Maple Lecture 20. Series, Approximations, and Limits

This lecture corresponds to [1, Chapter 11]. In this lecture we will work with series, which are a symbolic way to calculate approximately. We see how to define Maple functions from the generating function of a series and how to turn a truncated series approximation into a function. Maple can formally work with power series. We end this lecture considering limits.

20.1 Truncated Series

The most well-known series are Taylor series around a point:

```
[> tf := taylor(f(x),x=a,2);
[> whattype(tf);
```

Observe how the structure of a series differs from that of a polynomial:

```
[> dismantle(tf);
```

With series we can define functions. For example, you could define the Fibonacci numbers by

```
[> g := z/(1-z-z^2);                # generating function
[> fibser := taylor(g,z=0,15);
```

To extract the 10-th Fibonacci number, we use the coeff command:

```
[> coeff(fibser,z^10);
```

But we can only go as far as the series goes:

```
[> coeff(fibser,z^15);
```

Fortunately, Maple provides us with a nice function to compute only the coefficient :

```
> coeftayl(g,z=0,15);
```

Unfortunately, be advised that this is not so efficient...

```
[> fibfun := n -> coeftayl(g,z=0,n);
[> fibfun(10);
[> fibfun(17);
[> fibfun(20);
```

20.2 Approximations of Functions

Maple is powerful enough to compute series expansions of functions defined by integrals.

```
[> intfun := int(exp(sin(x)),x=0..t);
[> serfun := series(intfun,t=2,3);
```

To evaluate these series we have to convert first to a polynomial:

```
[> polser := convert(serfun,'polynom');
```

Then we turn the polynomial into a function

```
[> polfun := unapply(polser,t);
```

and finally we see there is still something missing

```
[> polfun(2.34);
```

because we would like to see an actual numerical value

```
[> myfun := (t,n) -> evalf(polfun(t),n);
[> myfun(2.34,30);
```

20.3 Power Series

We can view rational numbers as a completion of the integer numbers, fixing the problem of dividing two integers. Similarly, formal power series can be seen as an extension of polynomials. We speak of "formal" series, because we are not worried about convergence issues.

Maple offers a toolkit to operate with those formal power series, for example:

```
[> with(powseries);
[> powcreate(defexp(n) = 1/n!);
[> serexp := tpsform(defexp,n,10);
[> polexp := convert(serexp,`polynom`);
[> funexp := unapply(polexp,n);
[> evalf(funexp(1)); evalf(exp(1));
```

20.4 Limits

Series expansions are often used in the calculation of limits. For example, it is known that the series $1 - 1/3 + 1/5 - 1/7 + \cdots$ converges to $\pi/4$.

```
[> powcreate(f(n)=(-1)^(n)/(2*n+1));
[> fseries := tpsform(f,n,7);
[> p := convert(fseries,`polynom`);
[> fp := unapply(p,n);
```

Here we get a very crude approximation of π, the convergence of the series is very slow.

```
[> fp(1), 4*evalf(fp(1));
```

We get stuck if we want to take the limit of the series, instead we have to

```
[> s := Sum('(-1)^n/(2*n+1)','n'=0..infinity);
[> value(s);
[> s1 := n -> Sum('(-1)^i/(2*i+1)','i'=0..n);
```

Here we see the alternating convergence:

```
[> for n from 20 to 30 do
|>    evalf(s1(n));
[> end do;
```

The purpose of the above construction is to indicate the difference between a sum and a series. In a sum, the parameter is typically "n", used to indicate the place of the term in the sum. In a series, the parameter is typically "x", which can vary continuously and is to be seen as the argument of some function that was approximated by the series.

20.5 Assignments

1. Give the Maple commands

   (a) to create a fifth order Taylor series expansion of `exp(x)` about `x = 3`, and to assign this series to the variable `ts`;

   (b) to convert the series `ts` into a polynomial `pts`;

   (c) to convert the polynomial `pts` into a function `fts`;

   (d) What is the value of `fts(3.5)`?

2. Use Maple to verify $\displaystyle\frac{1}{\sqrt{1-4x}} = \sum_{n\geq 0} \left(\begin{array}{c} 2n \\ n \end{array} \right) x^n$. Give Maple commands

   (a) to compute a fifth order Taylor series of the left of the equality above about $x = 0$;

   (b) to create a function which for any n returns the coefficient of x^n in the Taylor series about $x = 0$;

   (c) to compare for $n = 40$ the result of the function you just created with the built-in command to compute $\left(\begin{array}{c} 2n \\ n \end{array} \right)$.

3. The binomial coefficient $\left(\begin{array}{c} n+k \\ k \end{array} \right)$, used to count all choices of k elements from a set of $n+k$ elements, can be defined as the k-th coefficient in the series expansion of $g(z) = \frac{1}{(1-z)^{n+1}}$ at $z = 0$. Use the series expansion of $g(z)$ at $z = 0$ to define the function $b(k)$, which returns this binomial coefficient for variable n.

   As $b(k)$ returns a formula, for some n, use $b(k)$ to define $c(n,k)$ which returns a number, when called like $c(5,3)$. Compare with `binomial(5,3)`.

4. Consider the expression

   $$\lim_{n\rightarrow\infty} \sum_{i=0}^{n} x^i$$

   (a) Give the Maple command to create the expression above. Assign the result to the variable `explim`.

   (b) Give the Maple command to convert the mathematical formula `explim` into the function `funlim` in x.

   (c) Evaluate `funlim` at x = 0.3 and give the value with 20 decimal places accuracy.

5. Consider $\displaystyle\sum_{n=1}^{\infty} \frac{(-1)^n e^{\beta n}}{n^2}$. Give the Maple commands

   (a) to compute the symbolic value of the sum;

   (b) to find a numerical approximation with 15 decimal places for $\beta = -2$;

   (c) to make a function that returns a numerical approximation (with the default precision of 10 decimal places) for the sum, for any value of β.

References

[1] A. Heck. *Introduction to Maple.* Springer-Verlag, third edition, 2003.