Defining L-functions in GP

A tutorial

B. Allombert

IMB
CNRS/Université de Bordeaux

10/01/2024

Riemann ζ function

```
? Zeta = lfuncreate(1)
%1 = [[Vecsmall([1]),1],0,[0],1,1,1,1]
? lfun(Zeta,2)
%2 = 1.6449340668482264364724151666460251892
? lfun(Zeta,0,1)
%3 = -0.91893853320467274178032973640561763986
? lfun(Zeta,1)
%4 = 1.0000000000*x^-1+O(x^0)
? lfun(Zeta,1+x+O(x^10))
%5 = 1.000000000*x^-1+0.5772156+0.0728158*x-0.00484
? lfunzeros(Zeta,20)
%6 = [14.134725141734693790457251983562470271]
? lfunlambda(Zeta,2)
%7 = 0.52359877559829887307710723054658381403
```

Dirichlet L functions

```
? G=znstar(4,1); G.clgp
%8 = [2, [2], [3]]
? Dir=lfuncreate([G,[1]]); lfunparams(Dir)
%9 = [4,1,[1]]
? lfunan(Dir,30)
%10 = [1,0,-1,0,1,0,-1,0,1,0,-1,0,1,0,-1,0,1,0,-1,0
? lfun(Dir,2)
%11 = 0.91596559417721901505460351493238411078
? Catalan
%12 = 0.91596559417721901505460351493238411077
? znconreyexp(G,[1])
%13 = 3
? lfun(Mod(3,4),2)
%14 = 0.91596559417721901505460351493238411078
```

Dedekind ζ functions

```
? Dedek = lfuncreate(x^2+1); lfunparams(Dedek)
%15 = [4,1,[0,1]]
? lfun(Dedek,2)
%16 = 1.5067030099229850308865650481820713960
? lfuneuler(Dedek,3)
%17 = 1/(-x^2+1)
? zeta(2)*Catalan
%18 = 1.5067030099229850308865650481820713960
? L=lfunmul(Zeta,Mod(3,4)); lfun(L,2)
%19 = 1.5067030099229850308865650481820713960
? L2=lfundiv(Dedek,1); lfun(L2,2)
%20 = 0.91596559417721901505460351493238411078
? lfuneuler(L2,3)
%21 = 1/(x+1)
```

Hecke L functions

```
? bnf = bnfinit(a^2+23);
? bnr = bnrinit(bnf, 1); bnr.clgp
%23 = [3,[3]]
? Hecke = lfuncreate([bnr,[1]]);
? lfunparams(Hecke)
%25 = [23,1,[0,1]]
? lfuneuler(Hecke,5)
%26 = 1/(-x^2+1)
? z=lfun(Hecke,0,1)
%27 = 0.28119957432296184651205076406787829979+0.E-
? algdep(exp(z),3)
%28 = x^3-x-1
```

Abelian extension

Let L/K an Abelian extension. It is possible to build ζ_L directly from class field theory parameters.

```
? bnr = bnrinit(bnf, 23); bnr.clgp
%29 = [759,[759]]
? ZL = lfuncreate([bnr,Mat(11)]);
? lfunparams(ZL)
%31 = [3947158412069548588724958962З,1,[0,0,0,0,0,0
? lfuneuler(ZL, 5)
%32 = 1/(-x^22+1)
? lfun(ZL,1)
%33 = 0.20543374142343149269037323981188134330*x^-1
```

Elliptic curves

For the elliptic curve $y^2 + y = x^3 - x^2 - 10x - 20$:

```
? E = ellinit([0,-1,1,-10,-20]);
? lfun(E,1)
%35 = 0.25384186085591068433775892335090946104
? lfuneuler(E,5)
%36 = 1/(5*x^2-x+1)
? ellbsd(E)
%37 = 0.25384186085591068433775892335090946104
? lfunparams(E)
%38 = [11,2,[0,1]]
? L = lfunsympow(E,2); lfunparams(L)
%39 = [121,3,[0,0,1]]
```

Elliptic curves over number fields

We define the elliptic curve $y^2 + xy + \phi x = x^3 + (\phi + 1)x^2 + x$ over the field $\mathbb{Q}(\sqrt{5})$ where $\phi = \frac{1+\sqrt{5}}{2}$.

```
? nf  = nfinit(a^2-5);
? phi = (1+a)/2;
? E   = ellinit([1,phi+1,phi,phi,0],nf);
? E.j
%43 = Mod(-53104/31*a-1649/31,a^2-5)
? E.disc
%44 = Mod(-8*a+17,a^2-5)
? N   = ellglobalred(E)[1]
%45 = [31,13;0,1]
? tor = elltors(E) \\ Z/8Z
%46 = [8,[8],[[Mod(-1,a^2-5),Mod(-1/2*a+1/2,a^2-5)]
```

Elliptic curves over number fields

We check the BSD conjecture for E.

```
? om = E.omega
%47 = [[3.05217315,-2.39884477*I],
%       [8.43805989,4.21902994-1.57216679*I]]
? per = om[1][1]*om[2][1];
? tam = elltamagawa(E)
%49 = 2
? bsd = (per*tam) / (tor[1]^2*sqrt(abs(nf.disc)))
%50 = 0.3599289594980394494002575466348575048
? ellbsd(E)
%51 = 0.3599289594980394494002575466348575048
? L1  = lfun(E,1)
%52 = 0.3599289594980394494002575466348575048
```

lfuntwist

lfuntwist allows to twist an L function by a Dirichlet character. The conductors need to be coprime.

```
? E = ellinit([0,-1,1,-10,-20]);
? L=lfuntwist(E,Mod(2,5));
? lfunan(E,10)
%55 = [1,-2,-1,2,1,2,-2,0,-2,-2]
? lfunan(Mod(2,5),10)
%56 = [1,I,-I,-1,0,1,I,-I,-1,0]
? lfunan(L,10)
%57 = [1,-2*I,I,-2,0,2,-2*I,0,2,0]
```

lfuntwist

We redefine the curve over $\mathbb{Q}(\zeta_5)$.

```
? nf=nfinit(polcyclo(5,'a));
? E2=ellinit(E[1..5],nf);
? localbitprec(64); lfun(E2,2)
%60 = 1.0543811873412420765
? L2=lfuntwist(E,Mod(4,5));
? lfun(E,2)*lfun(L2,2)*norm(lfun(L,2))
%62 = 1.0543811873410821651289745964738865962
```

Genus-2 curve

For the genus-2 curve $y^2 + (x^3 + 1)y = x^2 + x$:

```
? L=lfungenus2([x^2+x,x^3+1]);
? lfunparams(L)
%64 = [249,2,[0,0,1,1]]
? lfun(L,1)
%65 = 0.13154950701147875921340134301217526069
? lfunan(L,5)
%66 = [1,-2,-2,1,0]
```

Galois group

We start with a Galois extension of the rationals, here $\mathbb{Q}(\sqrt[3]{2}, \zeta_3) = \mathbb{Q}(\sqrt[6]{-108})$, with Galois group isomorphic to $\mathfrak{S}_3$.

```
? N = nfinit(x^6+108);
? G = galoisinit(N);
```

G is the Galois group of *N*.

Linear representation

```
? [T,o] = galoischartable(G);
? T~
%70 = [1  1  1]
%     [1  1 -1]
%     [2 -1  0]
```

T is the character table of $G \cong \mathfrak{S}_3$, which is defined over $\mathbb{Z}$. The first character is related to the trivial representation, the second to the signature, and the third to a faithful irreducible representation of dimension 2.
The ordering of the conjugacy classes is given by `galoisconjclasses(G)`.

```
? galoisconjclasses(G)
%71 = [[Vecsmall([1,2,3,4,5,6])],[Vecsmall([3,1,2,6
```

Artin L-function

We compute the Artin function associated to the 3rd character.

```
? L = lfunartin(N,G,T[,3],o);
? lfuncheckfeq(L)
%73 = -138
? L[2..5]
%74 = [0, [0, 1], 1, 108]
? z = lfun(L,0,1)
%75 = 1.3473773483293841009181878914456530463
? algdep(exp(z),3)
%76 = x^3-3*x^2-3*x-1
```

which suggests that this function is equal to a Hecke L-function.

```
? bnr = bnrinit(bnfinit(a^2+a+1),6);
? lfunan([bnr,[1]],100)==lfunan(L,100)
%78 = 1
```

A more interesting example

Let E be a model of the curve $X_0(11)$
$E : y^2 + y = x^3 - x^2 - 10x - 20$, we build the field $\mathbb{Q}(E[3])$ generated by the coordinates of the points of 3-torsions.

```
? E=ellinit([0,-1,1,-10,-20]);
  \\ or ellinit("11a1") if elldata is available
? P=elldivpol(E,3)
%80 = 3*x^4-4*x^3-60*x^2-237*x-21
? Q=polresultant(P,y^2-elldivpol(E,2))
%81 = 27*y^8+108*y^7-4813*y^6-14817*y^5+162543*y^4+
? R=nfsplitting(Q)
%82 = y^48-36*y^46+558*y^44-4588*y^42+24549*y^40-11
```

This defines a Galois extension of $\mathbb{Q}$ with Galois group $\mathrm{GL}_2(\mathbb{F}_3)$.

Non monomial representation

```
? N=nfinit(R); G=galoisinit(N);
? [T,o]=galoischartable(G); T~
%84 = [1,1,1,1,1,1,1,1;
%       1,1,-1,1,1,1,-1,-1;
%       2,0,-y^3-y,1,-1,-2,0, y^3+y;
%       2,0, y^3+y,1,-1,-2,0,-y^3-y;
%       2,2,0,-1,-1,2,0,0;
%       3,-1,-1,0,0,3,1,-1;
%       3,-1,1,0,0,3,-1,1;
%       4,0,0,-1,1,-4,0,0]
? o
%85 = 8
```

$o = 8$ means that the variable y denotes a 8-th root of unity.

Non monomial representation

```
? minpoly(Mod(y^3+y, polcyclo(o,y)))
%86 = x^2+2
```

So the coefficients are in $\mathbb{Q}(\sqrt{-2})$. We use the third irreducible representation.

```
? L = lfunartin(N,G,T[,3],o);
? L[2..5]
%88 = [1,[0,1],1,3267]
? lfuncheckfeq(L)
%89 = -127
```

Determinant

```
? dT = galoischardet(G,T[,3],o)
%90 = [1,1,-1,1,1,1,-1,-1]~
? dL = lfunartin(N,G,dT,o);
? dL[2..5]
%92 = [0,[1],1,3];
```

So L is associate to a modular form of weight 1, level 3267 and nebentypus $\left(\frac{-3}{\cdot}\right)$.

```
? mf=mfinit([3267,1,-3],1);
? M=mfeigenbasis(mf);
? C=mfcoefs(M[3],100);
? mfembed(M[3],C)[2][2..-1]==lfunan(L,100)
%96 = 1
```

Link to E

We reduce the coefficients of L modulo $1+\sqrt{-2}$ of norm 3.

```
? S = lfunan(L,1000); SE = lfunan(E,1000);
? Smod3 = round(real(S))+round(imag(S)/sqrt(2));
? [(Smod3[i]-SE[i])%3|i<-[1..#Smod3],gcd(i,33)==1]
%99 = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,...
```

The coefficients of L are congruent to the coefficients of the L-function associated to E modulo $1+\sqrt{-2}$.

Quotient of Hecke L-function

We will write L as L_1/L_2, where L_1 and L_2 are two Hecke L-functions.

```
? bnf6=bnfinit(a^6-3*a^5+6*a^4+4*a^3+6*a^2-3*a+1);
? bnr6=bnrinit(bnf6,1);
? L1=lfuncreate([bnr6,[1]]);
? L1[2..5]
%103 = [1, [0, 0, 0, 1, 1, 1], 1, 32019867]
? bnf4=bnfinit(a^4-a^3+3*a^2+a-1);
? pr4 = idealprimedec(bnf4,3)[1];
? bnr4=bnrinit(bnf4,[pr4,[0,1]]);
? L2=lfuncreate([bnr4,[1]]);
? L2[2..5]
%108 = [0, [0, 0, 1, 1], 1, 9801]
```

Quotient of Hecke L-function

```
? LL = lfundiv(L1,L2); LL[2..5]
%109 = [0,[0,1],1,3267]
? round(lfunan(L,1000)-lfunan(LL,1000),&e)
%110 = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,...
? e
%111 = -125
```

So L is equal to a quotient of two Hecke L-functions.