

# Lesson 22: Integrals and Series

```
> restart;  
with(Student[Calculus1]): with(IntegrationTools):
```

## An oscillatory integral

We were looking at this improper integral, which converges due to rapid oscillation.

```
> J := Int(x*cos(x^3), x=0..infinity);
```

$$J := \int_0^{\infty} x \cos(x^3) dx \quad (1.1)$$

Maple's symbolic value for this integral was

```
> Jtrue := value(J);
```

$$J_{true} := \frac{1}{6} \Gamma\left(\frac{2}{3}\right)$$

We split the interval into two parts, did a change of variables on the infinite part, and then some integrations by parts.

```
> with(IntegrationTools);
```

```
[Change, CollapseNested, Combine, Expand, ExpandMultiple, Flip, GetIntegrand,  
GetOptions, GetParts, GetRange, GetVariable, Parts, Split, StripOptions]
```

```
> S:=Split(J,1);
```

$$S := \int_0^1 x \cos(x^3) dx + \int_1^{\infty} x \cos(x^3) dx$$

```
> J1:= op(1,S); J2:= op(2,S);
```

$$J1 := \int_0^1 x \cos(x^3) dx$$

$$J2 := \int_1^{\infty} x \cos(x^3) dx \quad (1.2)$$

```
> V1:= evalf(ApproximateInt(x*cos(x^3),x=0..1,method=simpson,  
partition=50));
```

$$V1 := 0.4404076893 \quad (1.3)$$

```
> J3:= Change(J2,x=t^(1/3));
```

$$J3 := \int_1^{\infty} \frac{1}{3} \frac{\cos(t)}{t^{1/3}} dt$$

```
> J4:= Parts(J3,t^(-1/3));
```

$$J4 := -\frac{1}{3} \sin(1) - \left( \int_1^{\infty} \left( -\frac{1}{9} \frac{\sin(t)}{t^{4/3}} \right) dt \right) \quad (1.4)$$

> J5 := Parts(J4, t^(-4/3));

$$J5 := -\frac{1}{3} \sin(1) + \frac{1}{9} \cos(1) + \int_1^{\infty} \left( -\frac{4}{27} \frac{\cos(t)}{t^{7/3}} \right) dt$$

(1.5)

> for k from 0 to 7 do

    J5 := Parts(J5, t^(-k-7/3))

end do;

$$J5 := -\frac{5}{27} \sin(1) + \frac{1}{9} \cos(1) - \left( \int_1^{\infty} \frac{28}{81} \frac{\sin(t)}{t^{10/3}} dt \right)$$

$$J5 := -\frac{5}{27} \sin(1) - \frac{19}{81} \cos(1) + \int_1^{\infty} \frac{280}{243} \frac{\cos(t)}{t^{13/3}} dt$$

$$J5 := -\frac{325}{243} \sin(1) - \frac{19}{81} \cos(1) - \left( \int_1^{\infty} \left( -\frac{3640}{729} \frac{\sin(t)}{t^{16/3}} \right) dt \right)$$

$$J5 := -\frac{325}{243} \sin(1) + \frac{3469}{729} \cos(1) + \int_1^{\infty} \left( -\frac{58240}{2187} \frac{\cos(t)}{t^{19/3}} \right) dt$$

$$J5 := \frac{55315}{2187} \sin(1) + \frac{3469}{729} \cos(1) - \left( \int_1^{\infty} \frac{1106560}{6561} \frac{\sin(t)}{t^{22/3}} dt \right)$$

$$J5 := \frac{55315}{2187} \sin(1) - \frac{1075339}{6561} \cos(1) + \int_1^{\infty} \frac{24344320}{19683} \frac{\cos(t)}{t^{25/3}} dt$$

$$J5 := -\frac{23846485}{19683} \sin(1) - \frac{1075339}{6561} \cos(1) - \left( \int_1^{\infty} \left( -\frac{608608000}{59049} \frac{\sin(t)}{t^{28/3}} \right) dt \right)$$

$$J5 := -\frac{23846485}{19683} \sin(1) + \frac{598929949}{59049} \cos(1) + \int_1^{\infty} \left( -\frac{17041024000}{177147} \frac{\cos(t)}{t^{31/3}} \right) dt$$

> J6 := Change(J5, t=1/u);

$$J6 := -\frac{23846485}{19683} \sin(1) + \frac{598929949}{59049} \cos(1) - \frac{17041024000}{177147} \int_0^1 \cos\left(\frac{1}{u}\right) u^{25/3} du$$

This time the integrand has four continuous derivatives, so Simpson ought to work.

> diff(u^(25/3)\*cos(1/u), u\$4);

$$\cos\left(\frac{1}{u}\right) u^{1/3} - \frac{64}{3} \sin\left(\frac{1}{u}\right) u^{4/3} - \frac{608}{3} \cos\left(\frac{1}{u}\right) u^{7/3} + \frac{26752}{27} \sin\left(\frac{1}{u}\right) u^{10/3} \\ + \frac{167200}{81} \cos\left(\frac{1}{u}\right) u^{13/3}$$

Again, we can get around the fact that the integrand is undefined at  $u = 0$  by using **piecewise**.

> g := piecewise(u=0, 0, cos(1/u)\*u^(25/3));

$$g := \begin{cases} 0 & u = 0 \\ \cos\left(\frac{1}{u}\right) u^{25/3} & \text{otherwise} \end{cases} \quad (1.6)$$

```
> A50:= evalf(ApproximateInt(g,u=0..1,method=simpson,partition=50));
```

```
A50 := 0.04637355746
```

```
> s50:= evalf(V1 + eval(J6,Int(cos(1/u)*u^(25/3),u=0..1)=A50));
```

```
S50 := 0.224007
```

```
> A100:= evalf(ApproximateInt(g,u=0 .. 1,method=simpson,partition=100));
```

```
A100 := 0.04637354102
```

```
> s100:= evalf(V1 + eval(J6,Int(cos(1/u)*u^(25/3),u=0..1)=A100));
```

```
S100 := 0.225589
```

```
> SR := (2^4*s100-s50)/(2^4-1);
```

```
SR := 0.2256944667
```

```
> evalf(Jtrue-SR);
```

```
-0.0000081435
```

Not very accurate. Let's calculate A100 and A200. I'll also redo V1 with partition = 100 and partition=200; also increase Digits to reduce roundoff error.

```
> Digits:= 15; V1100 := evalf(ApproximateInt(x*cos(x^3),x=0..1,method=simpson,partition=100));
```

```
Digits := 15
```

```
V1100 := 0.440407691238742
```

(1.7)

```
> V1200:= evalf(ApproximateInt(x*cos(x^3),x=0..1,method=simpson,partition=200));
```

```
V1200 := 0.440407691373017
```

(1.8)

```
> A100:= evalf(ApproximateInt(g,u=0..1,method=simpson,partition=100));
```

```
A100 := 0.0463735410758500
```

```
> s100:= evalf(V1100 + eval(J6,Int(cos(1/u)*u^(25/3),u=0..1)=A100));
```

```
S100 := 0.22558103094
```

```
> A200:= evalf(ApproximateInt(g,u=0 .. 1,method=simpson,partition=200));
```

```
A200 := 0.0463735400497191
```

```
> s200:= evalf(V1200 + eval(J6,Int(cos(1/u)*u^(25/3),u=0..1)=A200));
```

```
S200 := 0.22567974187
```

```
> SR := (2^4*s200-s100)/(2^4-1);
```

```

SR := 0.225686322598667
> evalf(Jtrue-SR);
6.39067 10-10
> evalf(J);
0.225686323237734 (1.9)
> evalf(Jtrue-J);
0. (1.10)

```

## add, sum and Sum

Maple has three different commands for producing a sum: **add**, **sum** and **Sum**. The simplest is **add**, which just adds a given finite number of terms.

```

> restart;
add(t^j, j = 1..20);
t + t2 + t3 + t4 + t5 + t6 + t7 + t8 + t9 + t10 + t11 + t12 + t13 + t14 + t15 + t16 + t17 + t18 + t19
+ t20

```

Then there is **sum**, which looks for a formula for the sum. This can be used for both finite and infinite sums, i.e. series.

```

> sum(t^j, j = 1..20);
t + t2 + t3 + t4 + t5 + t6 + t7 + t8 + t9 + t10 + t11 + t12 + t13 + t14 + t15 + t16 + t17 + t18 + t19
+ t20
> sum(t^j, j=1..n);
tn+1
-----
t-1 - t
-----
t-1

```

```

> add(t^j, j=1..n);
Error, unable to execute add
> sum(t^j, j=1..infinity);

```

$$-\frac{t}{t-1}$$

If Maple can't find a formula it just returns the sum unevaluated.

```

> sum(exp(-k^2), k=0..n);
sum(exp(-k^2), k=0..infinity);

```

$$\sum_{k=0}^n e^{-k^2}$$

$$\sum_{k=0}^{\infty} e^{-k^2}$$

For a finite sum, **sum** and **add** generally return the same result, but there can be problems with "premature evaluation". Consider the following Vector:

```

> v := <1,2,3,4,5>;

```

$$V := \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix} \quad (2.1)$$

You want to add  $V_n$  for  $n$  from 1 to 5. This works fine:

```
> add(V[n], n=1..5);
```

$$15 \quad (2.2)$$

But this doesn't:

```
> sum(V[n], n=1..5);
```

Error, bad index into Vector

The problem is that **sum** starts out, like nearly all Maple procedures, by evaluating its inputs. In this case the first one is  $V[n]$ , so it tries to evaluate  $V_n$ . But  $n$  is a symbolic variable. There is no such thing (to Maple) as  $V_n$  just  $V_1, V_2, \dots, V_5$ . So it causes an error. This problem doesn't occur with **add**, because that has special evaluation rules so it only evaluates its first input with integer values substituted for the index variable.

And finally there is the inert form **Sum**.

```
> Sum(t^j, j=1..20);
```

$$\sum_{j=1}^{20} t^j$$

You can use **value** to turn **Sum** into **sum**:

```
> value(%);
```

$$t + t^2 + t^3 + t^4 + t^5 + t^6 + t^7 + t^8 + t^9 + t^{10} + t^{11} + t^{12} + t^{13} + t^{14} + t^{15} + t^{16} + t^{17} + t^{18} + t^{19} + t^{20}$$

To get numerical values, you can use **evalf**. This can deal with an infinite **Sum** (or a **sum** that Maple can't find a formula for) using numerical methods.

```
> evalf(Sum(1/k^2, k=1..infinity));
```

$$\text{sum}(1/k^2, k=1..infinity); \text{evalf}(\%);$$

$$1.644934067$$

$$\frac{1}{6} \pi^2$$

$$1.644934068$$

```
> sum(exp(-k^2), k=0..infinity); evalf(%);
```

$$\sum_{k=0}^{\infty} e^{-k^2}$$

$$1.386318602 \quad (2.3)$$

It's not always successful, though.

```
> evalf(Sum(1/(k^2+cos(k)), k=0..infinity));
```

$$\sum_{k=0}^{\infty} \frac{1}{k^2 + \cos(k)}$$

## Convergence of series

Consider an infinite series

$$S = \sum_{k=0}^{\infty} a_k$$

The n'th partial sum of the series is

$$S_n = \sum_{k=0}^n a_k$$

The main theoretical question about a series is whether or not it converges.

The sum of an infinite series is, by definition, the limit of the partial sums  $S_n$ , i.e.

$$S = \lim_{n \rightarrow \infty} S_n$$

That limit must exist (as a finite number) in order for the sum to exist. Otherwise, we say the series diverges.

In a case where the partial sums go to  $\infty$  we might write  $\sum_{k=0}^{\infty} a_k = \infty$ , but strictly speaking we should say this series diverges to  $\infty$ .

Maple can sometimes figure this out, but not always. For a series that diverges to  $\infty$ , it might give a result of  $\infty$ :

```
> sum(1/k,k=1..infinity);
```

$\infty$

But not always:

```
> sum(1/(k+ln(k)),k=1..infinity);
```

$$\sum_{k=1}^{\infty} \frac{1}{k + \ln(k)}$$

A series that diverges, but not to  $+\infty$  or  $-\infty$ , will likely just return unevaluated:

```
> sum((-2)^k,k=0..infinity);
```

$$\sum_{k=0}^{\infty} (-2)^k$$

But that's also what you get for a convergent series where Maple just doesn't know a formula for it. And **evalf** sometimes returns a finite answer for divergent series.

```
> evalf(%);
```

0.3333333333

There is an environment variable **\_EnvFormal** that is supposed to make **sum** try harder to detect divergent series when it's set to false. It isn't always reliable, though.

I'll do a restart so Maple won't remember previous results it got without **\_EnvFormal** set to false.

```
> restart; _EnvFormal:=false;
sum(1/(k+ln(k)),k=1..infinity);
```

```
_EnvFormal := false  
∞
```

```
> evalf(Sum((-2)^n, n=0..infinity));  
Float(undefined)
```

That answer of 1/3 wasn't complete nonsense. It is actually the continuation of a correct formula outside the region where it is correct.

```
> Sum(a^n, n=0..infinity) = sum(a^n, n=0..infinity);  

$$\sum_{n=0}^{\infty} a^n = -\frac{1}{a-1}$$

```

That is true if  $|a| < 1$ . If we plug in  $a = -2$ :

```
> eval(%, a=-2);  

$$\sum_{n=0}^{\infty} (-2)^n = \frac{1}{3}$$

```

If *\_EnvFormal* is set to true, Maple does even more of this sort of thing.

```
> restart; _EnvFormal := true;  
_EnvFormal := true  
> sum((-2)^n, n=0..infinity);  

$$\frac{1}{3}$$

```

So how do we tell whether a series converges or diverges?

There are a number of "convergence tests" that can be used: the integral test, comparison test, ratio test, root test and alternating series test are the ones that tend to be found in calculus texts. Maple can be used as a tool in applying any of them. For example, the **ratio test** says the following:

If  $\left| \frac{x_{n+1}}{x_n} \right|$  has a limit  $L < 1$  as  $n \rightarrow \infty$ , then  $\sum_{n=1}^{\infty} x_n$  converges.

If it has a limit  $L > 1$  (possibly  $\infty$ ), then the series diverges.

If  $L = 1$  or there is no limit, the ratio test is inconclusive.

Maple can find limits with the **limit** command.

For example, let's test the series  $\sum_{k=1}^{\infty} \frac{3^{k^2}}{k!}$

```
> x := k -> 3^(k^2)/k!;  

$$x := k \rightarrow \frac{3^{k^2}}{k!}$$
  
> ratio := x(k+1)/x(k);  

$$ratio := \frac{3^{(k+1)^2} k!}{(k+1)! 3^{k^2}}$$
  
> limit(ratio, k=infinity);  
∞
```

So this one diverges. It's not hard to see why.

```
> simplify(ratio);
```

$$\frac{3 \cdot 9^k}{k+1}$$

As  $k \rightarrow \infty$ ,  $9^k$  grows faster than  $k+1$ , so the ratio goes to  $\infty$ .

Next, I'll try  $\sum_{k=1}^{\infty} \frac{k^k}{k!^2}$

```
> x := k -> k^k/(k!)^2;
```

$$x := k \rightarrow \frac{k^k}{k!^2}$$

```
> ratio := x(k+1)/x(k);
```

$$\text{ratio} := \frac{(k+1)^{k+1} k!^2}{(k+1)!^2 k^k}$$

```
> limit(ratio, k=infinity);
```

0

```
> ratio := simplify(ratio);
```

$$\text{ratio} := (k+1)^{k-1} k^{-k}$$

This one converges. What is the sum?

```
> S:=sum(x(k),k=1..infinity);
```

$$S := \sum_{k=1}^{\infty} \frac{k^k}{k!^2}$$

Maple doesn't know a formula for it. We can get a numerical value using **evalf**.

```
> evalf(S);
```

3.548128226

## Approximating with the ratio test

Let's look at some partial sums of this series.

```
> seq( evalf(Sum(x(k), k=1 .. N)), N = 1 .. 10);
```

1., 2., 2.750000000, 3.194444444, 3.411458333, 3.501458333, 3.533879244, 3.544199224,  
3.547141317, 3.547900723

This makes Maple's answer look plausible. The partial sums appear to be approaching a limit that is approximately **evalf**'s answer. If we want to approximate the sum, we could use the partial sum for an appropriate  $N$ . What  $N$  should we use to get the sum  $S$  with an error of at most  $10^{-10}$ ?

We want to estimate  $R(N) = S - \left( \sum_{k=1}^N x(k) \right) = \sum_{k=N+1}^{\infty} x(k)$ , the "tail" of the series.

The ratio test gives us a clue for this: the ratio was  $\frac{x(k+1)}{x(k)} = (k+1)^{k-1} k^{-k}$

Note that  $(k+1)^k k^{-k} = \left(1 + \frac{1}{k}\right)^k$  goes to a limit of  $e$  as  $k \rightarrow \infty$ . It is in fact less than  $e$  for every  $k$ :  $\ln\left(\left(1 + \frac{1}{k}\right)^k\right) = k \ln\left(1 + \frac{1}{k}\right) \leq k \frac{1}{k} = 1$ .

So the ratio is less than  $\frac{e}{k+1}$ .

This means that for  $k \geq N \geq 10$ , we have the bound  $0 \leq x(k) \leq \left(\frac{e}{11}\right)^{k-N} x(N)$

And so for the tail of the series, we have an estimate:

$$0 \leq R(N) \leq \sum_{k=N+1}^{\infty} \left(\frac{e}{11}\right)^{k-N} x(N)$$

By the way, the letter  $e$  is not anything special in Maple input. To get the constant  $e$ , you have to use `exp(1)`.

The right side is a geometric series.

```
> sum((exp(1)/11)^(k-N)*x(N),k=N+1 .. infinity);
```

$$-\frac{11 \left(\frac{1}{11} e\right)^{-N} N^N \left(\frac{1}{11} e\right)^{N+1}}{N!^2 (e-11)}$$

```
> EstimatedR := simplify(%);
```

$$\text{EstimatedR} := -\frac{e N^N}{N!^2 (e-11)}$$

We want this estimated R to be less than  $10^{-10}$ .

```
> fsolve(EstimatedR = 10^(-10), N);
18.58532274
```

So  $N = 19$  should do it.

```
> evalf(eval(EstimatedR, N=19));
4.388369724 10^-11
```

```
> evalf(add(x(k),k=1..19));
3.548128226
```

```
> evalf(S);
3.548128226
```

```
> evalf(S-Sum(x(k),k=1..19), 20);
2.02590524 10^-11
```

That was a series that converged very quickly. For series that converge slowly, approximating the sum can be more difficult.

## Maple objects introduced in this lesson

sum

**Sum**  
**\_EnvFormal**