```
C       ============================================================
c       Sensitivity analysis for
C       EXAMPLE 3: a function of Gaussian variables (up to 100 dimensional)
C       METHOD: PDD-SPA option 2
c           MAIN CODE
C
C
C
C       N = 10 or 100
C       Y = frac{1}{1000+sum_{i=1}^{N}X_i} - frac{1}{1000+3*sqrt{N}}
c
C     XI ~ Gaussian(0, 1),    CONSISTENT POLYNOMIAL BY HERMITE
C     GAUSS-HERMITE QUADRATURE RULE
c
C
C     -------  command line parameters to run the code   -------------
C     PROGRAM.EXE NVAR NBAS_R
C
C*******************************************************************************
c     Copyright (c) 2010-2014 [Xuchun Ren], All Rights Reserved.




      IMPLICIT REAL*8 (A-H,O-Z)




        DOUBLE PRECISION X(200),G(200),MOMSEN_OUT(10,2),XMU_IN(200),
     1   OBJ(200), XSIG_IN(200),X0(200),XL01(200),XU01(200)

        REAL*8 XMU(200),SIG(200),NMU(200),NSIG(200)
        COMMON /SR02/ XMU,SIG, NMU, NSIG, NYQ, N
        INTEGER NBAS(200), NGAUSS(200),NGAUSS_SM(200),NBAS_SM(200),
     1   IFLAG_Y,
     1       I_EVAL

        COMMON /XR01/ N_F, N_GF, N_GC, N_EF,N_EC, N_EF1, N_EC1,
     1     N_EF2, N_EC2
         COMMON /XR04/ NBAS_R, NGAUSS_R, NVAR

         COMMON /XR09/ I_EVAL
         CHARACTER(80) FN
       COMMON /XR10/ FN
         COMMON /XR11/ XL01, XU01
         COMMON /XR20/ C1_SCAL, I_RECIPROCAL
      INTEGER IITER, IITER1, N_PREFEA, N_FEA, IITER_TOTAL, I_SQP
      COMMON /XR05/ IITER, IITER1, N_PREFEA, N_FEA, IITER_TOTAL, I_SQP

      REAL*8 PF, PF_MCS, PF_SEN(10,10), PF_SEN_MCS(100,10), TS_OUT,
     1    Y0, YU(10,10), WU(10,10)

      INTEGER NBAS_SPA, NGAUSS_SPA ,status

      COMMON /SPA01/ Y0, YU,WU, NBAS_SPA, NGAUSS_SPA
       CHARACTER(80) MSG,MSG1
       CHARACTER(320) buf
       INTEGER IW1
       COMMON /XR07/ IW1
        n1 = 1
        n2 = 2
        n3 = 3
        n4 = 4
        n5 = 5
        n6 = 6
        n7 = 7
        n8 = 8

        IW1 = 0

        MSG = 'PROGRAM NVAR NBAS_R'



        CALL GETARG(n1, buf, status)
        IF (status .EQ. -1) THEN
        PRINT*, 'WRONG NVAR!'
        PRINT*, MSG
C        PRINT*, MSG1
        STOP
        endif

        Read( buf, '(i10)' )  NVAR
        PRINT*, 'NVAR = '
        WRITE (*,*) NVAR

        CALL GETARG(n2, buf, status)
        IF (status .EQ. -1) THEN
        PRINT*, 'WRONG NBAS_R!'
        PRINT*, MSG
C        PRINT*, MSG1
        STOP
        endif

        Read( buf, '(i10)' )  NBAS_R
        PRINT*, 'NBAS_R = '
        WRITE (*,*) NBAS_R




      NBAS_SPA = 8
C      NGAUSS_SPA = 3

        c0 = 1.0

       IFLAG_AIJK  = 0
       N = 10
       IMCS = 0
c       NVAR = 1


        x0 = 0.

        NSAMP = 10**6
        IID = 1

        DO I = 1, N
        XMU_IN(I) = X0(I)
        XSIG_IN(I) = 1.0
        ENDDO




        NBAS = NBAS_R
        NBAS_SM = 2
        NGAUSS = NBAS_R+1
        NGAUSS_SM = 3
        NGAUSS_GQ = 3


c     -- modify main
      dcov =  0.02
      ncov  = (0.7-0.1)/dcov
      ncum  = 8
      ng_cum = 4
      ng_cum_sen = 4
      ntheta = 2


c     test PDD function


      do i = 0, 0

      CALL PDD_SEN_PF_SPA (X0, XMU_IN, XSIG_IN, NSAMP,NBAS,
     1   NBAS_SM, NGAUSS,
     1 NGAUSS_SM, IID,  NGAUSS_GQ, 1, IMCS, TS_OUT, PF,
     2     PF_SEN, PF_MCS, PF_SEN_MCS,
     3   IFLAG_AIJK, NVAR, ncum, ng_cum, ng_cum_sen, ntheta)


        enddo


C       PRINT*, "PF = ", PF
C       PRINT*, "PF_SEN = ", PF_SEN(1:3,1:2)
C
C       PRINT*, "PF_MCS = ", PF_MCS
C       PRINT*, "PF_SEN_MCS = ", PF_SEN_MCS(1:3,1:2)
C
C       PRINT*, "TS_OUT = ", TS_OUT
C
      END
```