

```

1 local b4={}; for k,_ in pairs(_ENV) do b4[k]=k end
2 local big,copy,cli,csv,egs,fmt,is,lt,map
3 local o,oo,push,splice,sort,string2thing
4 local the = {
5     also = 3, -- best=#t*min; rest=(also*best)
6     bins = 16,
7     cohen = .35,
8     file = "etc/data/aut93.csv",
9     how = "good",
10    keep = 256,
11    min = .5,
12    seed = 10019,
13 }
14 -----
15 big =1E32
16 fmt =string.format
17 rand=math.random
18 lt =function(x) return function(a,b) return a[x] < b[x] end end
19 map =function(t,f, u) u={};for k,v in pairs(t) do u[l+#u]=f(v) end; return u end
20 push=function(t,x) t[l+#t]=x; return x end
21 sort=function(t,f) table.sort(t,f); return t end
22
23 function cli(d)
24     for slot,x in pairs(d) do
25         x = tostring(x)
26         for n,flag in ipairs(arg) do
27             if flag=="-".slot or flag=="-".slot:sub(1,1) then
28                 x = x=="false" and"true" or x=="true" and"false" or arg[n+1] end end
29             d[slot] = string2thing(x) end
30         return d end
31
32 function copy(t, u)
33     if type(t) ~= "table" then return t end
34     u={};for k,v in pairs(t) do u[copy(k)]=copy(v) end
35     return setmetatable(u,getmetatable(t)) end
36
37 function csv(csvfile)
38     csvfile = io.input(csvfile)
39     return function(line, row)
40         line=io.read()
41         if not line then io.close(csvfile) else
42             row={}; for x in line:gmatch("[^,]+") do push(row,string2thing(x)) end
43             return row end end end
44
45 function oo(t) print(o(t)) end
46 function o(t, u)
47     if #t>0 then return "["..table.concat(map(t,tostring)," " ..")]" else
48         u={}; for k,v in pairs(t) do u[l+#u] = fmt("%.5s",k,v) end
49         return (t.is or "").."["..table.concat(sort(u," " ..")]" end end
50
51 function splice(t,i,j,k, u)
52     u={}; for n=(i or 1), (j or #t),(k or 1) do u[l+#u] = t[n] end return u end
53
54 function string2thing(x)
55     x = x:match("%s*(-)%s*$")
56     if x=="true" then return true elseif x=="false" then return false end
57     return math.tointeger(x) or tonumber(x) or x end
58
59 function is(name, t,new)
60     function new(kl,...) local x=setmetatable({},kl); kl.new(x,...); return x end
61     t = {__tostring=o, is=name or "", t.__index=t
62     return setmetatable(t, {__call=new}) end

```

```

63 -----
64 local SOME,NUM,SYM,EGS = is"SOME", is"NUM", is"SYM", is"EGS"
65
66 function SYM.new(i,at,name) i.n,i.txt,i.at,i.all = 0,txt or "",at or 0,{} end
67 function SYM.add(i,x,inc)
68     inc = inc or 1
69     if x=="?" then i.n = i.n+inc; i.all[x] = inc+(i.all[x] or 0) end end
70
71 function SYM.copy(i) return SYM(i.at,i.txt) end
72 function SYM.inject(i,...)
73     for _,more in pairs(...) do for x,n in pairs(more.all) do i:add(x,n) end end
74     return i end
75
76 function SYM.div(i, e)
77     e=0;for _,v in pairs(i.all) do if n>0 then e=e-v/i.n*math.log(v/i.n,2) end end
78     return e end
79
80 function SYM.range(i,x) return x end
81
82 function SYM.want(u,goal,B,R,how, b,r,z)
83     local how={
84         good=function(b,r) return ((b<r or b+r < .05) and 0) or b^2/(b+r) end,
85         bad=function(b,r) return ((r<b or b+r < .05) and 0) or r^2/(b+r) end,
86         novel=function(b,r) return 1/(b+r) end}
87     b, r, z = 0, 0, 1/big
88     goal = goal==nil and goal or true
89     for x,n in pairs(i.all) do
90         if x==goal then b=b+n else r=r+n end end
91         return how[the.how or "good"] (b/(B+z), z/(R+z)) end
92     -----
93 function SOME.new(i) i.all, i.ok, i.n = {}, false,0 end
94 function SOME.has(i) if not i.ok then sort(i.all) end;i.ok=true; return i.all end
95 function SOME.add(i,x)
96     i.n = 1 + i.n
97     if #i.all < the.keep then i.ok=false; push(i.all,x)
98     elseif rand() < the.keep/i.n then i.ok=false; i.all[rand(#i.all)]=x end end
99     -----
100 function NUM.new(i,at,txt)
101     i.n,i.mu,i.m2,i.sd,i.txt,i.at = 0,0,0,0,txt or "",at or 0
102     i.w,i.lo,i.hi,i.some = i.txt:find"-$" and -1 or 1,big,-big,SOME() end
103
104 function NUM.add(i,x, d)
105     if x=="?" then
106         i.some:add(x)
107         i.n = i.n+1
108         d = i.mu - x
109         i.mu = i.mu + d/i.n
110         i.m2 = i.m2 + d*(x - i.mu)
111         i.sd = (i.n<2 or i.m2<0) and 0 or (i.m2/(i.n-1))^0.5
112         i.lo = math.min(x, i.lo)
113         i.hi = math.max(x, i.hi) end end
114
115 function NUM.copy(i) return NUM(i.at,i.txt) end
116 function NUM.inject(i,...)
117     for _,more in pairs(...) do for x,n in pairs(more.some.all) do i:add(n) end end
118     return i end
119
120 function NUM.div() return i.sd end
121
122 function NUM.norm(i,x)
123     return (x=="?" and x) or (i.hi-i.lo<1E-9 and 0) or (x-i.lo)/(i.hi-i.lo) end
124
125 function NUM.range(i,x,n, b) b=(i.hi-i.lo)/n; return math.floor(x/b+0.5)*b end
126     -----
127 function EGS.new(i, names)
128     i.rows, i.names, i.all, i.x, i.y = {}, names, {}, {}, {}
129     for at,txt in pairs(names) do
130         local col = push(i.all, (txt:find"^[A-Z]" and NUM or SYM) (at,txt))
131         push(txt:find"^[+-]$" and i.y or i.x, col) end end
132
133 function EGS.add(i, row)
134     push(i.rows,row)
135     for _,col in pairs(i.all) do col:add(row[col.at]) end end
136
137 function EGS.bettters(i)
138     sort(i.rows, function(r1,r2)
139         local s1,s2,e,y,a,b = 0,0,math.exp(1),i.y
140         for _,col in pairs(y) do
141             a,b = col:norm(r1[col.at]), col:norm(r2[col.at])
142             s1 = s1 - e^(col.w * (a - b) / #y)
143             s2 = s2 - e^(col.w * (b - a) / #y) end
144         return s1/#y < s2/#y end)
145
146 function EGS.xx1(i,j,y,seen)
147     x=i.rows[j]
148     if x=="?" then
149         bin= col:range(x)
150         seen[bin] = seen[bin] or {x=NUM(), y=SYM()}
151         seen[bin].x:add(x)
152         seen[bin].y:add(y) end end
153
154 function EGS.xx(i)
155     i.rows = i.bettters()
156     n = (#i.rows)^the.min
157     step = (#i.rows - n1)/(the.also*n1)
158     for _,col in pairs(i.x) do
159         tmp={}
160         for j=1,n do i:xx1(j,true,tmp)
161             for j=n+1,#i.rows,step do i:xx1(j,false,tmp) end
162
163 function egs(f, i)
164     for row in csv(f or the.file) do
165         if i then i:add(row) else i=EGS(row) end end
166     return i end

```

```

168 the = cli(the)
169 math.random(the.seed or 10019)
170 local x=egs()
171 for i=1,5 do oo(x.rows[i]) end; print""
172 for i=#x.rows-5,#x.rows do oo(x.rows[i]) end
173 for k,v in pairs(_ENV) do if not b4[k] then print("?",k,type(v)) end end

```