Overview: Estimation Methods

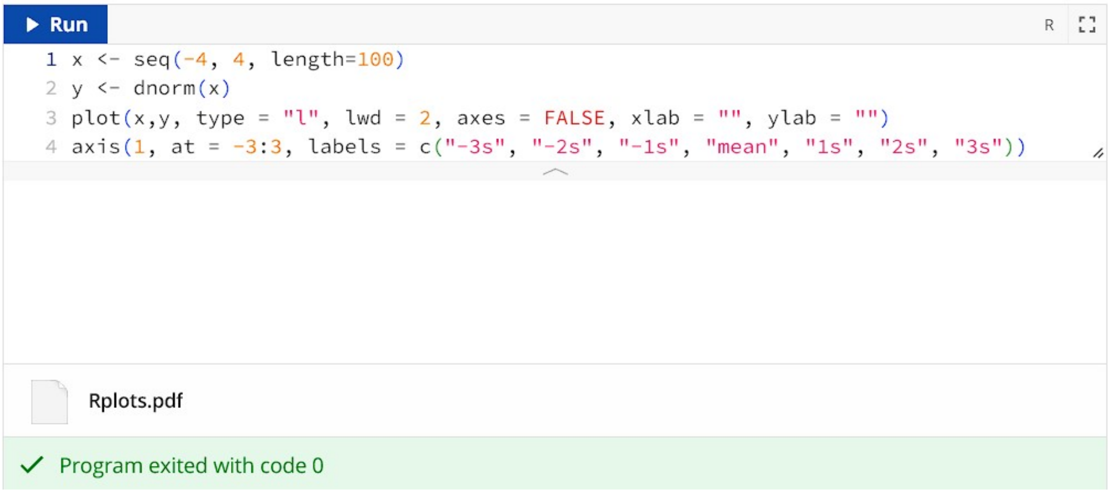
In predictive modelling or machine learning, we often use a model to describe the process which generates the observed data. For example, we can use a linear regression model to predict the selling price of houses in an area or we may use a decision tree to classify whether a customer will cancel a subscription from a telco service provider. Each model has a set of parameters which can determine its location, shape, scale and so on, and these parameters can be estimated from the observed data. The process of finding these parameter estimates is known as parameter estimation.

3.1 Maximum Likelihood Estimation

Introduction

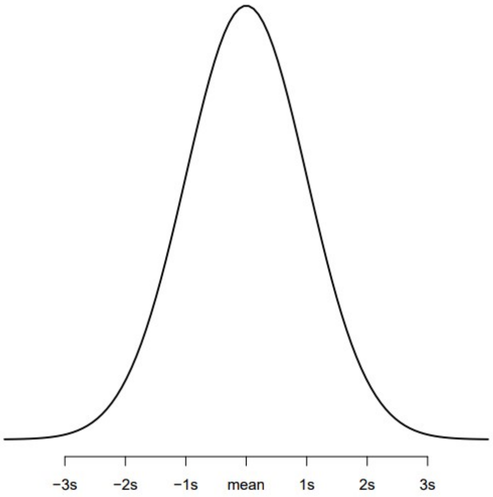
Maximum likelihood estimation (MLE) is one of the most popular parameter estimation methods in statistical modelling. MLE works by maximizing the likelihood function of the chosen model based on the observed data. The choice of the model is very important, and this decision could be based on the characteristics of the data or subject matter expertise.

For example, a variable with a bell-shaped distribution can be modelled using the normal distribution. Recall that a normal distribution has 2 parameters: the mean (location parameter) and the standard deviation (scale parameter). Therefore, there are two parameters which need to be estimated.



```r
x <- seq(-4, 4, length=100)
y <- dnorm(x)
plot(x,y, type = "l", lwd = 2, axes = FALSE, xlab = "", ylab = "")
axis(1, at = -3:3, labels = c("-3s", "-2s", "-1s", "mean", "1s", "2s", "3s"))
```

Program exited with code 0



Source: (Statology, 2019)
https://www.statology.org/plot-normal-distribution-r/

On the other hand, if a linear relationship is observed between a response variable and its predictor variable, then a linear regression model may be a good option. The parameters to be estimated in a linear regression model are the regression coefficients and the number of parameters depends on the number of predictors included in the model. For example, a linear regression model with two predictors would have a total of three parameters to be estimated from the fitted data set.

Maximum Likelihood Estimation

The parameter values estimated using maximum likelihood estimation are called maximum likelihood estimates. Watch the following video for an explanation on MLE using a very small data set as illustration.

Maximum Likelihood Estimation: How It Works
https://www.youtube.com/watch?v=RTFCh1QlHnY
Source: (selinalow, 2022)

The idea behind maximum likelihood estimation is to calculate the total probability of observing all of the data points, also known as the joint probability distribution. Before we further define the joint probability distribution in this case, an assumption to be made here is that the points are independent, which then simplifies the calculation considerably.

Recall from the probability axioms in Week 1 that if the independence assumption holds, then the joint probability is simply the product of the probability of observing each individual data point. This is also known as the product of the marginal probabilities.

For a normal distribution, we have seen in Week 1 that the probability density function for a single data point x is given by:

$$f(x) = \frac{e^{-(x-\mu)^2/2\sigma^2}}{\sigma\sqrt{2\pi}}$$

In our example, the joint probability density for the five independent points from our data set then becomes:

$$\prod_{i=1}^{n} l(\theta|x) = \prod_{i=1}^{5} f((\mu,\sigma)|x) = \frac{e^{-(x-\mu)^2/2\sigma^2}}{\sigma\sqrt{2\pi}}$$

In general, this joint probability formula given the observed values is known as the likelihood function $\mathrm{L} = \prod_{i=1}^{n} l(\theta|x)$. Maximum likelihood estimation then works by finding the values of μ and σ that gives the maximum of this likelihood function. If you have some experience with calculus, you may remember that this problem of finding the maxima of a function can be solved using differentiation. As the likelihood function itself is usually quite difficult to differentiate, the function is usually simplified by working with the natural logarithm of the function instead.

The natural logarithm of the likelihood function is known as the log-likelihood function.

Maximizing the log-likelihood function is equivalent to maximizing the likelihood function itself since the natural logarithm is a monotonically increasing function which ensures that the maximum of both functions concur. Therefore, in general we have the log-likelihood function defined as:

$$log\ L = log \prod_{i=1}^{n} l(\theta|x) = \sum_{i=1}^{n} log\ l(\theta|x)$$

As an example, for a set of n observations which are fitted with the normal distribution, the log-likelihood function will be:

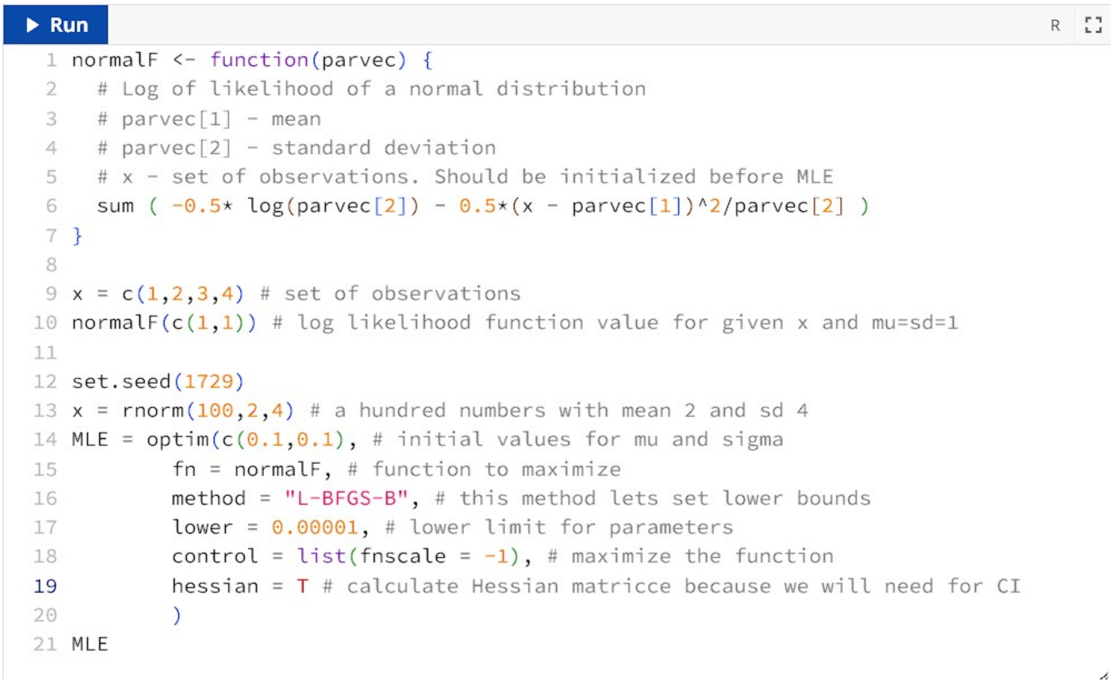
$$log\ L = log \prod_{i=1}^{n} f((\mu,\sigma)|x = \sum_{i=1}^{n} log \frac{e^{-(x-\mu)^2/2\sigma^2}}{\sigma\sqrt{2\pi}}$$

Maximum Likelihood For the Normal Distribution, step-by-step!!!

https://www.youtube.com/watch?v=Dn6b9fCIUpM

Source: (StatQuest with Josh Starner, 2018)

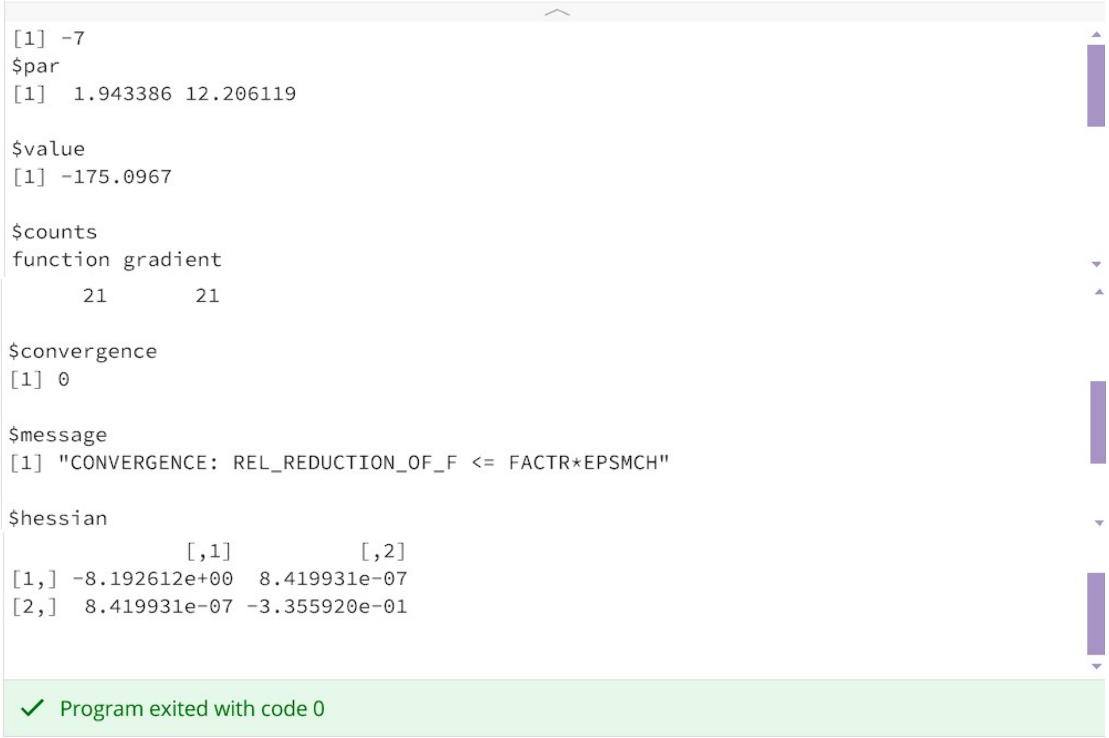
In R, the optim() function can be used together with a log-likelihood function to perform MLE. There are also other R functions to perform MLE but requires installation of packages. An example on MLE for normal distribution is in the R code below.



```r
normalF <- function(parvec) {
  # Log of likelihood of a normal distribution
  # parvec[1] - mean
  # parvec[2] - standard deviation
  # x - set of observations. Should be initialized before MLE
  sum ( -0.5* log(parvec[2]) - 0.5*(x - parvec[1])^2/parvec[2] )
}

x = c(1,2,3,4) # set of observations
normalF(c(1,1)) # log likelihood function value for given x and mu=sd=1

set.seed(1729)
x = rnorm(100,2,4) # a hundred numbers with mean 2 and sd 4
MLE = optim(c(0.1,0.1), # initial values for mu and sigma
            fn = normalF, # function to maximize
            method = "L-BFGS-B", # this method lets set lower bounds
            lower = 0.00001, # lower limit for parameters
            control = list(fnscale = -1), # maximize the function
            hessian = T # calculate Hessian matricce because we will need for CI
            )
MLE
```



```
[1] -7
$par
[1]  1.943386 12.206119

$value
[1] -175.0967

$counts
function gradient
      21       21

$convergence
[1] 0

$message
[1] "CONVERGENCE: REL_REDUCTION_OF_F <= FACTR*EPSMCH"

$hessian
              [,1]          [,2]
[1,] -8.192612e+00  8.419931e-07
[2,]  8.419931e-07 -3.355920e-01
```

Source: (Khitalishvili, 2017) [*https://rpubs.com/Koba/MLE-Normal*](https://rpubs.com/Koba/MLE-Normal)

In real world scenarios, more complex models may be required for the data set and more often than not the (log)-likelihood function for these complex models cannot be solved analytically as what we’ve seen in the video. In this case, approaches such as numerical optimization or Expectation-Maximization (EM) algorithms are used to find the maximum likelihood estimates.

3.2 Monte Carlo Estimation

Introduction

In Section 3.1, we discussed one of the most popular parameter estimation method known as maximum likelihood (ML) estimation which aims to maximize the (log)-likelihood function of a fitted model given the data set. Other commonly used parameter estimation methods include least squares estimation, method of moments and minimum distance estimation. When direct formulas for statistics are not available, numerical methods or approximate error analysis are adopted for nonlinear models. However, these approaches can give misleading results, such as when the numerical algorithm converges to a local optimum instead of the desired global maximum.

An alternative way to make numerical estimations of unknown parameters is to use Monte Carlo (MC) random sampling methods. Monte Carlo (MC) methods are computer intensive and rely heavily on random number generation.

Watch the following video for an introduction to Monte Carlo methods.

Monte Carlo Methods : Data Science Basics

https://www.youtube.com/watch?v=EaR3C4e600k

Source: (ritvikmath, 2022)

Applications of MC methods are found in various areas including portfolio optimization in finance (https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2438121), deep learning, computational biology and physical sciences. Before going into MC random sampling methods, we will first revisit Bayesian approach to statistics and estimation.

3.2.1 Bayesian Estimation

Introduction

In Bayesian parameter estimation, the parameter θ is interpreted as a random variable and the goal is to find the posterior distribution π(θ | x) of the parameters. Note here that the parameter θ is not fixed, but is treated as a random variable taking values in the parameter space Θ with its own probability density function. The posterior density for values of given the data set x is

$$\pi(\theta|x) = \frac{l(x|\theta)p(\theta)}{\int l(x|\theta)p(\theta)\, d\theta}$$

where $\mathrm{l}(\mathrm{x}|\theta)$ is the likelihood and $\mathrm{p}(\theta)$ is the prior distribution. The prior distribution is chosen based on information and beliefs about the possible values for θ, before any observation of data values. When the prior distribution is of the same type as the posterior density (which is quite common), it is known as conjugate prior. For example, the Beta distribution (https://chenxing.space/blog/beta-distribution-intuition-derivation-and-examples/) is the conjugate prior of binomial likelihood.

The integral in the denominator of this equation is the normalization constant. Whilst the terms in the numerator are usually easy to compute, the denominator poses a challenge especially when the parameter space is large.

The idea behind Bayesian estimation is that when the data values x are observed, the extra information about θ is combined with the prior to obtain the posterior distribution $\pi(\theta|\mathrm{x})$ for θ given x. Note that the left hand side of this equation is not a single point estimate of θ (which is what MLE gives us), but the entire posterior distribution.

Point-valued Estimates

Let $\mathrm{L}(\theta|a)$ be the loss incurred in estimating the value of a parameter to be a when the true value is θ . Common loss functions are:

- Quadratic loss, $L(\theta, a) = (\theta - a)^2$
- Absolute error loss, $L(\theta, a) = |\theta - a|$

When our estimate is a, the expected posterior loss is $h(a) = \int \mathrm{L}(\theta, a)\pi(\theta|\mathrm{x})\ \mathrm{d}\theta$. The Bayes' estimator minimizes the expected posterior loss. For example, when using the quadratic loss function we have $h(a) = \int (\theta - a)^2\pi(\theta|\mathrm{x})\ \mathrm{d}\theta$ and it can be shown that $h'(a) = 0$, if

$$a\int \pi(\theta|x)d\theta = \int \theta\pi(\theta|x)d\theta$$

Therefore, $\hat{\theta} = \int \theta\pi(\theta|x)d\theta$, the *posterior mean*, minimizes $h(a)$.

For absolute error loss, it can be shown that the Bayes' estimator $\hat{\theta}$ is the posterior median.

Another point estimate is the maximum a posteriori (MAP) estimator that maximizes the posterior density $\pi(\theta|y)$. On the other hand, maximum likelihood (ML) estimator maximizes the likelihood function $l(x|\theta)$ alone. Strictly speaking, this is not a Bayesian notion since it ignores the prior information.

Interval-ranged Estimates

In principle, the posterior distribution gives the solution to the parameter estimation problem in a fully probabilistic sense. We can find the peak of the probability density, and determine, for example, the 95% credibility regions for the parameters. However, working with the posterior density directly is challenging, since the normalization constant often cannot be computed analytically and classical numerical integration methods are not feasible if the number of parameters is large. This is where Markov chain Monte Carlo methods come into the picture.

Bayesian Statistics: An Introduction

https://www.youtube.com/watch?v=Pahyv9i_X2k

Source: (Zedstatistics 2018)

3.2.2 Markov Chain Monte Carlo

Markov chain Monte Carlo (MCMC) methods are used in statistical inference for model parameters without having to explicitly computing the difficult integral as seen earlier in the posterior density in Bayesian estimation. MCMC methods aim at generating a sequence of random samples $(\theta_1, \theta_2, \ldots, \theta_N)$, whose distribution asymptotically approaches the posterior distribution as *N* increases. This means that the posterior density is not used directly, but samples from the posterior distribution are produced instead. The samples are generated so that each new point only depends on the previous point , and the samples therefore form a Markov Chain. It can be shown that the samples approach the correct target (posterior). Watch the following video for an explanation on MCMC methods.

Markov Chain Monte Carlo (MCMC) : Data Science Concepts

https://www.youtube.com/watch?v=yApmR-c_hKU

Source: (ritvikmath, 2021)

Metropolis Algorithm

The Metropolis algorithm is one of the most widely used MCMC algorithms. It works by generating candidate parameter values from the proposal distribution and then either accepting or rejecting the proposed value according to a simple rule. With this algorithm, we only need to compute ratios of posterior densities and the normalization constant cancels out.

The main challenge in the implementation of the Metropolis algorithm is defining the proposal distribution *q*. The proposal should be chosen so that it is easy to sample from and 'close' to the underlying target distribution. An unsuitable proposal can lead to inefficient implementation:

- If the proposal is too large, the new candidates mostly miss the essential support and are only rarely accepted.
- If the proposal is too small, the new candidates are mostly accepted, but from a small neighbourhood of the previous point, and the chain moves slowly

Metropolis - Hastings : Data Science Concepts

https://www.youtube.com/watch?v=yCv2N7wGDCw

Source: (ritvikmath, 2021)

Case Study Activity: Estimation Methods

This activity provides opportunities to explore estimation methods in real-life applications.

Time: 45 minutes

Purpose: The purpose of this activity is to illustrate application of estimation methods in real-life applications.

Task: Choose and read one of the following articles below which illustrates the application of estimation methods in real life applications. Note down your observations on the following:

1. What is the data set or variables involved in the analysis?
2. What are the challenge(s) identified in the analysis?
3. What estimation method(s) have been discussed or proposed in the article?

Some of the articles include advanced statistical concepts in its explanation. Knowledge on these advanced concepts is optional in this subject. Instead, focus your reading on the answers to the questions above as listed in the task.

Suchoski, B., Stage, S., Gurung, H., & Baccam, P. (2022). GPU accelerated parallel processing for large-scale Monte Carlo analysis: COVID-19 Parameter Estimation and New Case Forecasting. Frontiers in Applied Mathematics and Statistics, 8. https://www.frontiersin.org/journals/applied-mathematics-and-statistics/articles/10.3389/fams.2022.818016

Figini, S. (2010). Penalized models are used to estimate customer survival. Statistical Methods & Applications, 19(1), 141–150. https://research.ebsco.com/linkprocessor/plink?id=26100098-000b-3b55-9b64-ef413bf41039

Hirz, J., Schmock, U., & Shevchenko, P. V. (2017). Actuarial applications and estimation of extended CreditRisk+. Risks, 5(2), 23. https://doi.org/10.3390/risks5020023

Liu, X., Xia, C., Tang, Y., Tu, J., & Wang, H. (2021). Parameter optimization and uncertainty assessment for rainfall frequency modeling using an adaptive Metropolis–Hastings algorithm. Water Science and Technology, 83(5), 1085–1102. https://doi.org/10.2166/wst.2021.032

Ban, Z., Ghaderi, A., Janatian, N., & Pfeiffer, C. F. (2022). Parameter estimation for a gas lifting oil well model using Bayes' rule and the Metropolis–Hastings algorithm. Modeling, Identification and Control, 43(2), 39–53. https://doi.org/10.4173/mic.2022.2.1