
Subject: Legendre Polynomials

Posted by [ganga](#) on Thu, 29 Dec 1994 23:27:08 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello,

Does anyone know where I might find IDL code for calculating Legendre Polynomials?

Thanks,

Ken

--

Ken Ganga | (510)642-3618
433 LeConte Hall | (510)643-5204 (Fax)
Berkeley, CA 94720 | ganga@physics.berkeley.edu

--

Ken Ganga | (510)642-3618
433 LeConte Hall | (510)643-5204 (Fax)
Berkeley, CA 94720 | ganga@physics.berkeley.edu

Subject: Re: Legendre Polynomials

Posted by [thompson](#) on Thu, 29 Dec 1994 23:58:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

ganga@physics23 (Ken Ganga) writes:

> Does anyone know where I might find IDL code for calculating Legendre
> Polynomials?

I wrote the following routine some years ago. I can't guarantee that it works,
or that it's even what you want. It's been a long time since I used it.

Sorry that it's undocumented. I've learned better since then. The technique
may be based on Abromowitz and Stegun--I tended to get that sort of thing from
there in those days.

Bill Thompson

```
=====
```

```
FUNCTION LEGENDRE,X,L,M
```

```
;
```

```
IF N_PARAMS(0) LT 2 THEN BEGIN
```

```
  PRINT,'*** LEGENDRE must be called with 2-3 parameters:'
```

```
  PRINT,'          X, L [, M ]'
```

```

RETURN,X
END ELSE IF N_PARAMS(0) EQ 2 THEN M = 0
;
IF M LT 0 THEN BEGIN
PRINT,'*** M must not be less than 0, routine LEGENDRE.'
RETURN,X
END ELSE IF M GT L THEN BEGIN
PRINT,'*** M must not be greater than L, routine LEGENDRE.'
RETURN,X
END ELSE BEGIN
S = SIZE(X)
IF S(0) EQ 0 THEN TEST = ABS(X) ELSE TEST = MAX(ABS(X))
IF TEST GT 1 THEN BEGIN
PRINT,'*** X must be in the range -1 to 1, routine LEGENDRE.'
RETURN,X
ENDIF
ENDELSE
;
PMM = 0.*X + 1.
IF M GT 0 THEN BEGIN
SOMX2 = SQRT( (1. - X) * (1. + X) )
FACT = 1.
FOR I = 1,M DO BEGIN
PMM = -PMM*FACT*SOMX2
FACT = FACT + 2.
ENDFOR
ENDIF
;
IF L EQ M THEN PLGNDR = PMM ELSE BEGIN
PMMP1 = X * (2*M + 1) * PMM
IF L EQ M+1 THEN PLGNDR = PMMP1 ELSE BEGIN
FOR LL = M+2,L DO BEGIN
PLL = (X*(2*LL-1)*PMMP1 - (LL+M-1)*PMM) / (LL-M)
PMM = PMMP1
PMMP1 = PLL
ENDFOR
PLGNDR = PLL
ENDELSE
ENDELSE
;
RETURN,PLGNDR
END

```
