Introduction to Maple for Mathcad Users

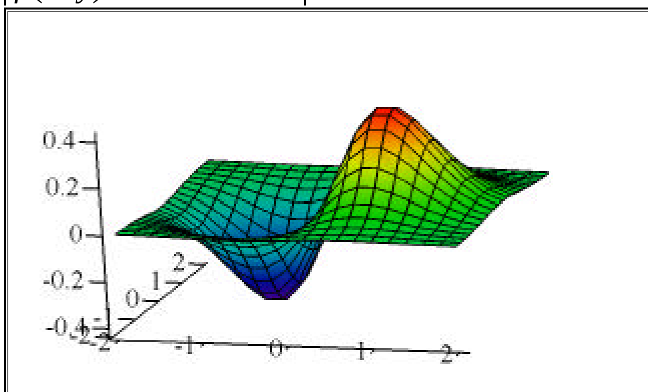
3-D Plots

```
> restart;
```

```
> f := (x,y) -> x*exp(-(x^2+y^2));
```

$$f(x, y) := x \cdot e^{-\left(x^2+y^2\right)}$$

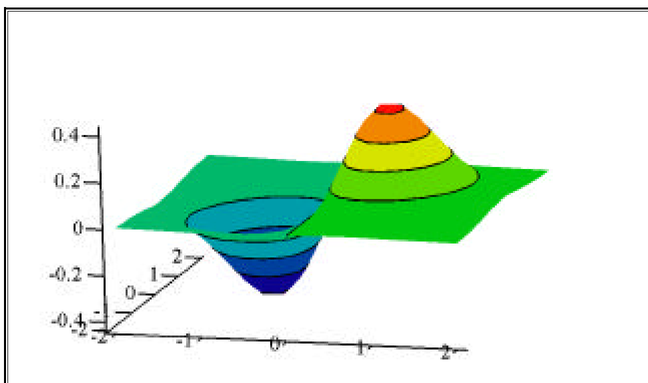
```
> plot3d(f(x,y), x=-2..2,
                 y=-2..2,
                 axes=boxed);
```



f

```
> with(plots);
```

```
> contourplot3d(f(x,y), x=-2..2,
                        y=-2..2,
                        axes=boxed);
```



f

Calculus

```
> P := (n) -> sum(i^3, i=1..n);
```

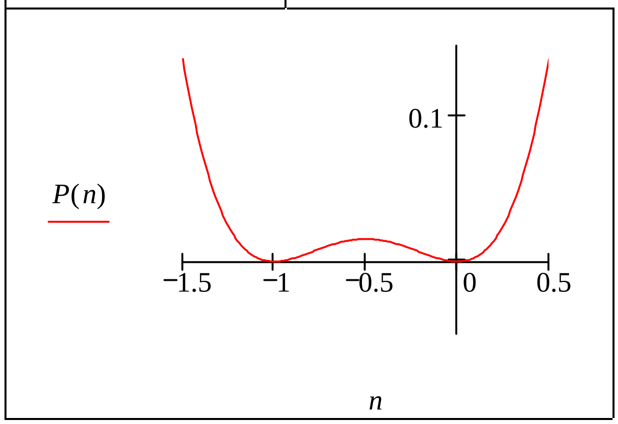
$$P(n) := \sum_{i=1}^{n} i^3 \rightarrow \frac{1}{4}\cdot(n+1)^4 - \frac{1}{2}\cdot(n+1)^3 + \frac{1}{4}\cdot(n+1)^2$$

```
> factor(P(n));
```

$$P(n)\ factor\ \rightarrow \frac{1}{4}\cdot n^2\cdot(n+1)^2$$

```
> plot(P(n), n=-1.5..0.5);
```

$$n := -1.5, -1.49 .. 0.5$$




```
> Pp := diff(P(n), n);
> Pp := factor(Pp);
> Pp := unapply(Pp, n);
> Pp(n);
```

$$P'(n) := \frac{d}{dn}P(n)\ factor\ \rightarrow \frac{1}{2}\cdot n\cdot(n+1)\cdot(2\cdot n+1)$$

```
> x := solve( Pp(n)=0, n );
```

$$x := (P'(n) = 0)\ solve, n \rightarrow \begin{pmatrix} 0 \\ -1 \\ \frac{-1}{2} \end{pmatrix}$$

```
> Ppp := diff(P(n), n, n);
> Ppp := factor(Ppp);
> Ppp := unapply(Ppp, n);
```

$$P''(n) := \frac{d^2}{dn^2}P(n)\ factor\ \rightarrow 3\cdot n^2 + 3\cdot n + \frac{1}{2}$$

```
> Ppp(x[1]), Ppp(x[2]), Ppp(x[3]);
            -or-
> map(Ppp, [x]);
```

$$P''(x) = \begin{pmatrix} 0.5 \\ 0.5 \\ -0.25 \end{pmatrix}$$

p" positive -> local minimum
p" negative -> local maximum

Linear Algebra

```
> x[1]; x[2]; x[3];
```

$ORIGIN := 1$

$x_1 = 0$ $x_2 = -1$ $x_3 = -0.5$

```
> with(linalg);
> A := array([[ 1 ,1/2,1/3],
              [1/2,1/3,1/4],
              [1/3,1/4,1/5]]);
```

$$A := \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{pmatrix}$$

```
> A[1,1];
```

$A_{1,1} = 1$

```
> x := array([[ x[1] ],
              [ x[2] ],
              [ x[3] ]]);

# x was a sequence, now an array.
```

$$x \rightarrow \begin{pmatrix} 0 \\ -1 \\ \frac{-1}{2} \end{pmatrix}$$

x was always an array, so just re-displaying

```
> b := evalm(A &* x);
```

$b := A \cdot x$

```
> eval(b);
```

$$b \rightarrow \begin{pmatrix} \frac{-2}{3} \\ \frac{-11}{24} \\ \frac{-7}{20} \end{pmatrix}$$

```
> evalf(eval(b), 3);
```

$$b = \begin{pmatrix} -0.667 \\ -0.458 \\ -0.35 \end{pmatrix}$$

Solving Linear Equations - 3 Methods

```
> M := augment(A, b);
```

$$M := augment(A,b) \rightarrow \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{-2}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{-11}{24} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{-7}{20} \end{pmatrix}$$

```
> R := rref(M);
```

$$R := rref(M) \rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & \frac{-1}{2} \end{pmatrix}$$

```
> x := col(R, 4);
```

$$x := R^{\langle 4 \rangle} \rightarrow \begin{pmatrix} 0 \\ -1 \\ \frac{-1}{2} \end{pmatrix}$$

```
> Ainv := inverse(A);
```

$$A_{inv} := A^{-1} \rightarrow \begin{pmatrix} 9 & -36 & 30 \\ -36 & 192 & -180 \\ 30 & -180 & 180 \end{pmatrix}$$

```
> x := evalm(Ainv &* b);
```

$$x := A_{inv} \cdot b \rightarrow \begin{pmatrix} 0 \\ -1 \\ \frac{-1}{2} \end{pmatrix}$$

```
> x := linsolve(A, b);
```

$$x := lsolve(A,b) \rightarrow \begin{pmatrix} 0 \\ -1 \\ \frac{-1}{2} \end{pmatrix}$$

Determinants and Eigenvalues

```
> det(A),          det(Ainv);
```

$|A| \rightarrow \frac{1}{2160}$ $\quad |A^{-1}| \rightarrow 2160$

```
> Id  := array([[1,0,0],
                [0,1,0],
                [0,0,1]]);
```

$I := identity(3) \rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

```
> p := det( evalm(A - lambda*Id) );
> p := unapply(p, lambda);
```

$p(l) := |A - l \cdot I| \rightarrow \frac{1}{2160} - \frac{127}{720} \cdot l + \frac{23}{15} \cdot l^2 - l^3$

```
> lambda := solve(p(lambda)=0, lambda):
```

$l := (p(l) = 0)\ solve, l \rightarrow \left[\begin{array}{l} \frac{-1}{360} \cdot (517148 + 405 \cdot i \cdot \sqrt{8 \\ \frac{-1}{360} \cdot (517148 + 405 \cdot i \cdot \sqrt{8 \end{array} \right.$

```
> evalf( [lambda], 6);
```

$l = \begin{pmatrix} 1.40832 \\ 0.00269 \\ 0.12233 \end{pmatrix}$

```
> check := product(lambda[i], i=1..3):
> simplify(expand(check));
```

$check := \prod_{i=1}^{3} l_i \left| \begin{array}{l} expand \\ simplify \end{array} \right. \rightarrow \frac{1}{2160}$

```
> evalf(eigenvals(A),6);
```

$eigenvals(A) = \begin{pmatrix} 0.12233 \\ 0.00269 \\ 1.40832 \end{pmatrix}$