

```

subroutine mars (n,p,x,y,w,nk,mi,lx,fm,im,sp,dp,mm)      1
integer p,lx(*),im(*),mm(*)                             2
real x(*),y(*),w(*),fm(*),sp(*)                       3
double precision dp(*)                                  4
im(3)=n                                                  5
im(4)=p                                                  6
im(5)=nk                                                 7
im(6)=mi                                                 8
im(7)=16                                                 9
im(8)=im(7)+5*nk                                       10
im(9)=im(8)+2*nk*mi                                    11
im(10)=im(9)+3*(nk+2)                                  12
im(2)=im(10)+nk*mi-1                                   13
im(11)=1                                                14
im(12)=2                                                15
im(13)=im(12)+5*nk                                    16
im(14)=im(13)+1                                        17
im(15)=im(14)+nk*(5*mi+1)                              18
call mars1(n,p,x,y,w,nk,mi,lx,fm(im(11)),fm(im(12)),fm(im(15)),im( 19
1im(7)), im(im(8)),im(im(9)),im(im(10)),fm(im(13)),fm(im(14)),sp,d 20
lp,mm)                                                  21
im(1)=im(15)+lcm(p,nk,fm(im(12)),fm(im(15)))-1       22
return                                                  23
end                                                      24
subroutine plot (m,x,fm,im,ngc,ngs,icx,nc,crv,ns,srf,sp,mm) 25
integer im(*),mm(*)                                     26
real x(*),fm(*),crv(*),srf(*),sp(*)                   27
if(m .ne. 1) go to 1                                    28
call plot1(im(3),im(4),x,im(5),im(im(7)),im(im(8)),im(im(9)),im(im 29
1(10)), fm(im(12)),fm(im(15)),ngc,ngs,icx,nc,crv,ns,srf,sp,mm) 30
return                                                  31
1 call plotc(im(3),im(4),x,im(5),im(im(7)),im(im(8)),im(im(9)),im(im 32
1(10)), fm(im(14)),fm(im(15)),ngc,ngs,icx,nc,crv,ns,srf,sp,mm) 33
return                                                  34
end                                                      35
subroutine catprt (m,fm,im,sp,mm)                      36
integer im(*),mm(*)                                     37
real fm(*),sp(*)                                       38
call ctp1(m,im(5),im(im(7)),im(im(8)),fm(im(12)),fm(im(15)),fm(i 39
1m(14)),sp,mm)                                         40
return                                                  41
end                                                      42
subroutine slice (flg,xs,x,fm,im,fmn,imn,sp,mm)         43
integer im(*),imn(*),mm(*)                             44
real xs(*),x(*),fm(*),fmn(*),sp(*)                   45
do 1 i=1,15                                             46
imn(i)=im(i)                                           47
1 continue                                             48
i=im(15)                                               49
go to 3                                                50
2 i=i+1                                                51
3 if((i).gt.(im(1))) go to 4                          52
fmn(i)=fm(i)                                           53
go to 2                                                54
4 call slice1(flg,xs,im(3),im(4),x,im(5),fm(im(11)),fm(im(12)),fm(im 55
1(15)), im(im(7)),im(im(8)),im(im(9)),im(im(10)),fm(im(13)),fm(im( 56
114)), fmn(im(11)),fmn(im(12)),imn(im(7)),imn(im(8)),imn(im(9)),im 57
1n(im(10)), fmn(im(13)),fmn(im(14)),sp,mm)            58
return                                                  59
end                                                      60
subroutine fmod (m,n,x,fm,im,f,sp)                    61
integer im(*)                                           62
real x(*),fm(*),f(*),sp(*)                            63

```

if(m .ne. 1) go to 1	64
call fmrs(n,x,im(5),fm(im(11)),fm(im(12)),fm(im(15)),f)	65
return	66
1 call cmrs(n,x,fm(im(15)),im(im(7)),im(im(8)),im(im(9)),im(im(10)),	67
1 fm(im(13)),fm(im(14)),f,sp)	68
return	69
end	70
subroutine print(it)	71
call printm(it)	72
call printg(it)	73
call printc(it)	74
call prtslc(it)	75
return	76
end	77
subroutine setint(i,j,k)	78
parameter(mlist=1000)	79
integer m(2,mlist)	80
save m	81
data il /0/	82
if((i .ne. 0) .and. (j .ne. 0)) go to 1	83
il=0	84
return	85
1 if(i.eq.j) return	86
m1=min0(i,j)	87
m2=max0(i,j)	88
if(k .ne. 0) go to 6	89
l=1	90
go to 3	91
2 l=l+1	92
3 if((l).gt.(il)) go to 4	93
if(m1.eq.m(1,l).and.m2.eq.m(2,l)) return	94
go to 2	95
4 il=il+1	96
if(il .le. mlist) go to 5	97
c write(6, '(' increase parameter mlist in subroutine setint to gr	98
c leater than'', i5,/, '' and recompile.'')'') il	99
stop	100
5 m(1,il)=m1	101
m(2,il)=m2	102
return	103
6 ig=0	104
l=1	105
go to 8	106
7 l=l+1	107
8 if((l).gt.(il)) go to 10	108
if(m1 .ne. m(1,l) .or. m2 .ne. m(2,l)) go to 7	109
ig=1	110
10 if(ig.eq.0) return	111
il=il-1	112
ll=l	113
go to 12	114
11 ll=ll+1	115
12 if((ll).gt.(il)) go to 13	116
m(1,ll)=m(1,ll+1)	117
m(2,ll)=m(2,ll+1)	118
go to 11	119
13 return	120
entry intlst(it)	121
if(it.le.0) return	122
if(il.eq.0) return	123
write(it,('/', '' interactions prohibited between:''))	124
c do 14 l=1,il	125
c write(it, '(' var('',i3, '') and var('',i3, '')'')'') m(1,l),m(2	126
c 1,l)	127
c 14 continue	128

```

        return 129
        entry intalw(i,j,k) 130
        k=1 131
        m1=min0(i,j) 132
        m2=max0(i,j) 133
        l=1 134
        go to 16 135
15 l=l+1 136
16 if((l).gt.(il)) go to 18 137
    if(m1 .ne. m(1,l) .or. m2 .ne. m(2,l)) go to 15 138
    k=0 139
18 return 140
    end 141
    subroutine mars1 (n,p,x,y,w,nk,mi,lx,az,tb,cm,kp,kv,lp,lv,bz,tc,sp 142
1,dp,mm) 143
    integer p,kp(5,*),kv(2,*),lp(3,*),lv(*),mm(n,*),lx(p) 144
    real x(n,p),y(n),w(n),tb(5,nk),cm(*),tc(*),sp(*) 145
    double precision dp(*) 146
    data ms,df,il,fv,it,ic,ix /0,3.0,0,0.0,6,0,0/ 147
c    if(it.gt.0) write(it,11) 148
c    if(it.gt.0) write(it,10) n,p,nk,ms,mi,df,il,fv,ic 149
c    if(it.gt.0) write(it,12) 150
c    if(it.gt.0) write(it,('' var: '' ,5('' '' ,20i3,/))'') (i,i=1,p) 151
c    if(it.gt.0) write(it,('' flag:'' ,5('' '' ,20i3,/))'') (lx(i),i=1,p) 152
c    print *, ' '
c    do 321 i = 1, n
c        print *, 'M1 ',x(i,1),' ',x(i,2),' ',x(i,3),' ',x(i,4),
c    1 ' ',y(i)
c321 continue
    call intlst(it) 153
    call nstlst(it) 154
    il=max0(n*(nk+1),2*n)+1 155
    im=il+n+max0(3*n+5*nk,2*p,4*n,2*n+5*nk+p) 156
    is=im+p 157
    i2=max0(n*nk,(nk+1)*(nk+1))+1 158
    call rspnpr(it,il,n,y,w,mm) 159
    do 2 j=1,p 160
    do 1 i=1,n 161
        mm(i,j)=i 162
1 continue 163
    call psort(x(1,j),mm(1,j),1,n) 164
2 continue 165
    call ordpr(it,n,p,x,lx,mm) 166
    call atoscl (n,p,w,x,lx,mm,sp(im),sp(is),cm,x) 167
    call catpr(it,n,p,x,cm,mm(1,p+1)) 168
    call oknest(it,p,lx,cm) 169
    if(ix.ne.0) call cvmars (ix,n,p,x,y,w,nk,ms,df,fv,mi,lx,it,sp(im), 170
1sp(is),tb,cm,sp,dp,dp(i2),mm, sp(is+p),sp(is+p+2*n)) 171
    call marsgo (n,p,x,y,w,nk,ms,df,fv,mi,lx,it,sp(im),sp(is),az,tb,c 172
1m,sp,dp,dp(i2),mm) 173
    if(il .le. 0) go to 6 174
    call logitl(n,x,y,w,nk,il,az,tb,cm,sp,dp) 175
    if(it .le. 0) go to 6 176
    sw=0.0 177
    wn=sw 178
    do 3 i=1,n 179
        sw=sw+w(i) 180
        wn=wn+w(i)**2 181
3 continue 182
    wn=sw**2/wn 183
    ef=1.0 184
    do 4 k=1,nk 185
        if(tb(1,k).ne.0.0) ef=ef+tb(5,k) 186
4 continue 187
    ef=1.0/(1.0-ef/wn)**2 188

```

s=0.0	189
t=s	190
call fmrs(n,x,nk,az,tb,cm,sp)	191
do 5 i=1,n	192
yh=1.0/(1.0+exp(-sp(i)))	193
gcv=ef*(y(i)-yh)**2	194
s=s+w(i)*gcv	195
t=t+w(i)*yh*(1.0-yh)	196
5 continue	197
s=s/sw	198
t=t/sw	199
c write(it,13) s,t	200
6 if(it .le. 0) go to 7	201
if(il.eq.0) call anova (n,x,y,w,nk,it,tb,cm,lp,lv,sp,dp)	202
if(il.gt.0) call anoval(n,x,y,w,nk,il,it,az,tb,cm,lp,lv,sp,dp)	203
7 call ccoll (nk,tb,cm,kp,kv,lp,lv,mm)	204
call cubic (n,p,x,y,w,nk,it,tb,cm,kp,kv,lp,lv,bz,tc,sp,sp(il),sp(i	205
11+2*p),mm,dp)	206
if(il .le. 0) go to 9	207
call logitc(n,x,y,w,nk,il,cm,kp,kv,lp,lv,bz,tc,sp,sp(il+4*n),dp)	208
if(it .le. 0) go to 9	209
call cmrs(n,x,cm,kp,kv,lp,lv,bz,tc,sp,sp(n+1))	210
s=0.0	211
t=s	212
do 8 i=1,n	213
yh=1.0/(1.0+exp(-sp(i)))	214
gcv=ef*(y(i)-yh)**2	215
s=s+w(i)*gcv	216
t=t+w(i)*yh*(1.0-yh)	217
8 continue	218
s=s/sw	219
t=t/sw	220
c write(it,14) s,t	221
9 if(it.gt.0) call varimp (n,p,x,y,w,nk,il,it,az,tb,cm,sp,sp(p+1),dp	222
1)	223
call orgpl(sp(im),sp(is),nk,tb,cm)	224
call orgpc(sp(im),sp(is),lp,lv,tc)	225
call sclato(n,p,x,sp(im),sp(is),cm,x)	226
return	227
entry setms(mal)	228
ms=mal	229
return	230
entry setdf(val)	231
df=val	232
return	233
entry printm(mal)	234
it=mal	235
return	236
entry logit(mal)	237
il=mal	238
return	239
entry setfv(val)	240
fv=val	241
return	242
entry setic(mal)	243
ic=mal	244
z00001=stelg(ic)	245
return	246
entry xvalid(mal)	247
ix=mal	248
call xvmrgo(ix)	249
return	250
10 format(/' input parameters (see doc.):',/, ' n p nk	251
lms mi df il fv ic',/, ' ',i5,i5,i6,i6,i6,f8.3,i5	252
1,f7.3,i6)	253

```

11 format(//, ' MARS modeling, version 3.6 (3/25/93)',/) 254
12 format(/' predictor variable flags:') 255
13 format(/' piecewise-linear logistic gcv =',g12.4,' ave var =',g1 256
12.4) 257
14 format(/' piecewise-cubic logistic gcv =',g12.4,' ave var =',g12 258
1.4) 259
end 260
subroutine plotc (n,p,x,nk,kp,kv,lp,lv,tc,cm,ngc,ngs,icx,nc,crv,ns 261
1,srf,sp,mm) 262
integer p,kp(5,*),kv(2,*),lp(3,*),lv(*),mm(*) 263
real x(n,p),tb(5,nk),tc(*),cm(*),crv(ngc,2,*),srf(ngs,ngs,*),sp(*) 264
1,zl(2),zu(2) 265
data big,it /1.e30,6/ 266
c if(it.gt.0) write(it,(/' mars graphics (piecewise-cubic):',/)) 267
jnt=2 268
go to 1 269
entry plotl (n,p,x,nk,kp,kv,lp,lv,tb,cm,ngc,ngs,icx,nc,crv,ns,srf, 270
1sp,mm) 271
c if(it.gt.0) write(it,(/' mars graphics (piecewise-linear):',/)) 272
c 1) 273
jnt=1 274
1 ngsq=ngs**2 275
iz=2*ngsq 276
d=1.0/(ngs-1) 277
dc=1.0/(ngc-1) 278
ll=1 279
nc=0 280
ns=nc 281
2 if(kp(1,ll).lt.0) go to 36 282
if(kp(3,ll) .gt. 0) go to 3 283
ll=ll+1 284
go to 2 285
3 nf=kp(3,ll) 286
k4=kp(4,ll)-1 287
k1=kp(1,ll) 288
k2=kp(2,ll) 289
if(it .le. 0) go to 7 290
if(k1 .ne. 0) go to 4 291
c write(it,(' pure ordinal contribution:')) 292
go to 7 293
4 continue
c write(it,(' categorical - ordinal interaction:')) 294
do 6 i=1,k1 295
jj=kv(1,k2+i-1) 296
j=iabs(jj) 297
k=kv(2,k2+i-1) 298
ncx=int(cm(2*j+1)+.1)-int(cm(2*j)+.1)+1 299
do 5 l=1,ncx 300
mm(l)=cm(k+l)+.1 301
if(jj.lt.0) mm(l)=mod(mm(l)+1,2) 302
5 continue 303
c write(it,(' x('',i3,'') =',70i1/80i1')) j,(mm(l),l=1,ncx) 304
6 continue 305
7 do 35 k=1,nf 306
l=lp(1,k+k4) 307
if(l.gt.2) go to 35 308
ko=lp(2,k+k4) 309
if(l .ne. 1) go to 17 310
j=0 311
jv=lv(ko) 312
do 9 m=k,nf 313
l1=lp(1,m+k4) 314
if(l1.eq.1) go to 9 315
l2=lp(2,m+k4)-1 316
do 8 i=1,l1 317

```

if(jv.eq.lv(l2+i)) j=1	318
8 continue	319
if(j.eq.1) go to 10	320
9 continue	321
10 if(j.eq.1) go to 35	322
nc=nc+1	323
zl(1)=big	324
zu(1)=-big	325
do 11 i=1,n	326
r=x(i,jv)	327
zl(1)=amin1(zl(1),r)	328
zu(1)=amax1(zu(1),r)	329
11 continue	330
dl=(zu(1)-zl(1))*dc	331
do 12 i=1,ngc	332
crv(i,1,nc)=zl(1)+dl*(i-1)	333
12 continue	334
if(jnt .ne. 1) go to 13	335
call fun(l,jv,ngc,crv(1,1,nc),nk,tb,cm,k1,kv(1,k2),crv(1,2,nc),mm)	336
go to 14	337
13 call cfun (l,jv,ngc,crv(1,1,nc),nf,lp(1,k4+1),lv,tc(kp(5,ll)), cr	338
lv(1,2,nc),sp,mm)	339
14 dl=big	340
do 15 i=1,ngc	341
dl=amin1(dl,crv(i,2,nc))	342
15 continue	343
fx=0.0	344
do 16 i=1,ngc	345
crv(i,2,nc)=crv(i,2,nc)-dl	346
fx=amax1(fx,crv(i,2,nc))	347
16 continue	348
c if(it.gt.0) write(it,39) nc,jv,fx	349
go to 35	350
17 j=0	351
mm(1)=lv(ko)	352
mm(2)=lv(ko+1)	353
do 19 m=k,nf	354
l1=lp(1,m+k4)	355
if(l1.le.2) go to 19	356
l2=lp(2,m+k4)-1	357
do 18 i=1,l1	358
if(mm(1).eq.lv(l2+i).or.mm(2).eq.lv(l2+i)) j=1	359
18 continue	360
if(j.eq.1) go to 20	361
19 continue	362
20 if(j.eq.1) go to 35	363
ns=ns+1	364
zl(1)=big	365
zl(2)=zl(1)	366
zu(1)=-big	367
zu(2)=zu(1)	368
do 22 j=1,2	369
do 21 i=1,n	370
r=x(i,mm(j))	371
zl(j)=amin1(zl(j),r)	372
zu(j)=amax1(zu(j),r)	373
21 continue	374
22 continue	375
do 23 j=1,2	376
dl=(zu(j)-zl(j))/(ngs-3)	377
zu(j)=zu(j)+dl	378
zl(j)=zl(j)-dl	379
23 continue	380
ne=0	381
dl=d*(zu(1)-zl(1))	382

```

      d2=d*(zu(2)-zl(2))
      do 25 j=1,ngs
      do 24 i=1,ngs
      ne=ne+1
      sp(iz+ne)=zl(1)+d1*(i-1)
      sp(iz+ngsq+ne)=zl(2)+d2*(j-1)
24 continue
25 continue
      dl=big
      if(jnt .ne. 1) go to 26
      call pair(mm,ngsq,sp(iz+1),nk,tb,cm,k1,kv(1,k2), srf(1,1,ns),sp,m
1m(3))
      go to 27
26 call cpair(mm,ngsq,sp(iz+1),nf,lp(1,k4+1),lv, tc(kp(5,ll)),srf(1,
1l,ns),sp)
27 if(icx .le. 0) go to 29
      call cvxhul(n,x(1,mm(1)),x(1,mm(2)),big,nh,sp)
      if(it .le. 0 .or. 3*nh .lt. iz) go to 28
      nxs=sqrt(float(3*nh)*0.5)+1.1
c      write(it,38) nxs
28 call hulset(ngsq,sp(iz+1),big,nh,sp,srf(1,1,ns))
29 do 31 j=1,ngs
      do 30 i=1,ngs
      if(i.eq.1.or.j.eq.1.or.i.eq.ngs.or.j.eq.ngs.or.srf(i,j,ns).ge.big)
1 go to 30
      dl=amin1(dl,srf(i,j,ns))
30 continue
31 continue
      fx=0.0
      do 34 j=1,ngs
      do 33 i=1,ngs
      if((i .ne. 1) .and. ((j .ne. 1) .and. ((i .ne. ngs) .and. ((j .ne.
1 ngs) .and. (srf(i,j,ns) .lt. big)))) go to 32
      srf(i,j,ns)=0.0
      go to 33
32 srf(i,j,ns)=srf(i,j,ns)-dl
      fx=amax1(fx,srf(i,j,ns))
33 continue
34 continue
c      if(it.gt.0) write(it,40) ns,mm(1),mm(2),fx
35 continue
      ll=ll+1
      go to 2
36 continue
c      if(it.gt.0) write(it,37) nc,ns
      return
      entry printg(nal)
      it=nal
      return
37 format(/,' ',i3,' curves and',i3,' surfaces.'/)
38 format(' plot: convex hull too large. increase ngs to',i6)
39 format('   crv',i3,':  x(',i2,').  max =',g12.4)
40 format('   srf',i3,':  x(',i2,'), x(',i2,').  max =',g12.4)
      end
      subroutine ctptrl (m,nk,kp,kv,tb,cm,tc,sc,js)
      integer kp(5,*),kv(2,*),js(*)
      real cm(*),tc(*),sc(*)
      data big,it /9.9e30,6/
      if(it.le.0) return
      nc=ncat(kp)
      if(nc.eq.0) return
      write(it,(/,' ' there are',i3,' purely categorical basis functio
1ns.'')) nc
c      write(it,(' ' purely additive and bivariate contributions follow'
c      1'))

```