

## Programme établi par: Laïd MESSAOUDI (2017)

```
> restart  
> with(CurveFitting) : with(StringTools) : with(LinearAlgebra) : with(plots) :  
>
```

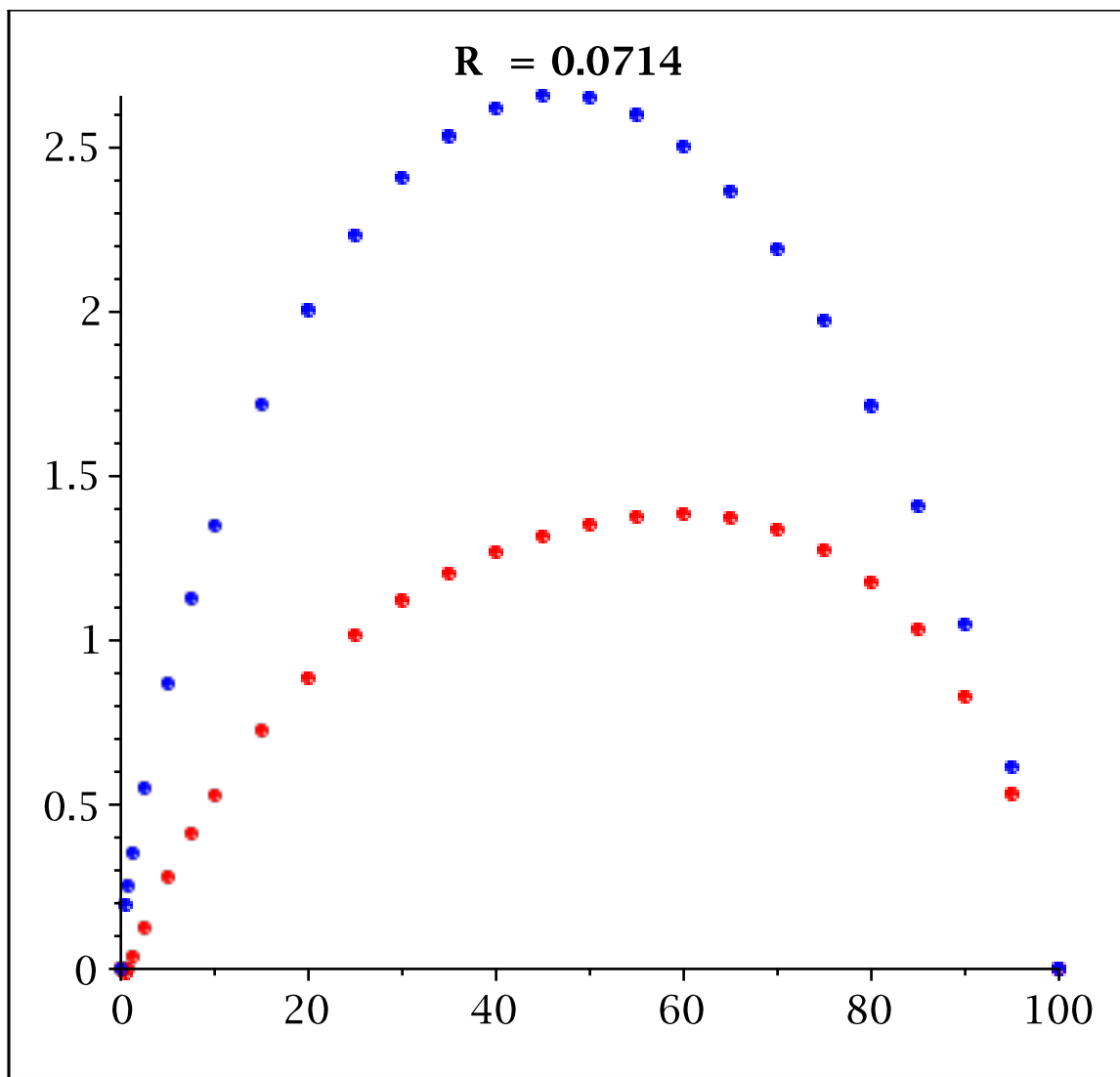
### ▼ Calculs PompAx:

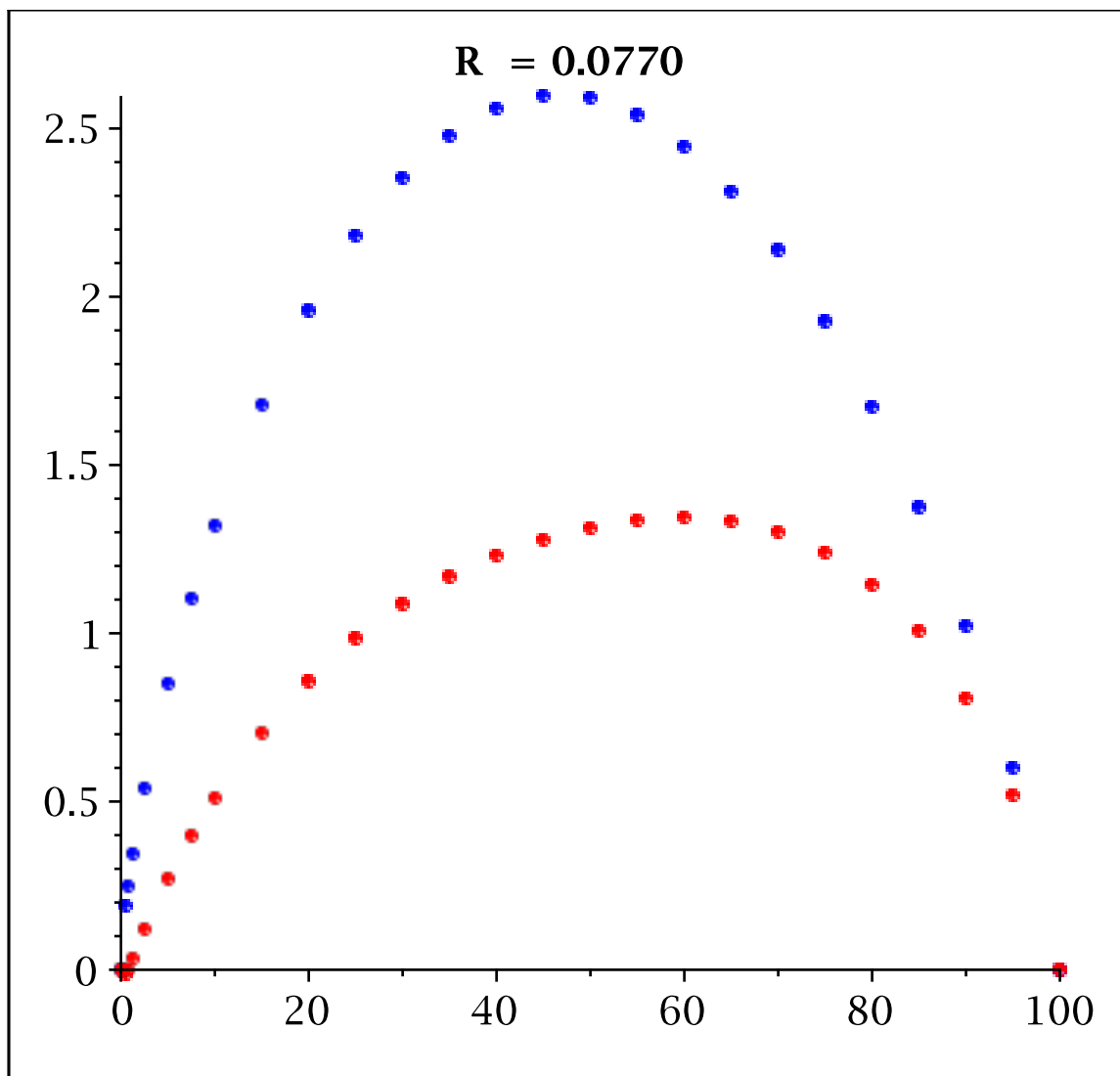
#### ▼ Lecture du fichier de donnée du stator:

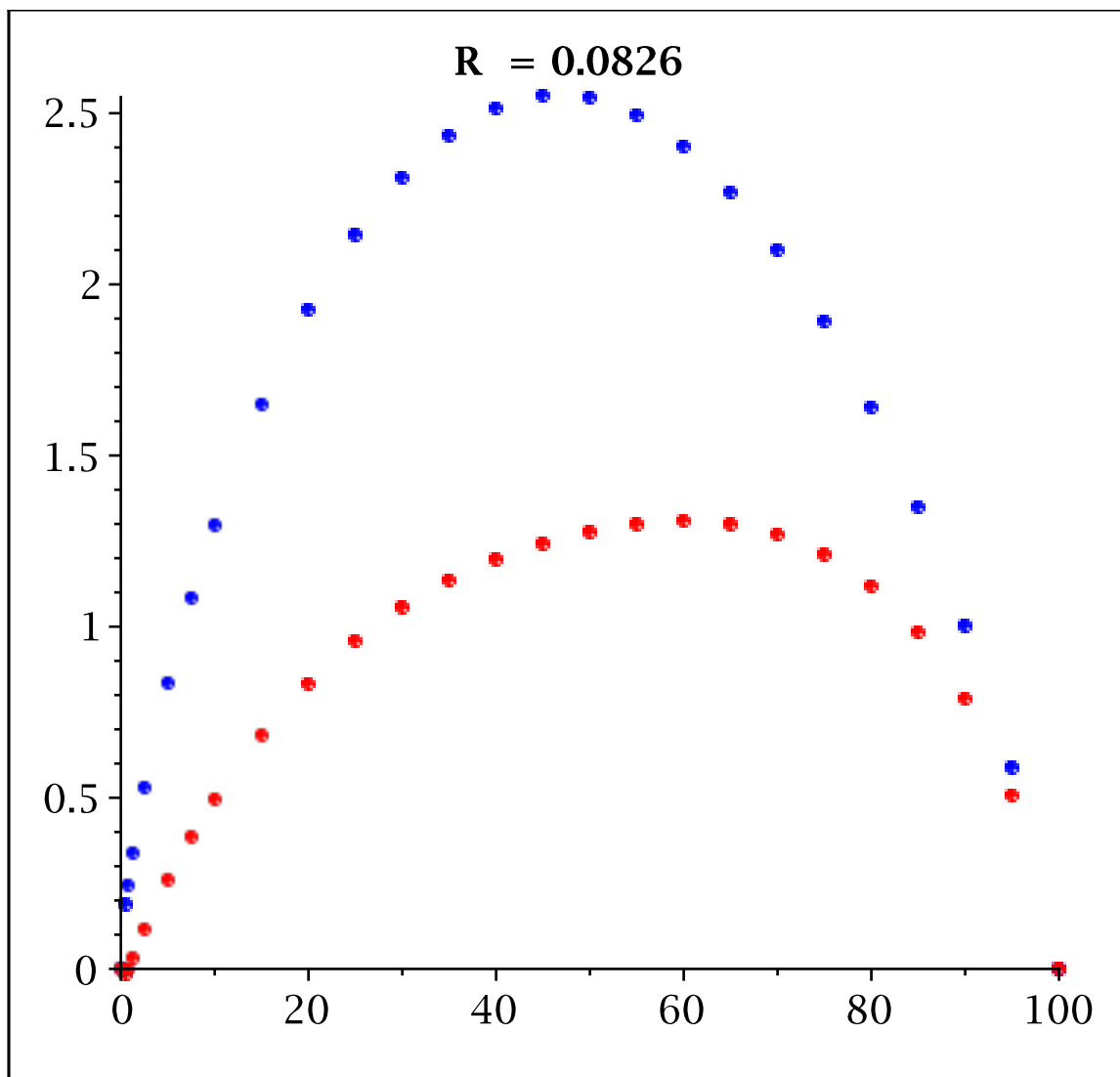
```
> Nc := 16 : Np := 26 :  
> FP := readdata("Aube-Stator.NAC", 4) :  
> Ld := 1 :  
  for k from 1 to Nc do  
    Lf := Ld + Np - 1 :  
    C[k] := [seq(FP[i], i = Ld..Lf) ] :  
    Ld := Lf + 1 :  
  end do:  
> X := [seq(C[1][i, 1], i = 1..Np) ] :  
> for k from 1 to Nc do  
  R[k] := C[k][1, 4];  
  Y1[k] := [seq(C[k][i, 2], i = 1..Np) ];  
  Y2[k] := [seq(C[k][i, 3], i = 1..Np) ];  
end do:
```

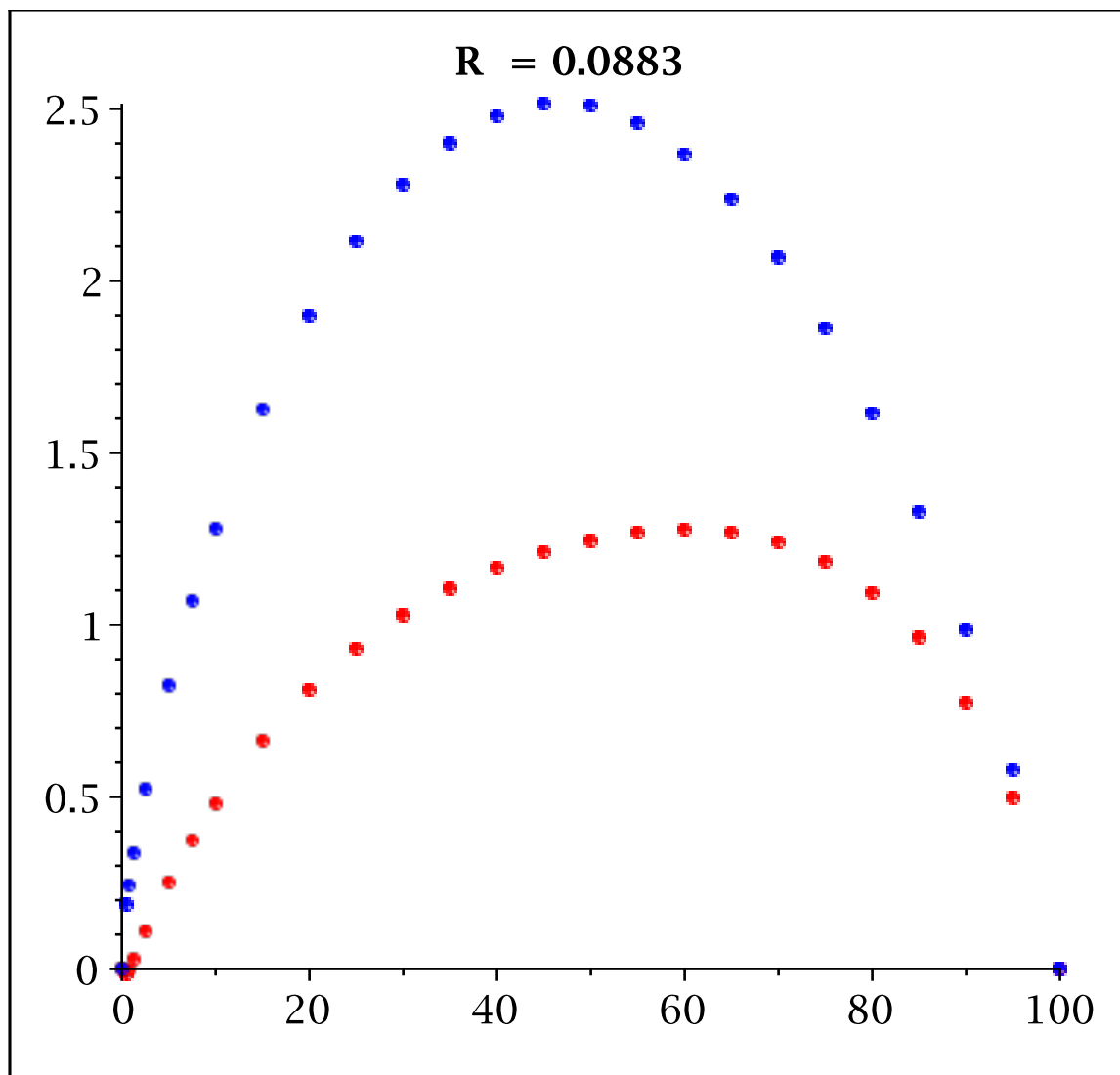
#### ▼ Tracé graphique des profils avant interpolation:

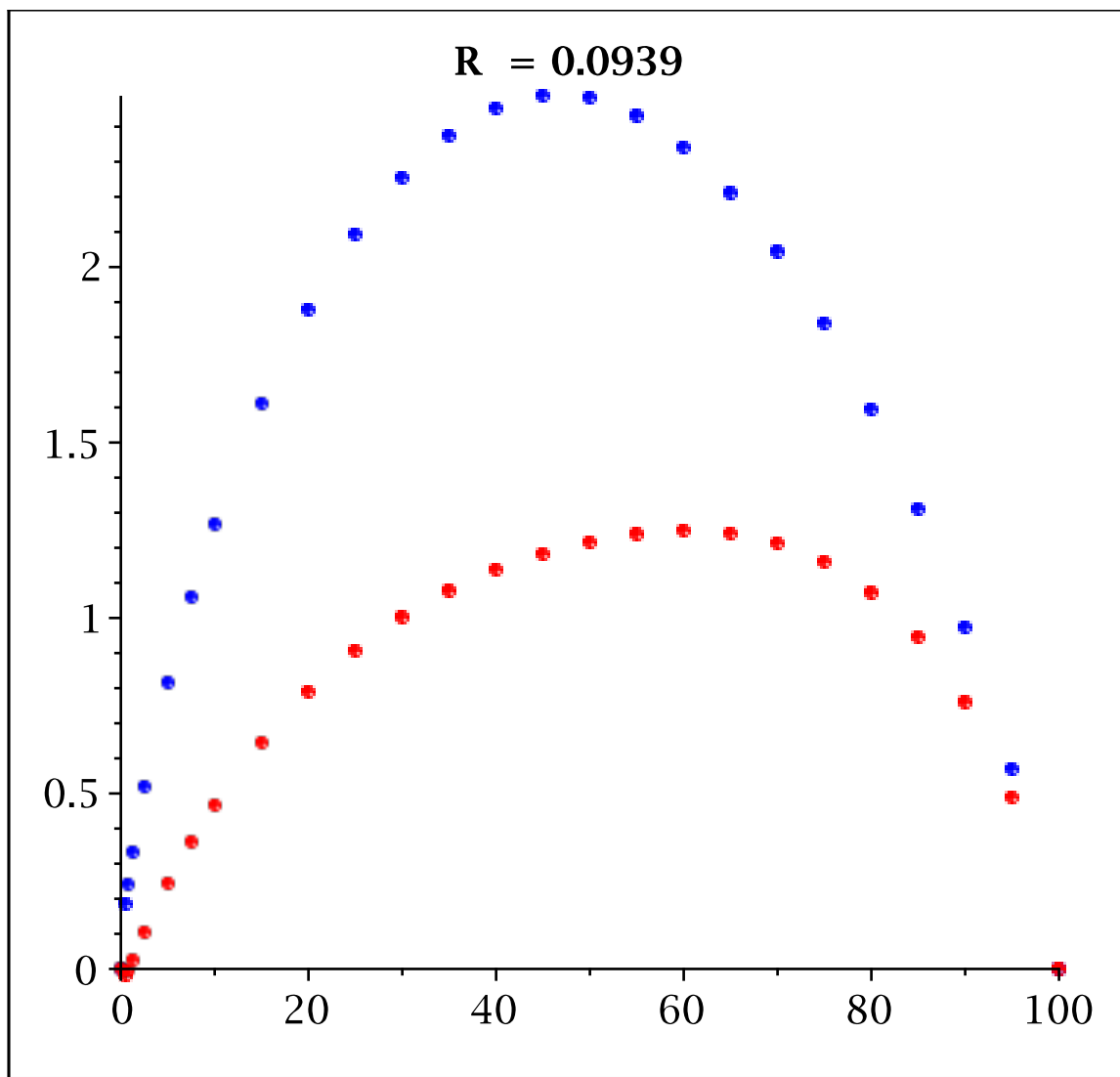
```
> A := Array(1..Nc, 1..1) :  
>  
> for k from 1 to Nc do  
  P[k] := pointplot(X, Y1[k], color = blue, symbol = solidcircle, symbolsize  
    = 8) :  
  P[k+1] := pointplot(X, Y2[k], color = red, symbol = solidcircle, symbolsize  
    = 8) :  
  A[k] := display( {P[k], P[k+1]}, title = [typeset("R = ", R[k]), font  
    = [TIMES, BOLD, 12]]) :  
end do:  
> display(A);
```

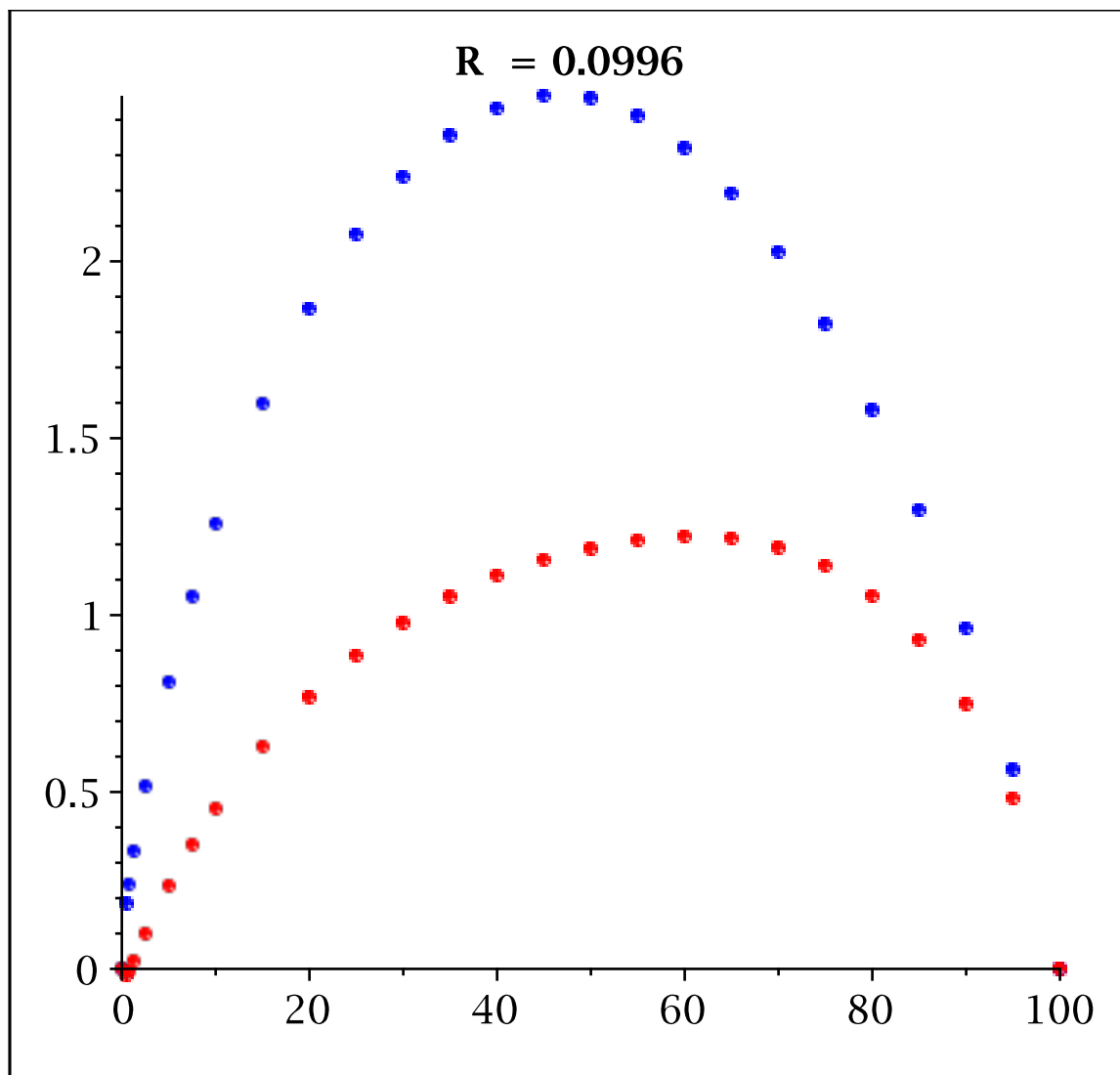


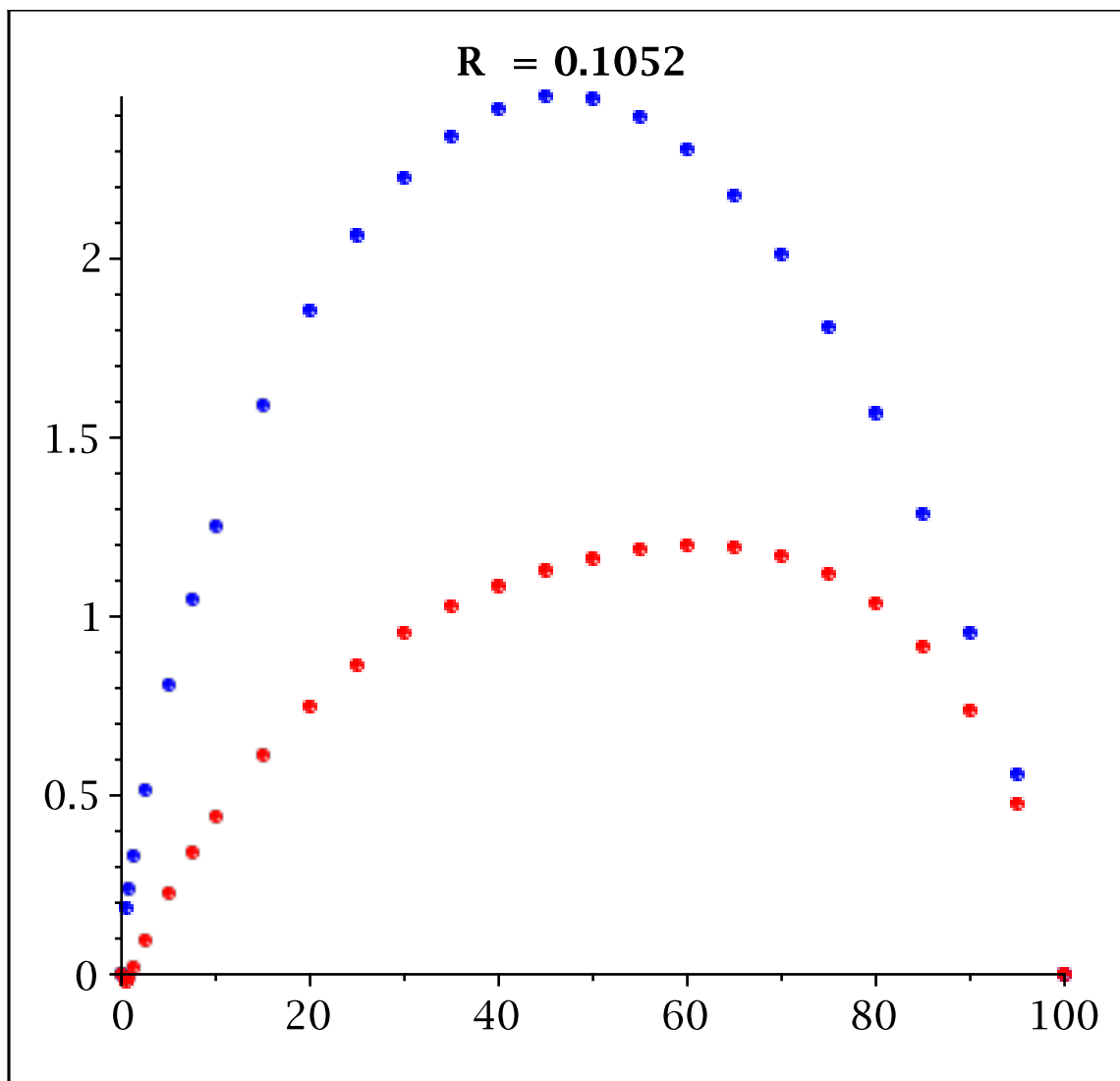




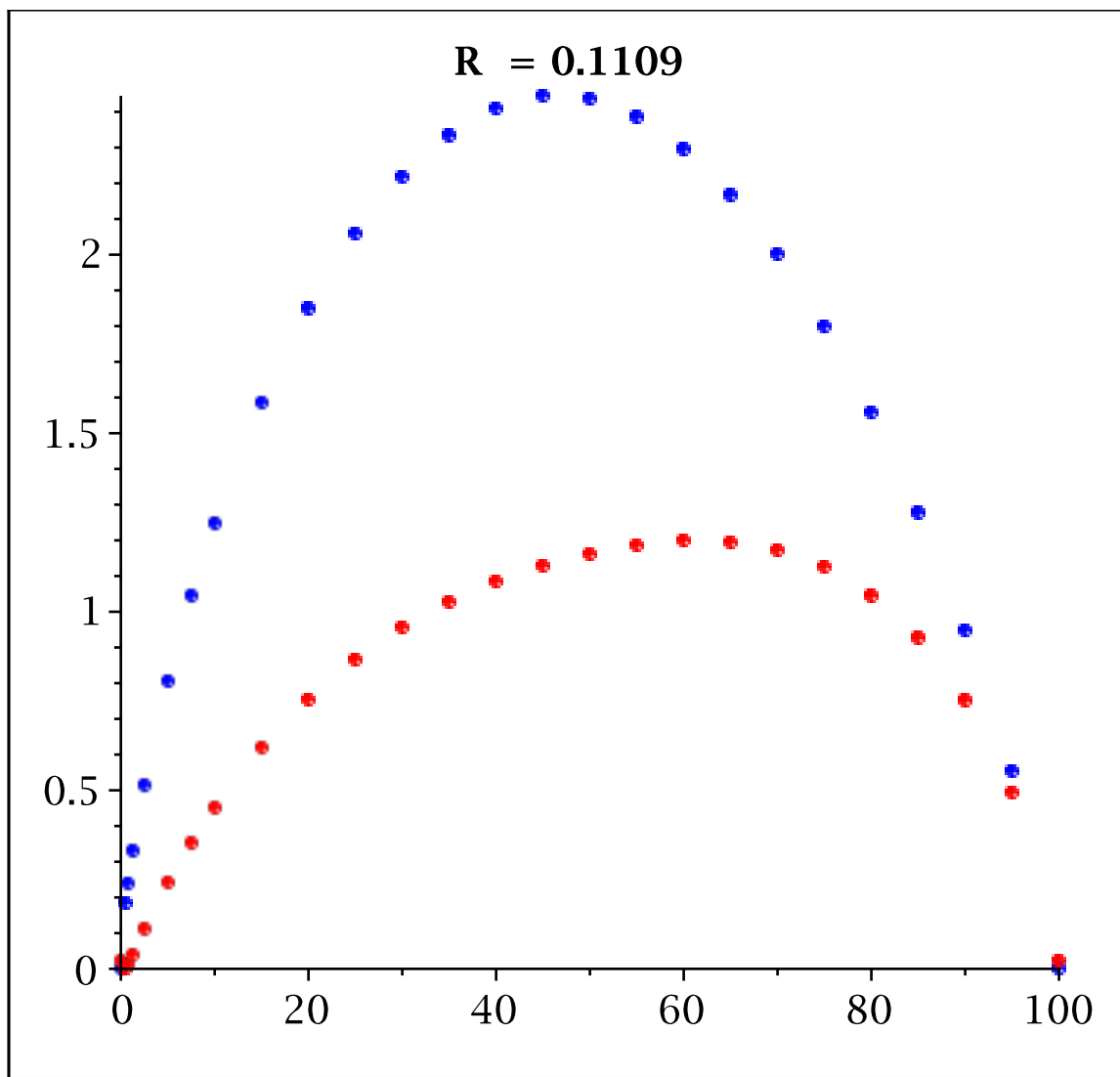


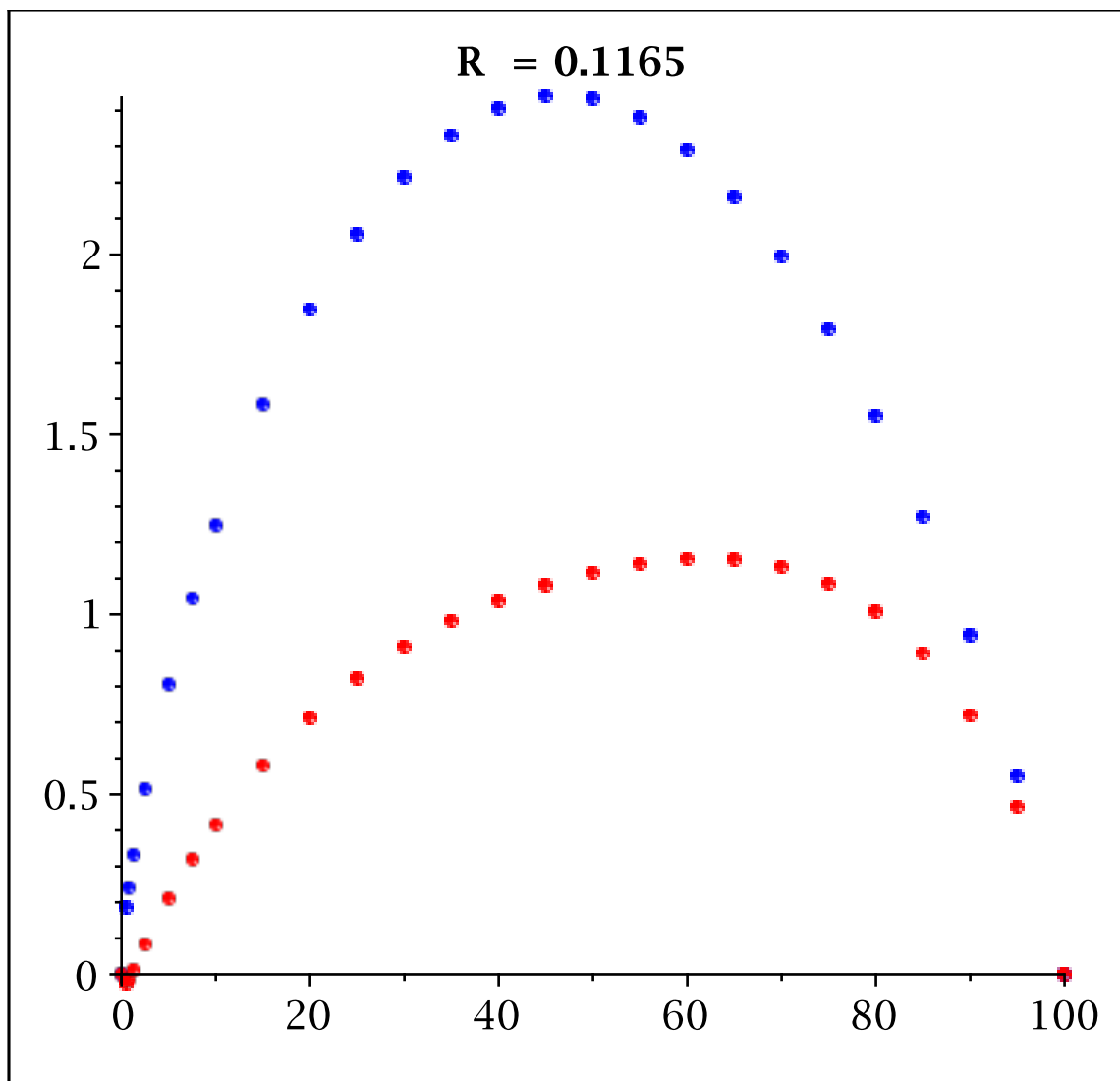


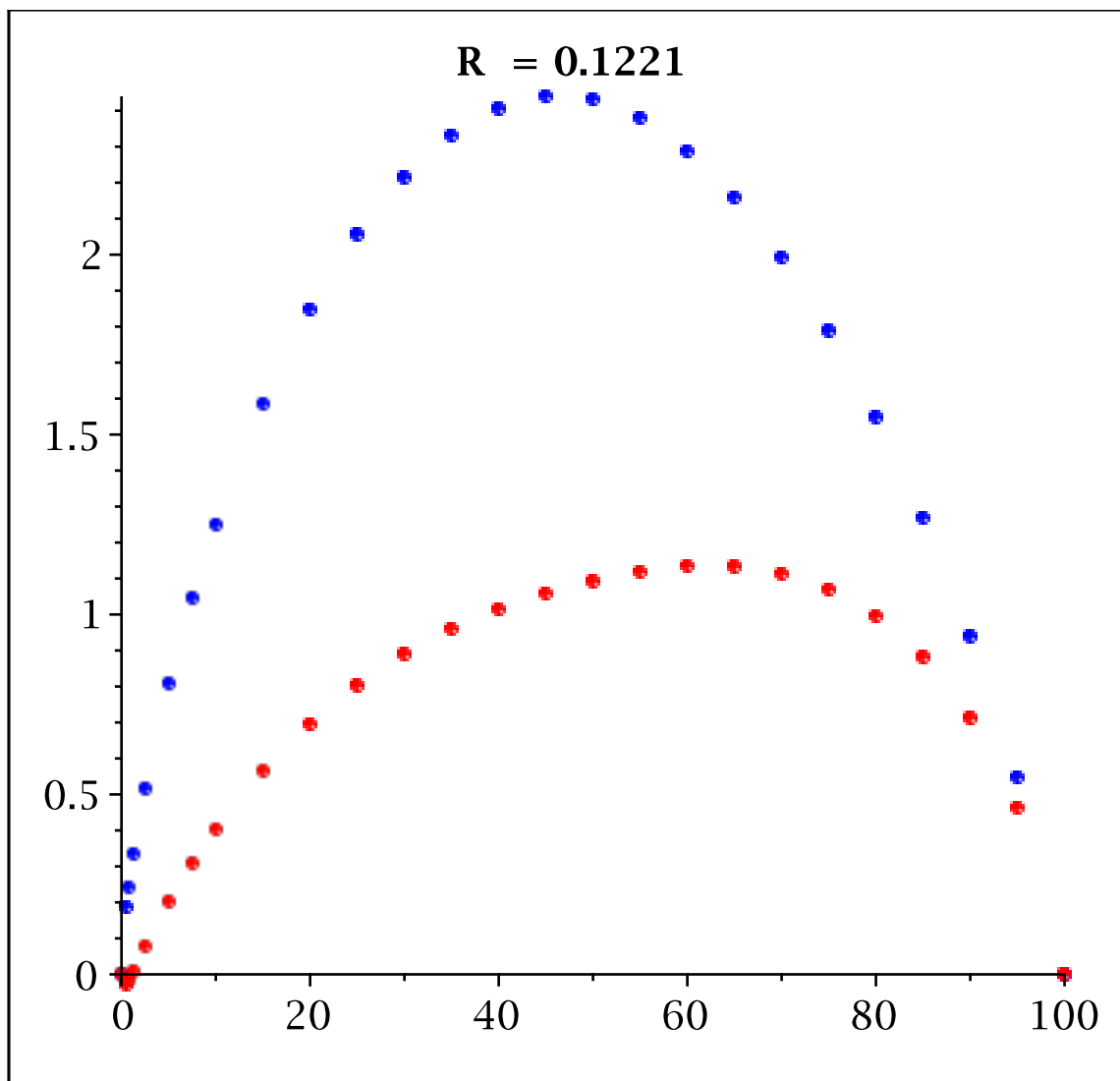


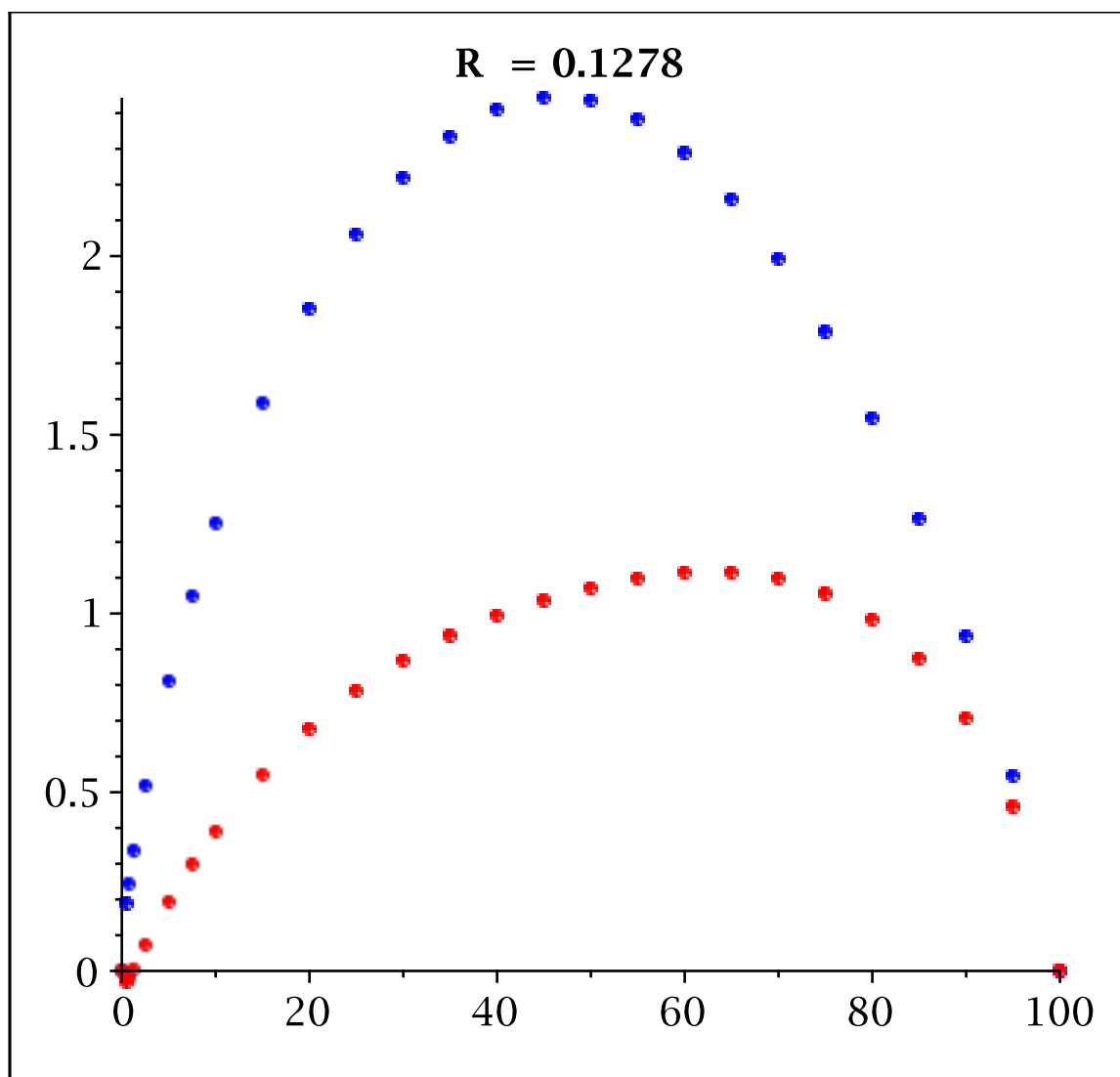


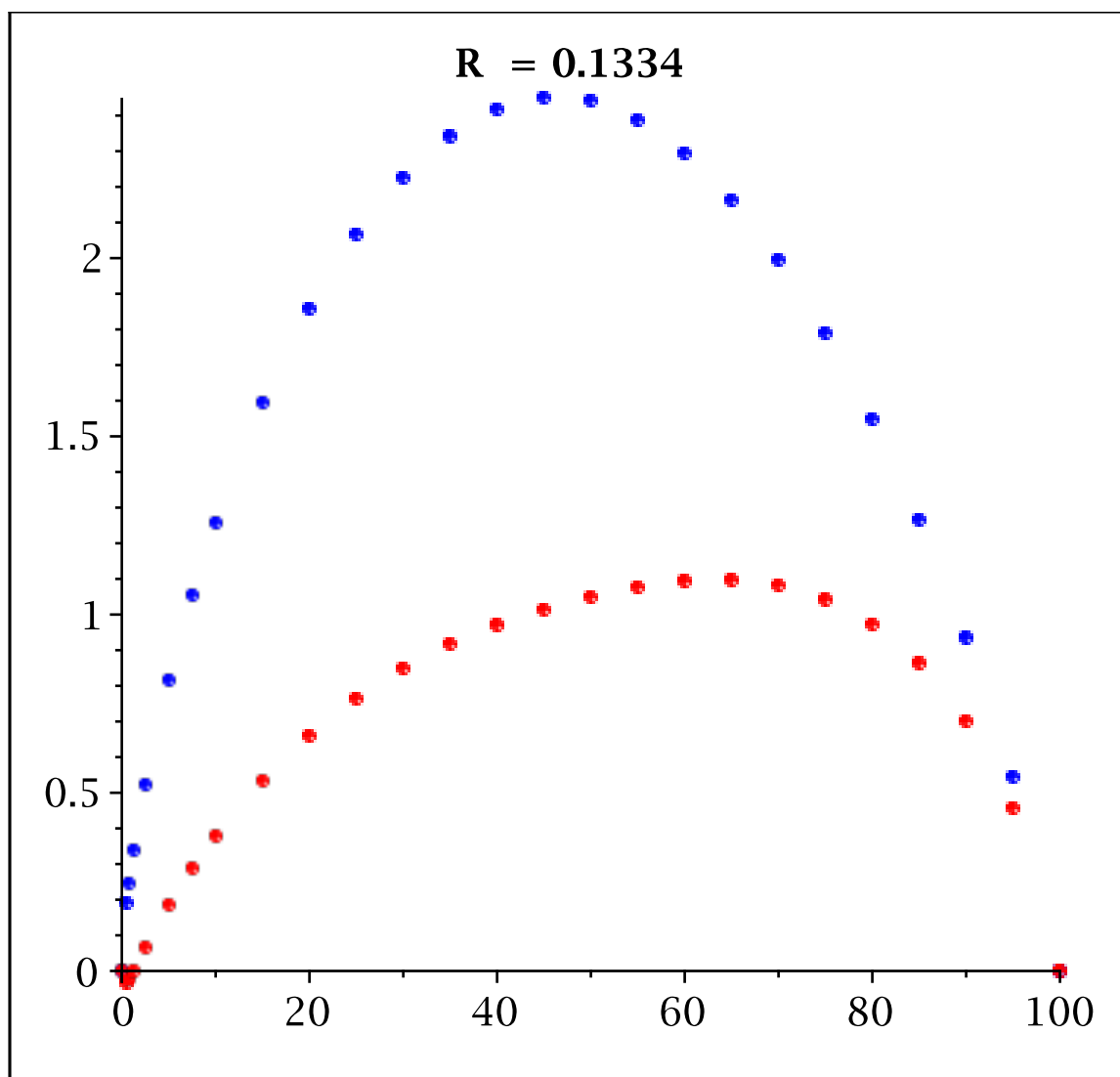


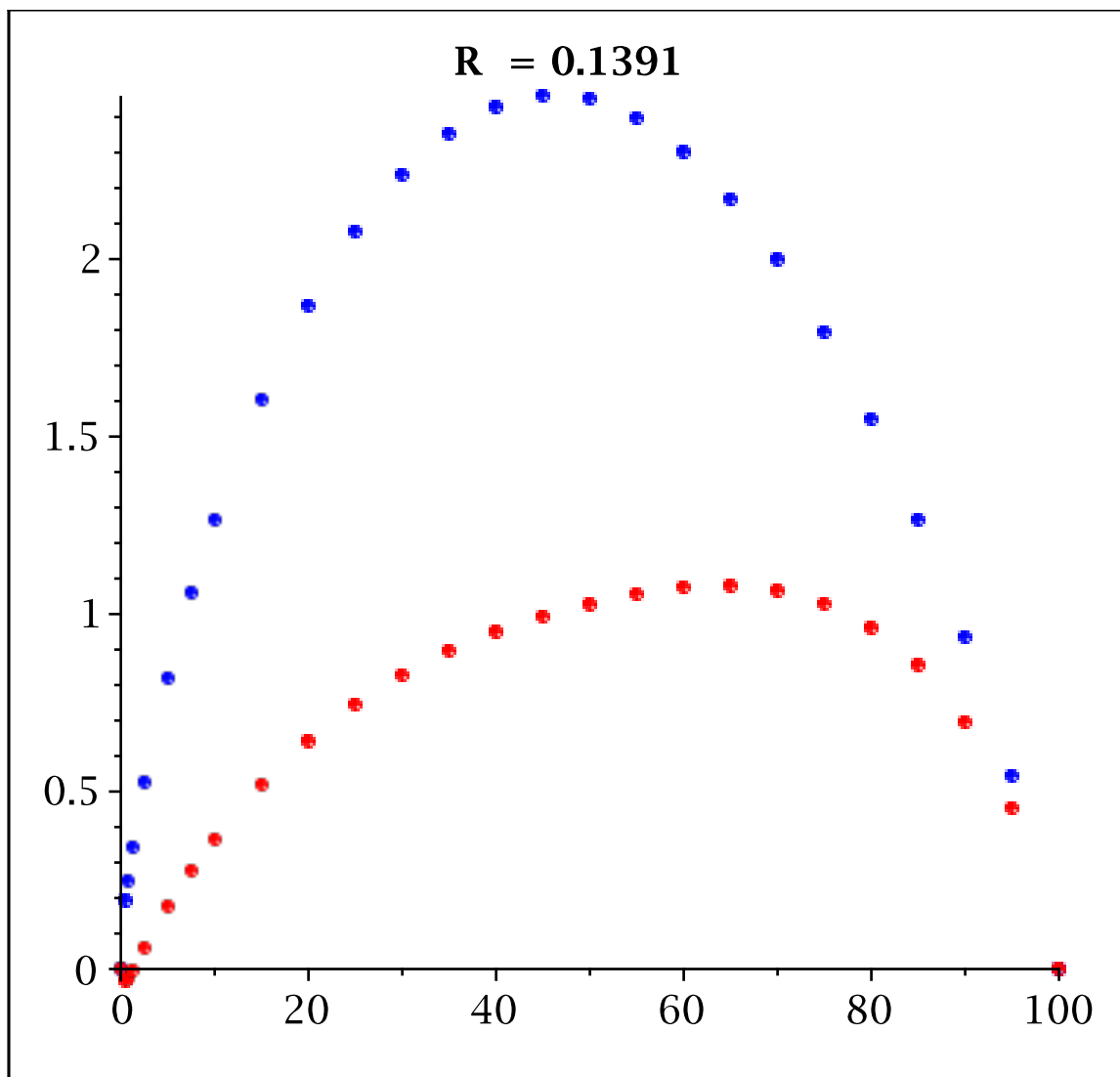


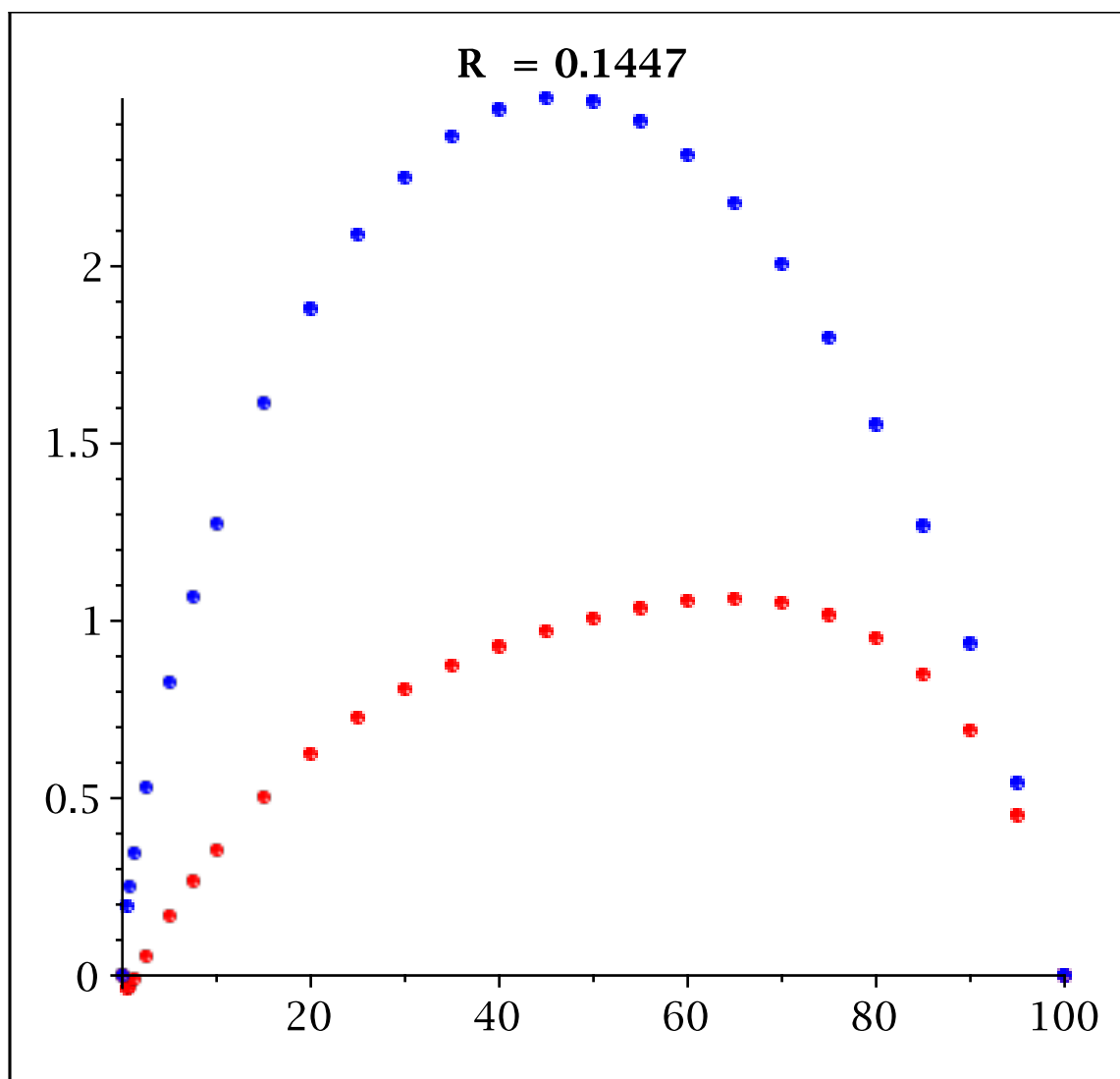


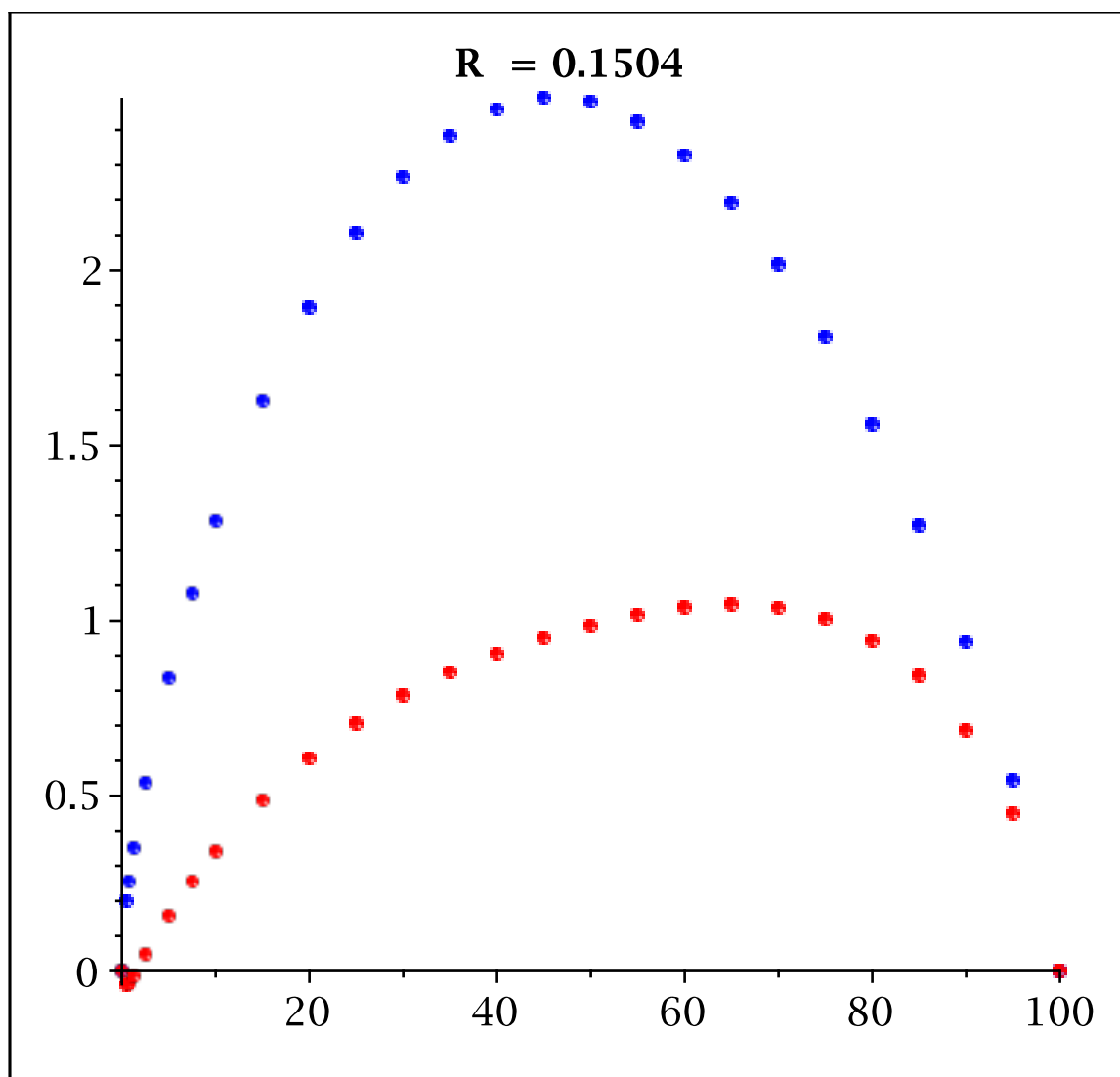




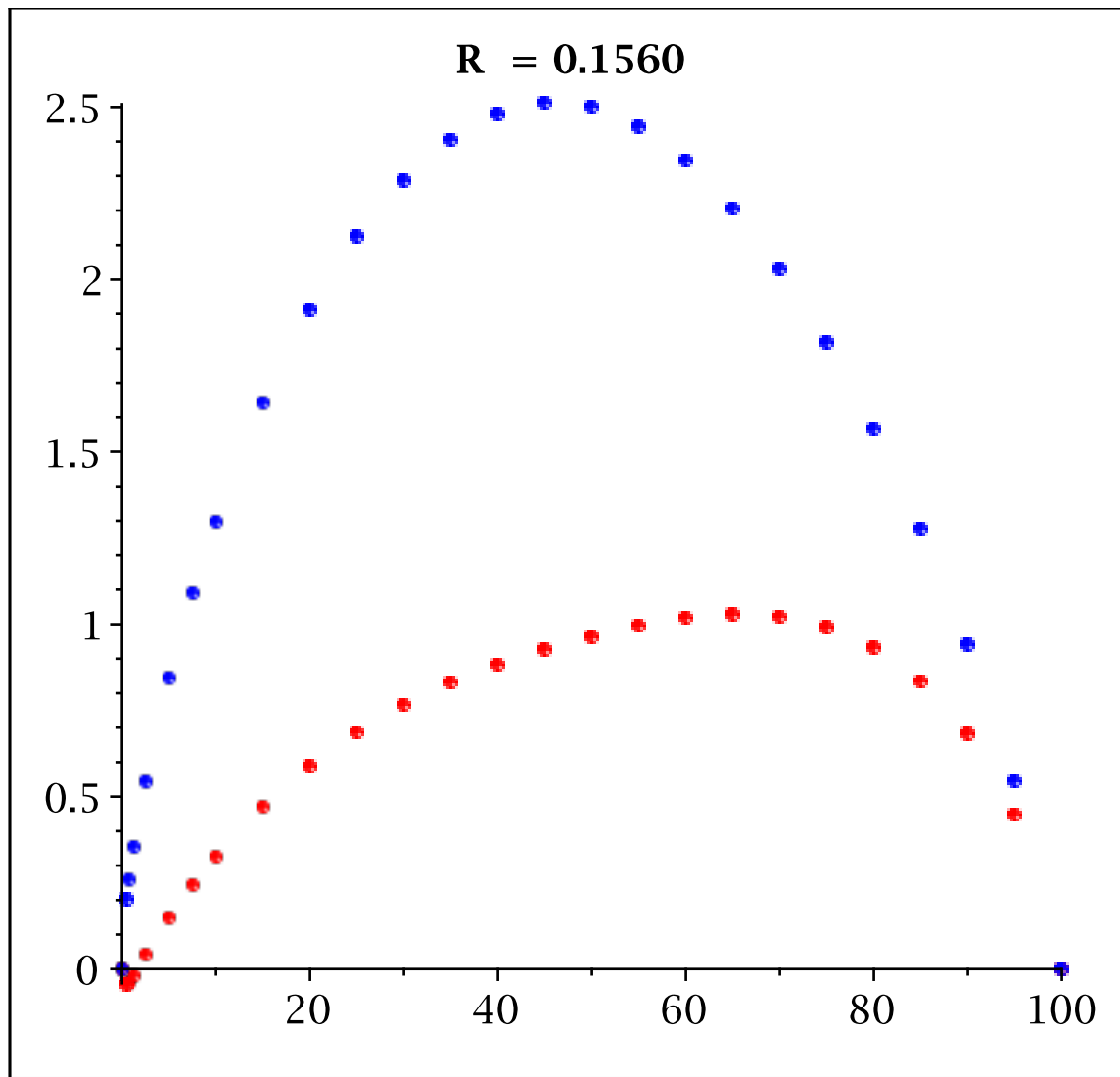












## Interpolation Spline:

```
[> Nps := 200 :
> XX := [ seq( (100. / Nps) * i, i = 0..Nps ) ] :
> for k from 1 to Nc do
  YY1[k] := ArrayInterpolation(X, Y1[k], XX, method = spline, degree = 5) :
  YY2[k] := ArrayInterpolation(X, Y2[k], XX, method = spline, degree = 5) :
end do:
```

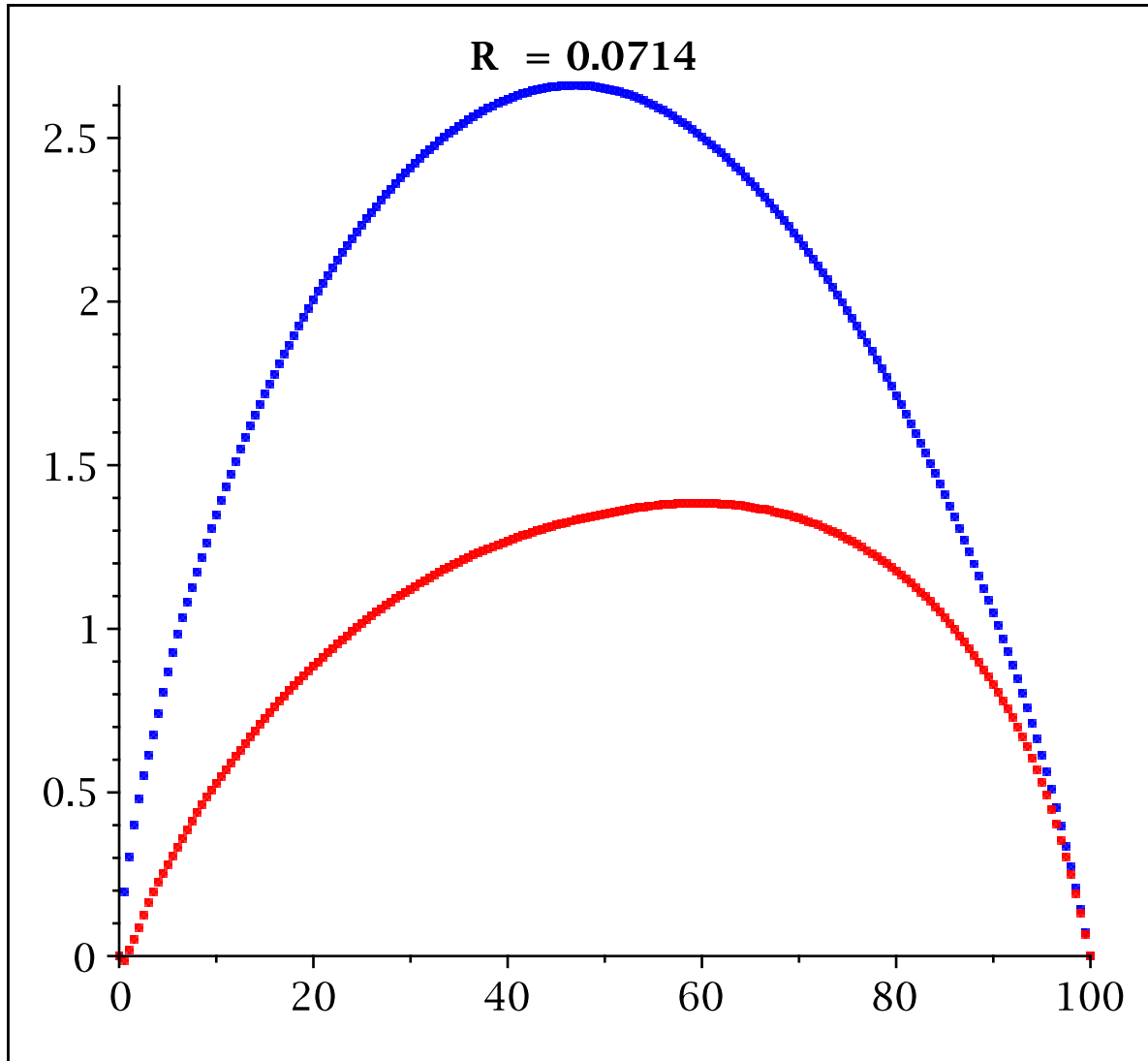
## Tracé graphique des profils après interpolation:

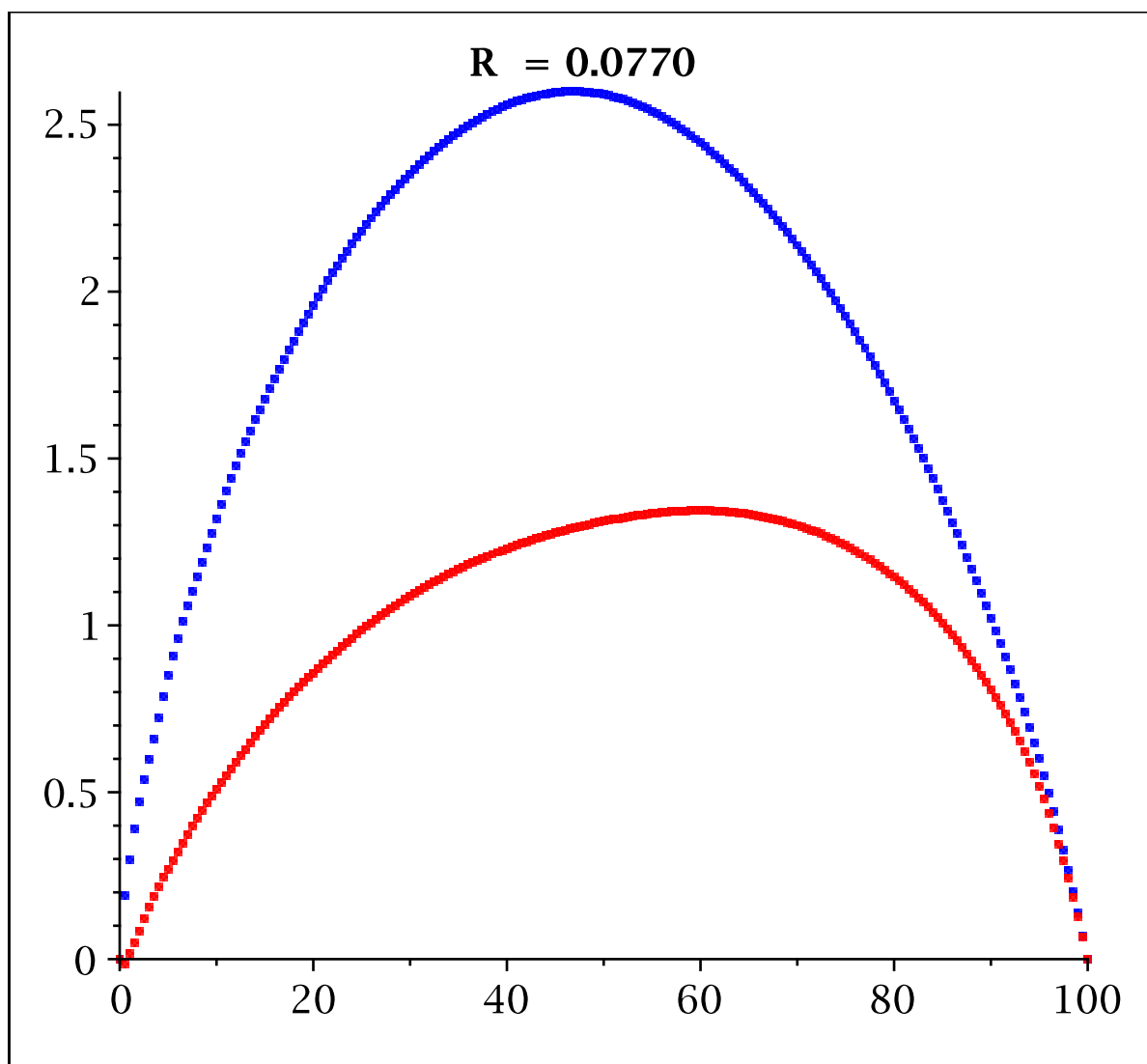
```
[>
> B := Array(1..Nc, 1..1) :
> for k from 1 to Nc do
  PP[k] := pointplot(XX, YY1[k], color = blue, symbol = solidcircle, symbolsize
    = 4) :
  PP[k+1] := pointplot(XX, YY2[k], color = red, symbol = solidcircle, symbolsize
```

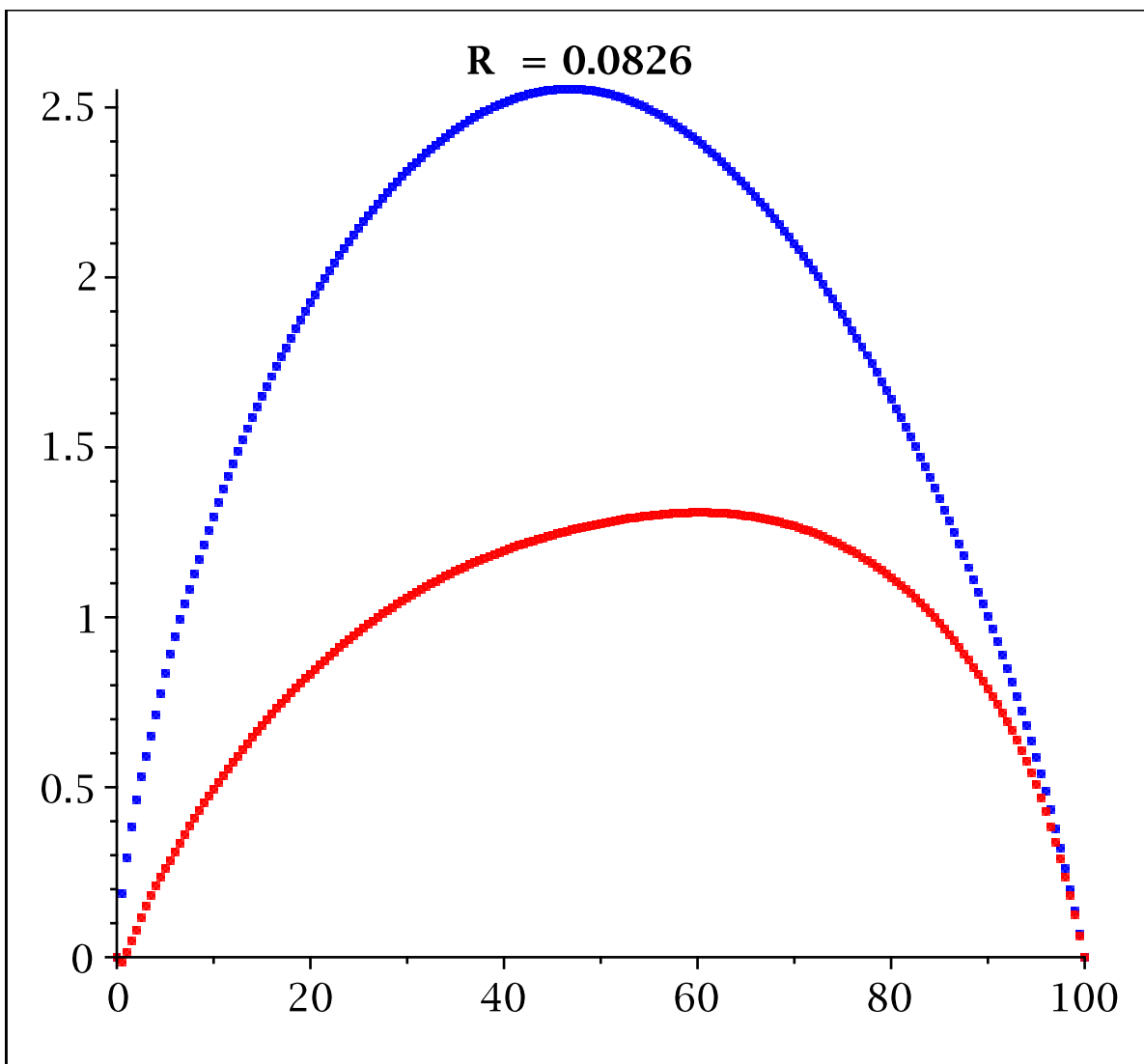
```

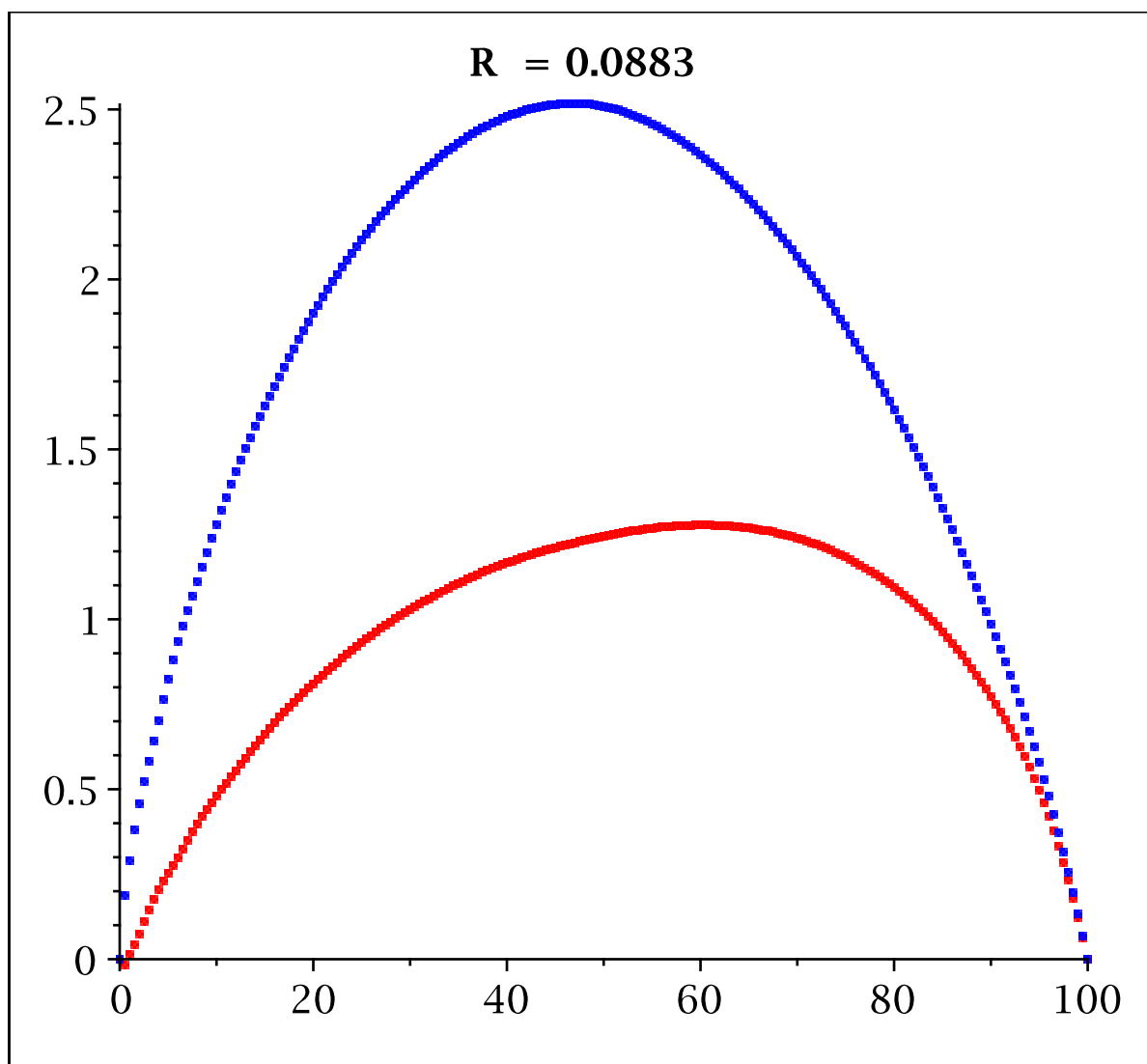
= 4) :
B[k] := display( {PP[k], PP[k + 1]}, title = [typeset("R = ", R[k]), font
= [TIMES, BOLD, 12]]) :
end do:
> display(B);

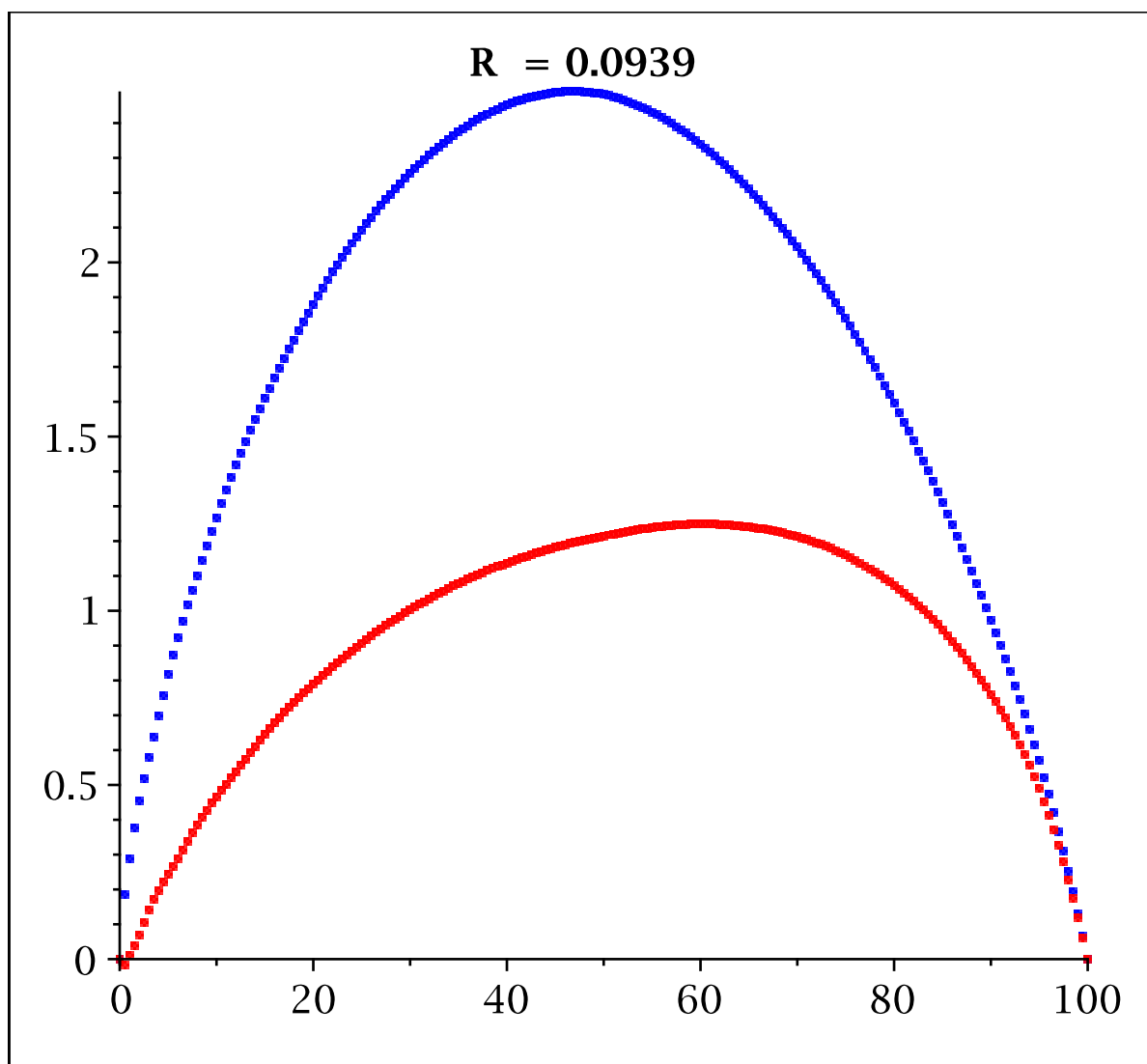
```

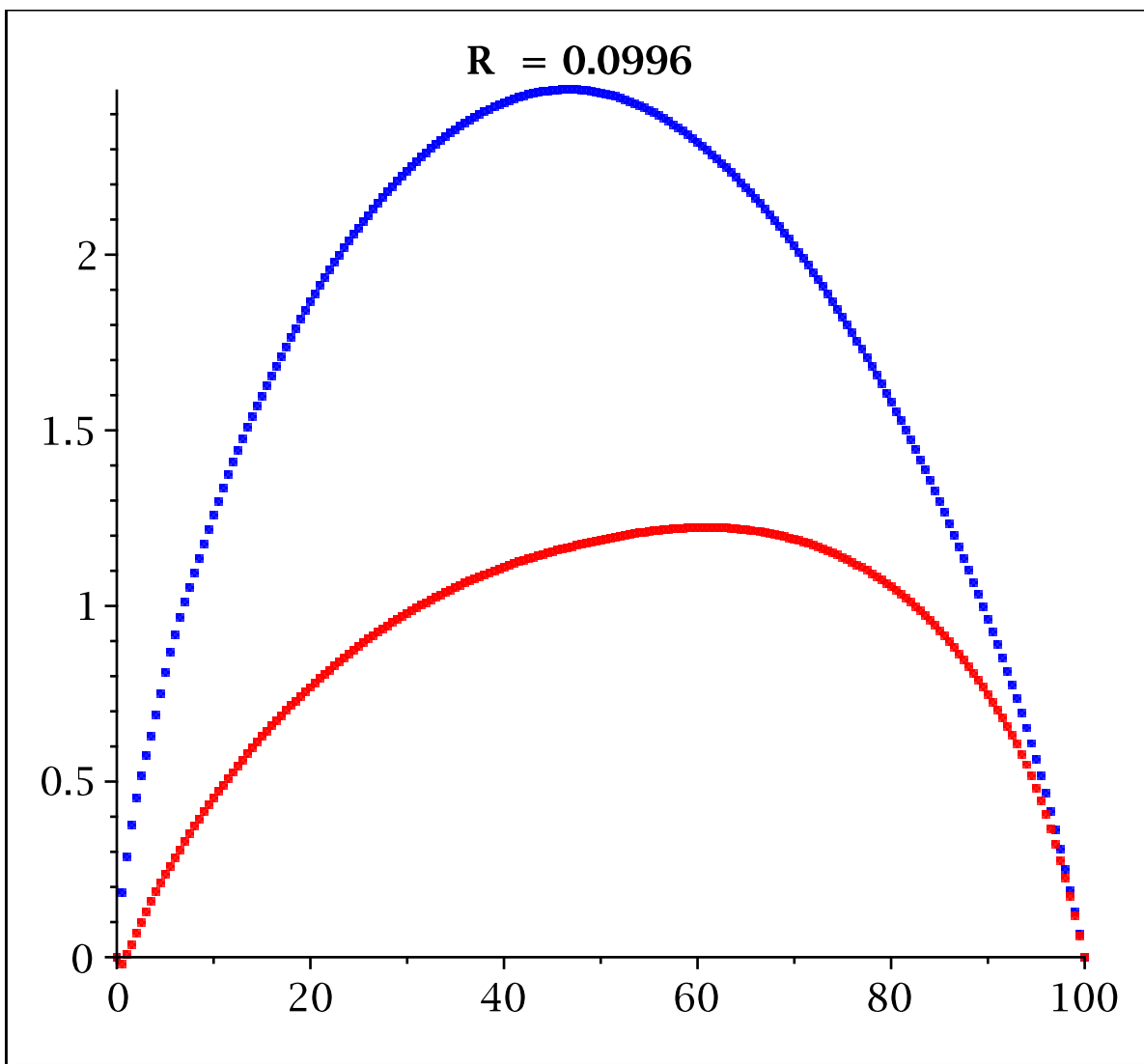


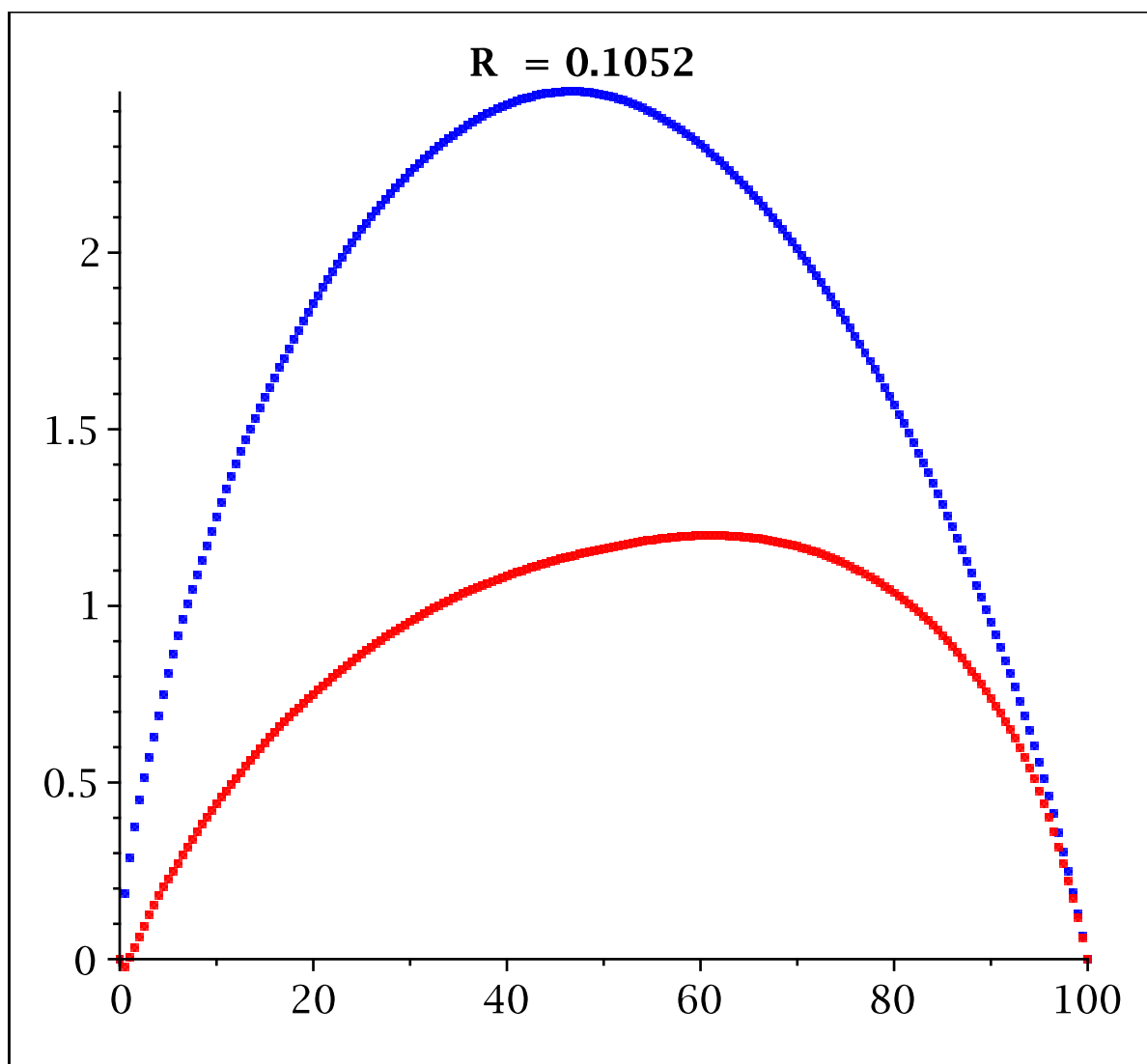




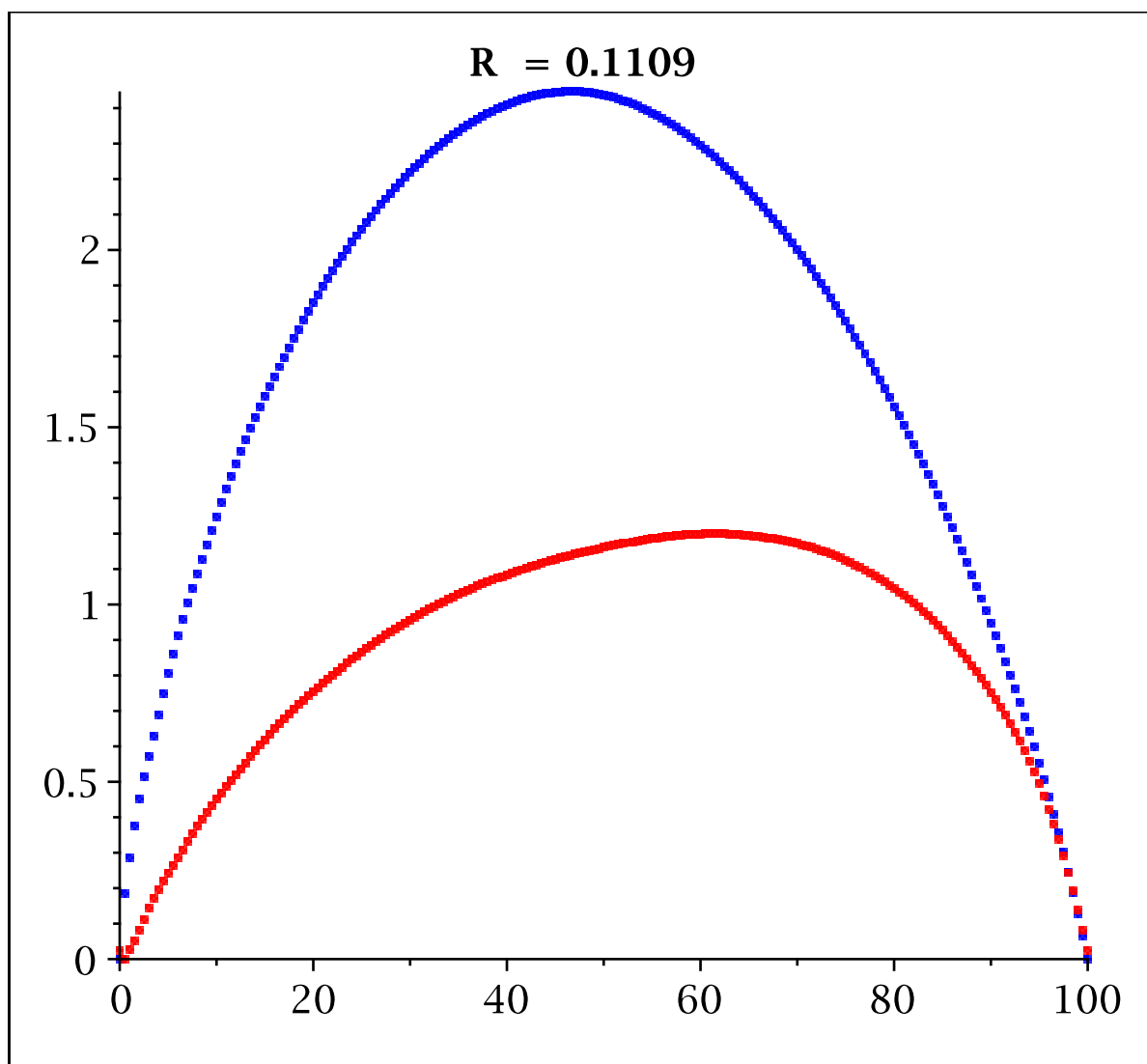


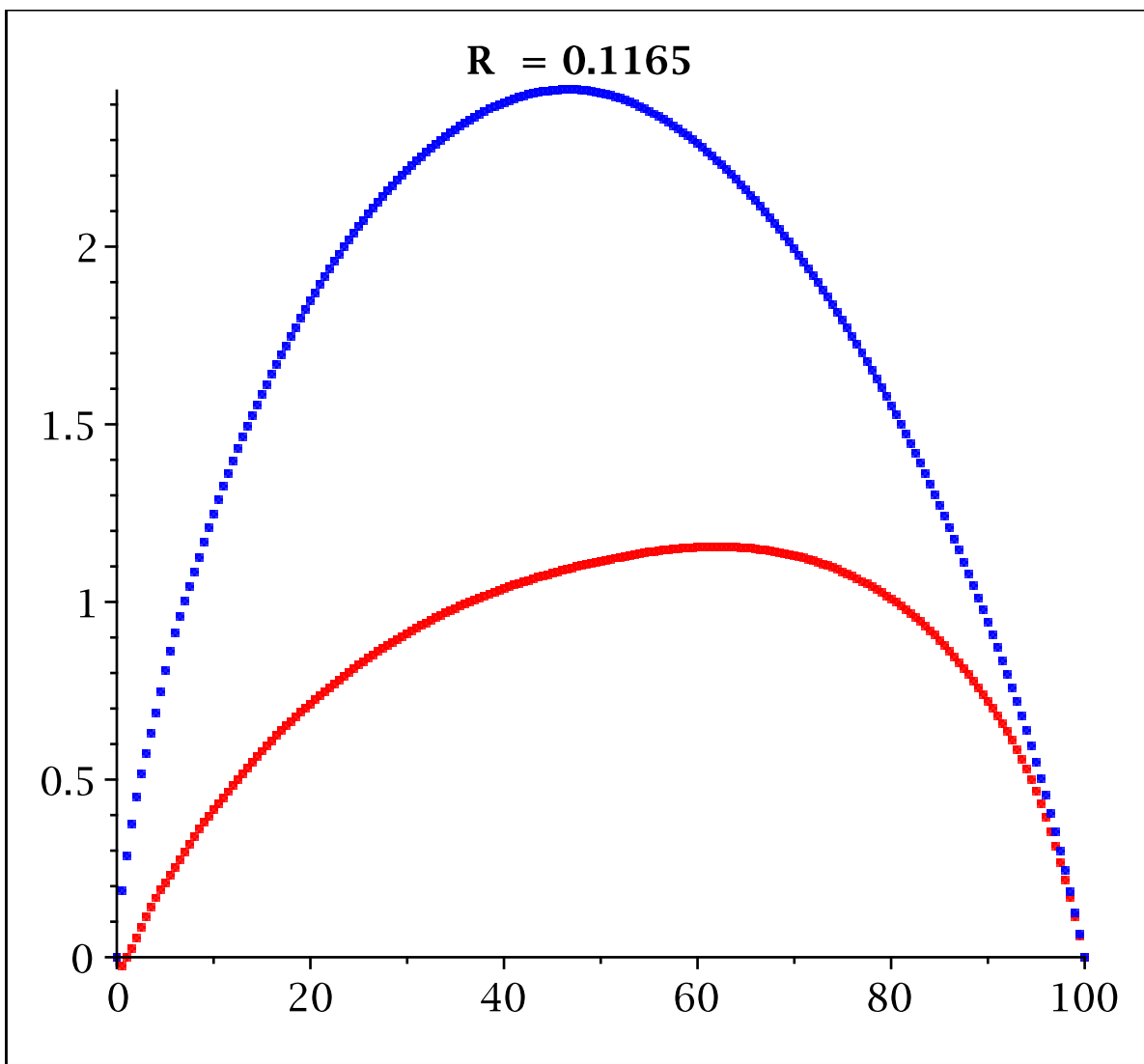


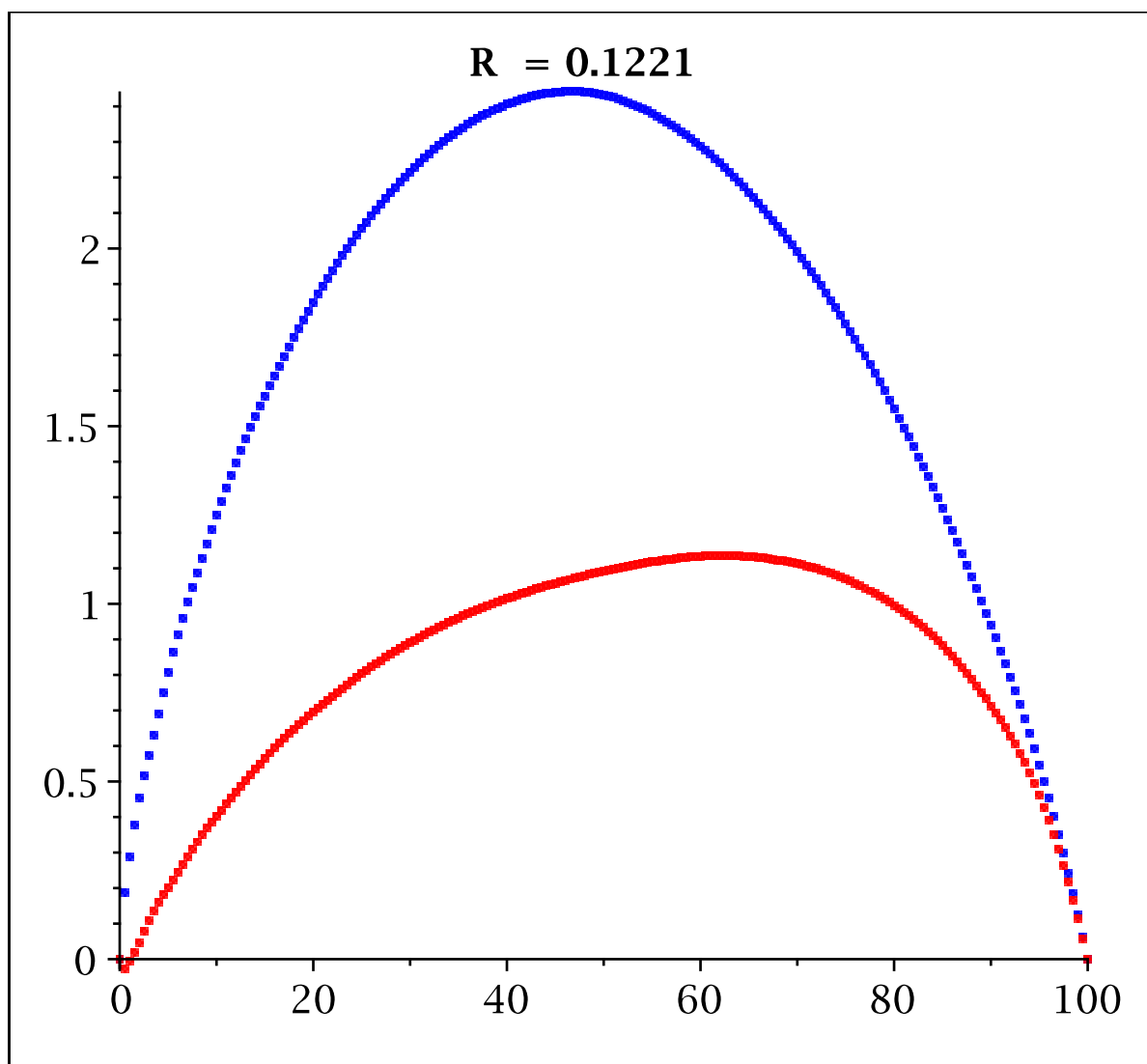


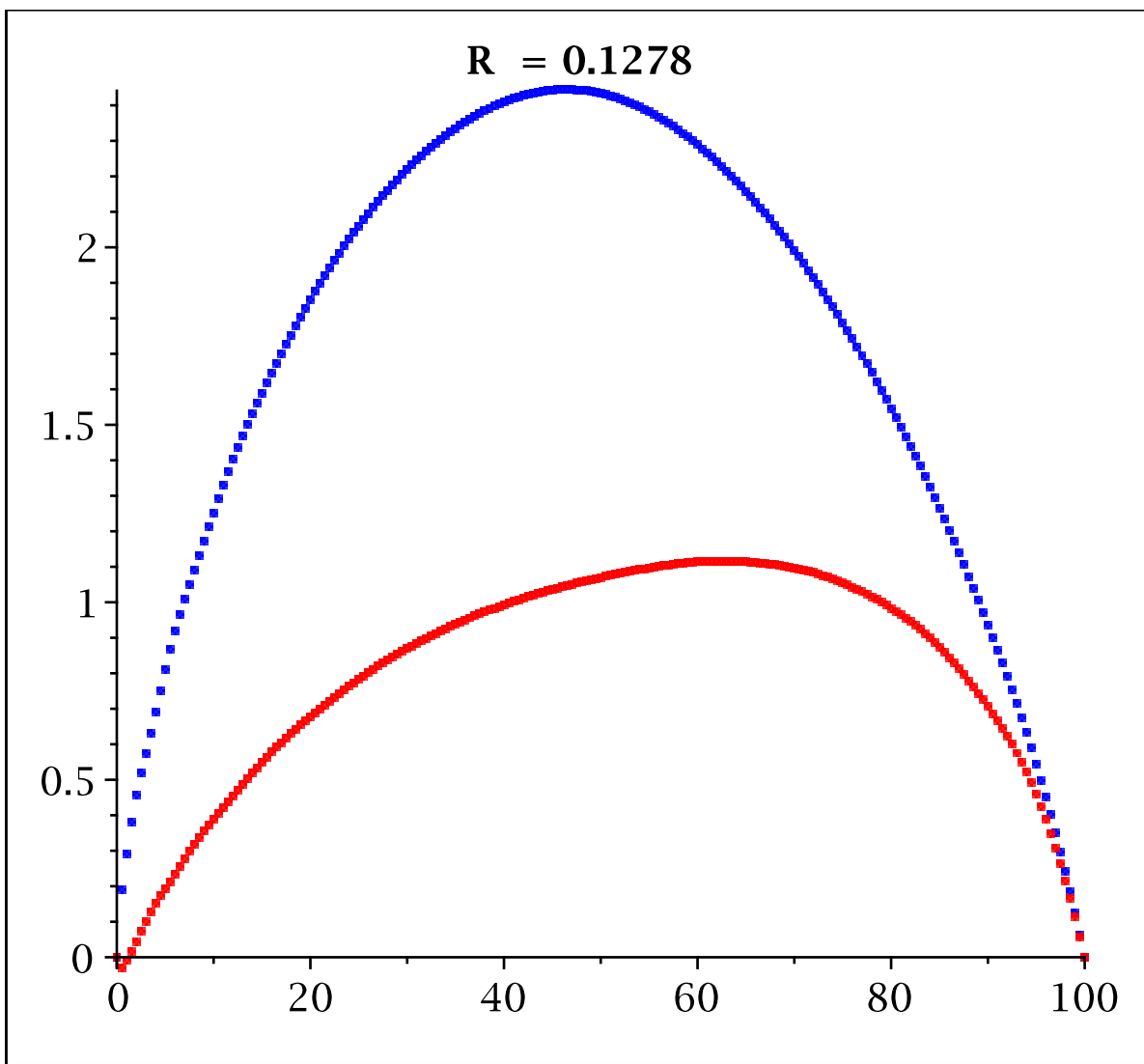


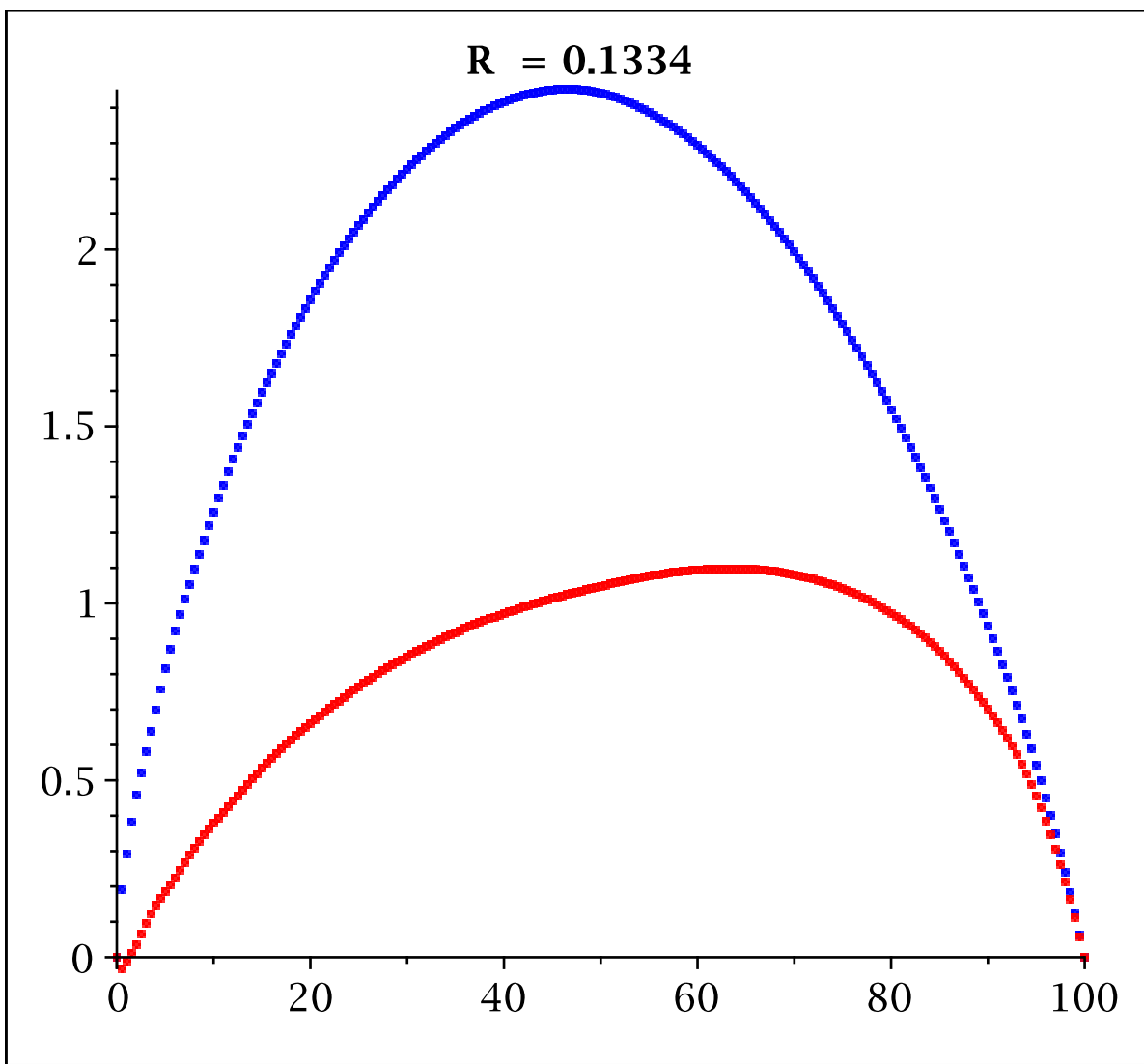


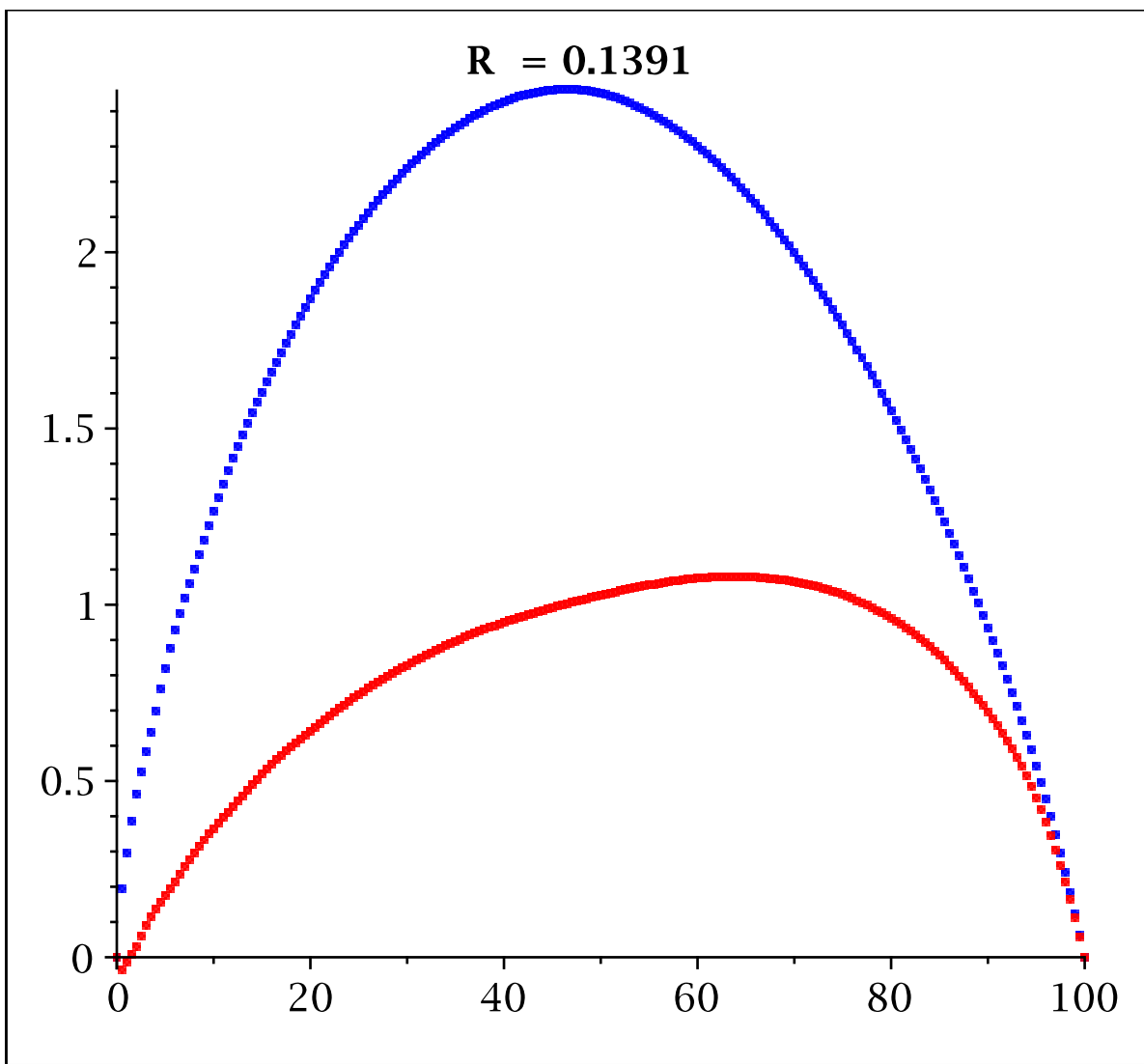


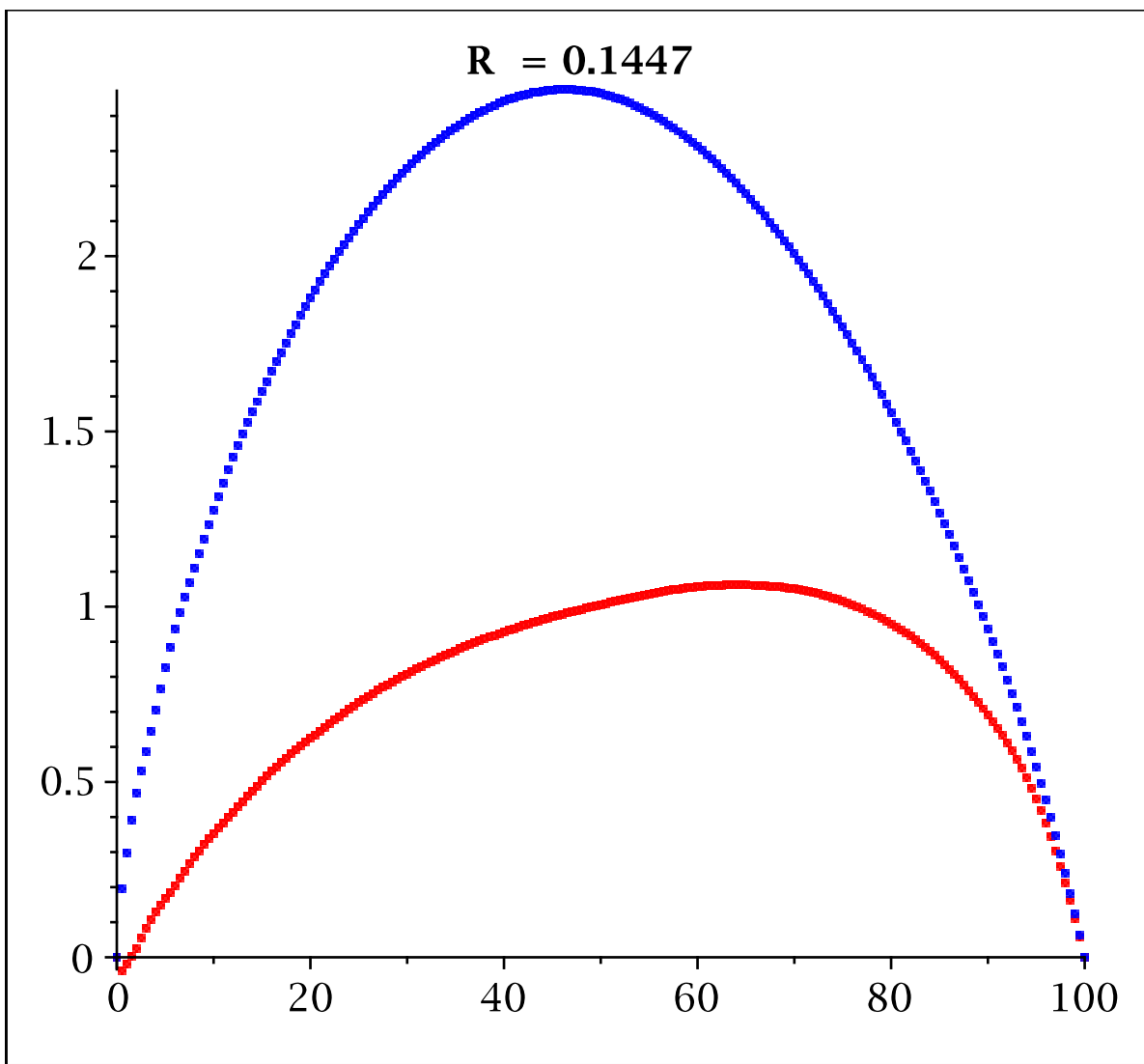


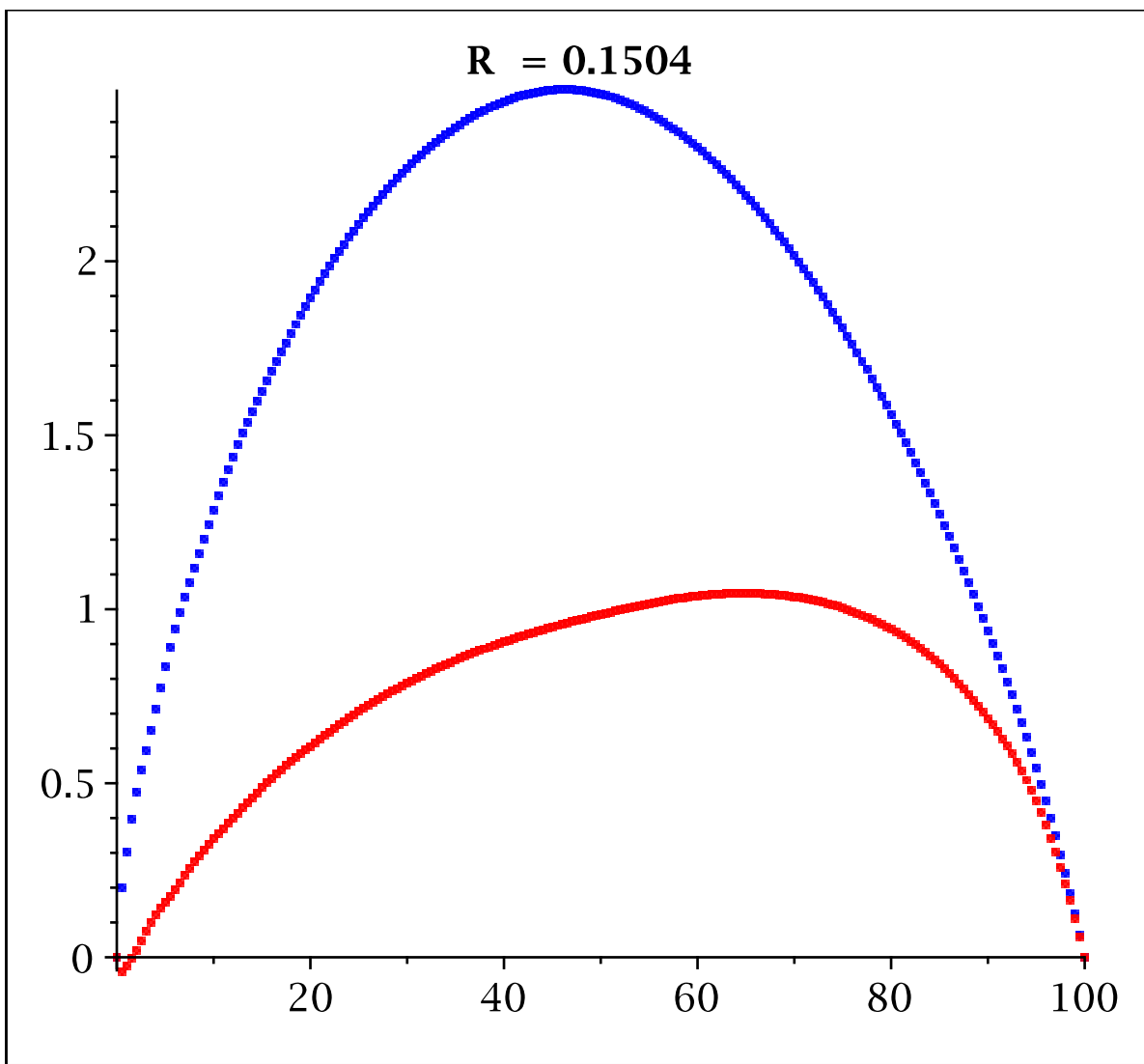




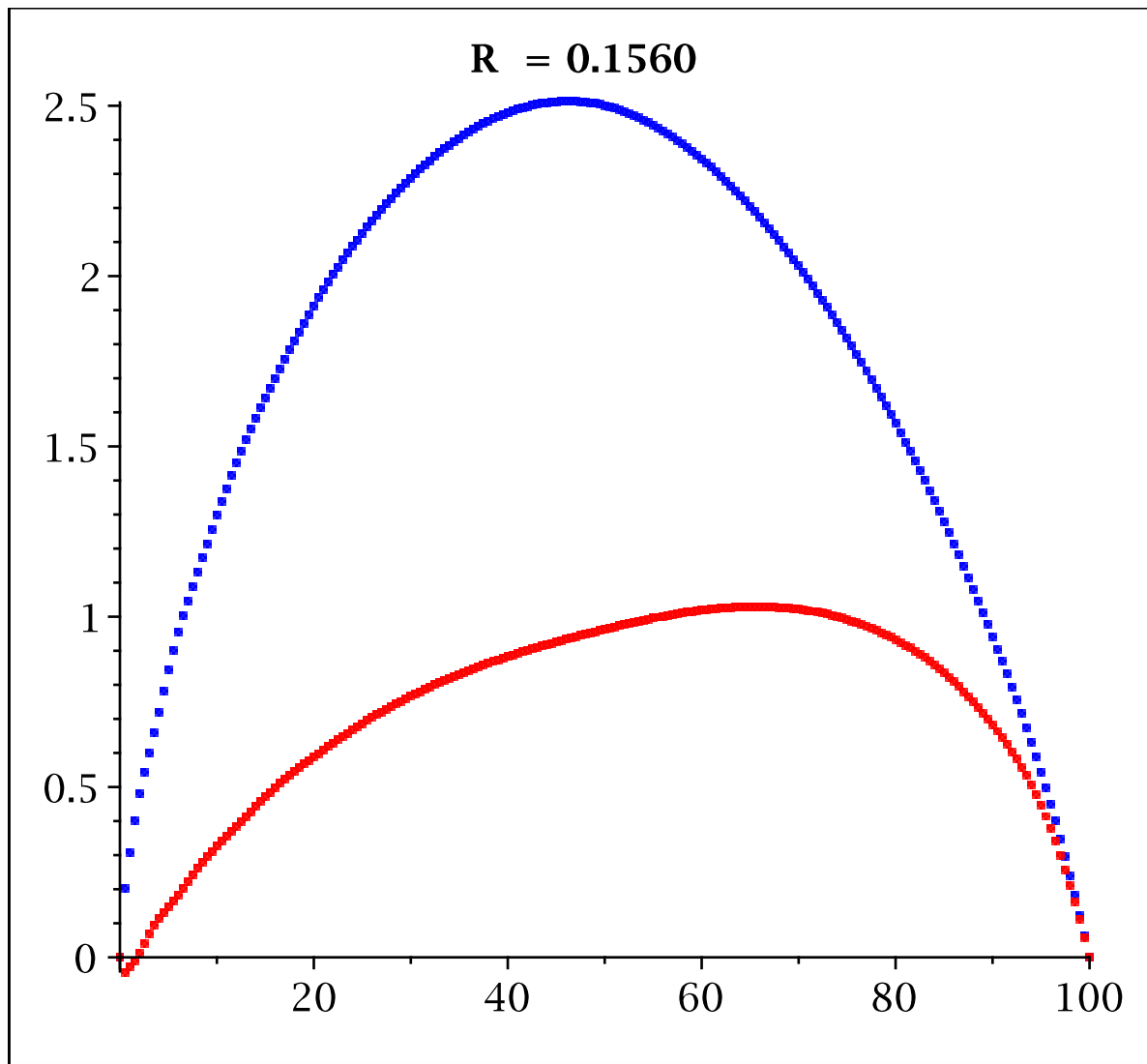












>

## ▼ Ecriture dans des fichiers:

>

```
> for k from 1 to Ncdo
  Strk := CaseJoin([convert(k, string), ".txt"]) :
  NomFich1 := CaseJoin(["ExtradosS", Strk]) :
  NomFich2 := CaseJoin(["IntradosS", Strk]) :

  Z[k] := Vector(1..Nps + 1, R[k]) : ZZ[k] := convert(Z[k], list) :

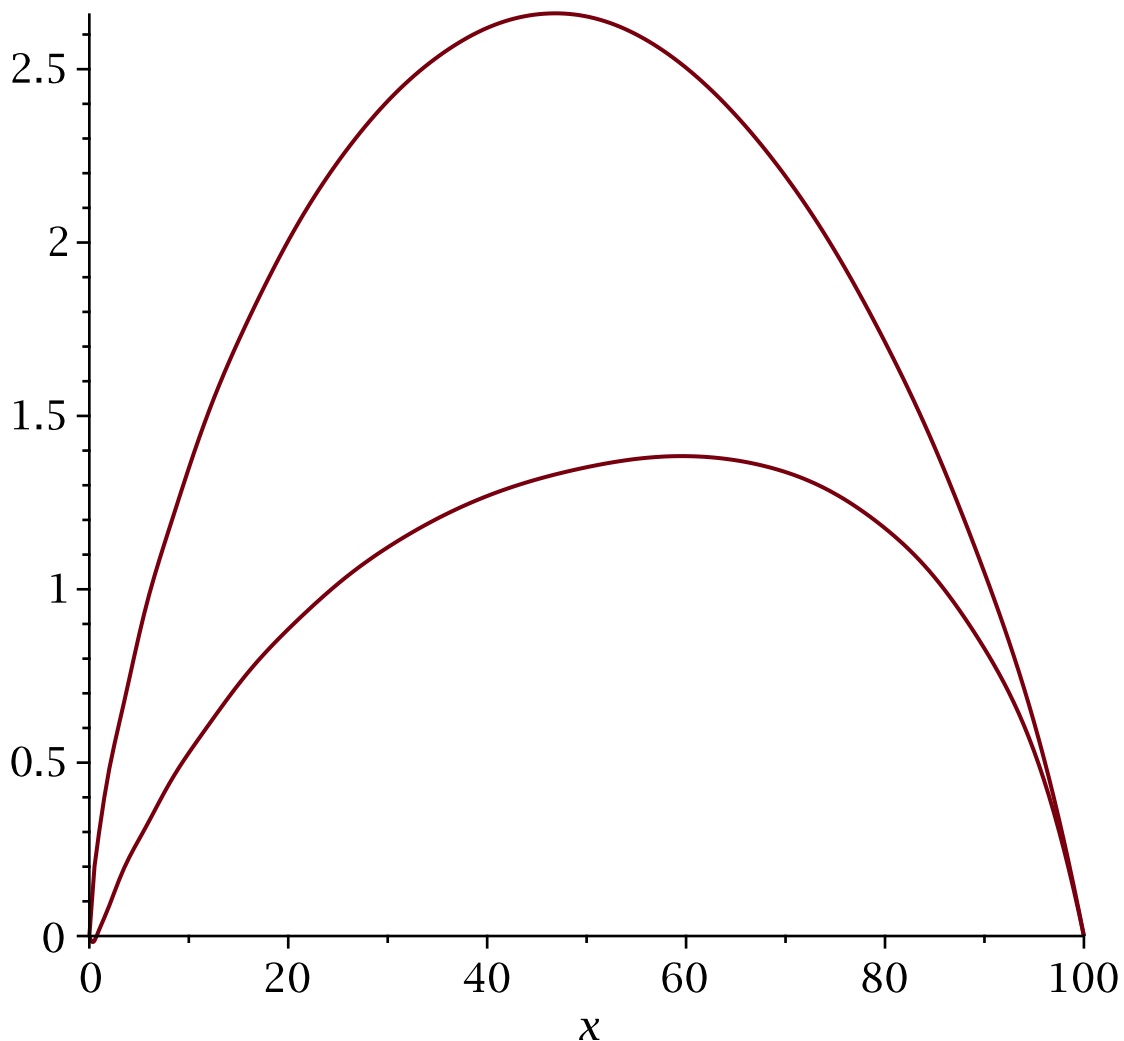
  Te := array(1..3, 1..Nps + 1, [XX, YY1[k], ZZ[k]]) :
  Ti := array(1..3, 1..Nps + 1, [XX, YY2[k], ZZ[k]]) :
  Me := convert(Te, Matrix) : MMe := convert(Transpose(Me), matrix) :
  Mi := convert(Ti, Matrix) : MMi := convert(Transpose(Mi), matrix) :

  writedata(NomFich1, MMe) :
```

```
writedata(NomFich2, MMi) :  
end do:
```

## ▼ Interpolation polynomiale:

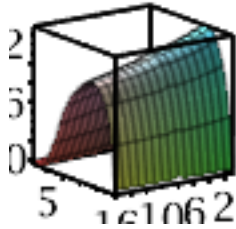
```
> for k from 1 to Nc do  
  Yip1[k] := Spline(X, Y1[1], x, degree = 5) :  
  Yip2[k] := Spline(X, Y2[1], x, degree = 5) :  
end do:  
  
>  
> Ip1 := plot(Yip1[1], x = 0..100) : Ip2 := plot(Yip2[1], x = 0..100) : display(Ip1,  
  Ip2)
```



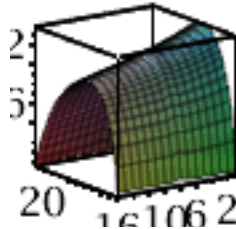
## ▼ Tracés 3D avant et après l'interpolation Spline puis après élimination de Nc=8

## Intrados

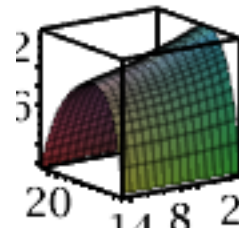
```
> Intra1 :=  
  listplot3d( [ seq(  
    Y2[k], k = 1  
    ..Nc) ] ) :  
  display(Intra1)
```



```
> Intra2 :=  
  listplot3d( [ seq(  
    YY2[k], k = 1  
    ..Nc) ] ) :  
  display(Intra2)
```



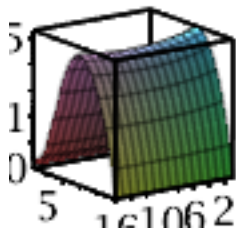
```
> IntCor :=  
  listplot3d( [ seq(  
    YY2[k], k = 1  
    ..7),  
    seq( YY2[k], k  
      = 9 ..Nc) ] ) :  
  display(IntCor)
```



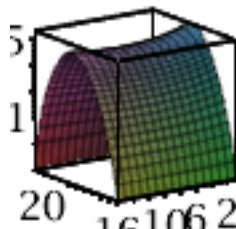
```
>
```

## Extrados

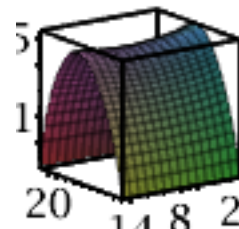
```
> Extra1 :=  
  listplot3d( [ seq(  
    Y1[k], k = 1  
    ..Nc) ] ) :  
  display(Extra1)
```



```
> Extra2 :=  
  listplot3d( [ seq(  
    YY1[k], k = 1  
    ..Nc) ] ) :  
  display(Extra2)
```



```
> ExtCor :=  
  listplot3d( [ seq(  
    YY1[k], k = 1  
    ..7),  
    seq( YY1[k], k  
      = 9 ..Nc) ] ) :  
  display(ExtCor)
```

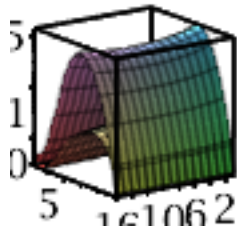


```
>
```

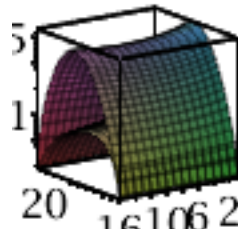
## Aube

```
>
```

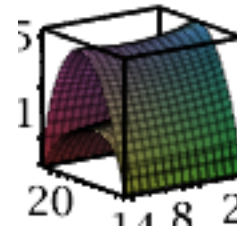
```
> display(Intra1,  
Extra1)
```



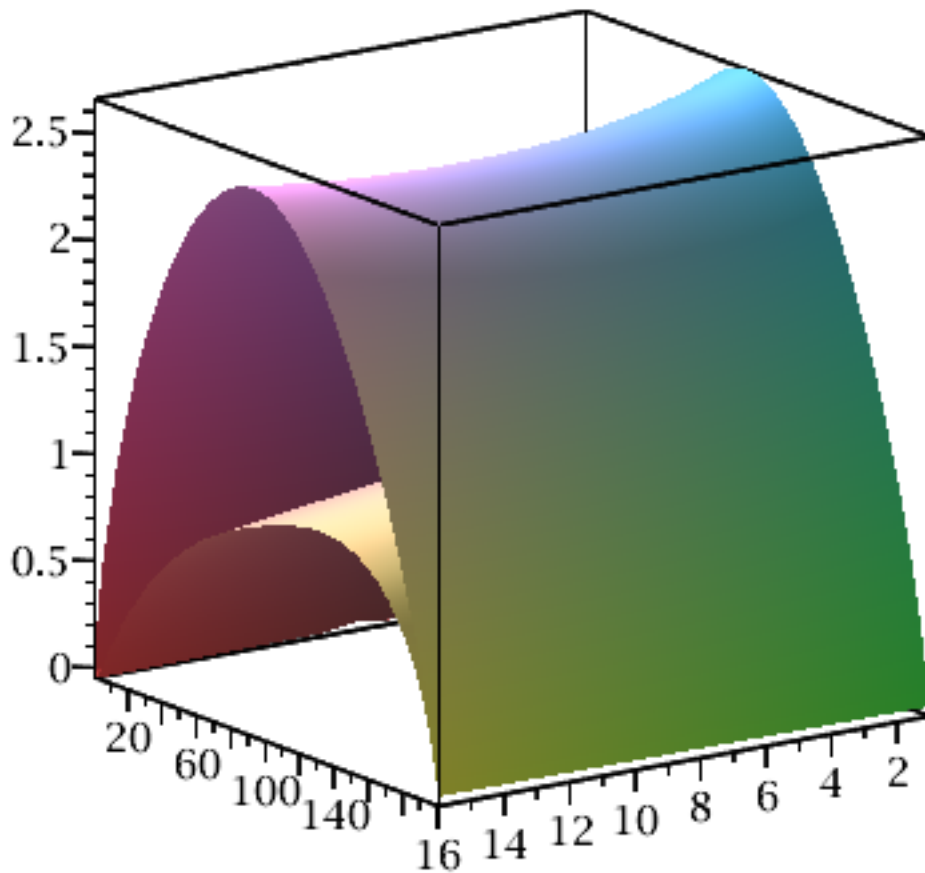
```
> display(Intra2,  
Extra2)
```



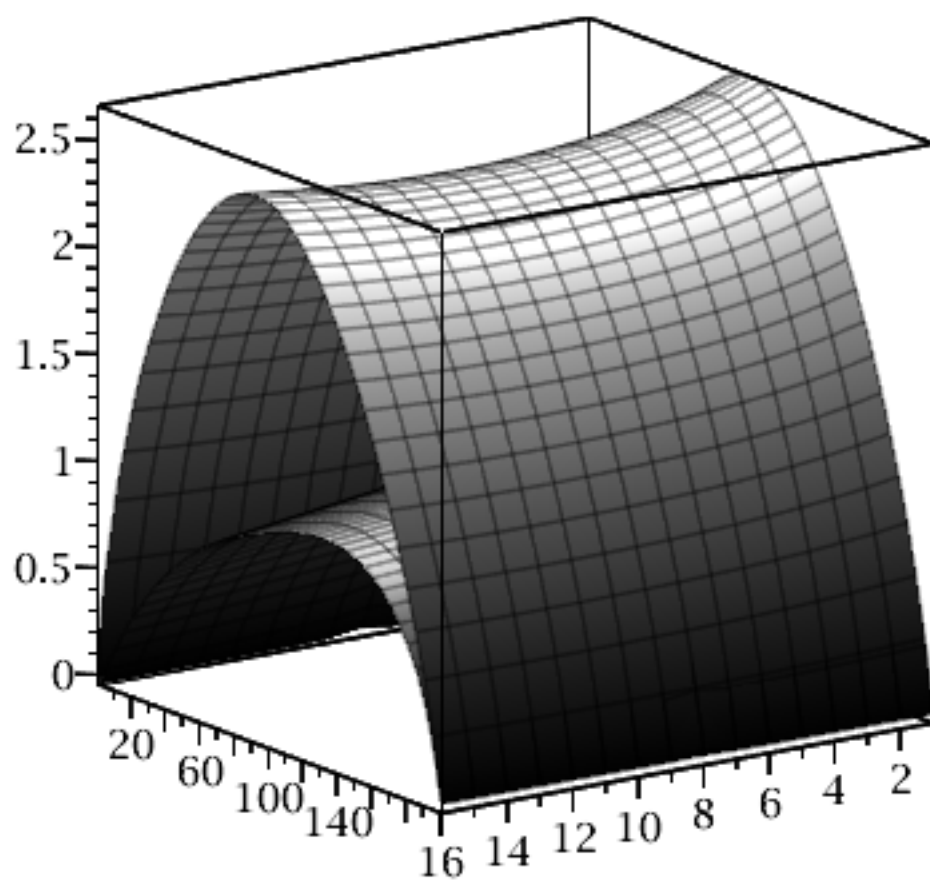
```
> display(IntCor,  
ExtCor)
```



```
> display(Intra2, Extra2, style = surface)
```



```
> display(Intra2, Extra2, shading = zgrayscale, lightmodel = light2)
```



>