

```

1 local the,help = {},{}
2 tweak: tries three weak learners for multi-objective optimization
3 (c) 2022, Tim Menzies, timm@ieee.org, opensource.org/licenses/Fair
4
5 USAGE:
6 alias twk="lua tweak.lua "
7 twk [OPTIONS]
8
9 OPTIONS:
10 --bins      -b  max bins              = 16
11
12 OPTIONS (other):
13 --file      -f  where to find data      = ../etc/data/auto2.csv
14 --dump      -d  dump stack+exit on error = false
15 --help      -h  show help               = false
16 --go        -g  start up action         = nothing[]
17
18 local rows, aotm = {}
19 R=math.random
20 Big=math.huge
21
22 function map(t,f, u)  u={}; for _,v in pairs(t) do u[l+#u]=f(v) end;return u end
23 function push(t,x)    t[l+#t]=x; return x end
24
25 function thing(x)
26   x = x:match("^%s*(-)%s*$"
27   if x=="true" then return true elseif x=="false" then return false end
28   return math.tointeger(x) or tonumber(x) or x end
29
30 help:gsub("u ([-~|]|([^%s+)][%s]+)(-[^%s+)]^%u)%s%([%s+])",function(f1,k,f2,x)
31   for n,flag in ipairs(arg) do if flag==f1 or flag==f2 then
32     x = x=="false" and"true" or x=="true" and"false" or arg[n+1] end end
33   the[k] = thing(x) end)
34
35 function csv(f)
36   f = io.input(f)
37   return function(t, u)
38     t=io.read()
39     if not t then io.close(f) else
40       u={}; for x in t:gmatch("[^,]+") do u[l+#u]=thing(x) end
41       return u end end end
42
43 function o(t, u)
44   u={}; for k,v in pairs(t) do u[l+#u] = string.format("%.%.%.s",k,v) end
45   return (t.is or "").."{"..table.concat(sort(u)," ").."}" end
46
47 function obj(name, t,new)
48   function new(kl,...)
49     local x=setmetatable(({id=id()},{kl}; kl.new(x,...)); return x end
50   t = {__toString=o, is=name or ""}; t.__index=t
51   _ = t
52   return setmetatable(t, {__call=new}) end

```

```

53 -----
54 SYM=obj"SYM"
55 function __.new(i,at,txt) i.at, i.txt, i.has,i.bins = at,txt,{},{} end
56 function __.add(i,x)      if x=="?" then i.has[x] = 1+(i.has[x] or 0) end end
57 function __.addxy(i,x,y)  if x=="?" then i.bins[x]=y+(i.bins[x] or 0) end end
58
59 __.mid=function(i, m,x)
60   m=0; for y,n in pairs(i.has) do if n>m then m,x=y,n end end; return x end
61
62 __.div=function(i, n,e)
63   n=0; for _,m in pairs(i.has) do n = n + m end
64   e=0; for m in pairs(i.has) do e = e - m/n*math.log(m/n,2) end
65   return e end
66 -----
67 function BIN.new(i,t) i.pos,i.txt,i.lo,i.hi,i.y = t.pos,t.txt,t.lo,t.hi,t.y end
68 function BIN.of(i,x) return i.y.has[x] or 0 end
69
70 function BIN.select(i,t, x)
71   t = t.cells and t.cells or t
72   x = t[i.pos]
73   return x=="?" or i.lo == i.hi and i.lo == x or i.lo <= x and x < i.hi end
74
75 function BIN.__toString(i)
76   local x,lo,hi,big = i.txt, i.lo, i.hi, Big
77   if lo == hi then return fmt ("%s==%s",x, lo)
78   elseif hi == big then return fmt ("%s>=%s",x, lo)
79   elseif lo == -big then return fmt ("%s< %s", x, hi)
80   else return fmt ("%s<= %s< %s",lo,x,hi) end end
81 -----
82 local NUM=obj"NUM"
83 function NUM.new(i,n,s) i.at,i.txt, i.lo,i.hi,i.bins=n,s,Big,-Big,{} end
84 function NUM.norm(i,x) return x=="?" and x or (x-i.lo)/(i.hi - i.lo) end
85
86 function NUM.addx(i,x)
87   if x=="?" then return x end
88   if x > i.hi then i.hi=x elseif x<i.lo then i.lo=x end end
89
90 function NUM.addxy(i,x,y)
91   if x=="?" then return x end
92   x = math.max(1, math.min(the.bins, the.bins*i:norm(x) // 1))
93   i.bins[x] = i.bins[x] or Sym()
94   i.bins[x]:add(y) end
95
96 function ROW.new(i,egs,t) i.cells,i.data = t,egs end
97
98 local COLS=obj"COLS"
99 function COLS:new(names, col)
100   self.all,self.x,self.y,self.names={},(),{},names
101   for i,t in pairs(names) do
102     col = push(self.all, txt.find"^[A-Z]+" and Num or Sym)(i,t)
103     if not txt.find"$" then
104       push(txt.find"[^!]+$" and self.y or self.x,col) end end end
105
106 local EGS=obj"EGS"
107 function EGS:new() self.rows,self.cols = {},nil end
108 function EGS:add(t)
109   if self.cols
110     then t = push(self.rows, t.cells and t or ROW(self,t)).cells
111     for _,col in pairs(self.cols.all) do col:add(t[col.pos]) end
112     else self.cols = COLS(t) end end
113
114 function EGS:file(f) for row in csv(f) do self.add(row) end end
115 -----
116

```