbols μ, Σ with μ_0, Σ_0 may be slightly confusing. We can think as such: μ, Σ are random variables that determine the parameters of some Gaussian distribution. But the values μ, Σ are uncertain, and their possible values with probabilities take the form of another distribution $\mathcal{N}(\mu_0, \Sigma_0)$. The posterior distribution is given by the familiar formula

$$p(\mu, \Sigma \mid \mathbf{X}) \propto p(\mathbf{X} \mid \mu, \Sigma)\, p(\mu, \Sigma) \tag{13}$$

which is another Gaussian $p(\mu \mid \mathbf{X}) = \mathcal{N}(\mu, \Sigma \mid \mu_N, \Sigma_N)$. Let us place a few conditions for simplification. Since every Gaussian density can be represented as a product of independent univariate Gaussians, we can work with univariate Gaussians. Furthermore, let us assume that the true value of σ is known, so all we have to do is find the posterior distribution of μ using the prior density $\mathcal{N}(\mu \mid \mu_0, \sigma_0^2)$. We have our prior and likelihood to be the following. Note that while the likelihood distribution is pretty much given, we have the flexibility to choose what our prior distribution is. We have only set the prior as a Gaussian simply because it is a conjugate form and therefore will greatly simplify calculations.

$$\begin{aligned} p(\mu) = \mathcal{N}(\mu \mid \mu_0, \sigma_0^2) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(\mu - \mu_0)^2) \right) \\ p(\mathbf{X} \mid \mu) = \prod_{n=1}^{N} p(x_n \mid \mu) &= \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{n=1}^{N} (x_n - \mu)^2 \right) \end{aligned}$$

which gives a posterior $p(\mu \mid \mathbf{X}) = \mathcal{N}(\mu \mid \mu_N, \sigma_N^2)$ where

$$\mu_N = \frac{\sigma^2}{N\sigma_0^2 + \sigma^2}\mu_0 + \frac{N\sigma_0^2}{N\sigma_0^2 + \sigma^2}\mu_{ML}$$
$$\frac{1}{\sigma_N^2} = \frac{1}{\sigma_0^2} + \frac{N}{\sigma^2}$$

and μ_{ML} is the maximum likelihood solution for μ given by the sample mean $\mu_{ML} = \frac{1}{N}\sum_{n=1}^N x_n$. These values make sense. We can see that the mean of the posterior distribution μ_N is a compromise between the prior mean μ_0 and maximum likelihood solution μ_{ML}. If the number of observed data points $N = 0$, then it is simply the prior mean, but for $N \to \infty$, the posterior mean is given by the maximum likelihood solution since the data "overpowers" the prior mean assumption.

Now, suppose that the mean of the Gaussian over the data is known and we wish to infer the variance. For convenience, let us work with the precision $\lambda = \frac{1}{\sigma^2}$ over the variance. The likelihood function for λ is

$$p(\mathbf{X} \mid \lambda) = \prod_{n=1}^N \mathcal{N}(x_n \mid \mu, \lambda^{-1}) \propto \lambda^{N/2} \exp\bigg(-\frac{\lambda}{2}\sum_{n=1}^N (x_n - \mu)^2 \bigg) \tag{14}$$

Note that since this is a function of λ, it behaves differently than the likelihood function of μ, even though they are of the same form. Since the likelihood function is proportional to the product of a power of λ and the exponential of a linear function of λ, we must find a prior distribution $p(\lambda)$ with precisely these proportional properties identical to that of the likelihood. Fortunately, the Gamma distribution satisfies them, defined by

$$p(\lambda \mid a_0, b_0) = \text{Gamma}(\lambda \mid a_0, b_0) = \frac{1}{\Gamma(a_0)} b_0^{a_0} \lambda^{a_0 - 1} \exp(-b_0 \lambda) \tag{15}$$

Using Bayes rule and multiplying gives the posterior density

$$p(\lambda \mid \mathbf{X}) \propto \lambda^{a_0 - 1} \lambda^{N/2} \exp\bigg(-b_0\lambda - \frac{\lambda}{2}\sum_{n=1}^N (x_n - \mu)^2 \bigg) \tag{16}$$

which is indeed the density of a $\text{Gamma}(\lambda \mid a_N, b_N)$ distribution, where

$$a_N = a_0 + \frac{N}{2}$$
$$b_N = b_0 + \frac{1}{2}\sum_{n=1}^N (x_n - \mu)^2 = b_0 + \frac{N}{2}\sigma_{ML}^2$$

where σ_{ML}^2 is the maximum likelihood estimator of the variance. Now, suppose that both the mean and precision are unknown. To find a conjugate prior, we consider the dependence of the likelihood function on μ and λ.

$$\begin{aligned} p(\mathbf{X} \mid \mu, \lambda) &= \prod_{n=1}^N \left(\frac{\lambda}{2\pi}\right)^{1/2} \exp\bigg(-\frac{\lambda}{2}(x_n - \mu)^2 \bigg) \\ &\propto \bigg(\lambda^{1/2} \exp\Big(-\frac{\lambda\mu^2}{2} \Big) \bigg)^N \exp\bigg(\lambda\mu \sum_{n=1}^N x_n - \frac{\lambda}{2}\sum_{n=1}^N x_n^2 \bigg) \end{aligned}$$

We now wish to identify a prior distribution $p(\mu, \lambda)$ that has the same functional dependence on μ and λ as the likelihood function and that should therefore take the form

$$\begin{aligned} p(\mu, \lambda) &\propto \left(\lambda^{1/2} \exp\Big(-\frac{\lambda\mu^2}{2}\Big)\right)^{\beta} \exp\big(c\lambda\mu - d\lambda\big) \\ &= \exp\left(-\frac{\beta\lambda}{2}\Big(\mu - \frac{c}{\beta}\Big)^2\right) \lambda^{\beta/2} \exp\left(-\Big(d - \frac{c^2}{2\beta}\Big)\lambda\right) \end{aligned}$$

where c, d, β are constants. Since we can always write $p(\mu, \lambda) = p(\mu \,|\, \lambda)p(\lambda)$, we can find $p(\mu \,|\, \lambda)$ and $p(\lambda)$ by inspection. We have just shown that $p(\mu \,|\, \lambda)$ is a Gaussian whose precision is a linear function of λ and that $p(\lambda)$ is a gamma distribution, so the normalized prior takes the form

$$p(\mu, \lambda) = \mathcal{N}(\mu \,|\, \mu_0, (\beta\lambda)^{-1}) \,\text{Gamma}(\lambda \,|\, a, b) \tag{17}$$

which is called the **Gaussian-Gamma distribution**. Note that this is not simply the product of an independent Gaussian prior over μ and a gamma prior over λ, because the precision of μ is a linear function of λ. The extension of this to multivariate random variables is straightforward.

2.3 Inference over Periodic Distributions

Although Gaussian distributions are of great significance, there are situations in which they are inappropriate as density models for continuous variables (e.g. wind direction or quantities periodic over 24 hours). Such quantities are conveniently represented using an angular (polar) coordinate $0 \leq \theta < 2\pi$. Let us consider the problem of evaluating the mean of a set of observations $\theta = \{\theta_1, \theta_2, \ldots, \theta_N\}$ of a periodic variable measured in radians. The simple average $(\theta_1 + \ldots + \theta_N)/N$ is strongly coordinate dependent. To find an invariant measure of the mean, we can see that the observations can be viewed as points on the unit circle and can therefore be described instead by two-dimensional unit vectors $x_1, \ldots, x_N$, where $x_n = (\cos\theta_n, \sin\theta_n)$. We can average these vectors and compute its angle to find this average angle.

$$\overline{x} = \frac{1}{N}\sum_{n=1}^{N} x_n = \Big(\frac{1}{N}\sum_{n=1}^{N}\cos\theta_n, \frac{1}{N}\sum_{n=1}^{N}\sin\theta_n\Big) \implies \overline{\theta} = \tan^{-1}\left(\frac{\sum_{n=1}^{N}\sin\theta_n}{\sum_{n=1}^{N}\cos\theta_n}\right) \tag{18}$$

In general, any distribution $p(\theta)$ that have period 2π must be defined such that it is nonnegative, integrate to 1, and be periodic.

$$\begin{aligned} p(\theta) &\geq 0 \\ \int_0^{2\pi} p(\theta)\, d\theta &= 1 \\ p(\theta + 2\pi) &= p(\theta) \end{aligned}$$

We can obtain a Gaussian-like distribution that satisfies these three properties. Consider a 2-dimensional Gaussian over variables x_1, x_2 having mean $\mu = (\mu_1, \mu_2)$ and a covariance matrix $\Sigma = \sigma^2 I$. This gives us

$$p(x_1, x_2) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{(x_1 - \mu_1)^2 + (x_2 - \mu_2)^2}{2\sigma^2}\right) \tag{19}$$

Now, suppose that we consider the value of this distribution along a circle of fixed radius. Then, this distribution will be periodic, although it will not be normalized. We can determine the form of this distribution by transforming from Cartesian coordinates to polar coordinates (r, θ) (so that $x_1 = r\cos\theta, x_2 = r\sin\theta$) and keeping r constant.

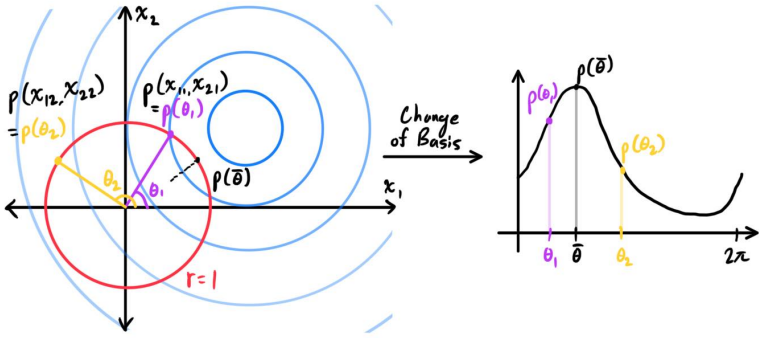

Figure 1: Circular normal change of basis

This transformation from $\mathbb{R}^2 \longrightarrow [0, 2\pi)$ defined

$$(x_1, x_2) \mapsto \tan^{-1} \frac{y}{x} \tag{20}$$

simply takes the "circular" cross section of the Gaussian and maps those values.

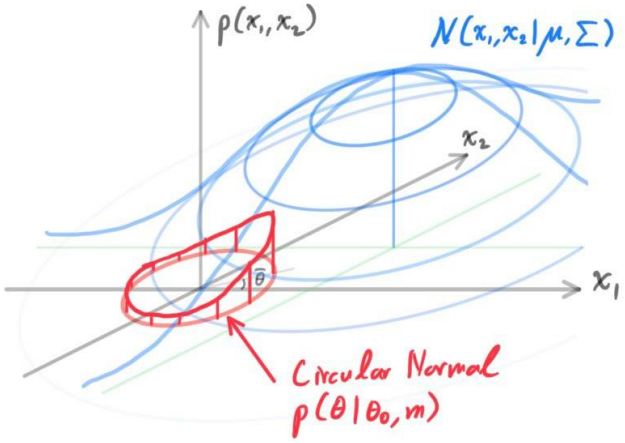

Figure 2: Circular cross section visualization

The value of r is not important so we assume $r = 1$. With some algebra and trig identities, we have the **circular normal**, or **von Mises distribution**, of form

$$p(\theta \,|\, \theta_0, m) = \frac{1}{2\pi I_0(m)} \exp\big(m \cos(\theta - \theta_0)\big) \tag{21}$$

where the parameter θ_0 corresponds to the mean of the distribution while m is analogous to the precision for the Gaussian. The normalization coefficient $I_0(m)$ is the zeroth-order Bessel function of the first kind, defined by

$$I_0(m) = \frac{1}{2\pi} \int_0^{2\pi} \exp\big(m \cos\theta\big)\, d\theta \tag{22}$$

For large m, the distribution becomes approximately Gaussian. Considering the maximum likelihood esti-

mators for the parameters θ_0 and m for the circular normal, the log likelihood function is given by

$$\ln p(\theta \mid \theta_0, m) = -N \ln(2\pi) - N \ln \big(I_0(m)\big) + m \sum_{n=1}^{N} \cos(\theta_n - \theta_0) \tag{23}$$

The maximum estimator for the mean is

$$\theta_0^{ML} = \tan^{-1} \left(\frac{\sum_n \sin \theta_n}{\sum_n \cos \theta_n} \right) \tag{24}$$

while that of m can be evaluated numerically.

2.4 Exponential Family of Distributions

The probability distributions so far are contained within the **exponential family** of distributions, which have important properties in common. The exponential family of distributions over $x \in \Omega \subset \mathbb{R}^D$, given parameters η, is defined to be the set of distributions of the form

$$p(x \mid \eta) = h(x) g(\eta) \exp \big(\eta^T u(x)\big) \tag{25}$$

where x may be a scalar or vector, discrete or continuous. Here, η are called the **natural parameters** of the distribution, and $u(x)$ is some function of x. The function $g(\eta)$ can be interpreted as the normalizing coefficient and therefore satisfies

$$g(\eta) \int_{x \in \Omega} h(x) \exp \big(\eta^T u(x)\big) \, dx = 1 \tag{26}$$

with the integration replaced by a summation if x is discrete.

Now, consider a set of iid data denoted by $\mathbf{X} = \{x_1, \dots, x_n\}$, for which the likelihood function is given by

$$p(X \mid \eta) = \left(\prod_{n=1}^{N} h(x_n) \right) g(\eta)^N \exp \left(\eta^T \sum_{n=1}^{N} u(x_n) \right) \tag{27}$$

Setting the gradient of $\ln p(\mathbf{X} \mid \eta)$ with respect to η to 0, we can the following condition to be satisfied by the maximum likelihood estimator η_{ML}:

$$-\nabla \ln g(\eta_{ML}) = \frac{1}{N} \sum_{n=1}^{N} u(x_n) \tag{28}$$

which can in principle be solved to obtain η_{ML}. The solution for the maximum likelihood estimator depends on the data only through $\sum_n u(x_n)$, which is therefore called the sufficient statistic of this distribution. Therefore, we do not need to store the entire data set itself but its sufficient statistic.

In general, for a given probability distribution $p(\mathbf{X} \mid \eta)$, we can seek a prior that is conjugate to the likelihood function, so that the posterior distribution has the same functional form as the prior. Given that the likelihood function is in the exponential family, there exists a conjugate prior that can be written in the form

$$p(\eta) = p(\eta \mid \chi, \nu) = f(\chi, \nu) g(\eta)^\nu \exp \big(\nu \eta^T \chi\big) \tag{29}$$

where $f(\chi, \nu)$ is a normalization coefficient, and $g(\eta)$ is the same function as the one appearing in the exponential family form of likelihood function. Indeed, multiplying this conjugate with the exponential family likelihood gives

$$p(\eta \mid \mathbf{X}, \chi, \nu) \propto g(\eta)^{\nu + N} \exp \left(\eta^T \left(\sum_{n=1}^{N} u(x_n) + \nu \chi \right) \right) \tag{30}$$

3 Linear Regression

3.1 Bayesian Regression: Modeling with Hierarchical Priors

Given a training data set $\mathcal{D} = (\mathbf{X}, \mathbf{Y})$ comprised of N pairs of observations with corresponding target variables $\{(x_i, y_i)\}_{i=1}^N$ ($x_i \in \mathbb{R}^D, y_i \in \mathbb{R}$), the goal is to predict the value of y for a new value of x. We first construct a *statistical model* (more explained in next next subsection) by assuming that there exists some function $f(x)$ of some form such that the y_i's have been generated by inputting the x_i's into f, followed by a random residual term. We assume that the data $\mathcal{D}$ has been sampled independently, but this may not always be a justifiable assumption in practice. Under this model, which we denote $\mathcal{M}_i$, we further assume that f can be parameterized by a vector θ, so therefore, we assume that

$$y = f(x, \theta) + \epsilon, \qquad \epsilon \sim \text{Residual}(\beta) \tag{31}$$

where β is some collection of parameters that determine the error function.

- The frequentist perspective reduces this problem to finding the value of θ that maximizes the likelihood. That is, we must find

$$\theta^* = \arg\max_\theta p(\mathcal{D} \mid \theta) = \arg\max_\theta \prod_{i=1}^N p(y_i \mid x_i, \theta) \tag{32}$$

  and claiming that $y = f(x, \theta^*)$ is the function of best fit. This is a quite straightforward (hopefully convex) optimization problem, which can be done in many ways (e.g. batch/sequential gradient descent, solving normal equations, etc.).

- The Bayesian approach attempts to construct a *distribution* of the values of θ. Clearly, this vector θ would be an element in some multidimensional Euclidean space, and we want to define a posterior density $p(\theta \mid \mathcal{D})$ across this space that tells us the probability of θ. Using Bayes rule,

$$p(\theta \mid \mathcal{D}) \propto p(\mathcal{D} \mid \theta)\, p(\theta) \tag{33}$$

  we see that we must define some prior distribution $p(\theta)$ on θ. We can assume that this prior is defined with some distribution

$$\theta \sim \text{Dist}_\theta(\gamma) \tag{34}$$

  where γ is a collection of parameters on θ. Knowing this prior of θ will allow us to get the posterior of $\theta \mid \mathcal{D}$. The not-so-complete Bayesian treatment would treat this γ as a known constant. But note that there is still uncertainty of whether θ comes from $\text{Dist}_\theta(\gamma)$ for one value of γ, compared to another value of γ. This uncertainty requires us to treat γ as now a **hyperparameter**, that is a parameter for the distribution of a parameter, and this distribution of γ, which we can denote

$$\gamma \sim \text{Dist}_\gamma(\xi) \tag{35}$$

  is called a **hyperprior**. We can construct higher and higher level hyperpriors on top of this as much as we want, which will lead to more flexibility in our model (but more computationally expensive). This is known as **hierarchical priors**. Generally, we will only go up to the level of one hyperparameter.

3.2 Computing the Posterior Parameter Distribution by Initially Marginalizing over Hyperparameters

Let us summarize how we would conduct the Bayesian method step by step. We first have to determine how many levels of hierarchical priors we are accounting for. Say that we will treat ξ as a constant, and consider the parameter θ along with its hyperparameter γ. Our goal is to compute the posterior $p(\theta \mid \mathcal{D})$.

1. Since there is uncertainty over the value of θ depending on γ, we can marginalize over γ to get

$$p(\theta \mid \mathcal{D}) = \int p(\theta \mid \mathcal{D}, \gamma)\, p(\gamma \mid \mathcal{D})\, d\gamma \tag{36}$$

If the situation calls for it, we could also compute the posterior by doing Bayes rule first to get $p(\theta \mid \mathcal{D}) \propto p(\mathcal{D} \mid \theta)\ p(\theta)$, but then we would have to calculate both $p(\mathcal{D} \mid \theta)$ and $p(\theta)$ by marginalizing each over γ, which would lead to complications.

2. To calculate $p(\theta \mid \mathcal{D}, \gamma)$, note that the formula for the posterior density of θ given $\mathcal{D}$ is $p(\theta \mid \mathcal{D}) \propto p(\mathcal{D} \mid \theta)p(\theta)$, where $p(\theta)$ is a density function of θ and parameter γ, which means that $p(\theta \mid \mathcal{D})$ would be a density function of θ and parameter γ. But since γ is fixed, the posterior

$$p(\theta \mid \mathcal{D}, \gamma) \propto p(\mathcal{D} \mid \theta, \gamma)p(\theta \mid \gamma) \tag{37}$$

is a density function of θ with fixed constant γ. This can be easily calculated because the prior $p(\theta \mid \gamma)$ is of distribution $\text{Dist}_\theta(\gamma)$ and the likelihood $p(\mathcal{D} \mid \theta, \gamma)$ is the product of densities of y given fixed θ.

3. To calculate $p(\gamma \mid \mathcal{D})$, we first use Bayes rule to get

$$p(\gamma \mid \mathcal{D}) \propto p(\mathcal{D} \mid \gamma)\, p(\gamma) \tag{38}$$

This can be easily calculated because the prior $p(\gamma)$ is of distribution $\text{Dist}_\gamma(\xi)$ of given ξ. The likelihood can be marginalized over θ to get

$$p(\mathcal{D} \mid \gamma) = \int p(\mathcal{D} \mid \theta, \gamma)\, p(\theta \mid \gamma)\ d\theta \tag{39}$$

where $p(\theta \mid \gamma)$ is a function of θ with given parameter γ, and $p(\mathcal{D} \mid \theta)$ is the product of the individual likelihoods.

But remember that this was all assumed under model $\mathcal{M}_i$, so the posterior density $p(\theta^i \mid \mathcal{D})$ of the θ^i parameterizing our best-fit function is really

$$p(\theta^i \mid \mathcal{D}, \mathcal{M}_i) \tag{40}$$

where we index the parameter of model $\mathcal{M}_i$ to be θ^i, with a superscript (since we may mistake subscript indices to be the components of θ).

3.3 Computing Posterior Distribution by Initially Applying Bayes Rule

There is another way we can approach to calculating the posterior $p(\theta \mid \mathcal{D})$.

1. We directly apply Bayes rule to get

$$p(\theta \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \theta)\, p(\theta)}{p(\mathcal{D})} = \frac{p(\mathcal{D} \mid \theta)\, p(\theta)}{\int p(\mathcal{D} \mid \theta)\, p(\theta)\ d\theta} \tag{41}$$

Since we are working under a specific model $\mathcal{M}_i$, it would be more accurate to say

$$p(\theta^i \mid \mathcal{D}, \mathcal{M}_i) = \frac{p(\mathcal{D} \mid \theta^i, \mathcal{M}_i)\, p(\theta^i \mid \mathcal{M}_i)}{p(\mathcal{D} \mid \mathcal{M}_i)} = \frac{p(\mathcal{D} \mid \theta^i, \mathcal{M}_i)\, p(\theta^i \mid \mathcal{M}_i)}{\int p(\mathcal{D} \mid \theta^i, \mathcal{M}_i)\, p(\theta^i \mid \mathcal{M}_i)\ d\theta^i} \tag{42}$$

2. Since $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ consists of N independent observations, we can calculate

$$p(\mathcal{D} \mid \theta^i, \mathcal{M}_i) = \prod_{j=1}^{N} p(y_j \mid x_j, \theta^i, \mathcal{M}_i) \tag{43}$$

since the form of the likelihood is determined by our model $\mathcal{M}_i$ that says $y = f(x, \theta^i) + \epsilon$.

3. To calculate $p(\theta^i \mid \mathcal{M}_i)$, we would have to condition over the hyperparameter γ, which gives

$$p(\theta^i \mid \mathcal{M}_i) = \int p(\theta^i \mid \gamma, \mathcal{M}_i)\, p(\gamma \mid \mathcal{M}_i)\ d\gamma \tag{44}$$

where $p(\theta^i \mid \gamma, \mathcal{M}_i)$ is the density of $\text{Dist}_{\theta^i}(\gamma)$ where γ is constant, and $p(\gamma \mid \mathcal{M}_i)$ is the prior distribution $\text{Dist}_\gamma(\xi)$ with fixed ξ.

Multiplying the two would get the proportional term, and integrating them over θ^i would get the marginalization constant

$$p(\mathcal{D} \mid \mathcal{M}_i) = \int p(\mathcal{D} \mid \theta^i, \mathcal{M}_i)\, p(\theta^i \mid \mathcal{M}_i)\, d\theta^i \tag{45}$$

entirely defining the posterior. Upon closer inspection, these two methods of deriving the posterior parameter are not that different. One just uses Bayes rule first and then marginalizes, while the other marginalizes and then uses Bayes rule.

3.4 Constructing a Predictive Function from Parameter Density

We can then construct a **predictive distribution** that calculates the probability of y given x. That is, given a new input x, the probability of getting a value y, given our dataset $\mathcal{D}$, is

$$\begin{aligned} p(y \mid x, \mathcal{D}, \mathcal{M}_i) &= \int p(y \mid \theta^i, x, \mathcal{D}, \mathcal{M}_i)\, p(\theta^i \mid x, \mathcal{D}, \mathcal{M}_i)\, d\theta^i \\ &= \int p(y \mid \theta^i, x, \mathcal{M}_i)\, p(\theta^i \mid \mathcal{D}, \mathcal{M}_i)\, d\theta^i \end{aligned}$$

but $p(\theta^i \mid \mathcal{D}, \mathcal{M}_i)$ is completely defined by what we just calculated, and $p(y \mid \theta^i, x, \mathcal{D}, \mathcal{M}_i)$ is defined by the random variable generated by

$$y \sim f(x, \theta^i) + \epsilon \tag{46}$$

3.5 Basis Functions

For *linear* regression, we usually denote the parameters θ of function $f(x, \theta)$ as w, so we can treat them as equivalent. The simplest linear model for regression is one that involves a linear combination of the input variables

$$f(x, \theta) = f(x, w) = w_0 + w_1 x_1 + \ldots + w_D x_D \tag{47}$$

where $x = (x_1, \ldots, x_D)^T$. The key property of this model is that it is a linear function of the parameters $w_0, \ldots, w_D$. But the fact that it linear with respect to the input variables x_i imposes significant limitations. Therefore, we can extend the class of models by considering combinations of fixed nonlinear functions of the input variables of the form

$$f(x, w) = w_0 + \sum_{j=1}^{M-1} w_j \phi_j(x) \tag{48}$$

where each **basis function** $\phi_j : \mathbb{R}^D \longrightarrow \mathbb{R}$. By denoting the maximum value of the index j by $M - 1$, the total number of parameters in this model will be M. Note that the above form can be written in the form

$$f(x, w) = \sum_{j=0}^{M-1} w_j \phi_j(x) = w^T \phi(x) \tag{49}$$

by introducing a "dummy" basis function $\phi_0(x) = 1$. The reason this is still called a linear model is because the function is linear in w.

We can choose many different types of basis functions. The following examples are for 1-dimensional x.

1. The **polynomial basis functions** form powers of x such that

$$\phi_j(x) = x^j \tag{50}$$

   One limitation of polynomial basis function is that they are global functions on the input variable, so that changes in one region of input space affect all other regions. This can be resolved by dividing up the input space up into regions and fit a different polynomial in each region, leading to **spline functions**.

2. The **Gaussian basis functions** (which can be, but not necessarily must be interpreted in the probabilistic way), have the form
$$\phi_j(x) = \exp\left(-\frac{(x-\mu_j)^2}{2s^2}\right) \tag{51}$$
where the μ_j govern the locations of the basis functions in input space, and the parameter s governs their spatial scale.

3. The **sigmoidal basis functions** are of form
$$\phi_j(x) = \sigma\left(\frac{x-\mu_j}{s}\right), \text{ where } \sigma(a) = \frac{1}{1+e^{-a}} \tag{52}$$
Rather than using the sigmoid function σ, we could also use the hyperbolic tangent $\tanh(a) = 2\sigma(a) - 1$.

4. The **Fourier basis functions** leads to an expansion in sinusoidal functions, which has specific frequency and infinite spatial extent. By contrast, basis functions that are localized to finite regions of input space necessarily comprise a spectrum of different spatial frequencies. In many signal processing applications, it is of interest to consider basis functions that are localized in both space and frequency, leading to a class of functions known as **wavelets**.

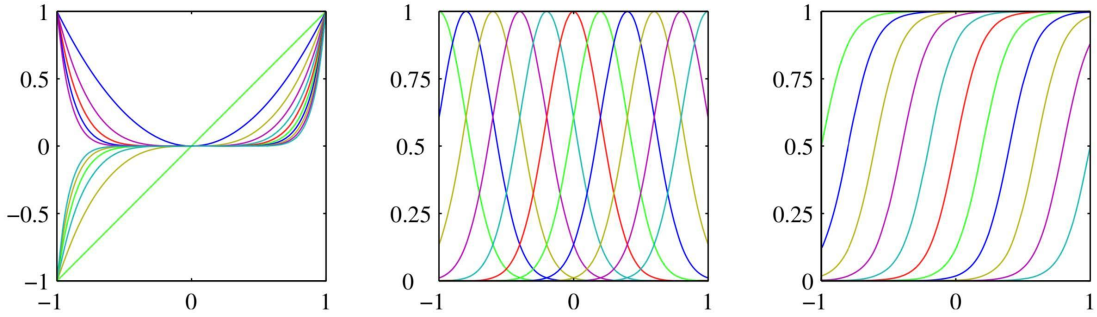

Figure 3: Different types of basis functions

3.6 Bayesian Model Selection

Note that up until now, we have assumed that we *knew* the **statistical model** describing the process of how the data $\mathcal{D}$ was generated. The definition of a model is often used loosely without explicit definition, but we can define it as such: A model completely defines the *form* of the function f that we assume is generating y for values of x. This does not mean that the model corresponds to a parameter value of w. It defines the *entire form* of f for
$$y = f(x, \theta) + \epsilon \tag{53}$$
The model then defines the form of $p(y_i \mid x_i, \theta)$ according to the above, which then defines the form of the likelihood function
$$p(\mathcal{D} \mid \theta) = \prod_{i=1}^{N} p(y_i \mid x_i, \theta) \tag{54}$$
Here are some examples of different models for different problems. Note that for every model $\mathcal{M}_i$, the set of parameters θ^i is *different*, since the basis functions do not need to necessarily be the same for these models.

1. For linear regression, we assume that the distribution is of form
$$y = w^T \phi(x) + \epsilon \tag{55}$$
and thus our models have different forms which are completely dependent on the basis functions $\phi_j(x)$ we choose. Assuming that we have scalar inputs $x \in \mathbb{R}$, we may choose

- a purely linear model of x, which we will call $\mathcal{M}_1$ with $\theta^1 = (w_0, w_1)$.

$$y = \begin{pmatrix} w_0 & w_1 \end{pmatrix} \begin{pmatrix} \phi_0(x) \\ \phi_1(x) \end{pmatrix} + \epsilon = \begin{pmatrix} w_0 & w_1 \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix} + \epsilon \tag{56}$$

  Therefore the form is $f(x, w) = w_0 + w_1 x$.

- a quadratic model of x, which we will call $\mathcal{M}_2$ with $\theta^2 = (w_0, w_1, w_2)$.

$$y = \begin{pmatrix} w_0 & w_1 & w_2 \end{pmatrix} \begin{pmatrix} \phi_0(x) \\ \phi_1(x) \\ \phi_2(x) \end{pmatrix} + \epsilon = \begin{pmatrix} w_0 & w_1 & w_2 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ x^2 \end{pmatrix} + \epsilon \tag{57}$$

  Therefore the form is $f(x, w) = w_0 + w_1 x + w_2 x^2$.

- a cubic model of x called $\mathcal{M}_3$ with form $f(x, \theta) = w_0 + w_1 x + w_2 x^2 + w_3 x^3$, and so on...

2. More examples to be updated.

A fully Bayesian approach would condition over all possible models when predicting y given x. Suppose that we have a finite set of Bayesian models $\{\mathcal{M}_i\}$ (each with their own parameters θ^i) that we could use to explain the observed data $\mathcal{D}$. Then, as shown above, for each ith model, we would calculate the posterior density of the parameter $p(\theta^i \,|\, \mathcal{D}, \mathcal{M}_i)$ and then construct a predictive distribution

$$p(y \,|\, x, \mathcal{D}, \mathcal{M}_i) \tag{58}$$

for each model in $\{\mathcal{M}_i\}$. Then we calculate the posterior probabilities of the models $p(\mathcal{M}_i \,|\, \mathcal{D})$, and conditioning over all possible models, we get the ultimate predictive distribution over all models

$$p(y \,|\, x, \mathcal{D}) = \sum_i p(y \,|\, x, \mathcal{D}, \mathcal{M}_i)\, p(\mathcal{M}_i \,|\, \mathcal{D}) \tag{59}$$

This is called a **mixture model** or **Bayesian model averaging**, but in practice this is not used due to computational overhead. A more common practice is simply to calculate all the $p(\mathcal{M}_i \,|\, \mathcal{D})$, pick the $\mathcal{M}_i$ that has the highest posterior probability, and build out predictive distribution assuming $\mathcal{M}_i$. This is called **model selection**, since we are throwing away all other models that are deemed to overfit or underfit and selecting the best one.

Now, the problem of model selection (and averaging) reduces to just finding the posterior model probabilities $p(\mathcal{M}_i \,|\, \mathcal{D})$, since we know how to do everything else. We can work out the posterior probability over the models via Bayes rule

$$p(\mathcal{M}_i \,|\, \mathcal{D}) \propto p(\mathcal{D} \,|\, \mathcal{M}_i)\, p(\mathcal{M}_i) \tag{60}$$

$p(\mathcal{M}_i)$ is the prior distribution over models that we have selected, which is conventionally set to the uniform: $p(\mathcal{M}_i) \propto 1$. Therefore, calculating the posterior probability of the models reduces to calculating $p(\mathcal{D} \,|\, \mathcal{M}_i)$ which is called the **model evidence**. By marginalizing over the parameter θ^i, we have

$$p(\mathcal{D} \,|\, \mathcal{M}_i) = \int p(\mathcal{D} \,|\, \theta^i, \mathcal{M}_i)\, p(\theta^i \,|\, \mathcal{M}_i)\, d\theta^i \tag{61}$$

To calculate this, we evaluate each component of the integral:

- Remember that $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ is composed of independent data. So

$$p(\mathcal{D} \,|\, \theta^i, \mathcal{M}_i) = \prod_{i=1}^N p(y_i \,|\, x_i, \theta^i, \mathcal{M}_i) \tag{62}$$

  which is well-defined since we can simply use our model $y = f(x, \theta^i) + \epsilon$.

- Furthermore, we see that $p(\theta^i \,|\, \mathcal{M}_i)$ should be conditioned over its hyperparameter γ, so

$$p(\theta^i \,|\, \mathcal{M}_i) = \int p(\theta^i \,|\, \gamma, \mathcal{M}_i)\, p(\gamma \,|\, \mathcal{M}_i)\, d\gamma \tag{63}$$

  where $\theta^i \,|\, \gamma \sim \text{Dist}_\theta(\gamma)$ for constant γ and $\gamma \sim \text{Dist}_\gamma(\xi)$ for constant ξ.

Note that if we had calculated the posterior densities of the parameters $p(\theta_i \,|\, \mathcal{D}, \mathcal{M}_i)$ by applying Bayes rule first, we would have calculated the posterior as

$$p(\theta^i \,|\, \mathcal{D}, \mathcal{M}_i) = \frac{p(\mathcal{D} \,|\, \theta^i, \mathcal{M}_i)\, p(\theta^i \,|\, \mathcal{M}_i)}{p(\mathcal{D} \,|\, \mathcal{M}_i)} = \frac{p(\mathcal{D} \,|\, \theta^i, \mathcal{M}_i)\, p(\theta^i \,|\, \mathcal{M}_i)}{\int p(\mathcal{D} \,|\, \theta^i, \mathcal{M}_i)\, p(\theta^i \,|\, \mathcal{M}_i)\, d\theta^i} \tag{64}$$

Note that the marginalization term that we've already calculated is the model evidence! So this shortcut may save us a lot of computation.

3.7 Intuition Behind Model Evidence

Let us take a closer look at the model evidence term and try to develop an intuition for it.

$$p(\mathbf{Y} \,|\, \mathcal{M}_i) = \int p(\mathbf{Y} \,|\, w, \mathcal{M}_i) p(w \,|\, \mathcal{M}_i)\, dw \tag{65}$$

Note that the evidence tells us the probability of getting $\mathbf{Y}$ from a given model $\mathcal{M}_i$, and we want this to be as large as possible. It does this by conditioning over all possible values of w for that given model. Consider first the case of a model having a single parameter w. Let us make two assumptions:

- The posterior distribution $p(\mathbf{Y} \,|\, w, \mathcal{M}_i)$ is sharply peaked around the most probable value w_{MAP}, with width $\Delta w_{\text{posterior}}$.
- The prior distribution $p(w \,|\, , \mathcal{M}_i)$ is flat with width Δw_{prior}, so that $p(w) = 1/\Delta w_{\text{prior}}$.

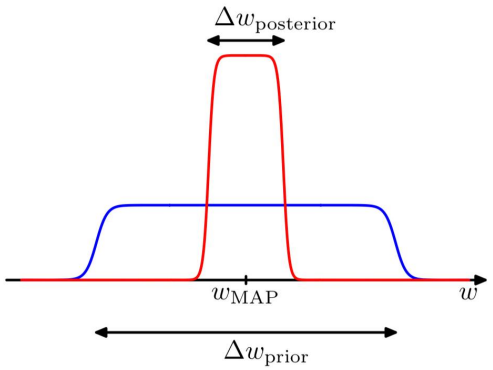

Figure 4: Approximation of posterior and prior distributions

Then, we can approximate

$$p(\mathbf{Y} \,|\, \mathcal{M}_i) = \int p(\mathbf{Y} \,|\, w, \mathcal{M}_i) p(w \,|\, \mathcal{M}_i)\, dw \approx p(\mathbf{Y} \,|\, w_{MAP})\, \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}} \tag{66}$$

Note two things:

- The term $p(\mathbf{Y} \,|\, w_{MAP})$ gives the fit to the data given the most probable parameter values w_{MAP}. If this fit is better (i.e. this term becomes larger), then the evidence also increases.

- However, the ratio $\Delta w_{\text{posterior}}/\Delta w_{\text{prior}}$ should be less than 1, meaning that the more "squished" the posterior distribution is, the smaller this fraction becomes, decreasing the evidence.

For a model having a set of M parameters, we can make a similar approximation. Assuming that all parameters have the same ratio $\Delta w_{\text{posterior}}/\Delta w_{\text{prior}}$, we get

$$p(\mathbf{Y} \mid \mathcal{M}_i) = p(\mathbf{Y} \mid w_{MAP}, \mathcal{M}_i) \left(\frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}} \right)^M \tag{67}$$

Therefore, we can see that the size of the complexity penalty increases with the number M of adaptive parameters in the model. Therefore, given two models $\mathcal{M}_i$ and $\mathcal{M}_j$ with the latter having more parameters (e.g. higher degree polynomial model), the model evidence $p(\mathbf{Y} \mid \mathcal{M}_j)$ will decrease at a faster rate as the posterior gets more fine-tuned to the data.

3.8 Frequentist Linear Regression Using Maximum Likelihood: Gaussian Error w/ OLS & Laplacian Error w/ LAV

Now, given dataset $\mathcal{D} = (\mathbf{X}, \mathbf{Y})$, we fix a model $\mathcal{M}$ and assume that $f(x, w) = w^T \phi(x)$ for a given collection (determined by $\mathcal{M}$) and the noise is Gaussian $\epsilon \sim \mathcal{N}(0, \beta^{-1})$. Therefore,

$$y = w^T \phi(x) + \epsilon = \begin{pmatrix} w_1 & \dots & w_D \end{pmatrix} \begin{pmatrix} \phi_0(x) \\ \vdots \\ \phi_D(x) \end{pmatrix} + \epsilon \implies p(y \mid x, w, \beta) = \mathcal{N}\big(y \mid w^T \phi(x), \beta^{-1}\big) \tag{68}$$

Then, the likelihood function is

$$p(\mathcal{D} \mid w, \beta) = \prod_{n=1}^{N} p(y_i \mid x_i, w, \beta) = \prod_{n=1}^{N} \mathcal{N}\big(y_n \mid w^T \phi(x_n), \beta^{-1}\big) \tag{69}$$

Taking the logarithm of it and a bit of algebra gives

$$\begin{aligned} \ln p(\mathcal{D} \mid w, \beta) &= \frac{N}{2} \ln \beta - \frac{N}{2} \ln 2\pi - \beta E_D(w) \\ &= \frac{N}{2} \ln \beta - \frac{N}{2} \ln 2\pi - \beta \cdot \frac{1}{2} \sum_{n=1}^{N} \big(y_n - w^T \phi(x_n)\big)^2 \end{aligned}$$

which we can see is very dependent on the **sum-of-squares error term** $E_D(w)$. This is the motivation behind the least squares function as the cost function for modeling functions with Gaussian errors. Moving on, maximizing this likelihood gives us

$$\begin{aligned} w_{ML} &= (\Phi^T \Phi)^{-1} \Phi^T \mathbf{Y} \\ \beta_{ML} &= \left(\frac{1}{N} \sum_{n=1}^{N} \big(y_n - w_{ML}^T \phi(x_n)\big)^2 \right)^{-1} \end{aligned}$$

where $\mathbf{Y}$ is the N-vector of target values y_i in the data $\mathcal{D}$ and Φ is the $N \times M$ matrix of basis functions evaluated for each x_n.

$$\Phi = \begin{pmatrix} \phi_1(x_1) & \phi_2(x_1) & \dots & \phi_{M-1}(x_1) & \phi_M(x_1) \\ \phi_1(x_2) & \phi_2(x_2) & \dots & \phi_{M-1}(x_2) & \phi_M(x_2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \phi_1(x_{N-1}) & \phi_2(x_{N-1}) & \dots & \phi_{M-1}(x_{N-1}) & \phi_M(x_{N-1}) \\ \phi_1(x_N) & \phi_2(x_N) & \dots & \phi_{M-1}(x_N) & \phi_M(x_N) \end{pmatrix} \tag{70}$$

Note that even if there were a hyperparameter of θ, the frequentist approach would not care about this because all it looks at is the likelihood of $\mathcal{D}$ *given* θ. Note that if we assumed that the residual noise distribution was $\epsilon \sim \text{Laplace}(0, \beta)$, the likelihood function would turn out to be

$$p(\mathcal{D} \mid w, \beta) = \prod_{n=1}^{N} \text{Laplace}(y_n \mid w^T \phi(x_n), \beta) = \prod_{n=1}^{N} \frac{1}{2\beta} \exp\left(- \frac{|y_n - w^T \phi(x_n)|}{b} \right) \tag{71}$$

and taking the logarithm of it gives

$$\begin{aligned} \ln p(\mathcal{D} \mid w, \beta) &= -N \ln(2\beta) - \frac{2}{\beta} E_D(w) \\ &= -N \ln(2\beta) - \frac{1}{\beta} \sum_{n=1}^{N} \left| y_n - w^T \phi(x_n) \right| \end{aligned}$$

which is now dependent on the **sum-of-residuals error term** $E_D(w)$.

3.9 Regularization: Gaussian Parameter Prior w/ L2 Regularizers & Laplacian Parameter Prior w/ L1 Regularizers

In some cases of solving the least squares problem, it may be case that our model with optimized parameters w, β may be either:

- too fine-tuned to the data, i.e. may overfit. This happens when the number of basis functions exceeds the number of observations, which makes the least squares problem ill-posed and is therefore impossible to fit because the associated optimization problem has infinitely many solutions. RLS allows the introduction of further constraints that uniquely determine the solution.
- suffering from poor generalization.

Therefore, we can add a **regularization term** $E_W(w)$ to our residual-squared error function $E_D(w)$.

$$E_D(w) + \lambda E_W(w) \tag{72}$$

The idea is that as the model becomes more complex and as w's values increase, the $E_W(w)$ term will also increase, nullifying the minimization of $E_D(w)$. Two common regularization terms are:

- The **L1 regularization term** is

$$E_W(w) = \frac{1}{2} \sum_{j=0}^{M-1} |w_j| \tag{73}$$

  which leads us to find

$$\begin{aligned} &\arg\min_w \left\{ \frac{1}{2} \sum_{n=1}^{N} \left(y_n - w^T \phi(x_n) \right)^2 + \frac{\lambda}{2} \sum_{j=0}^{M-1} |w_j| \right\} \text{ if } \epsilon \text{ is Gaussian} \\ &\arg\min_w \left\{ \frac{1}{2\beta} \sum_{n=1}^{N} \left| y_n - w^T \phi(x_n) \right| + \frac{\lambda}{2} \sum_{j=0}^{M-1} |w_j| \right\} \text{ if } \epsilon \text{ is Laplacian} \end{aligned}$$

- The **L2 regularization term**

$$E_W(w) = \frac{1}{2} \sum_{j=0}^{M-1} w_j^2 = \frac{1}{2} ||w||^2 = \frac{1}{2} w^T w \tag{74}$$

which leads us to find

$$\arg\min_w \left\{ \frac{1}{2}\sum_{n=1}^{N} \big(y_n - w^T\phi(x_n)\big)^2 + \frac{\lambda}{2}\sum_{j=0}^{M-1} w_j^2 \right\} \text{ if } \epsilon \text{ is Gaussian}$$

$$\arg\min_w \left\{ \frac{1}{2\beta}\sum_{n=1}^{N} \big|y_n - w^T\phi(x_n)\big| + \frac{\lambda}{2}\sum_{j=0}^{M-1} w_j^2 \right\} \text{ if } \epsilon \text{ is Laplacian}$$

But how do we know which regularization term $E_W(w)$ to use?

- Remember that our *assumption* of the form of the error distribution ϵ led to least error term. A Gaussian ϵ led to a OLS cost function, and a Laplace ϵ led to a LAV cost function.
- Similarly, our assumption of the form of the prior density $p(w)$ will naturally lead to the form of the regularization term. A Gaussian prior $p(w)$ leads to the L2 regularizer, and a Laplace $p(w)$ leads to the L1 regularizer.

We must step out of the frequentist setting and let Bayesian statistics take over. Unlike simply getting the point estimate from the maximum likelihood, i.e. calculating $\arg\max_w p(\mathcal{D}\,|\,w)$, we must calculate

$$\begin{aligned}
\arg\max_w p(w\,|\,\mathcal{D}) &= \arg\max_w p(\mathcal{D}\,|\,w)\,p(w) \\
&= \arg\max_w \log\big(p(\mathcal{D}\,|\,w)\,p(w)\big) \\
&= \arg\max_w \Big(\log p(\mathcal{D}\,|\,w) + \log p(w)\Big)
\end{aligned}$$

Note that the frequentist calculations is the Bayesian approach with the prior $p(w)$ set to uniform. Previously, we have assumed that $p(w) = \mathcal{N}(w\,|\,0, \alpha^{-1}I)$. We can simplify this assumption by further assuming that it is a product of univariate distributions for each of its parameters w_i, which can be done with a change of basis. So, we will write

$$p(w) = \prod_{j=0}^{M-1} p(w_j) = \begin{cases} \prod_{j=0}^{M-1} \mathcal{N}(w_j\,|\,0, \alpha^{-1}) & \text{if assuming Gaussian} \\ \prod_{j=0}^{M-1} \text{Laplace}(w_j\,|\,0, \alpha^{-1}) & \text{if assuming Laplace} \end{cases} \tag{75}$$

Remember that $\epsilon \sim \mathcal{N}(0, \beta^{-1})$, and the priors $\mathcal{N}(0, \alpha^{-1})$ and $\text{Laplace}(0, b)$ have fixed and known parameters α and b.

- If we assume that each $p(w_j)$ is Gaussian, we have

$$\begin{aligned}
\arg\max_w p(w\,|\,\mathcal{D}) &= \arg\max_w \Big(\log p(\mathcal{D}\,|\,w) + \log p(w)\Big) \\
&= \arg\max_w \left(\log \prod_{n=1}^{N} \mathcal{N}\big(y_n\,|\,w^T\phi(x_n), \beta^{-1}\big) + \log \prod_{j=0}^{M-1} \mathcal{N}(w_j\,|\,0, \alpha^{-1})\right) \\
&= \arg\max_w \left(\log \prod_{n=1}^{N} \frac{1}{\beta^{-1}\sqrt{2\pi}} e^{-\frac{(y_n - w^T\phi(x_n))^2}{2(\beta^{-1})^2}} + \log \prod_{j=0}^{M-1} \frac{1}{\alpha^{-1}\sqrt{2\pi}} e^{-\frac{w_j^2}{2(\alpha^{-1})^2}}\right) \\
&= \arg\min_w \frac{1}{2(\beta^{-1})^2}\left(\sum_{n=1}^{N} \big(y_n - w^T\phi(x_n)\big)^2 + \frac{(\beta^{-1})^2}{(\alpha^{-1})^2}\sum_{j=0}^{M-1} w_j^2\right) \\
&= \arg\min_w \left(\sum_{n=1}^{N} \big(y_n - w^T\phi(x_n)\big)^2 + \lambda\sum_{j=0}^{M-1} w_j^2\right)
\end{aligned}$$

So, we can see that having a Gaussian prior of the parameter naturally leads to us minimizing the L2-regularized cost function. Furthermore, we have the optimal value $\lambda = (\beta^{-1})^2/(\alpha^{-1})^2$.

- If we assume that each $p(w_j)$ is Laplace, we have [Similar derivation for Laplace case...]

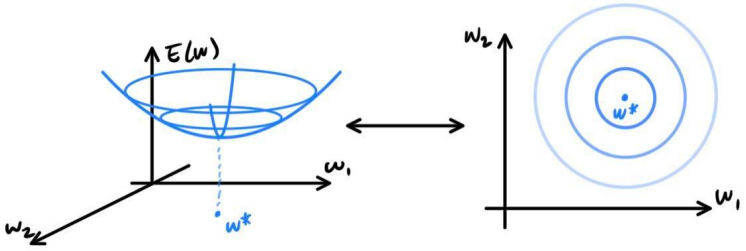

Figure 5: Error function contours in $\mathbb{R}^2$

To develop an intuition for this, let us visualize what this regularization term does. Setting $w \in \mathbb{R}^2$ for visual purposes, we can visualize the (unregularized) error function $E_D(w)$ as being defined over $\mathbb{R}^2$ with contours, where darker lines represent lower values. Clearly, the minimum value of w lies at the dot w^*.

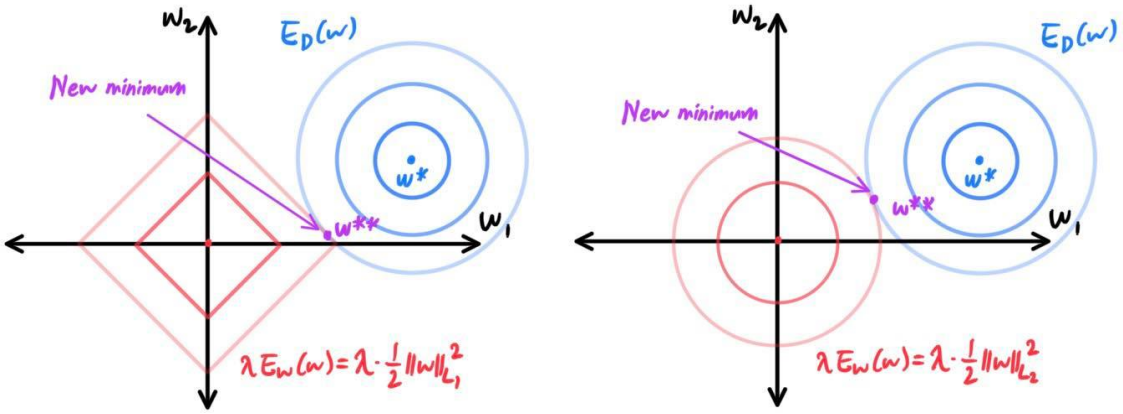

Figure 6: Comparison of L1 and L2 regularization effects

[Continuing with the rest of the visualizations and explanations...]

In summary:

- The Laplace prior promotes sparsity, i.e. zeroes out some of the coefficients due to its greater peak around 0.
- The Gaussian prior is more diffused around 0, allowing non-zero values to have greater probability mass.

Other possibilities for robust priors are Cauchy or t-distributions.

3.10 Bayesian Linear Regression with Gaussian Priors

To perform linear regression in the Bayesian setting, let us start off with a collection of potential models $\{\mathcal{M}_i\}_{i=1}^L$ and dataset $\mathcal{D}$. For each model $\mathcal{M}_i$ with

$$y = w^T \phi(x) + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \beta^{-1}) \tag{76}$$

We will state our unknowns:

- The value of β that determines the variance of the error ϵ will have a *fixed* prior distribution $p(\beta)$ (with no hyperparameter).

- The parameter w has a (not fixed) prior distribution $p(w) = \mathcal{N}(w \mid 0, \alpha^{-1} I)$ with hyperparameter α.
- The value of α that determines the covariance matrix of the prior of w will have a fixed prior distribution $p(\alpha)$, with no further hyperparameters.

Now, our final goal is to construct the predictive function $p(y \mid x, \mathcal{D})$. But since the predictive function is completely determinant on the values of w, β (since $y = w^T \phi(x) + \epsilon \sim \mathcal{N}\big(w^T \phi(x), \beta^{-1}\big)$, we simply marginalize over the two parameters to simplify it into

$$\begin{aligned} p(y \mid x, \mathcal{D}) &= \iint p(y \mid x, w, \beta, \mathcal{D})\, p(w, \beta \mid \mathcal{D})\, dw\, d\beta \\ &= \iint \mathcal{N}\big(y \mid w^T \phi(x), \beta^{-1}\big)\, p(w, \beta \mid \mathcal{D})\, dw\, d\beta \end{aligned}$$

Therefore, we now need to calculate the joint posterior distribution of w, β given $\mathcal{D}$. To marginalize this over the proper parameters, we need more insight.

- Let us first calculate $p(w \mid \mathcal{D})$ to see what parameters the posterior density is dependent on.

$$\begin{aligned} p(w \mid \mathcal{D}) &\propto p(\mathcal{D} \mid w)\, p(w) \\ &= \left(\int p(\mathcal{D} \mid w, \beta)\, p(\beta)\, d\beta \right) \cdot \left(\int p(w \mid \alpha)\, p(\alpha)\, d\alpha \right) \\ &= \int \left(\prod_{n=1}^{N} p(y_i \mid x_i, w, \beta) \right) p(\beta)\, d\beta \cdot \left(\int \mathcal{N}(w \mid 0, \alpha^{-1} I)\, p(\alpha)\, d\alpha \right) \\ &= \int \left(\prod_{n=1}^{N} \mathcal{N}\big(y \mid w^T \phi(x), \beta^{-1}\big) \right) p(\beta)\, d\beta \cdot \left(\int \mathcal{N}(w \mid 0, \alpha^{-1} I)\, p(\alpha)\, d\alpha \right) \end{aligned}$$

  Note that in this case, we marginalized over all β and α, so $p(w \mid \mathcal{D})$ is parameterized by both α and β. If we kept them fixed, we would have

$$\begin{aligned} p(w \mid \alpha, \beta, \mathcal{D}) &\propto p(\mathcal{D} \mid w, \alpha, \beta)\, p(w \mid \alpha, \beta) \\ &= p(\mathcal{D} \mid w, \beta)\, p(w \mid \alpha) \\ &= \left(\prod_{n=1}^{N} \mathcal{N}\big(y \mid w^T \phi(x), \beta^{-1}\big) \right) \cdot \mathcal{N}(w \mid 0, \alpha^{-1} I) \\ &= \mathcal{N}\big(w \mid m_N = \beta S_N \Phi^T \mathbf{Y}, S_N = (\alpha I + \beta \Phi^T \Phi)^{-1}\big) \end{aligned}$$

  which itself is a multivariate Gaussian.

Knowing this, we know we should marginalize $p(w, \beta \mid \mathcal{D})$ so that the term $p(w \mid \alpha, \beta, \mathcal{D})$ exists. We can do this by

$$\begin{aligned} p(w, \beta \mid \mathcal{D}) &= \int p(w, \beta \mid \alpha, \mathcal{D})\, p(\alpha \mid \mathcal{D})\, d\alpha \\ &= \int p(w \mid \beta, \alpha, \mathcal{D})\, p(\beta \mid \alpha, \mathcal{D})\, p(\alpha \mid \mathcal{D})\, d\alpha \\ &= \int p(w \mid \beta, \alpha, \mathcal{D})\, p(\alpha, \beta \mid \mathcal{D})\, d\alpha \end{aligned}$$

where in the second row we simply used the conditional probability rule $p(a, b \mid c) = p(a \mid b, c)\, p(b \mid c)$. Finally,

substituting this into the double integral above gives

$$\begin{aligned} p(y \mid x, \mathcal{D}) &= \iint p(y \mid x, w, \beta, \mathcal{D}) \left(\int p(w \mid \beta, \alpha, \mathcal{D})\, p(\alpha, \beta \mid \mathcal{D})\, d\alpha \right) dw\, d\beta \\ &= \iiint p(y \mid x, w, \beta, \mathcal{D})\, p(w \mid \beta, \alpha, \mathcal{D})\, p(\alpha, \beta \mid \mathcal{D})\, d\alpha\, dw\, d\beta \\ &= \iiint \mathcal{N}\big(y \mid w^T \phi(x), \beta^{-1}\big) \left(\prod_{n=1}^{N} \mathcal{N}\big(y \mid w^T \phi(x), \beta^{-1}\big) \right) \cdot \mathcal{N}(w \mid 0, \alpha^{-1} I)\, p(\alpha, \beta \mid \mathcal{D})\, d\alpha\, dw\, d\beta \end{aligned}$$

Our Gaussian assumption on the priors will greatly simplify this term when written in vector notation. Now, the only thing to do is figure out what $p(\alpha, \beta \mid \mathcal{D})$ is. We will use Bayes rule and assume that the prior $p(\alpha, \beta)$ is relatively flat.

$$\begin{aligned} p(\alpha, \beta \mid \mathcal{D}) &\propto p(\mathcal{D} \mid \alpha, \beta)\, p(\alpha, \beta) \\ &\propto p(\mathcal{D} \mid \alpha, \beta) \end{aligned}$$

where $p(\mathcal{D} \mid \alpha, \beta)$ is another evidence function (like the model evidence), which we will call the *hyperparameter evidence*. We can simply condition over w to get the following. Hopefully, the realizations of the probabilities into the densities function make sense to the reader.

$$\begin{aligned} p(\alpha, \beta \mid \mathcal{D}) &\propto p(\mathcal{D} \mid \alpha, \beta) \\ &= \int p(\mathcal{D} \mid w, \beta)\, p(w \mid \alpha)\, dw \\ &= \int \left(\prod_{n=1}^{N} \mathcal{N}\big(y_n \mid w^T \phi(x_n), \beta^{-1}\big) \right) \cdot \mathcal{N}(w \mid 0, \alpha^{-1} I)\, dw \end{aligned}$$

If we know that $p(\alpha, \beta \mid \mathcal{D})$ is sharply peaked, then we can try maximizing the evidence function with respect to α, β using maximum likelihood, and simply fixing them in further calculations.

By substituting in the densities, the evidence function reduces to

$$p(\mathcal{D} \mid \alpha, \beta) = \left(\frac{\beta}{2\pi} \right)^{N/2} \alpha^{M/2} \exp\big(- E(m_N) \big)\, |A|^{-1/2} \tag{77}$$

where

$$\begin{aligned} \Phi = (\Phi)_{nj} &= \phi_j(x_n) \\ S_N^{-1} &= \alpha I + \beta \Phi^T \Phi \\ m_N &= \beta S_N \Phi^T \mathbf{Y} \\ E(m_N) &= \frac{\beta}{2} ||\mathbf{Y} - \Phi m_N||^2 + \frac{\alpha}{2} m_N^T m_N \end{aligned}$$

Taking the log gives us

$$\log p(\mathcal{D} \mid \alpha, \beta) = \frac{M}{2} \log \alpha + \frac{N}{2} \log \beta - E(m_N) - \frac{1}{2} \log |S_N^{-1}| - \frac{N}{2} \log 2\pi \tag{78}$$

This evidence can also be used as the model evidence. Remember that given data $\mathcal{D}$, a (linear) model $\mathcal{M}_i$ determines the collection of basis function, i.e. determines Φ. Let us denote the Φ determined by model $\mathcal{M}_i$ as $\Phi_{\mathcal{M}_i}$. Therefore, we can treat $p(\mathcal{D} \mid \alpha, \beta)$ as a function of Φ and write

$$p(\mathcal{D} \mid \alpha, \beta, \Phi) \tag{79}$$

To determine which model from $\{\mathcal{M}_i\}_{i=1}^L$ to choose, we first fix $\Phi_{\mathcal{M}_i}$ and maximize $p(\mathcal{D} \mid \alpha, \beta, \Phi_{\mathcal{M}_i})$ with respect to α, β, for $i = 1, \dots, L$.

$$\begin{aligned} \text{Assume model } \mathcal{M}_1 &\implies \text{Find } \max_{\alpha,\beta} p(\mathcal{D} \mid \alpha, \beta, \Phi_{\mathcal{M}_1}) \\ \text{Assume model } \mathcal{M}_2 &\implies \text{Find } \max_{\alpha,\beta} p(\mathcal{D} \mid \alpha, \beta, \Phi_{\mathcal{M}_2}) \\ \dots &\implies \dots \\ \text{Assume model } \mathcal{M}_L &\implies \text{Find } \max_{\alpha,\beta} p(\mathcal{D} \mid \alpha, \beta, \Phi_{\mathcal{M}_L}) \end{aligned}$$

Then, find

$$\arg\max_{\mathcal{M}_i} \{\max_{\alpha,\beta} p(\mathcal{D} \mid \alpha, \beta, \Phi_{\mathcal{M}_i})\} \tag{80}$$

For each model $\mathcal{M}_i$, we have optimized α, β to maximize the evidence function $p(\mathcal{D} \mid \alpha, \beta, \Phi_{\mathcal{M}_i})$. The model with the highest max evidence should be the best model, and by Occam's razor, we should choose simpler models if their predictive powers are equal. Again, we restate the big takeaway: *The Φ represents the model, and therefore maximizing the evidence function $p(\mathcal{D} \mid \alpha, \beta, \Phi_{\mathcal{M}_i})$ with respect to the Φ will tell us what the correct model is.* That is,

1. $p(\mathcal{D} \mid \alpha, \beta, \Phi_{\mathcal{M}_i})$ interpreted as a function of α, β is the **hyperparameter evidence**.
2. $p(\mathcal{D} \mid \alpha, \beta, \Phi_{\mathcal{M}_i})$ interpreted as a function of $\Phi_{\mathcal{M}_i}$ (or more accurately, of $\mathcal{M}_i$) is the **model evidence**.

3.11 Equivalent Kernel

The posterior mean solution $m_N = \beta S_N \Phi^T \mathbf{Y}$ is a point-estimate prediction of what w is. We can substitute it into the linear equation $f(x, w) = w^T \phi(x)$ to get

$$f(x, m_N) = m_N^T \phi(x) = \beta \phi(x)^T S_N \Phi^T \mathbf{Y} = \sum_{n=1}^N \beta \phi(x)^T S_N \phi(x_n) y_n \tag{81}$$

which is a linear combination of the training set target variables y_n, written as

$$f(x, m_N) = \sum_{n=1}^N k(x, x_n) y_n, \qquad k(x, x_n) \equiv \beta \phi(x)^T S_N \phi(x_n) \tag{82}$$

That is, the mean of the predictive distribution at a point x is given by a linear combination of the y_n's. The function $k(x, x_n)$ is known as the **smoother matrix**, or **equivalent kernel**.

4 Bias Variance Decomposition

Determination of the predictive distribution $p(y \mid x)$ given data $\mathcal{D}$ is the goal of statistical inference, as we have seen. That is, posterior $p(y \mid x, \mathcal{D})$ tells us the distribution of y if we have a new data point x. But after this inference step, we must look now at the **decision step**: we must determine a function $h(x)$ that deterministically predicts a value y, without predictions. That is, we must have some algorithm to make a decision.

Let us zoom out for a better overview. Let $\mathcal{D}$ be our training data of N points. We can assume that each point $(x_i, y_i) \in \mathcal{D}$ was *generated* independently by a joint distribution $p(x, y)$. If we were to get another data point, we would just generate one from the density $p(x, y)$. Usually, we have fixed input data x and knew that the output y given x would be $p(y \mid x)$. But if we loosen our constraint on x, we would get

$$p(x, y) = p(y \mid x) p(x) \tag{83}$$

which states that each data point in $\mathcal{D}$ is gotten by generating a value of x with probability $p(x)$, and then generating a y given this x. Let us also denote $\mathcal{A}$ as our machine learning algorithm, which we can interpret as a function that takes in data $\mathcal{D}$ and outputs the hypothesis function $h_{\mathcal{D}}$.

$$\mathcal{A}(\mathcal{D}) = h_{\mathcal{D}} \tag{84}$$

Then, given that the next new data point (x, y) is generated, we can set our **test error**, or **loss/cost function**, of $h_{\mathcal{D}}$ to be

$$L\big(h_{\mathcal{D}}, (x, y)\big) = \big[h_{\mathcal{D}}(x) - y\big]^2 \tag{85}$$

This loss function basically calculates the inaccuracy of whatever hypothesis function $h_{\mathcal{D}}$ we have on the data (x, y), which in this case is the square of the residual. There can be other types of loss functions, but we will consider the squares loss function for now. Given $h_{\mathcal{D}}$, we can also calculate the expected test error by conditioning over all x, y drawn from P.

$$\text{Expected Test Error given } h_{\mathcal{D}} \implies \mathbb{E}_{x,y,\sim P}\big[(h_{\mathcal{D}}(x) - y)^2\big] = \int_x \int_y (h_{\mathcal{D}}(x) - y)^2 \, p(x, y) \, dy \, dx \tag{86}$$

However, note that we can treat the N data points $\mathcal{D}$ also as a random variable coming from the joint distribution of N P's. Therefore, we can take each possible dataset $\mathcal{D}$, calculate $h_{\mathcal{D}} = \mathcal{A}(\mathcal{D})$ with our algorithm, and average them out to get the expected hypothesis function $\overline{h}$. We can interpret $\overline{h}$ as the "ideal regressor" that we are trying to build, but with limited data $\mathcal{D}$, we can only build $h_{\mathcal{D}}$ that deviates from $\overline{h}$.

$$\overline{h} = \mathbb{E}_{\mathcal{D} \sim P^N}\big[\mathcal{A}(\mathcal{D})\big] = \int_{\mathcal{D}} h_{\mathcal{D}} P(\mathcal{D}) \, d\mathcal{D} \tag{87}$$

So, we can compute the expected error of the *entire algorithm* $\mathcal{A}$ by marginalizing over all x, y given $h_{\mathcal{D}}$ and marginalizing over all $\mathcal{D}$. Remember that $D \sim P^N$ is our training data of N points, and $(x, y) \sim P$ is our $(n+1)$th data point. Therefore, the expected test error of our *algorithm* for the $(n+1)$th data point is

$$\mathbb{E}_{(x,y)\sim P, \mathcal{D}\sim P^N}\big([h_{\mathcal{D}}(x) - y]^2\big) = \int_{\mathcal{D}} \int_x \int_y [h_{\mathcal{D}}(x) - y]^2 \, p(x, y) \, p(\mathcal{D}) \, dy \, dx \, d\mathcal{D} \tag{88}$$

The integral above looks quite intimidating, so let us decompose it. We just have do use a trick where we subtract and add the same term $\overline{h}(x)$.

$$\begin{aligned}
\mathbb{E}_{(x,y),\mathcal{D}}\big([h_{\mathcal{D}}(x) - y]^2\big) &= \mathbb{E}_{(x,y),\mathcal{D}}\big([(h_{\mathcal{D}}(x) - \overline{h}(x)) + (\overline{h}(x) - y)]^2\big) \\
&= \mathbb{E}_{(x,y),\mathcal{D}}\big([h_{\mathcal{D}}(x) - \overline{h}(x)]^2\big) + \mathbb{E}_{(x,y),\mathcal{D}}\big([\overline{h}(x) - y]^2\big) \\
&\quad + 2\mathbb{E}_{(x,y),\mathcal{D}}\big([h_{\mathcal{D}}(x) - \overline{h}(x)]\,[\overline{h}(x) - y]\big)
\end{aligned}$$

But I claim that the last term vanishes. It is easy to see why because

$$\begin{aligned}
\mathbb{E}_{(x,y),\mathcal{D}}\left[\big(h_{\mathcal{D}}(x) - \bar{h}(x)\big)\big(\bar{h}(x) - y\big)\right] &= \mathbb{E}_{(x,y)}\left[E_{\mathcal{D}}\left[h_{\mathcal{D}}(x) - \bar{h}(x)\right]\big(\bar{h}(x) - y\big)\right] \\
&= \mathbb{E}_{(x,y)}\left[\big(E_{\mathcal{D}}\left[h_{\mathcal{D}}(x)\right] - \bar{h}(x)\big)\big(\bar{h}(x) - y\big)\right] \\
&= \mathbb{E}_{(x,y)}\left[\big(\bar{h}(x) - \bar{h}(x)\big)\big(\bar{h}(x) - y\big)\right] \\
&= \mathbb{E}_{(x,y)}\left[0\right] \\
&= 0
\end{aligned}$$

Therefore, we can see that the expected value of the error of an algorithm consists of two terms: the variance and the second term.

$$\mathbb{E}_{(x,y),\mathcal{D}}\big([h_{\mathcal{D}}(x) - y]^2\big) = \mathbb{E}_{(x,y),\mathcal{D}}\big([h_{\mathcal{D}}(x) - \overline{h}(x)]^2\big) + \mathbb{E}_{(x,y),\mathcal{D}}\big([\overline{h}(x) - y]^2\big) \tag{89}$$

The second term is the expected value of the average prediction minus the y-value of the new point. Now, we do the same trick: Let the expected value of y given x be $\overline{y}(x) = \mathbb{E}_{y\,|\,x}(y) = \int y\, p(y\,|\,x)\, dx$. This function $\overline{y}(x)$ is the ideal regressor predicting y from x. Then, we have

$$\begin{aligned}\mathbb{E}_{x,y}\left[\left(\bar{h}(x) - y\right)^2\right] &= \mathbb{E}_{x,y}\left[\left(\bar{h}(x) - \bar{y}(x)\right) + \left(\bar{y}(x) - y\right)^2\right] \\ &= \underbrace{\mathbb{E}_{x,y}\left[\left(\bar{y}(x) - y\right)^2\right]}_{\text{Noise}} + \underbrace{\mathbb{E}_{x}\left[\left(\bar{h}(x) - \bar{y}(x)\right)^2\right]}_{\text{Bias}^2} + 2\,\mathbb{E}_{x,y}\left[\left(\bar{h}(x) - \bar{y}(x)\right)\left(\bar{y}(x) - y\right)\right]\end{aligned}$$

where the third term vanishes since

$$\begin{aligned}\mathbb{E}_{x,y}\left[\left(\bar{h}(x) - \bar{y}(x)\right)\left(\bar{y}(x) - y\right)\right] &= \mathbb{E}_{x}\left[\mathbb{E}_{y|x}\left[\bar{y}(x) - y\right]\left(\bar{h}(x) - \bar{y}(x)\right)\right] \\ &= \mathbb{E}_{x}\left[\left(\bar{y}(x) - \mathbb{E}_{y|x}\left[y\right]\right)\left(\bar{h}(x) - \bar{y}(x)\right)\right] \\ &= \mathbb{E}_{x}\left[\left(\bar{y}(x) - \bar{y}(x)\right)\left(\bar{h}(x) - \bar{y}(x)\right)\right] \\ &= \mathbb{E}_{x}\left[0\right] \\ &= 0\end{aligned}$$

Therefore, the expected test error is precisely the sum of three things.

$$\underbrace{\mathbb{E}_{x,y,\mathcal{D}}\left[\left(h_{\mathcal{D}}(x) - y\right)^2\right]}_{\text{Expected Test Error}} = \underbrace{\mathbb{E}_{x,\mathcal{D}}\left[\left(h_{\mathcal{D}}(x) - \bar{h}(x)\right)^2\right]}_{\text{Variance}} + \underbrace{\mathbb{E}_{x,y}\left[\left(\bar{y}(x) - y\right)^2\right]}_{\text{Noise}} + \underbrace{\mathbb{E}_{x}\left[\left(\bar{h}(x) - \bar{y}(x)\right)^2\right]}_{\text{Bias}^2}$$

To understand this term a bit deeper, recall the following: The function $\overline{y}(x)$, which outputs the expected value of y given x, is the best possible regressor we can have. There are many different algorithms that we can choose to approximate $\overline{y}(x)$, so let us choose one learning algorithm $\mathcal{A}$. We just feed an arbitrary dataset $\mathcal{D}$ to $\mathcal{A}$, which outputs a hypothesis function $h_{\mathcal{D}}$. But this hypothesis function $h_{\mathcal{D}}$ is really just an approximation of the *ideal* hypothesis function $\overline{h}$, which is the expectation of all hypotheses $h_{\mathcal{D}}$ (i.e. the hypothesis that $\mathcal{A}$ should generate when we feed it an infinite amount of data). So, by feeding $\mathcal{D}$ to $\mathcal{A}$, it generates a hypothesis function $h_{\mathcal{D}}(x)$, which approximates $\overline{h}(x)$, which hopefully is a good estimate of $\overline{y}(x)$.

1. The difference between a generated hypothesis function $h_{\mathcal{D}}(x)$ and the ideal hypothesis that it is trying to estimate according to learning algorithm $\mathcal{A}$ is represented by the variance. The variance term tells us how far each generated hypothesis $h_{\mathcal{D}}$ deviates from the ideal $\overline{h}$.
2. The difference between the ideal hypothesis $\overline{h}(x)$ (according to algorithm $\mathcal{A}$) and the ideal regressor *in general* $\overline{y}(x)$ is captured in the bias term. The bias term tells us how far our algorithm's ideal hypothesis deviates from the expectation of the conditional $p(y\,|\,x)$.
3. The noise term represents the difference between the true value of y and the best possible regressor $\overline{y}(x)$. But since the best we can do is find the expectation of the conditional $p(y\,|\,x)$, the deviation of the true values y from the mean $\overline{y}$ is simply the noise. For example, if we have $p(y\,|\,x) = \mathcal{N}\big(y\,|\,w^T\phi(x), \epsilon\big)$, then the noise would simply be ϵ. If the variance of ϵ is large, the noise would be large. Therefore, the same ideal regressor function $\overline{y}(x)$ would perform worse with a higher noise.

If we are comparing this to the throwing-darts analogy, we can imagine the ideal function $\overline{y}(x)$ to be the bull's eye that we must hit. The different algorithms $\mathcal{A}$ represent different players throwing the darts. When one algorithm (player) is chosen, their vision can be skewed (perhaps their glasses is off), leading them to think that the target $\overline{h}(x)$ is somewhere else. If their target is far away from the bull's eye (i.e. $[\overline{h}(x) - \overline{y}(x)]^2$ is high), then their bias is high. Their skills in darts may just be bad, so even if their vision is good and they have a good sense of where to hit (low bias), for each time they throw the dart (i.e. each time the regressor function $h_{\mathcal{D}}$ is generated from data), it may be very off from their ideal target $\overline{h}(x)$.

Therefore, if you are a data scientist and you find that your regression function is not accurate enough, it is your job to find out whether your bias is too high, your variance is too high, or whether there is too much

noise, and fix the proper component. Generally, we would try to minimize this cost function, visualized below.

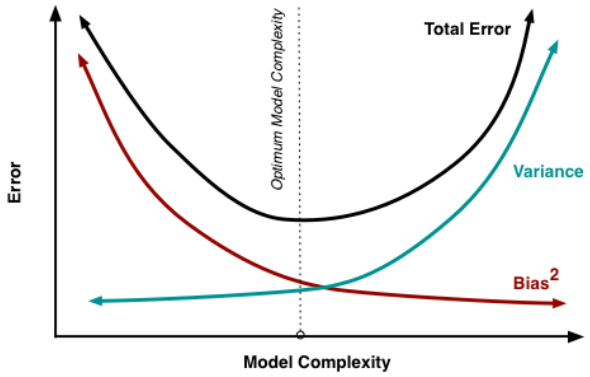

Figure 7: Visualization of the bias-variance tradeoff showing how model complexity affects error components

5 Markov Chain Monte Carlo (MCMC)

Monte carlo algorithms is a general term for computational techniques that use random numbers, which can be used both in classical and Bayesian statistics. This is extremely important when working with distributions that are cannot be simply stated using elementary densities (Gaussian, Beta, etc.). The entire goal of Bayesian inference is to maintain a full posterior probability distribution over a set of random variables. However, maintaining and using this distribution requires computing integrals which, for most non-trivial models, is intractable.

The basic idea of MCMC is that we want to construct a Markov chain which will travel between different possible states (e.g. the hypotheses/parameter values in a Bayesian analysis), where the amount of time spent in any particular state is proportional to the posterior probability of the state. That is, the stationary distribution of the chain is the posterior distribution. As a result, the computer explores the set of possible parameter values, spending a lot of time in the regions with high posterior probability, and only rarely visiting regions of low posterior probability.

5.1 Metropolis-Hastings: General Algorithm

Say that with initial distribution $p(\theta)$, we have calculated the posterior as

$$p(\theta \mid x) \propto p(\theta)\ p(x \mid \theta) \tag{90}$$

It is often the case that the set of possible values of θ is very large, so it is computationally inefficient to compute the normalizing factor

$$p(x) = \sum_{\theta} p(\theta)\ p(x \mid \theta) \tag{91}$$

Therefore, we only have this function $f(\theta) = p(\theta)\ p(x \mid \theta)$ that is directly proportional to $p(\theta \mid x)$. That is, we don't know the normalizing constant c such that

$$p(\theta \mid x) = \frac{f(\theta)}{c} \tag{92}$$

Using this information, we wish to construct and run an algorithm that converges onto the true posterior distribution $p(\theta \mid x)$ at a sufficiently fast rate.

We begin by constructing a discrete-time irreducible Markov chain with state space $\mathcal{S} = \{1, 2, \dots, N\}$ representing the set of possible parameter values (the labels for the elements of $\mathcal{S}$ does not matter, since we can construct whatever bijection we want from the actual states to a subset of $\mathbb{N}$). Like a normal Markov chain, we will choose the next state to go to at each step, *but now, we will then choose to accept this proposal to go to that step with an additional probability.* That is, we will construct two matrices:

- An $|\mathcal{S}| \times |\mathcal{S}|$ **proposal transition matrix** Q_{prop}, with
$$p(\text{propose } i \mapsto j) = (Q_{prop})_{ij} = q_{prop}(i, j) \tag{93}$$
being the probability of getting a *proposal* to transition from state i to state j. This matrix is constructed by the user and is completely well-defined and known; this choice may also affect the convergence rate. Note that with this formulation, the rows will sum up to 1 and Q^T is a stochastic matrix. We can also construct Q_{prop} to be symmetric, that is $q_{prop}(i, j) = q_{prop}(j, i)$, for easier calculations.
- An $|\mathcal{S}| \times |\mathcal{S}|$ **acceptance probability matrix** A, with
$$\begin{aligned}(\text{accept proposal } i \mapsto j \,|\, \text{propose } i \mapsto j) = (A)_{ij} &= \alpha(i, j) \\ &= \min\left(1, \frac{p(\theta = j \,|\, x)\; q_{prop}(j, i)}{p(\theta = i \,|\, x)\; q_{prop}(i, j)}\right) \\ &= \min\left(1, \frac{f(\theta = j)\; q_{prop}(j, i)}{f(\theta = i)\; q_{prop}(i, j)}\right) \\ &= \min\left(1, \frac{f(\theta = j)}{f(\theta = i)}\right) \qquad (\text{if } Q_{prop} \text{ symmetric})\end{aligned}$$

Then, we element-wise multiply the two matrices, except the diagonals, to get the **true transition matrix** Q defined

$$(Q)_{ij} = q(i, j) = \begin{cases} q_{prop}(i, j) \cdot \alpha(i, j) = q_{prop}(i, j) \cdot \min\left(1, \frac{f(\theta=j)\; q(j,i)}{f(\theta=i)\; q(i,j)}\right) & \text{if } i \neq j \\ 1 - \sum_{j \neq i} q(i, j) & \text{if } i = j \end{cases} \tag{94}$$

where $q(i, j)$ represents the **true transition probability** of going from state i to state j. Note that we have element-wise multiplied every non-diagonal element, and we have defined $(Q)_i i$ such that the sum of each row is 1 (so that this becomes a viable transition matrix). Note also that this element-wise multiplication makes sense because

$$\begin{aligned}p(\theta_{k+1} = j \,|\, \theta_k = i) &= p(\text{accept proposal } i \mapsto j,\ \text{propose } i \mapsto j) \\ &= p(\text{accept proposal } i \mapsto j \,|\, \text{propose } i \mapsto j)\; p(\text{propose } i \mapsto j) \\ &= \alpha(i, j) \cdot q_{prop}(i, j)\end{aligned}$$

This is the Markov chain we wish to get, where "one" step is really a two-step process of proposing and accepting/rejecting. We wish to get the steady state distribution $\pi(\theta)$ of this chain, which can be found in two well-known ways:

- Calculate the left-eigenvector of Q with eigenvalue 1.
- Randomly initialize θ_0 and run the chain for a sufficiently long time to record where it lands at each step
$$\theta_0 = i_0, \theta_1 = i_1, \theta_2 = i_2, \theta_3 = i_3, \dots, \theta_n = i_n \tag{95}$$
which can be used to approximate $\pi(\theta)$ by defining
$$\pi(\theta = i) = \frac{\text{proportion of states in state } i \text{ in the n-step process}}{n} \tag{96}$$

Finally, we claim that this steady state distribution $\pi(\theta)$ is precisely the posterior we are looking for: $p(\theta \,|\, x)$.

5.2 Detailed Balance: Justification of the Metropolis Algorithm

But why does $\pi(\theta) = p(\theta \mid x)$? Given a Markov chain θ_0 with transition matrix Q, the chain is said to satisfy **detailed balance** with respect to a distribution $\pi(\theta)$ if

$$\pi(\theta = i)q(i,j) = \pi(\theta = j)q(j,i) \tag{97}$$

for all $i, j \in \mathcal{S}$. In fact, we claim that θ_i does satisfy detailed balance with respect to $p(\theta \mid x)$. That is, it satisfies

$$p(\theta = i \mid x)q(i,j) = p(\theta = j \mid x)q(j,i) \tag{98}$$

This case is trivial for when $i = j$, so assume $i \neq j$. A transition from i to a different j can only be achieved with an accepted proposed step, which happens with probability

$$\begin{aligned} q(i,j) &= q_{prop}(i,j) \cdot \alpha(i,j) \\ &= q_{prop}(i,j) \cdot \min\left(1, \frac{p(\theta = j \mid x)\ q_{prop}(j,i)}{p(\theta = i \mid x)\ q_{prop}(i,j)}\right) \\ &= \frac{q_{prop}(i,j)}{p(\theta = i \mid x)} \min\left(p(\theta = i \mid x), \frac{p(\theta = j \mid x)\ q_{prop}(j,i)}{q_{prop}(i,j)}\right) \\ &= \frac{1}{p(\theta = i \mid x)} \min\left(p(\theta = i \mid x)\ q_{prop}(i,j), p(\theta = j \mid x)\ q_{prop}(j,i)\right) \end{aligned}$$

Applying the same method from transitioning from j to i gives the same equation, but with the i and j's switched.

$$q(j,i) = \frac{1}{p(\theta = j \mid x)} \min\left(p(\theta = j \mid x)\ q_{prop}(j,i), p(\theta = i \mid x)\ q_{prop}(i,j)\right) \tag{99}$$

But switching the i and j leaves the terms inside the minimum invariant. Therefore, we can see that

$$p(\theta = i \mid x)\ q(i,j) = \min\left(p(\theta = j \mid x)\ q_{prop}(j,i), p(\theta = i \mid x)\ q_{prop}(i,j)\right) = p(\theta = j \mid x)\ q(j,i) \tag{100}$$

proving detailed balance. Now, we can sum the left hand side of the detailed balance equation over i to get

$$\sum_i p(\theta = i \mid x)q(i,j) = \sum_i p(\theta = j \mid x)q(j,i) = p(\theta = j \mid x) \sum_i q(j,i) = p(\theta = j \mid x) \tag{101}$$

which in matrix form, says

$$p(\theta \mid x)Q = p(\theta \mid x) \tag{102}$$

where $p(\theta \mid x) = \big(p(\theta = 1 \mid x) \dots p(\theta = N \mid x)\big)$ and $Q_{ij} = q(i,j)$. This implies that $p(\theta \mid x)$ is a stationary distribution, and therefore, computing the stationary distribution is equivalent to computing $p(\theta \mid x)$.

The intuition behind detailed balance is quite easy to understand, too. Suppose we start a chain in the stationary distribution, so that the respective probabilities $\theta_0 \sim \pi(\theta)$ of starting at position are "smeared" across all states i. Then, the quantity $\pi(\theta = i)q(i,j)$ represents the "amount" of probability that flows down edge $i \to j$ in one time step. If detailed balance holds, then the amount of probability flowing from $i \to j$ equals the amount that flows from $j \to i$ (which is $\pi(\theta = j)q(j,i)$). Therefore, there is no *net* flux of probability along the edge $i \leftrightarrow j$ during one time step (remember this holds only for when the chain is in the stationary distribution).

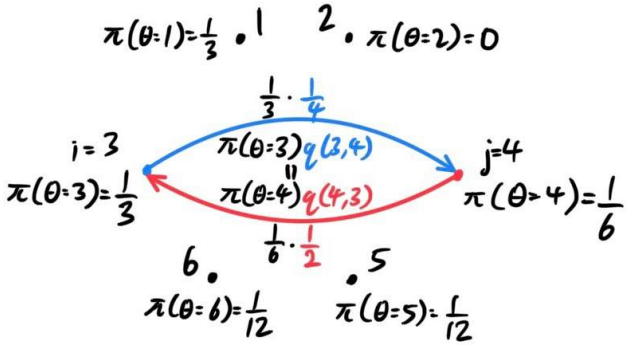

Figure 8: Visualization of detailed balance in Markov chain

5.3 Metropolis-Hastings: Example

Suppose we want a Markov chain of state space $\mathcal{S} = \{1, 2\}$ with the steady state distribution

$$\pi = \begin{pmatrix} \frac{3}{4} & \frac{1}{4} \end{pmatrix} \iff \pi(\theta = 1) = \frac{3}{4},\ \pi(\theta = 2) = \frac{1}{4} \tag{103}$$

To implement the Metropolis-Hastings algorithm, we calculate the proposal matrix and acceptance matrix

$$Q_{prop} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix} \text{ and } A = \begin{pmatrix} 1 & \frac{1}{3} \\ 1 & 1 \end{pmatrix} \tag{104}$$

which is calculated since $\alpha(1, 2) = \min\left(1, \frac{1/4}{3/4}\right) = 1/3$ and $\alpha(2, 1) = \min\left(1, \frac{3/4}{1/4}\right) = 1$. We multiply the nondiagonal entries together and fill in the diagonals to get

$$\begin{pmatrix} & \frac{1}{2} \cdot \frac{1}{3} \\ \frac{1}{2} \cdot 1 & \end{pmatrix} \implies Q = \begin{pmatrix} \frac{5}{6} & \frac{1}{6} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix} \tag{105}$$

Which can be visualized as an object jumping between two nodes with the following transitions.

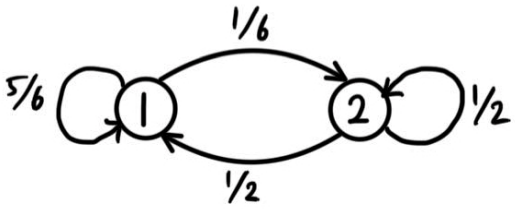

Figure 9: Two-state Markov chain with transition probabilities

5.4 Gibbs Sampling: General Algorithm

Gibbs Sampling is a special case of the Metropolis-Hastings in which the newly proposed state is accepted with probability one. With observed data x, say that we have calculated the D-dimensional posterior

$$p(\theta \,|\, x) \propto f(\theta) = p(\theta)\, p(x \,|\, \theta) \tag{106}$$

where the parameter $\theta = (\theta^1, \ldots, \theta^D)$ is an element of the D-dimensional state space $\mathcal{S} = \{1, \ldots, n\}^D$ (actually, each θ^i does not need to be derived from the same $\{1, \ldots, n\}$ and we can generalize this algorithm to account for this). Remember that:

- It is hard to calculate $p(\theta \,|\, x) = p(\theta^1, \dots, \theta^D \,|\, x)$ because calculating the constant c that normalizes $f(\theta)$ is hard (since D may be large). This makes it difficult to sample from the posterior.
- It is easy to calculate $f(\theta) = f(\theta^1, \dots, \theta^D)$. We just don't know how to scale the individual values appropriately and so this function is useless in of itself, even though it is directly proportional to $p(\theta \,|\, x)$.

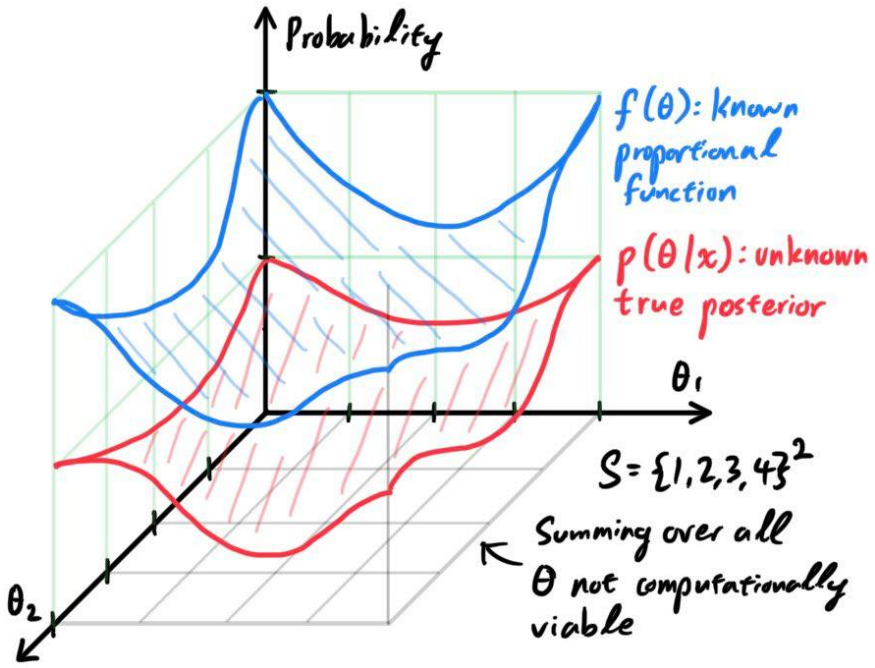

Figure 10: Comparison of unknown posterior vs known proportional function

With the D-dimensional state space $\mathcal{S}$, we construct the true transition matrix. Say that the ith state of the chain is located at node θ_i with given coordinates

$$\theta_i = (\theta_i^1, \theta_i^2, \dots, \theta_i^D) \tag{107}$$

The step to transition from this given θ_i to the next θ_{i+1} consists of two parts:

1. Pick a component index $j = d \in \{1, 2, \dots, D\}$ uniformly at random. Many algorithms also pick $d = 1$ for the first step, $d = 2$ for the second, and so on.

$$p(\text{Index } d \text{ chosen}) = \frac{1}{D} \tag{108}$$

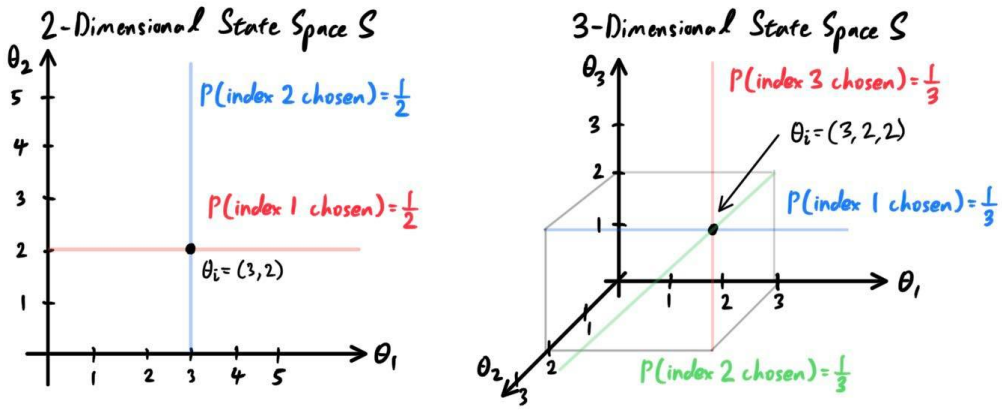

Figure 11: Choosing a component index in Gibbs sampling

2. With this well-defined d, we would like to update the Markov chain from state θ_i to θ_{i+1} by updating only the dth component of θ_i, and keeping every component fixed. When θ_i^d is updated, the new θ_{i+1}^d must take some value of $k^* \in \{1, \dots, n\}$. As expected, it chooses which value k^* to update to according to the marginal distribution of $p(\theta \,|\, x)$ given $\theta_i^1, \dots, \theta_i^{d-1}, \theta_i^{d+1}, \dots, \theta_i^D$.

$$\begin{aligned} p(\theta_i^d \mapsto \theta_{i+1}^d = k^* \,|\, \text{Index } d \text{ chosen}) &= p(\theta_{i+1}^d = k^* \,|\, \theta_i^1, \dots, \theta_i^{d-1}, \theta_i^{d+1}, \dots, \theta_i^D, x) \\ &= \frac{p(\theta_i^1, \dots, \theta_i^{d-1}, k^*, \theta_i^{d+1}, \dots, \theta_i^D \,|\, x)}{\sum_{k=1}^n p(\theta_i^1, \dots, \theta_i^{d-1}, k, \theta_i^{d+1}, \dots, \theta_i^D \,|\, x)} \\ &= \frac{f(\theta_i^1, \dots, \theta_i^{d-1}, k^*, \theta_i^{d+1}, \dots, \theta_i^D)}{\sum_{k=1}^n f(\theta_i^1, \dots, \theta_i^{d-1}, k, \theta_i^{d+1}, \dots, \theta_i^D)} \end{aligned}$$

where the last step is justified by the proportionality of f and p. It turns out that the probability of where θ_{i+1}^d will land on does not actually depend on where θ_i^d is currently.

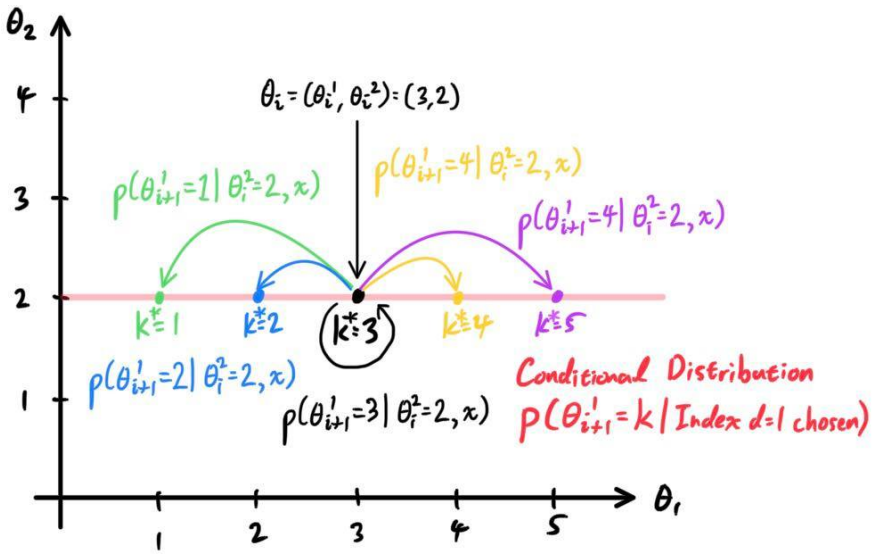

Figure 12: Example for $D = 2, n = 5$ showing possible states (within red line) that θ_{i+1} can transition to

Do not be daunted by the notation. Just remember that $p(\theta_{i+1}^d = k^* \,|\, \theta_i^1, \dots, \theta_i^{d-1}, \theta_i^{d+1}, \dots, \theta_i^D, x)$ is just the conditional probability of $p(\theta \,|\, x)$ given that every $\theta_i^j, j \neq d$ are constant. This is easily visualized by taking the 1-dimensional cross section of the density $p(\theta \,|\, x)$ defined on $\mathcal{S}$.

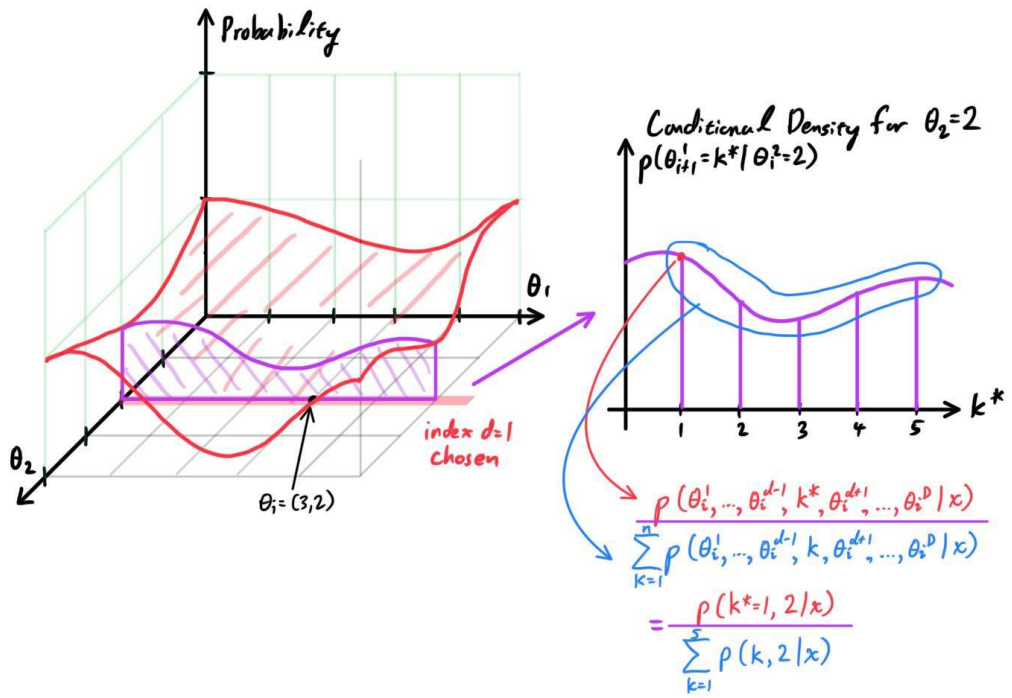

Figure 13: Cross-sectional view of density in Gibbs sampling

Therefore, we can construct a Markov chain with the following transition probabilities. Given two states $\theta_r, \theta_s \in \mathcal{S}$, if θ_s differs in θ_r in at most one component, call it the dth component (i.e. $\theta_r^j = \theta_s^j$ for all $j \neq d$), then the probability of transition from θ_r to θ_s is

$$\begin{aligned} p(\theta_r, \theta_s) &= p(\theta_r^d \mapsto \theta_s^d \,|\, \text{Index } d \text{ chosen})\; p(\text{Index } d \text{ chosen}) \\ &= \frac{f(\theta_r^1, \ldots, \theta_r^{d-1}, \theta_s^d, \theta_r^{d+1}, \ldots, \theta_r^D)}{\sum_{k=1}^n f(\theta_r^1, \ldots, \theta_r^{d-1}, k, \theta_r^{d+1}, \ldots, \theta_r^D)} \cdot \frac{1}{D} \end{aligned}$$

Therefore, given that the chain is in state $\theta_i = (3, 2)$ in state space $\mathcal{S} = \{1, 2, 3, 4, 5\}^2$, it may be able to get to the point in red or blue, depending on which index d was chosen. But it is impossible to go to any of the yellow states, so the transition probabilities are all 0.

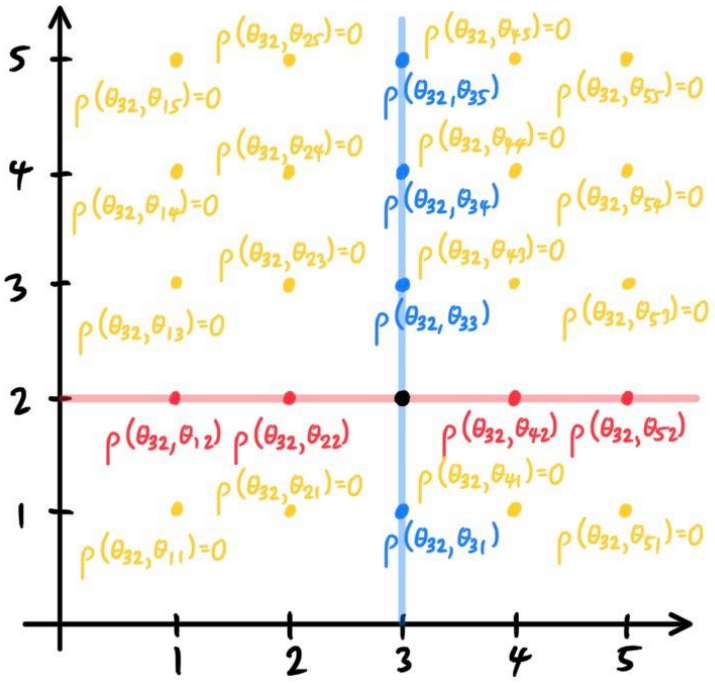

Figure 14: Possible transitions from state $(3, 2)$ in Gibbs sampling

With this, we can calculate the stationary distribution by either:

- Calculating the left-eigenvector of the transition matrix defined $p(\theta_r, \theta_s)$ with eigenvalue 1.
- Randomly initialize $\theta_0 = (\theta_0^1, \dots, \theta_0^D)$ and run the chain for sufficiently long time to find out the proportion of steps in which a Markov chain lands on each $\theta \in \mathcal{S}$.

Now, it is easy to see why Gibbs sampling is a special case of Metropolis-Hastings. The Gibbs transition algorithm that we just mentioned is clearly a Markov chain, and within the context of Metropolis, we can interpret it as the proposal transition matrix with acceptance probability 1. By the same justification for Metropolis, we can prove that the stationary distribution of Gibbs sampling is $p(\theta \mid x)$.