```lua
1   local the, help = {}, [[
2
3   ween.lua [OPTIONS]
4   ween (vb), archaic. To think or imagine.
5
6   A small sample multi-objective optimizer / data miner.
7   (c)2021 Tim Menzies <timm@ieee.org> unlicense.org
8
9   OPTIONS:
10    -best   X  Best end  of the examples.         = .5
11    -debug  X  Run one test, show stack dumps on fail. = ing
12    -file   X  Read data from files.              = ../../data/auto93.csv
13    -h         Show help.                         = false
14    -hints  X  How many  to evaluate each iteration.  = 4
15    -p      X  Coefficient on distance calculation.   = 2
16    -seed   X  Random number seed.                = 10019
17    -todo   X  Demos to run at start-up. 'all'=run all. = ing]]
18
19  local b4={}; for k,_ in pairs(_ENV) do b4[k]=k end
20  local function rogues()
21    for k,v in pairs(_ENV) do if not b4[k] then print("?:",k,type(v)) end end end
22  -------------------------------------------------------------------------------
23  -------------------------------------------------------------------------------
24  local randi,rand,Seed -- remember to set seed before using this
25  function randi(lo,hi) return math.floor(0.5 + rand(lo,hi)) end
26  function rand(lo,hi)
27    lo, hi = lo or 0, hi or 1
28    Seed = (16807 * Seed) % 2147483647
29    return lo + (hi-lo) * Seed / 2147483647 end
30
31  local pop,csv,fmt,map,keys,sort,copy,norm,push,color,coerce,shuffle
32  fmt = string.format
33  function coerce(x)      return tonumber(x) or x end
34  function color(n,s)     return fmt("\27[1m\27[%sm%s\27[0m",n,s) end
35  function copy(t,  u)    u={}; for k,v in pairs(t) do u[k]=v end; return u end
36  function keys(t,u)      u={};for k,_ in pairs(t) do u[1+#u]=k end;return sort(u);end
37  function norm(lo,hi,x)  return math.abs(lo-hi)<1E-32 and 0 or (x-lo)/(hi-lo) end
38  function pop(t)         return table.remove(t) end
39  function push(t,x)      table.insert(t,x); return x end
40  function sort(t,f)      table.sort(t,f); return t end
41  function shuffle(t,   j)
42    for i=#t,2,-1 do j=randi(1,i); t[i],t[j]=t[j],t[i] end; return t end
43  function map(t,f,   u)
44    u,f = {},f or same; for k,v in pairs(t) do push(u, f(k,v)) end; return u end
45
46  function csv(file)
47    file = io.input(file)
48    return function(    t,x)
49      x = io.read()
50      if x then
51        t={};for y in x:gsub("%s*","")gmatch("[^,]+") do push(t,coerce(y)) end
52        if #t>0 then return t end
53      else io.close(file) end end end
54
55  local shout,out
56  function shout(x)  print(out(x)) end
57  function out(t,      u,key,val)
58    function key(_,k) return string.format(":%s %s", k, out(t[k])) end
59    function val(_,v) return out(v) end
60    if type(t) ~= "table" then return tostring(t) end
61    u = #t>0 and map(t, val) or map(keys(t), key)
62    return "{"..table.concat(u,"")..."}" end
63  -------------------------------------------------------------------------------
64  local slurp,sample,dist,ordered,hint
65  function slurp(  i) for eg in csv(the.file) do i=sample(i,eg) end; return i end
66
67  function sample(i,eg)
68    local numeric,independent,dependent,head,data,datum
69    i = i or {n=0,xs={},nys=0,ys={},lo={},hi={},w={},egs={},heads={},divs={}}
70    function head(n,x)
71      function numeric()     i.lo[n]= math.huge; i.hi[n]= -i.lo[n] end
72      function independent() i.xs[n]= x end
73      function dependent()
74        i.w[n]   = x:find"-" and -1 or 1
75        i.ys[n]  = x
76        i.nys    = i.nys+1 end
77      if not x:find":" then
78        if x:match"^[A-Z]" then numeric() end
79        if x:find"-" or x:find"+" then dependent() else independent() end end
80      return x end
81    function datum(n,x)
82      if x ~= "?" then
83        if i.lo[n] then
84          i.lo[n] = math.min(i.lo[n],x)
85          i.hi[n] = math.max(i.hi[n],x) end end
86      return x end
87    if #i.heads==0 then i.heads=map(eg,head) else push(i.egs,map(eg,datum)) end
88    i.n = i.n + 1
89    return i end
90
91  function ordered(i,egs)
92    local function left_is_best(left,right,     a,b,lefts,rights)
93      lefts,rights=0,0
94      for n,_ in pairs(i.ys) do
95        a  = norm(i.lo[n], i.hi[n], left[n])
96        b  = norm(i.lo[n], i.hi[n], right[n])
97        lefts  = lefts  - 2.71828^(i.w[n] * (a-b)/i.nys)
98        rights = rights - 2.71828^(i.w[n] * (b-a)/i.nys) end
99      return lefts/i.nys < rights/i.nys end
100   return sort(egs or i.egs, left_is_best),i end
101
102 function dist(i,eg1,eg2)
103   local function dist1(lo,hi,a,b)
104     if lo then
105       if     a=="?" then b=norm(lo,hi,b); a = b>.5 and 0 or 1
106       elseif b=="?" then a=norm(lo,hi,a); b = a>.5 and 0 or 1
107       else    a,b = norm(lo,hi,a), norm(lo,hi,b) end
108       return math.abs(a-b)
109     else
110       return a==b and 0 or 1 end end
111   local d,n,a,b,inc = 0,0
112   for col,_ in pairs(i.xs) do
113     a,b = eg1[col], eg2[col]
114     inc = a=="?" and b=="?" and 1 or dist1(i.lo[col],i.hi[col],a,b)
115     d   = d + inc^the.p
116     n   = n + 1 end
117   return (d/n)^(1/the.p) end
118
119 function hint(i,egs)
120   function hint1(egs, all, min, evals)
121     local scoreds,nearest,best = {}
122     function nearest(_,eg)
123       return sort(map(scoreds,function(rank,scored)
124                return {rank+dist(i,eg,scored)/10^6,eg} end),firsts)[1] end
125     if    #egs <= 2*min
126     then scoreds = egs
127          return map(sort(map(all,nearest),firsts),second)
128     else for j=1,the.hints do push(scoreds, pop(egs)) end
129          best, scoreds = {}, ordered(i,scoreds)
130          print(2)
131          egs = map(egs,nearest)
132          print(1)
133          egs = sort(map(egs,nearest),firsts)
134          for j=1,(#egs)//2 do push(best, egs[j][2]) end
135          hint1(best, all, min, evals+the.hints) end
136   end -----------------
137   egs = egs or i.egs
138   return hint1( copy(shuffle(egs)), egs, (#egs)^the.best, 0) end
139
```

```lua
140 -------------------------------------------------------------------------------
141 local go={}
142 function go.ing() return true end
143 function go.the() shout(the) end
144 function go.csv() for eg in csv(the.file) do shout(eg) end end
145 function go.more(   u,t)
146   t= {10,20,30,40}
147   u= copy(shuffle(t))
148   t[1]=100
149   assert(u[1] ~= t[1])
150   assert(u[1] ~= 100) end
151
152 function go.sample(  s)
153   s=slurp()
154   assert(398 == #s.egs)
155   assert(3 == s.lo[1]) end
156
157 function go.ordered(  _,i,egs)
158   egs,i = ordered(slurp())
159   shout(i.heads)
160   for j=1,5 do shout(egs[j]) end
161   print("#")
162   for j=#egs-5,#egs do shout(egs[j]) end end
163
164 function go.dist(  i,dist1,t)
165   function dist1(_,eg) return {dist(i,i.egs[1],eg), eg} end
166   i = slurp()
167   t=map(i.egs,dist1)
168   for j=1,5 do print(j,fmt("%5.3f",t[j][1]),out(t[j][2])) end
169   print("#")
170   for j=(#t)-5,#t do print(j,fmt("%5.3f",t[j][1]),out(t[j][2])) end end
171
172 function go.hint(  i)
173   hint(slurp()) end
174
175 -- Run demos, each time resetting random seed and the global config options.
176 -- Return to the operating system then number of failing demos.
177 local function main()
178   local no,defaults,reset = 0,copy(the)
179   function reset(x) Seed=the.seed or 10019; the= copy(defaults) end
180   reset()
181   go[ the.debug ]()
182   for _,it in pairs(the.todo=="all" and keys(go) or {the.todo}) do
183     if type(go[it]) ~= "function" then return print("NOFUN:",it) end
184     reset()
185     local ok,msg = pcall( go[it] )
186     if ok then print("PASS ",it) else no=no+1; print("FAIL "..it,msg) end end
187   rogues()
188   os.exit(no) end
189
190 -------------------------------------------------------------------------------
191 -- Make 'the' options array from help string and any updates from command line.
192 (help or ""):gsub("^.*OPTIONS:","")gsub("\n%s*-([^%s]+)[^\n]*%s([^%s]+)",
193     function(flag,x)
194       for n,word in ipairs(arg) do if word=="-"..flag) then
195         x = x=="false" and "true" or tonumber(arg[n+1]) or arg[n+1] end end
196       if x=="false" then x=false elseif x=="true" then x=true end
197       the[flag]=x end)
198
199 if the.h then print(help) else main() end
```