Computing $H^1(SL(2,k), \mathrm{End}^0(k^2))$ on a computer.

Kevin Buzzard

February 9, 2012

Last modified 08/06/2010.

Here's how I computed some group cohomology for $\mathrm{End}^0(k^2)$ in MAGMA. First I wrote a program by hand which outputted how a 2×2 matrix in $\mathrm{SL}(2,k)$ acted on the trace zero endomorphisms on k^2:

```
end0:=function(g)
  alp:=g[1,1];bet:=g[1,2];gam:=g[2,1];del:=g[2,2];
  return [del*alp+gam*bet,-gam*alp,del*bet,-2*bet*alp,alp^2,-bet^2,2*del*gam,-gam^2,del^2];
end function;
```

(theoretically you need to coerce the result into the appropriate general linear group or matrix algebra for it to make sense), and secondly I wrote the function which computed the cohomology (H^1) of such a module using magma's inbuilt cohomology calculator:

```
h1:=function(q)
  k:=GF(q);
  G:=SL(2,#k);
  M:=MatrixAlgebra(k,3);
  Y:=GModule(G,[M!end0(G.i):i in [1..#Generators(G)]]);
  D:=CohomologyModule(G,Y);
  return CohomologyGroup(D,1);
end function;
```

NOTE: `Generators(G)` returns a *set*, despite the fact that $G.i$ makes sense for $1 \leq i \leq$`#Generators(G)`, so it could have been a sequence.

I could then write a loop which computed lots of H^1s and experimentally verified Lemma 2.48 of DDT:

```
for q:=3 to 100000000 by 2 do
  if IsPrimePower(q) then print q,h1(q); end if;
end for;
```

As expected, the output looks like

```
3
Full Vector space of degree 0 over GF(3)
5
Full Vector space of degree 1 over GF(5)
7
Full Vector space of degree 0 over GF(7)
9
Full Vector space of degree 0 over GF(3^2)
11
Full Vector space of degree 0 over GF(11)
```

(and then lots more zeros).