STAT 460/560 Class 9: Parametric frequentist inference

Daniel McDonald & Ben Bloem-Reddy

Reading: Chapter 9, [Was04]; Chapters 4-5, [EH21].

This class and the next are devoted to some of the core approaches from the classical theory and practice of statistical inference. They all have to do with parametric models, and differ only in their approaches to carrying out the inference. Although the differences are historically philosophical, today their distinction is largely practical, though the philosophical aspects are still of primary importance when interpreting the results of the statistical analysis.

1. Parametric models

Recall that in a **parametric model**, the parameter is finite-dimensional. That is,

$$\mathcal{F} = \{P_\theta : \theta \in \Theta \subseteq \mathbb{R}^k\} \ , \tag{9.1}$$

where Θ is the **parameter space**. We saw some examples in Class 5.

Note that sometimes the **parameter of interest** is not one of the parameters in the model, but a function of multiple parameters. For example, from the probabilities of infection for two diseases, p and q, we may be interested in the log odds ratio, $\ln(p/(1-p)) - \ln(q/(1-q))$.

2. Frequentist inference

In the **frequentist** interpretation of probability, all probability statements refer to limiting relative frequencies, with these limiting frequencies then applied verbatim *to particular realizations* of the events to which they refer. For example, if X_i is the outcome of a coin flip (heads/tails), then the limiting frequency

$$p = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \mathbf{I}\{X_i \text{ is heads}\} \tag{9.2}$$

is applied to a single coin flip variable X. We won't take up any philosophical arguments here, but just point out that the move of "limiting frequency property" to "distributional property of a particular realization" makes the entire pursuit of frequentist inference possible. In particular, it allows one to estimate distributional quantities from a single sample of data.

Note also that the frequentist approach implies that parameters of the underlying probability distributions are fixed and unknown, not subject to randomness, and *therefore no probability statements can be made about them.* As a consequence, frequentist statistical procedures should be designed to have well-defined long-run frequency properties, in the sense that as the sample size grows, any probabilistic statements about the procedure should become approximately true in the long-run frequency sense. Equivalently, across many repeated trials. Confidence intervals are the most obvious example.

Efron and Hastie [EH21, Ch. 2] offer the useful working definition of frequentist statistics: "the probabilistic properties of a procedure of interest are derived and then applied verbatim to the procedure's output for the observed data." They point out that this requires calculating probabilistic properties of the procedure under the *unknown distribution.* In practice, various methods are used to get around the problem. Plug-in estimators, the delta method (and other Taylor series-based approximations), and bootstrapping are three methods that we've already seen. Parametric models are another approach, which sometimes allow things to be computed in closed form, and other times lead to relatively simple ways to implement the other methods.

3. Method of moments

The goal of estimation is to use the data to make inferences about the unknown parameters of the model. There are various ways to do this, but the most useful are (1) method of moments, (2) maximum likelihood estimation, and (3) Bayesian estimation. We'll focus on the first two here, and the third in Class 10. (There's a common but less useful way of estimating parameters: guessing. Don't do that.)

The method of moments is a simple and intuitive approach to estimation. The idea is to equate the sample moments (e.g., sample mean, sample variance) to the corresponding population moments (e.g., population mean, population variance) and solve for the unknown parameters. This method is often straightforward to apply. In particular, there are cases where method of moment estimators are much easier to compute than other methods, and they can be used as initial estimates for iterative methods that find the MLE.

Suppose that $\theta = (\theta_1, \ldots, \theta_k)$. Define

$$\begin{aligned} m_1 &= \frac{1}{n}\sum_{i=1}^n X_i, \quad \mu_1(\theta) = \mathbb{E}(X_i) \\ m_2 &= \frac{1}{n}\sum_{i=1}^n X_i^2, \quad \mu_2(\theta) = \mathbb{E}(X_i^2) \\ &\vdots \qquad\qquad\qquad\qquad \vdots \\ m_k &= \frac{1}{n}\sum_{i=1}^n X_i^k, \quad \mu_k(\theta) = \mathbb{E}(X_i^k). \end{aligned}$$

Let $\widehat{\theta} = (\widehat{\theta}_1, \ldots, \widehat{\theta}_k)$ solve:

$$m_j = \mu_j(\widehat{\theta}), \quad j = 1, \ldots, k.$$

In other words, we equate the first k sample moments with the first k theoretical moments. This defines k equations with k unknowns.

Sometimes rather than the sample moments $m_1, \ldots, m_k$, we use the sample mean, sample variance, and higher-order central moments.

Exercise 9.1. Suppose $X_1, \ldots, X_n \sim_{\text{IID}} \text{Gamma}(\alpha, \beta)$, where α is the shape parameter and β is the rate parameter. Find the method of moments estimators for α and β.

4. Maximum likelihood estimation

Let $X_1, \ldots, X_n$ be IID with PDF/PMF $f(x; \theta)$, where $\theta \in \Theta \subseteq \mathbb{R}^k$. The **likelihood** function is

$$\mathcal{L}_n(\theta) = \prod_{i=1}^n f(X_i; \theta) \ , \tag{9.3}$$

and the **log-likelihood** function is $\ell_n(\theta) = \ln \mathcal{L}_n(\theta)$. This is the joint PMF/PDF of the data, *viewed as a function of the parameter*. That is, $\mathcal{L}_n \colon \Theta \to [0, \infty)$.

The **maximum likelihood estimator** (MLE), denoted $\hat{\theta}_n$, is the value of θ that maximizes $\mathcal{L}_n(\theta)$. Note that since logarithm is a strictly increasing function, the maximum of $\mathcal{L}_n$ occurs at the same place as the maximum of the log-likelihood, ℓ_n. Often, ℓ_n is easier to work with (analytically and numerically).

In many cases, the maximum can be found by taking the gradient of ℓ_n with respect to θ, setting the resulting expressions to zero, and solving that system of equations. Just be sure to check the second derivatives (Hessian matrix) to confirm that you've found a maximum rather than a minimum.

Exercise 9.2. Let $X_1, \ldots, X_n \sim_{\text{IID}} \mathcal{N}(\mu, \sigma^2)$. Find the MLE of $\theta = (\mu, \sigma)$.

Sometimes we can't solve the problem analytically.

Exercise 9.3. Let $X_1, \ldots, X_n \sim_{\text{IID}} \text{Gamma}(\alpha, \beta)$, where β is the rate parameter so that the PDF is

$$f(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x} \ . \tag{9.4}$$

Show that the MLE $(\hat{\alpha}, \hat{\beta})$ solves

$$\hat{\beta} = \frac{\hat{\alpha}}{\bar{X}_n} \quad \text{and} \quad \ln \hat{\alpha} - \psi(\hat{\alpha}) = \ln \bar{X}_n - \frac{1}{n} \sum_{i=1}^n \ln x_i \ , \tag{9.5}$$

where $\psi(\alpha) = \frac{d}{d\alpha} \ln \Gamma(\alpha)$ is the digamma function.

In such cases, how do we find the MLE in practice? Numerical optimization. See the section at the end of these notes (and STAT 535C) for some details.

We'll focus here on some theoretical aspects of ML estimators. These properties are a major reason that the use of MLEs is so widespread.

5. Consistency

We would like to show that if we maximize the likelihood, then as $n \to \infty$, $\hat{\theta}_n \xrightarrow{\text{P}} \theta_*$, where θ_* is the true parameter value. This is a little more involved than proving the consistency of an estimator that is the sample average of some function (though in many cases, the MLE is a sample average).

The proof relies on the **Kullbeck–Leibler (KL) divergence** between two PDFs f and g,

$$D(f, g) = \int f(x) \ln \left(\frac{f(x)}{g(x)} \right) \, dx \ . \tag{9.6}$$

This is non-negative, and $D(f, g) = 0$ if and only if $f = g$ (up to null sets of F). For any two parameters $\theta, \theta' \in \Theta$, we write $D(\theta, \theta') = D(f(x; \theta), f(x; \theta'))$.

A model $\mathcal{F}$ is **identifiable** if $\theta \neq \theta'$ implies that $D(\theta, \theta') > 0$, i.e., the two parameters correspond to different distributions in $\mathcal{F}$. We will assume that the model is identifiable.

Maximizing $\ell_n(\theta)$ is equivalent to maximizing

$$M_n(\theta) = \frac{1}{n} \sum_{i=1}^n \ln \frac{f(X_i; \theta)}{f(X_i; \theta_*)} \ . \tag{9.7}$$

Since θ_* is the true parameter value, $M_n(\theta) \xrightarrow{\text{P}} -D(\theta_*, \theta) =: M(\theta)$ for each $\theta \in \Theta$ (pointwise). This is not quite enough, however, to imply that $\hat{\theta}_n \xrightarrow{\text{P}} \theta_*$.

Here's the basic idea for proving consistency. We want to show that $M(\hat{\theta}_n) \xrightarrow{\text{P}} M(\theta_*)$, and also find conditions on M so that if M can be used to "separate points" in Θ near θ_*, then the convergence of M can be used to induce convergence of $\hat{\theta}_n$.

Formally, we assume that for every $\epsilon > 0$,

$$\sup_{\theta \colon \|\theta - \theta_*\| \geq \epsilon} M(\theta) < M(\theta_*) \ . \tag{9.8}$$

Then there is some $\delta > 0$ so that

$$\mathbb{P}(\|\hat{\theta}_n - \theta_*\| \geq \epsilon) \leq \mathbb{P}(M(\hat{\theta}_n) < M(\theta_*) - \delta) \ . \tag{9.9}$$

If $M(\hat{\theta}_n) \xrightarrow{\text{p}} M(\theta_*)$ then the right-hand term goes to zero. We still need to show $M(\hat{\theta}) \xrightarrow{\text{p}} M(\theta_*)$. The key additional assumption is uniform convergence of M_n.

Activity 9.1. Assume that $\sup_\theta |M_n(\theta) - M(\theta)| \xrightarrow{\text{p}} 0$. Show that $M(\hat{\theta}) \xrightarrow{\text{p}} M(\theta_*)$.

Theorem 9.1. *Suppose that* $\sup_\theta |M_n(\theta) - M(\theta)| \xrightarrow{p} 0$*, and that* (9.8) *holds. Then* $\hat{\theta}_n \xrightarrow{p} \theta_*$ *as* $n \to \infty$*.*

6. Equivariance

Suppose we're interested in some function of the parameter, say $\tau = g(\theta)$. Then it turns out that $\hat{\tau}_n = g(\hat{\theta}_n)$. This is easy to prove if g is one-to-one. If it is not, then we can define the **induced likelihood** as

$$\mathcal{L}_n^*(\tau) = \sup_{\theta\,:\; g(\theta)=\tau} \mathcal{L}_n(\theta) \; . \tag{9.10}$$

Theorem 9.2. *Let* $\tau = g(\theta)$*. Then* $\hat{\tau}_n = g(\hat{\theta}_n)$*, where the likelihood as a function of* τ *is as defined in* (9.10)*.*

7. Asymptotic normality and the delta method

The asymptotic properties of MLEs are characterized by the **score function**,

$$s(X;\theta) = \nabla_\theta \ell(\theta) \; , \tag{9.11}$$

and the **Fisher information** matrix

$$[\mathcal{I}_n(\theta)]_{i,j} = \mathbb{E}\left(\frac{\partial \ell_n(\theta)}{\partial \theta_i}\frac{\partial \ell_n(\theta)}{\partial \theta_j}\right) =^* -\mathbb{E}\left(\frac{\partial^2 \ell_n(\theta)}{\partial \theta_i \partial \theta_j}\right) \; . \tag{9.12}$$

The $=^*$ indicates that the equality is true only under certain conditions; namely that we can interchange differentiation and integration (expectation). For the situations we encounter in this course, we can assume that the condition is met.

Exercise 9.4. Prove that $\mathbb{E}_\theta[s(X;\theta)] = 0_k$, where 0_k is a k-dimensional vector of zeros. (You can assume that you can interchange differentiation with respect to θ and expectation.) Use this fact to show that

$$[\mathcal{I}(\theta)]_{i,j} = \text{Cov}(s(X;\theta)_i, s(X;\theta)_j) \; . \tag{9.13}$$

Theorem 9.3. *Under appropriate regularity conditions,*[1]

$$\mathcal{I}_n(\hat{\theta}_n)^{1/2}(\hat{\theta}_n - \theta) \rightsquigarrow \mathcal{N}(0, \mathbb{I}_k) \; . \tag{9.14}$$

The approximate covariance between components of $\hat{\theta}_n$ *is* $Cov(\hat{\theta}_n^i, \hat{\theta}_n^j) \approx [\mathcal{I}_n(\hat{\theta}_n)^{-1}]_{i,j}$*.*

The regularity conditions here are essentially the ones needed for the consistency of $\hat{\theta}_n$, plus those that allow us to interchange differentiation and expectation. See, for example, Schervish [Sch95, Thm. 7.63], for sufficient regularity conditions.

We'll look at the proof of this for $\Theta \subseteq \mathbb{R}$, just to keep the notation light. But the technique generalizes to higher dimensional Θ.

Proof. Let ℓ_n' denote the derivative of ℓ_n with respect to θ, ℓ_n'' the second derivative, and so on. The basic idea is to perform a Taylor expansion of $\ell_n'(\theta)$ around θ_*, and analyze as $n \to \infty$. In particular,

$$\ell_n'(\theta) = \ell'(\theta_*) + (\theta - \theta_*)\ell_n''(\theta_*) + \cdots \; , \tag{9.15}$$

[1] We'll look at these in more detail later in the term. For now, existence of second derivatives, continuity of the Fisher information, and the assumptions needed for consistency are sufficient.

We'll neglect the higher-order terms because, under the regularity conditions, the error from doing so becomes negligible as $n \to \infty$. (See Schervish [Sch95, Thm. 7.63].) Substituting $\hat{\theta}_n$, noting that $\ell_n'(\hat{\theta}_n) = 0$, and rearranging yields

$$\sqrt{n}(\hat{\theta}_n - \theta_*) = -\frac{\ell_n'(\theta_*)/\sqrt{n}}{\ell_n''(\theta_*)/n} \ . \tag{9.16}$$

We'll finish the proof as activities. □

Activity 9.2. Show that the numerator of (9.16) converges in distribution as

$$-\frac{\ell_n'(\theta_*)/\sqrt{n}}{\ell_n''(\theta_*)/n} \rightsquigarrow \mathcal{N}(0, 1/\mathcal{I}(\theta_*)) \ .$$

Activity 9.3. Assume that $\mathcal{I}(\theta)$ is a continuous function of θ. Argue that $\mathcal{I}(\hat{\theta}_n) \xrightarrow{\text{P}} \mathcal{I}(\theta_*)$, and therefore that the conclusion of the theorem holds.

For differentiable functions of the parameter $\tau = g(\theta)$, the delta method can be used to establish an asymptotic normality result for $\sqrt{n}(\hat{\tau}_n - \tau_*)$.

A Numerical optimization: a primer (optional)

When we have an optimization problem such as

$$\max_{\alpha>0,\beta>0} \ell_n(\alpha, \beta) \tag{A.1}$$

that we can't solve analytically, we can rely on any number of numerical/algorithmic optimization techniques. We'll just touch the surface here; there are entire courses on optimization (focused on numerical methods) and much of modern statistics and machine learning would be impossible without good numerical optimization methods.

A large subclass of numerical optimization methods are gradient-based. The basic idea is as follows. Suppose our problem is to maximize some function $f(\theta)$ with respect to $\theta \in \mathbb{R}^k$, and that f has at least one derivative. That is,

$$\theta^* = \arg\max_{\theta} f(\theta) = \arg\min_{\theta} -f(\theta) \ . \tag{A.2}$$

By using only "local information" about f (e.g., calls to f, ∇f, $\nabla^2 f$, etc., at a particular value), we can find minimum of $-f$ (and therefore a maximum of f) by iteratively following the negative gradient of f. That is, we start with some guess θ_0 and set

$$\theta_{t+1} \leftarrow \theta_t - s_t \nabla_\theta f(\theta_t) \ , \quad t = 0, 1, 2, \ldots, \tag{A.3}$$

until some termination criterion is met, e.g., $|\theta_{t+1} - \theta_t| \leq \epsilon$, or $\|\nabla_\theta f(\theta_t)\| \leq \epsilon$. In the update (A.3), s_t is a *step size* (also called a *learning rate* in some areas) that can make the optimization method more efficient, stable, etc.

The gradient-based update above is a **first-order** method, because it uses only the first-order derivative of f. It can be converted into a **second-order** method by setting s_t as the inverse of the Hessian matrix,

$$s_t = [\nabla^2_\theta f(\theta_t)]^{-1} \ , \tag{A.4}$$

which yields the **Newton–Raphson** method (also known as Newton's method). This uses the curvature (encoded by the inverse Hessian) to adjust the direction and magnitude of the step indicated by the gradient. As a rule of thumb, second-order methods converge in fewer iterations than first-order methods, but are computationally more expensive. For parametric models with parameter spaces of relatively small dimension, Newton–Raphson is the default optimization method. In many cases, the gradient and Hessian matrix can be derived analytically, allowing for easy numerical optimization. For a model with a large number of parameters (e.g., a deep neural network), inverting the Hessian matrix is not feasible (and the number of observations is so large that evaluating every term of the log-likelihood is also not feasible), and other methods (e.g., first-order methods, stochastic optimization, etc.) are used.

Exercise A.1. Let $f(\theta) = a\theta^2 + b\theta + c$, for $a > 0$ and $b, c, \theta \in \mathbb{R}$. Show that for arbitrary θ_0, the Newton–Raphson method converges to the minimum of f in one iteration.

Example A.1. The following is example code for maximizing the Gamma log-likelihood from Exercise 9.3.

A.1 Define functions

```r
#' Computes log-likelihood of Gamma distribution
#'
#' @param theta (alpha, beta) shape and rate parameters
#' @param x data
#' @return log-likelihood
loglik <- function(theta, x) {
  if (any(theta <= 0)) {
    return(Inf)
  } ## return Inf if any parameter is non-positive
  alpha <- theta[1]
  beta <- theta[2]
  mean(dgamma(x, shape = alpha, rate = beta, log = TRUE))
}

#' Computes gradient of log-likelihood of Gamma distribution
#'
#' @param theta (alpha, beta) shape and rate parameters
#' @param x data
#' @return gradient of log-likelihood
loglik_grad <- function(theta, x) {
  # computes gradient of log-likelihood; theta = (alpha,beta)
  alpha <- theta[1]
  beta <- theta[2]
  c(
    log(beta) - digamma(alpha) + mean(log(x)), # derivative w.r.t. alpha
    alpha / beta - mean(x) # derivative w.r.t. beta
  )
}

#' Computes Hessian of log-likelihood of Gamma distribution
#'
#' @param theta (alpha, beta) shape and rate parameters
#' @param x data (unused)
#' @return Hessian of log-likelihood
loglik_hess <- function(theta, x) {
  ## function computes Hessian of log-likelihood; theta = (alpha,beta)
  alpha <- theta[1]
  beta <- theta[2]
  hess_aa <- -trigamma(alpha)
  hess_bb <- -alpha / beta^2
  hess_ab <- 1 / beta

  matrix(c(hess_aa, hess_ab, hess_ab, hess_bb), nrow = 2, ncol = 2)
}

#' Wrapper function for optimization
#'
#' @param theta (alpha, beta) shape and rate parameters
#' @param x data
#' @return negative log-likelihood with gradient and Hessian attributes
f_optim <- function(theta, x) {
  ## wrapper function for loglik, gradient, hessian
  ## minus signs are because we will minimize rather than maximize
  ll <- -loglik(theta, x)
  attr(ll, "gradient") <- -loglik_grad(theta, x)
  attr(ll, "hessian") <- -loglik_hess(theta, x)
  ll
}
```

A.2 Run a simulation study to see how well the MLE performs

```r
### simulate data
suppressMessages(library(tidyverse))
set.seed(560)
n <- c(50, 500, 5000)
alpha_star <- 3
beta_star <- 4
X <- map(n, ~ rgamma(.x, shape = alpha_star, rate = beta_star))

# initialize with method of moments (try other initializations)
azero <- map(X, ~ mean(.x)^2 / (mean(.x^2) - mean(.x)^2))
bzero <- map2(azero, X, ~ .x / mean(.y))

for (i in seq_along(n)) {
  cat("n =", n[i], "\n")
  cat("true parameters: alpha =", alpha_star, "beta =", beta_star, "\n")
  cat("initial guess: alpha =", azero[[i]], "beta =", bzero[[i]], "\n")
  theta_hat <- nlm(
    f_optim,
    p = c(azero[[i]], bzero[[i]]),
    x = X[[i]],
    hessian = TRUE
  )
  cat(
    "MLE: alpha =",
    theta_hat$estimate[1],
    "beta =",
    theta_hat$estimate[2],
    "\n"
  )
  cat("eigenvalues of Hessian:", eigen(theta_hat$hessian)$values, "\n\n")
}
```

```
n = 50
true parameters: alpha = 3 beta = 4
initial guess: alpha = 3.720394 beta = 5.133903
MLE: alpha = 3.51822 beta = 4.854916
eigenvalues of Hessian: 0.4633945 0.01420146

n = 500
true parameters: alpha = 3 beta = 4
initial guess: alpha = 3.140145 beta = 3.997289
MLE: alpha = 3.285252 beta = 4.182005
eigenvalues of Hessian: 0.5248998 0.01820249

n = 5000
true parameters: alpha = 3 beta = 4
initial guess: alpha = 3.009751 beta = 4.047141
MLE: alpha = 3.012678 beta = 4.051078
eigenvalues of Hessian: 0.5563647 0.02011649
```

Exercise A.2. Derive the initial values of α, β used above using method of moments.

Exercise A.3. Use Exercise 9.3 to find the ML estimates by numerically optimizing only α.

Exercise A.4. Implement the Newton–Raphson method to find the MLE for $\theta = (\mu, \sigma)$ for $X_1, \ldots, X_n \sim_{\text{IID}} \mathcal{N}(\mu, \sigma^2)$, and compare to your answer to Exercise 9.2.

References

[EH21] B. Efron and T. Hastie. *Computer Age Statistical Inference*. Cambridge University Press, 2021.

[Sch95] M. J. Schervish. *Theory of Statistics*. Springer-Verlag New York, 1995.

[Was04] L. Wasserman. *All of Statistics*. Springer New York, 2004.