Computational methods

Thomas Achia

2/9/2022

## Analytical and numerical approaches to problems

See [https://math.stackexchange.com/questions/935405/what-s-the-difference-between-analytical-and-numerical-approaches-to-problems#](https://math.stackexchange.com/questions/935405/what-s-the-difference-between-analytical-and-numerical-approaches-to-problems):~:text=Numerical%20methods%20use%20exact%20algorithms,the%20use%20of%20numerical%20methods.

**Analytical approach example:**

Find the root of

**Analytical solution**: add to both sides to get the answer

**Numerical solution:**

let’s guess A negative number.

Let’s guess A positive number.

The answer must be between them. Let’s try

So it must be between 72 and 6…etc.

This is called **bisection method.**

* Numerical solutions are extremely abundant.
* The main reason is that sometimes we either don’t have an analytical approach (try to solve or that the analytical solution is too slow and instead of computing for 15 hours and getting an exact solution, we rather compute for 15 seconds and get a good approximation.

# NOTE

* Numerical methods use exact algorithms to present numerical solutions to mathematical problems.
* Analytic methods use exact theorems to present formulas that can be used to present numerical solutions to mathematical problems with or without the use of numerical methods.
* Analytical method gives exact solutions, more time consuming and sometimes impossible.
* Whereas numerical methods give approximate solution with allowable tolerance, less time and possible for most cases

**Analytical Method**

* When a problem is solved by means of analytical method its solution may be exact.
* it doesn’t follow any algorithm to solve a problem
* This method provides exact solution to a problem
* These problems are easy to solve and can be solved with pen and paper

**Numerical Method**

* When a problem is solved by mean of numerical method its solution may give an approximate number to a solution
* It is the subject concerned with the construction, analysis and use of algorithms to solve a problem
* It provides estimates that are very close to exact solution
* It can’t be solved with pen and paper but can be solved via computer tools like R, Python, FORTRAN or C++

# Bisection Method

* The bisection method is the easiest to numerically implement and almost always works.
* The main **disadvantage** is that **convergence is slow**.
* If the bisection method results in a computer program that runs too slow, then other faster methods may be chosen; otherwise it is a good choice of method.

We want to construct a sequence that converges to the root that solves

* We choose and such that
* We say that and bracket the root.
* With we want and to be of opposite sign, so that

# Bisection Method-Cont.

We then assign to be the midpoint of and that is

or

The sign of can then be determined.

The value of is then chosen as either the midpoint of and or as the midpoint of and depending on whether and bracket the root, or and bracket the root.

The root, therefore, stays bracketed at all times. The algorithm proceeds in this fashion and is typically stopped when the increment to the left side of the bracket (above, given by

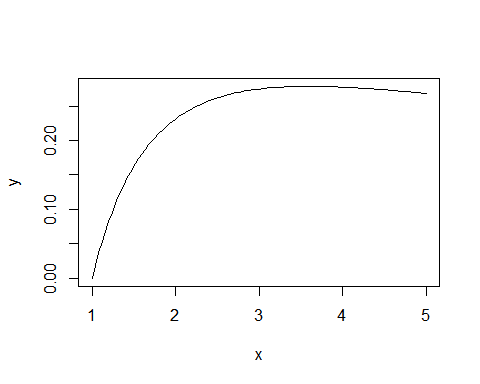
is smaller than some required precision.

# EXAMPLE-Bisection method

Suppose that we wish to implement the bisection method to maxima of the function

The plot of the function suggests that the maxima is potentially between and See the following:

curve(log(x)/(1+x), from=1, to=5, , xlab="x", ylab="y")

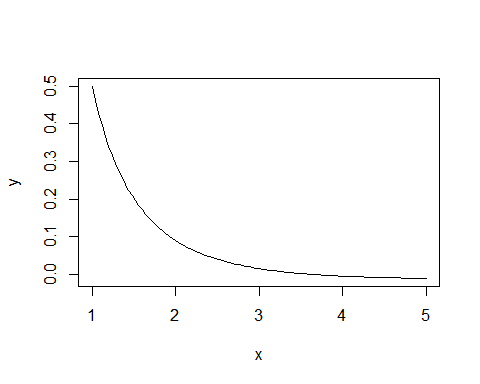


# Example-Cont.

Turning points for this curve will satisfy That is,

A plot of this first derivatives is:

curve((1+(1/x)-log(x))/(1+x)^2, from=1, to=5, , xlab="x", ylab="y")



# Example-R code

To find maxima of the equation is thus equivalent to finding the root of A plot the function has helped identify the bounds wherein the solution lies.

In this code **a** is the initial left endpoint, **b** is the initial right endpoint, **x** is the initial value, **itr** provides the number of iterations to run, **g** is the objective function, and **g.prime** is first derivative of objective function.

## INITIAL VALUES  
a = 1  
b = 5  
x = a+(b-a)/2  
itr = 40  
  
## FUNCTIONS  
f = function(x){log(x)/(1+x)}  
f.prime = function(x){(1+(1/x)-log(x))/((1+x)^2)}  
  
## MAIN  
for (i in 1:itr){  
 if (f.prime(a)\*f.prime(x) < 0) {b = x}  
 else {a = x}  
 x = a+(b-a)/2  
}

# The Output

x # FINAL ESTIMATE

## [1] 3.591121

f(x) # OBJECTIVE FUNCTION AT ESTIMATE

## [1] 0.2784645

f.prime(x) # GRADIENT AT ESTIMATE

## [1] -1.121895e-14

# Newton’s Method

* This is a faster method, but requires analytical computation of the derivative of
* Also, the method **may not always converge** to the desired root.
* We can derive Newton’s Method graphically, or by a Taylor series.
* We again want to construct a sequence that converges to the root
* Consider the member of this sequence, and Taylor series expand about the point We have
* To determine we drop the higher-order terms in the Taylor series, and assume That is,

Solving for we have

Starting Newton’s Method requires a guess for hopefully close to the root

# EXAMPLE-NEWTON’S METHOD

We continue with the previous example where we seek the maxima for the function which is equivalent to seeking the root of it’s first derivative

To solve using Newton’s method requires the we find so that our iterative formula is:

Now

# The R-code

In the code **x** is the initial value, **itr** is the number of iterations to run, **g** is the objective function, **g.prime** is the first derivative of objective function, and **g.2prime** is second derivative of objective function.

## INITIAL VALUES  
x = 3  
itr = 40  
  
## FUNCTIONS  
f = function(x){log(x)/(1+x)}  
f.prime = function(x){(1+(1/x)-log(x))/((1+x)^2)}  
f.2prime = function(x){(-1/((x^2)+(x^3)))-2\*(1+(1/x)-log(x))/((1+x)^3)}  
  
## MAIN  
for(i in 1:itr){x = x - f.prime(x)/f.2prime(x)}

# OUTPUT

x # FINAL ESTIMATE

## [1] 3.591121

f(x) # OBJECTIVE FUNCTION AT ESTIMATE

## [1] 0.2784645

f.prime(x) # GRADIENT AT ESTIMATE

## [1] 1.053423e-17

# Secant Method

* The Secant Method is second best to Newton’s Method, and is used when a faster convergence than Bisection is desired, but it is too difficult or impossible to take an analytical derivative of the function .

We write in place of

so that our iterative formula becomes

Starting the Secant Method requires a guess for both and

# Estimate using Newton’s Method

The is the zero of the function

To implement Newton’s Method, we use Therefore, Newton’s Method is the iteration

We take as our initial guess Then

# Order of convergence

Let be the root and be the th approximation to the root.

Define the error as

If for large we have the approximate relationship

with a positive constant, then we say the root-finding numerical method is of order Larger values of correspond to faster convergence to the root. The order of convergence of bisection is one: the error is reduced by approximately a factor of 2 with each iteration so that

We now find the order of convergence for Newton’s Method and for the Secant Method.

# Newton’s Method

We start with Newton’s Method

Subtracting both sides from we have

or

We use Taylor series to expand the functions and about the root r, using We have

To make further progress, we will make use of the following standard Taylor series:

which converges for

Therefore, we have shown that

as with

provided Newton’s method is thus of order at simple roots.

# Secant Method

Determining the order of the Secant Method proceeds in a similar fashion. We start with

We subtract both sides from r and make use of

and the Taylor series

so that

Since

we have

or to leading order

The order of convergence is not yet obvious from this equation, and to determine the scaling law we look for a solution of the form

From this ansatz, we also have

and therefore

This leads to

Equating the coefficient and the power of en−1 results in

and

The order of convergence of the Secant Method, given by p, therefore is determined to be the positive root of the quadratic equation

or

which coincidentally is a famous irrational number that is called The Golden Ratio, and goes by the symbol We see that the Secant Method has an order of convergence lying between the Bisection Method and Newton’s Method.

# MULTIVARIATE PROBLEMS

In a multivariate optimization problem we seek the optimum of a real-valued function of a -dimensional vector At iteration denote the estimated optimum as

Many of the general principles discussed above for the univariate case also apply for multivariate optimization. Algorithms are still iterative. Many algorithms take steps based on a local linearization of derived from a Taylor series or secant approximation. Convergence criteria are similar in spirit despite slight changes in form. To construct convergence criteria, let be a distance measure for dimensional vectors.

Two obvious choices are

and

Then absolute and relative convergence criteria can be formed from the inequalities

# Newton’s Method and Fisher Scoring

To fashion the Newton’s method update, we again approximate by the quadratic Taylor series expansion

and maximize this quadratic function with respect to to find the next iterate.

Setting the gradient of the right-hand side equal to zero yields

This provides the update

Alternatively, note that the left-hand side of (2.32) is in fact a linear Taylor series approximation to g(x∗), and solving (2.32) amounts to finding the root of this linear approximation.

From either viewpoint, the multivariate Newton increment is

# Elements of statistical inference

# Random sample

* If are independent random variables having a common distribution , then we say that they constitute a sample (sometimes called a random sample) from the distribution .
* This means that:

# Statistic

Suppose that is a random sample from

* If the function is such that is a random variable, then
* is called a **statistic**.
* A **dimensional statistic** is a vector
* where is a dimensional statistic.

# Parametric point estimation

Point estimation of an unknown parameter is done in the following way:

* The observed values of a random sample are use to suggest an approximation for of the form where is such that is a statistic.
* The random variable
* is called an **estimator** of .
* a numeric value of is called an **estimate** for

# Problems

* What are methods to construct estimators for a given parameter?
* How do we construct an estimator which is *‘best’* in some sense?

# Some general properties of estimators

* Unbiasedness
* Consistency
* Asymptotic normality
* Sufficiency
* Efficiency
* Uniform minimum variance estimator

# Unbiased estimators

* An estimator is said to be an **unbiased estimator** for if
* for all

# The multidimensional generalization

An estimator is said to be an **unbiased estimator** for if

for all

# Example

Suppose that is a random sample from and that and

The **sample mean** is:

and the **sample variance** is:

# Example-Expected value of

The expected value of the **sample mean** is:

Since the sample mean is an unbiased estimator of the population mean.

# Example-Variance of

The variance of the **sample mean** is:

# Example-Expected value of

The **sample variance** is:

Then the expected value of is:

Thus the sample variance is an unbiased estimator of the population variance

# Bias

If is an estimator of then the quantity:

is called the **Bias** of

**Example**

Consider the estimator:

This estimator is usually referred to as the Maximum likelihood estimator of

Now so that

The bias of is:

**Weakly consistent estimator**

An estimator is said to be a **weakly consistent estimator** for if, for

## Maximum likelihood estimation

Method of Moments

Let be a where is a vector of parameters. The likelihood function:

The log-likelihood function

The maximum likelihood estimator of should satisfy:

Thus

Now is usual called the score vector. That is,

And so the MLE for can be found by equating the score to zero. That is,

The variance of an ML estimator, , is calculated by the inverse of the Information matrix:

The Information matrix is the negative of the expected value of the Hessian matrix:

where Hessian is the matrix of second derivatives of the likelihood with respect to the parameters:

Thus, the variance-covariance matrix of is:

Example. The Bernoulli distribution.

Consider a random sample of size from the distribution. The probability density function here is:

Our likelihood and log-likelihood functions are:

and

The score function is:

That is,

Equating this to zero, we find

Example. **The Poisson distribution.**

Consider a random sample of size from the distribution. The probability density function here is:

Our likelihood and log-likelihood functions are:

and

The score function is:

That is,

Equating this to zero, we find