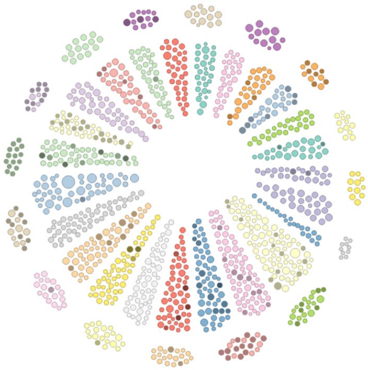
Numerical Integration

Marvin Gauer

Numerical Introductory Course
Humboldt University to Berlin

Table of Contents



Motivation

Solving integrals in a closed form is often not applicable or consumes to much time in real world applications. Therefore procedures need to be in place which solve definite integrals numerically to a certain point certainty.
In statistics one might for example want to know the probabilities of a normal distribution. Since the probability density function of the normal distribution has no antiderivative, one needs to solve this integral numerically.

Review: The Riemann Integral

Aim: Calc. the area below a certain limited function $f:[a,b]\to\mathbb{R}$.

Steps:

- Create a decomposition Z from $[a,b]$ with $a=x_0<x_1<\ldots<x_n=b$
- Calculate the upper and lower sum
  $U(Z):=\sum_{k=1}^{n}((x_k-x_{k-1})\cdot\sup\limits_{x_{k-1}<x<x_k} f(x))$
  $L(Z):=\sum_{k=1}^{n}((x_k-x_{k-1})\cdot\inf\limits_{x_{k-1}<x<x_k} f(x))$
- f is Riemann integrable if $\sup\limits_{Z} L(Z)=\inf\limits_{Z} U(Z)$

Review: The Riemann Integral

If a certain function f is integrable, then one might use the following rules to find the antiderivative:

- Power Rule: $\int x^n dx = \frac{x^{n+1}}{n+1} + C$
- Integration by Parts
  $\int f'(x) \cdot g(x) dx = f(x) \cdot g(x) - \int f(x) \cdot g'(x) dx$
- Integration by Substitution $\int f(x) dx = \int f(\phi(u)) \cdot \phi'(u) du$

Review: The Riemann Integral

⚠ This might get messy!

Review: The Riemann Integral

⚠ This might get messy!

Therefore one might need/prefer numerical methods

Methods

The methods presented can be classified in the following ways:

- One-Dimensional Integrals
- Multidimensional Integrals

- Deterministic Methods
- Probabilistic Methods

One-Dimensional Integrals

For one-dimensional integrals lots of deterministic numerical procedurees were established. The ones we want to focuse on are:

- Midpoint- or Rectangular-Quadrature
- Simpson-Rule
- Adaptive Algorithm

Midpoint- or Rectangular-Quadrature

Algorithm:

- Partition $[a,b]$ into m subintervals with $a = x_0 < x_1 < ... < x_m = b$
- Calc. the midpoint $x^{(k)}$ of each subinterval $[x_i, x_{i+1}]$ with $i \in \{0,1,..,m\}$
- Then it holds $\int_a^b f(x)dx \approx \sum_{k=1}^m f(x^{(k)}) \cdot (x_{i+1} - x_i)$

Simpson-Rule

Algorithm:

- Partition $[a,b]$ into m subintervals with $a = x_0 < x_1 < ... < x_m = b$
- Calc. the midpoint $x^{(k)}$ of each subinterval $[x_i, x_{i+1}]$ with $i \in \{0, 1, .., m\}$
- Interpolate a quadratic function through the points $(x_i, f(x_i))$, $(x^{(k)}, f(x^{(k)}))$ and $(x_{i+1}, f(x_{i+1}))$
- Then it holds $\int_a^b f(x)dx \approx \frac{\Delta x}{6} \cdot (f(x_o) + 2 \cdot \sum_{j=1}^{m-1} f(x_j) + f(x_m) + 4 \cdot \sum_{k=1}^{m} f(x^{(k)}))$

Adaptive Algorithm

When integrating numerically one has to decide on the number of partitions m of $[a, b]$. A to high m will consume a lot of computational power, while a to low m might yield a quite unsatisfactoring estimate of our integral of interest.

.

Adaptive Algorithm

When integrating numerically one has to decide on the number of partitions m of $[a, b]$. A to high m will consume a lot of computational power, while a to low m might yield a quite unsatisfactoring estimate of our integral of interest.

Solution: Adaptively refine the number of subintervals m until the error ϵ reaches the desired level.

Adaptive Algorithm

Algorithm:

- Initialize m and calculate $Q \approx \int_a^b f(x)dx$ with a method of choice
- Calc. the approximation error $\epsilon \approx |Q - \int_a^b f(x)dx|$
- If ϵ is to large increase m and repeat until ϵ is as desired

Multidimensional Integrals

Aim: Calc. $\int_\Omega f(\mathbf{x})d\mathbf{x} = \int_{a_1}^{b_1}\int_{a_2}^{b_2}...\int_{a_m}^{b_m} f(x_1, x_2, ..., x_m)dx_1 dx_2...dx_m$

Idea: Use presented Quadrature rules, like Rectangular Quadrature, in more dimensions!

Multidimensional Integrals

Aim: Calc. $\int_\Omega f(\mathbf{x})d\mathbf{x} = \int_{a_1}^{b_1}\int_{a_2}^{b_2}...\int_{a_m}^{b_m} f(x_1, x_2, ..., x_m)dx_1 dx_2...dx_m$

~~Idea: Use presented Quadrature rules, like Rectangular Quadrature, in more dimensions!~~

Curse of dimensionality $\Rightarrow$ others methods need to be used

Multidimensional Integrals

For multidimensional integrals lots of stochastic/deterministic numerical proceduree were developed. The ones we want to focuse on are:

- Monte Carlo Integration
  - Hit or Miss
  - Crude

Monte Carlo - Hit or Miss

Algorithm:

- Step 1: Create n m+1-dimensional uniformly distributed points $\mathbf{r}_i = (x_1, ..., x_m, y_i)\ \forall\ i = 1, .., n$
- Step 2: Determine the percentage of points for which holds $y_i \leq f(x_1, ..., x_m)$ denoted as p
- Step 3: Calculate the area of the rectangular $[a, b] \times [0, max(f(\mathbf{x}))]$ denoted as A
- Step 4: $\int_a^b f(\mathbf{x}) dx \approx A \cdot p$

Monte Carlo - Crude

Algorithm:

- Step 1: Create k bins $[a_{1,j}, b_{1,j}] \times [a_{2,j}, b_{2,j}] \times ... \times [a_{m,j}, b_{m,j}]$ $\forall$ $j = 1, ..., k$ and n m-dimensional uniformly distributed points $\mathbf{x}_i = (x_1, ..., x_m)$ $\forall$ $i = 1, .., n$
- Step 2: Calculate $f(\mathbf{x}_i)$ $\forall$ $i = 1, ..., n$ and the corresponding mean m_i of the simulated points within each bin
- Step 3: $\int_a^b f(\mathbf{x})dx \approx \sum_{t=1}^{k} (\prod_{l=1}^{m} (b_{l,t} - a_{l,t}) \cdot m_t)$

Integrals over infinite intervals

All formulas presented so far are applicable for definite integrals, but one might also be interested in numerically solving improper integrals.
We want to present two methods to overcome this issue:

- Truncation
- Change of Variables

Integrals over infinite intervals - Truncation

Method: Instead of integrating to infinity one chooses a large fixed number in order to use methods presented earlier

Pros:

- Easy to understand and implement
- Further knowledge of the function might be necessary

Cons:

- Not very scientific
- Might be hazardous in certain applications

Integrals over infinite intervals - CoV

Method: Transform the integrand such that the boundaries are finite

Case 1: $\int_{-\infty}^{\infty} f(x)dx = \int_{-1}^{1} f\left(\frac{t}{1-t}\right) \cdot \frac{1+t^2}{(1-t^2)^2} dt$

Case 2: $\int_{a}^{\infty} f(x)dx = \int_{0}^{1} f\left(a + \frac{t}{1-t}\right) \cdot \frac{1}{(1-t)^2} dt$

Case 3: $\int_{-\infty}^{a} f(x)dx = \int_{0}^{1} f\left(a - \frac{1-t}{t}\right) \cdot \frac{1}{t^2} dt$

Applications

Numerical Integration Methods are widely used by practitioners. Some of the most well known/most used applications are:

- Approximation of probabilities of a normal distribution
- Approximation of antiderivatives
- Calculation of Moments
- Calculation of the Value at Risk and Expected Shortfall

Approx. of the normal distributions CDF

Algorithm:

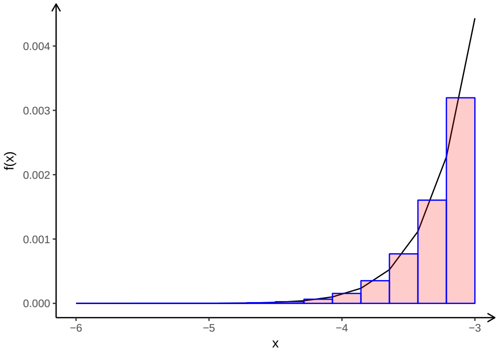
- Step 1: Decide on the sampling points $(x_1, .., x_N)$ and quadrature method
- Step 2: Use the pdf to approx. evaluate $y_i = \int_{-\infty}^{x_i} f(x)dx \; \forall \; i = 1, .., N$
- Step 3: Decide on a suited interpolation function and interpolate the points to retrieve $\hat{\mathrm{F}}(\mathrm{x})$

Approx. of the normal distributions CDF

Step 1: Midpoint Rule for $(-3, -2, -1, 0, 1, 2, 3)$ as sampling points and corresponding $(15, 30, 45, 60, 75, 90, 105)$ bins. Trunction is used to handle the improper integral (cut off at -6)

Approx. of the normal distributions CDF

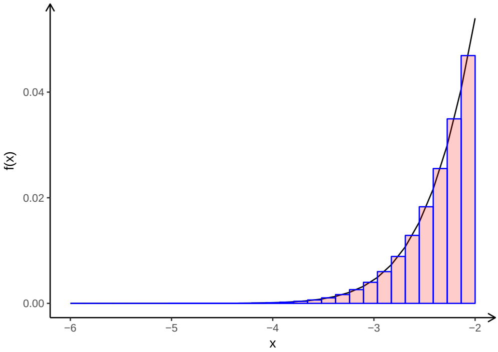
Step 2: Calculate $y_i = \int_{-\infty}^{x_i} f(x)dx \ \forall \ i = 1,..,N$



$x_1 = -3$ and $y_1 = 0.001324663$

Approx. of the normal distributions CDF

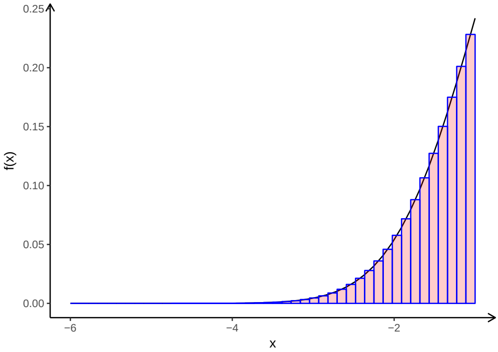
Step 2: Calculate $y_i = \int_{-\infty}^{x_i} f(x)dx \ \forall \ i = 1,..,N$



$x_1 = -2$ and $y_1 = 0.022664581$

Approx. of the normal distributions CDF

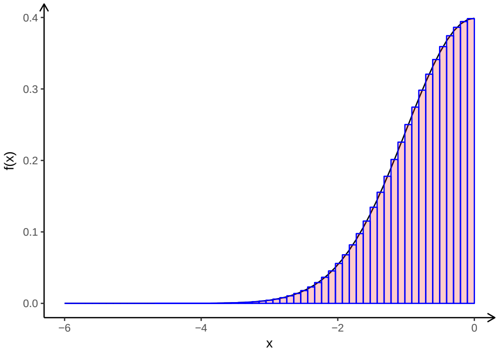
Step 2: Calculate $y_i = \int_{-\infty}^{x_i} f(x)dx \;\forall\; i = 1, .., N$



$$x_1 = -1 \text{ and } y_1 = 0.158524962$$

Approx. of the normal distributions CDF

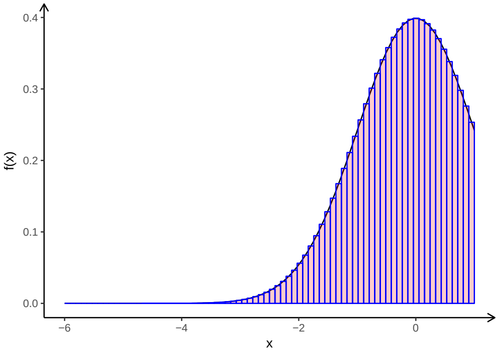
Step 2: Calculate $y_i = \int_{-\infty}^{x_i} f(x)dx \ \forall \ i = 1,..,N$



$x_1 = 0$ and $y_1 = 0.499999999$

Approx. of the normal distributions CDF

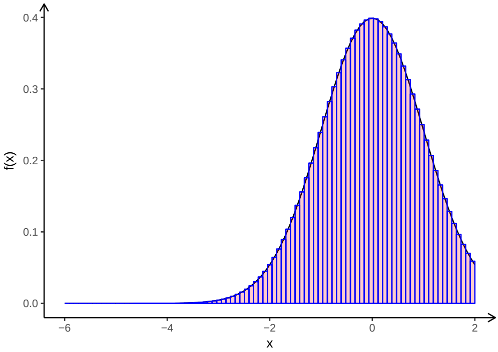
Step 2: Calculate $y_i = \int_{-\infty}^{x_i} f(x)dx \ \forall \ i = 1,..,N$



$x_1 = 1$ and $y_1 = 0.841435008$

Approx. of the normal distributions CDF

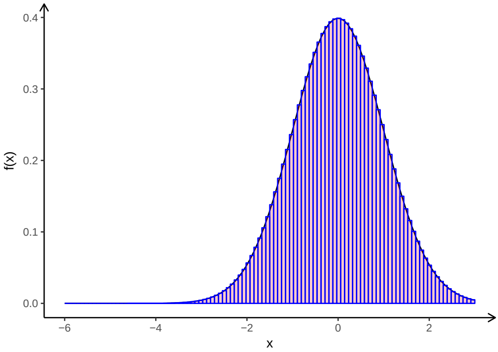
Step 2: Calculate $y_i = \int_{-\infty}^{x_i} f(x)dx \ \forall \ i = 1,..,N$



$x_1 = 2$ and $y_1 = 0.977286211$

Approx. of the normal distributions CDF

Step 2: Calculate $y_i = \int_{-\infty}^{x_i} f(x)dx \ \forall \ i = 1,..,N$



$x_1 = 3$ and $y_1 = 0.998654244$

Approx. of the normal distributions CDF

Step 3: Decide on a suited interpolation function and interpolate the points to retrieve $\hat{F}(x)$

In our example we use $\widehat{F}(x) = d + c \cdot x + b \cdot x^2 + a \cdot x^3$ for simplicity, which yields:

$$\widehat{F}(x) = 0.4999835 + 0.3239174 \cdot x - 0.01778339 \cdot x^3$$

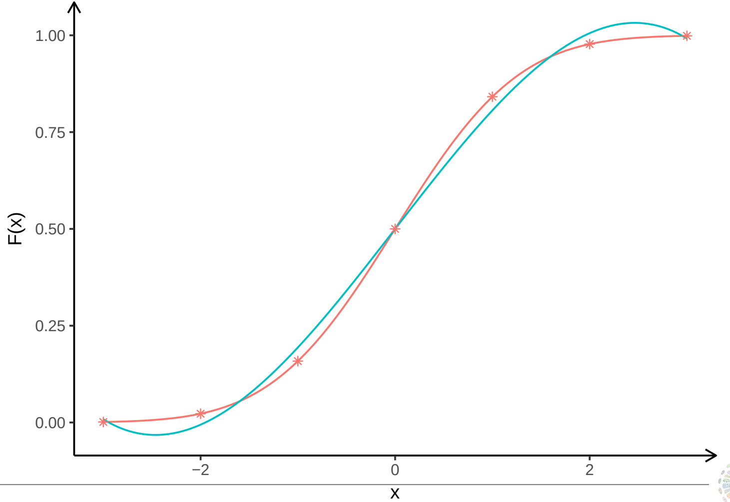
Approx. of the normal distributions CDF

x	-3	-2	-1	0	1	2
$F(x)$[1]	0.0013	0.0227	0.1586	0.5	0.8413	0.9772
$\hat{F}(x)$	0.0083	-0.005	0.1938	0.4999	0.8061	1.0055
Difference	-0.007	0.0283	-0.035	0	0.0352	-0.0283

r	1.47	-2.49	-3.43	-2.82	2.60	2.15
$F(x)$[1]	0.929	0.006	0.0003	0.002	0.995	0.984
$\hat{F}(x)$	0.919	-0.031	0.108	-0.014	1.029	1.019
Difference	0.009	0.038	-0.107	0.016	-0.034	-0.035

[1] Based on R's pnorm()

Approx. of the normal distributions CDF



R-Code

Bibliography

- **Monte-Carlo-Integration**. Retrieved from https://www.mathematik.tu-clausthal.de/interaktiv/integration/montecarlo/
- **Das Riemann-Integral**. Retrieved from http://www.mathematik.uni-wuerzburg.de/ kanzow/analysis1/Kapitel7.pdf
- **Numerical Integration: One Dimension**. Retrieved from https://www.value-at-risk.net/numerical-integration-in-one-dimension/