Analiza numeryczna

Wyklad 9. Aproksymacja średniokwadratowa na zbiorze dyskretnym

Pawel Wozny

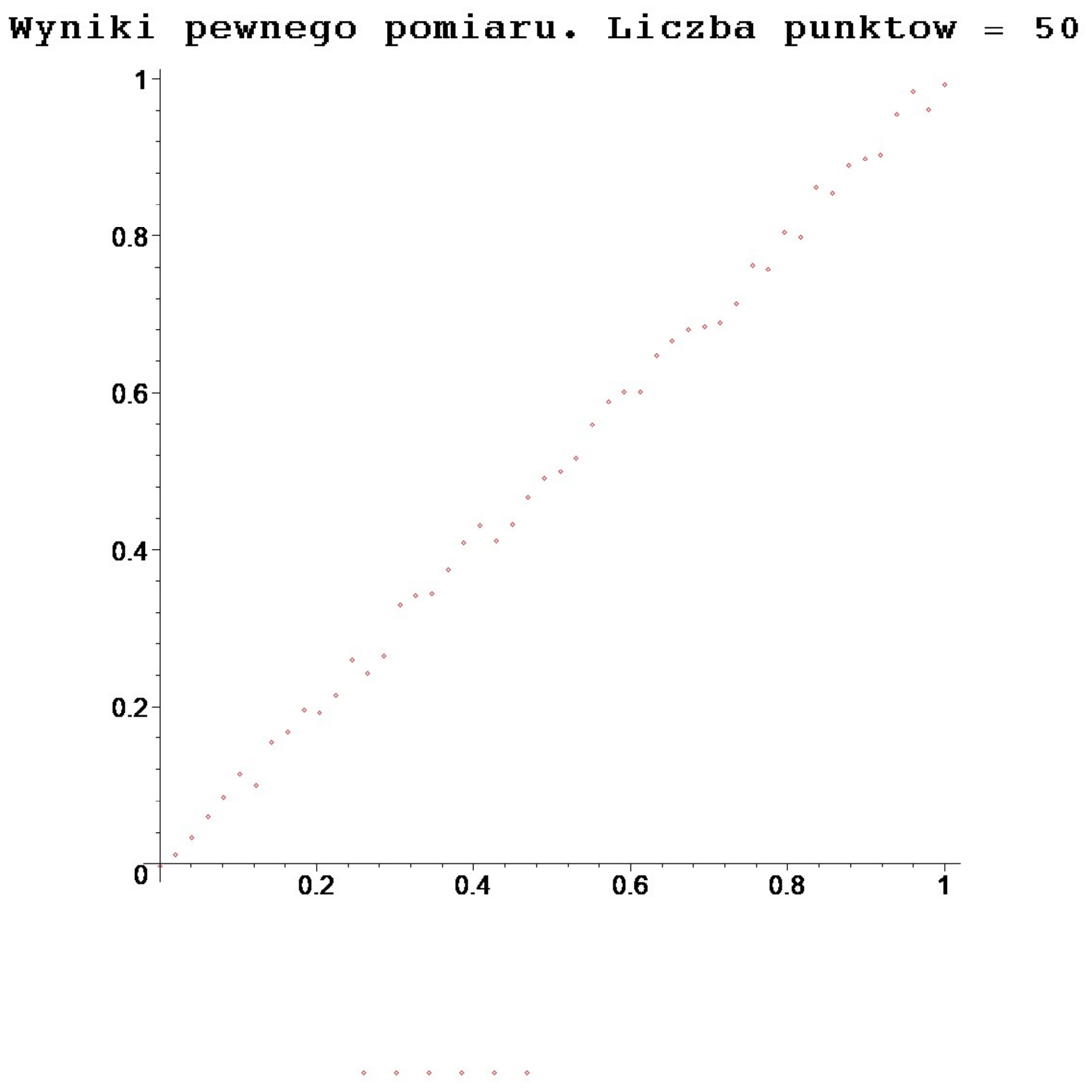
Wroclaw, 13 grudnia 2023 r.

Przykady

1. Idea aproksymacji

1.1. Wyniki pewnego pomiaru

```
>
> restart:
>
> Digits:=128:
>
>
> N:=49:
>
> wezly:=[seq(evalf(i/N),i=0..N)]:
>
> f_wezly:=[seq(i/N+stats[random,uniform[-0.025,0.025]](1),i=0..N)]:
>
> p:=[seq([wezly[i],f_wezly[i]],i=1..N+1)]:
>
> tytul:="Wyniki pewnego pomiaru. Liczba punktow = "||(N+1):
>
> pomiar:=plot(p,style=POINT,title=tytul,titlefont=[COURIER,BOLD,15],
             legend="Wyniki pomiaru",scaling=CONSTRAINED):
>
> plots[display](pomiar);
>
```



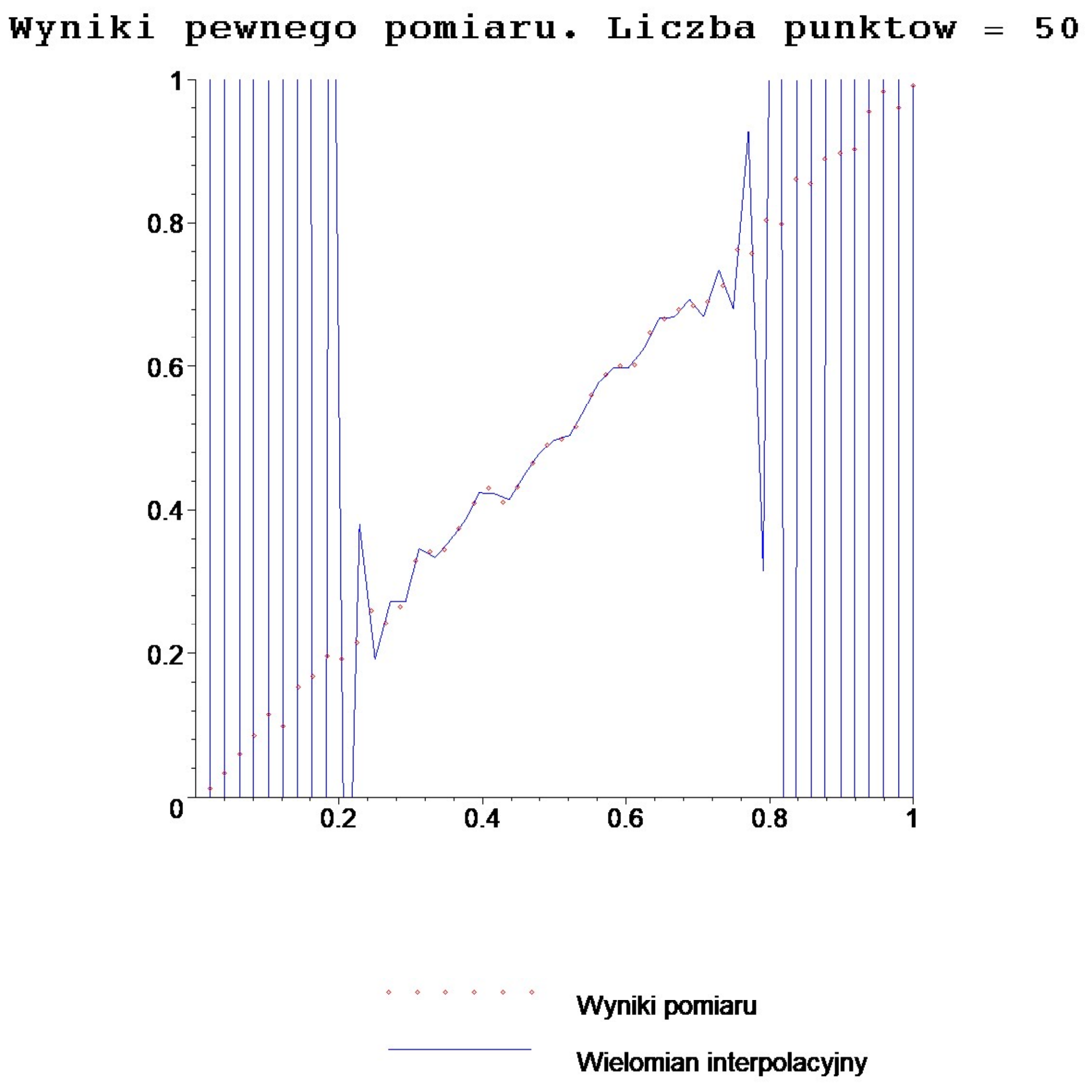
>

1.2. Interpolacja Lagrange'a

>

>

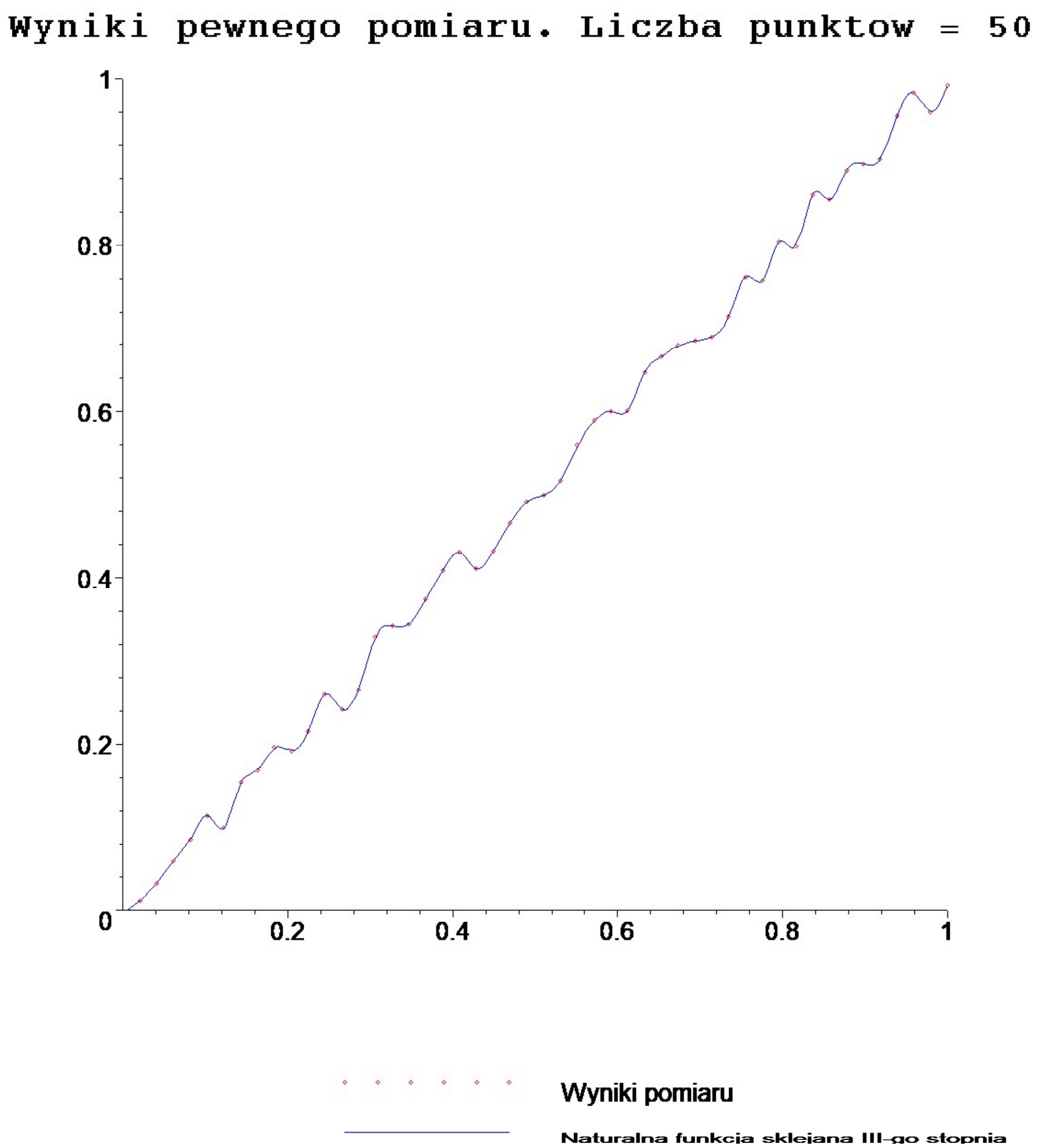
```
> L[N]:=unapply(interp(wezly,f_wezly,x),x):
>
> interpolacja:=plot(L[N](x),x=0..1,y=0..1,legend="Wielomian interpolacyjny",color=blue):
>
> plots[display](pomiar,interpolacja);
>
>
```




```
>
```

1.3. Naturalna funkcja sklejana III-go stopnia

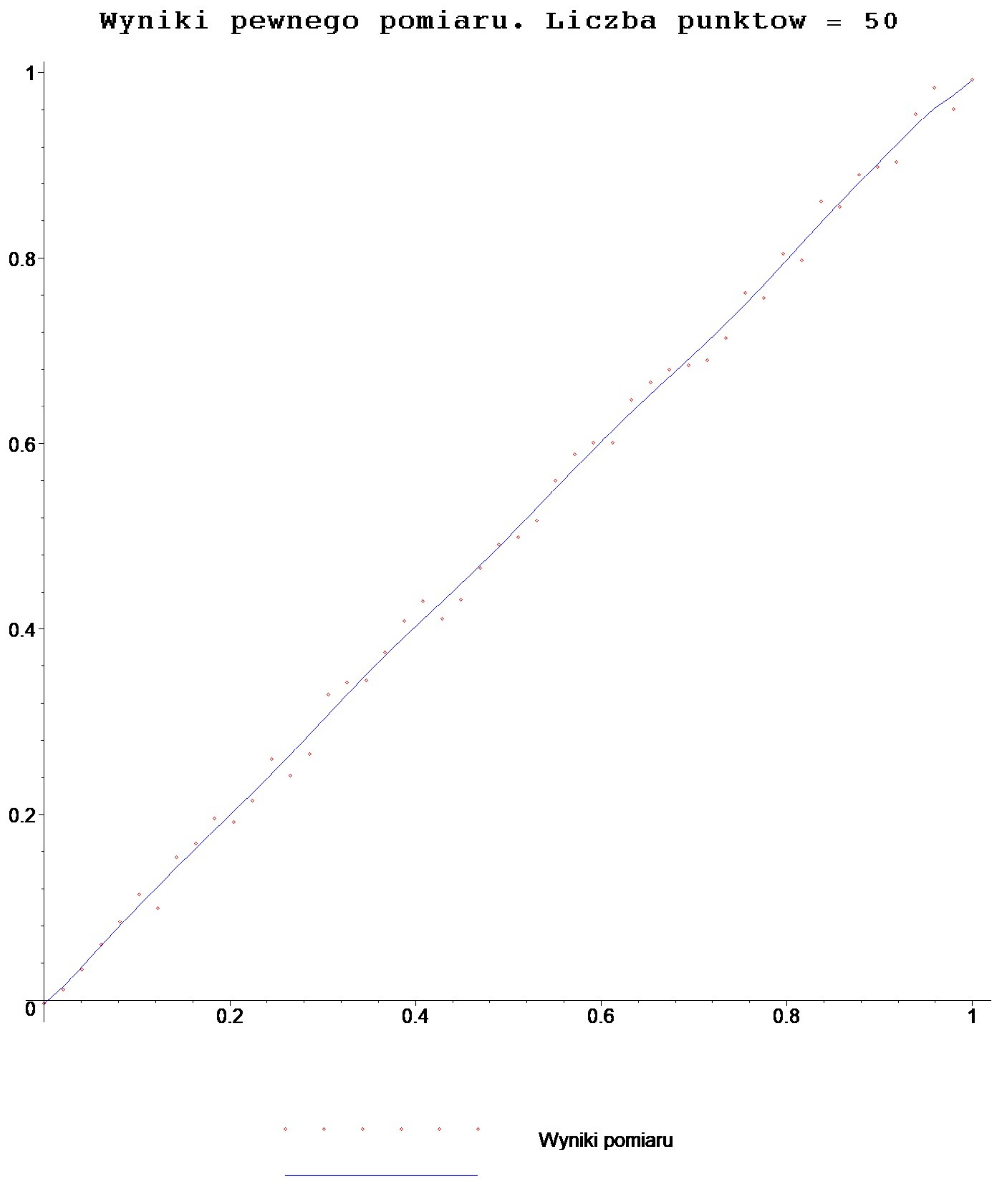
```
>
>
> s[N]:=unapply(spline(wezly,f_wezly,x),x):
>
> NFS3:=plot(s[N](x),x=0..1,y=0..1,
          legend="Naturalna funkcja sklejana III-go stopnia",color=blue):
>
> plots[display](pomiar,NFS3);
>
>
```




```
>
```

1.4 Krzywe Beziera

```
>
>
> P[N]:=unapply(add(f_wezly[i+1]*binomial(N,i)*x^i*(1-x)^(N-i),i=0..N),x):
>
> Bezier:=plot([x,P[N](x),x=0..1],
           legend="Krzywa Beziera stopnia 49",color=blue):
>
> plots[display](pomiar,Bezier);
>
>
```

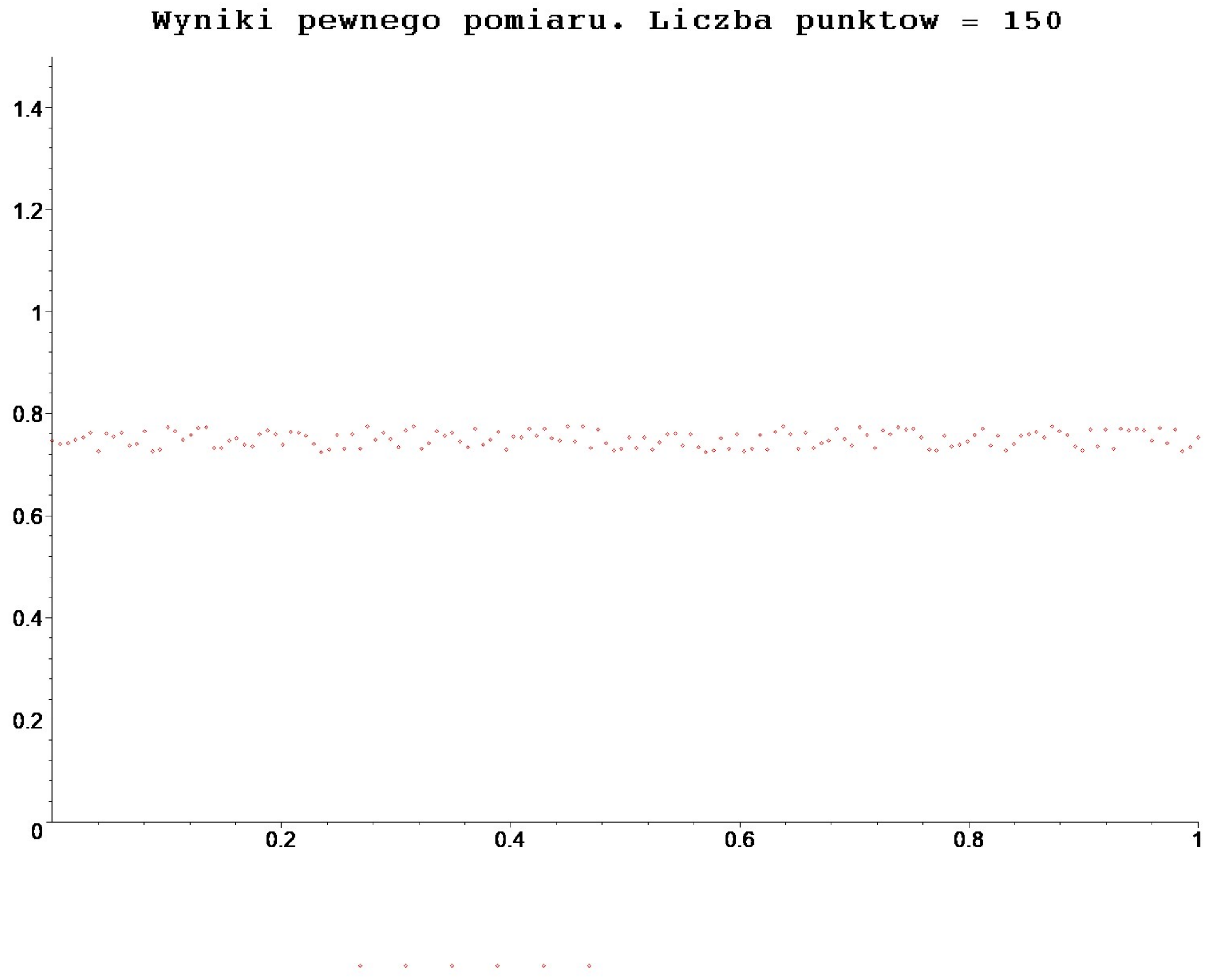


>

2. Przyklad: model w(x)=a

```
>
> restart:
>
> Digits:=32:
>
```

```
>
> N:=149:
>
> wezly:=[seq(evalf(i/N),i=0..N)]:
>
> f_wezly:=[seq(0.75+stats[random,uniform[-0.025,0.025]](1),i=0..N)]:
>
> p:=[seq([wezly[i],f_wezly[i]],i=1..N+1)]:
>
> tytul:="Wyniki pewnego pomiaru. Liczba punktow = "||(N+1):
>
> pomiar:=plot(p,style=POINT,title=tytul,titlefont=[COURIER,BOLD,15],
             legend="Wyniki pomiaru",scaling=UNCONSTRAINED,view=[0..1,0..1.5]):
>
> plots[display](pomiar);
>
```

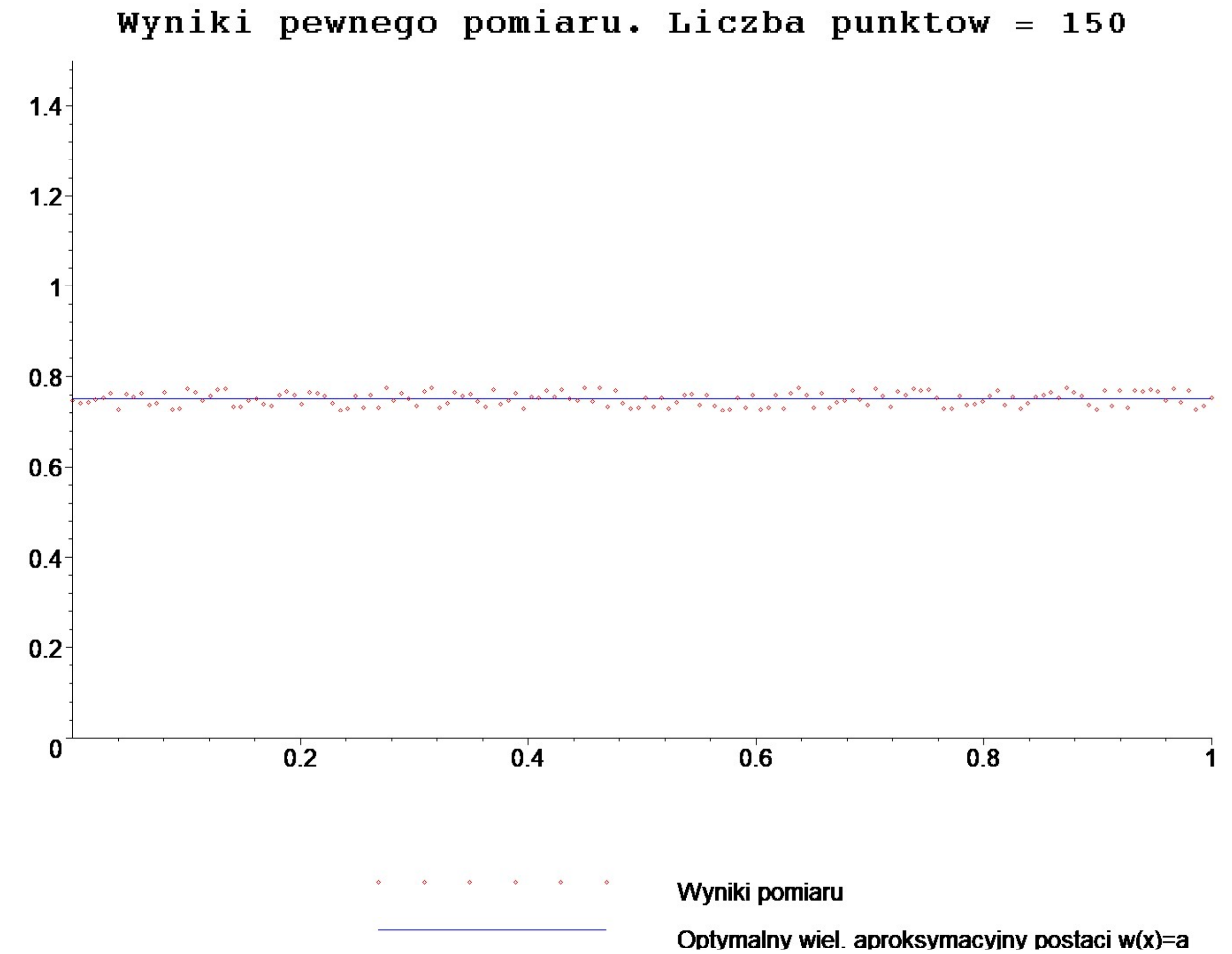



```
>
>
> w:=x->stats[describe,mean](f_wezly):
>
> printf("\n\n");
> 'w(x)'=w(x);
> Blad_sredniokwadratowy:=sqrt(sum((f_wezly[i]-w(wezly[i]))^2,i=1..N+1));
```

```
> printf("\n");
>
> aproksymacja:=plot(w(x),x=0..1,y=0..1.5,
                    legend="Optymalny wiel. aproksymacyjny postaci w(x)=a",color=blue):
>
> plots[display](pomiar,aproksymacja);
>
```

$w(x) = 0.7504377845186906666666666666667$

$Blad\_sredniokwadratowy := 0.1846296295235071810126774488657 4$

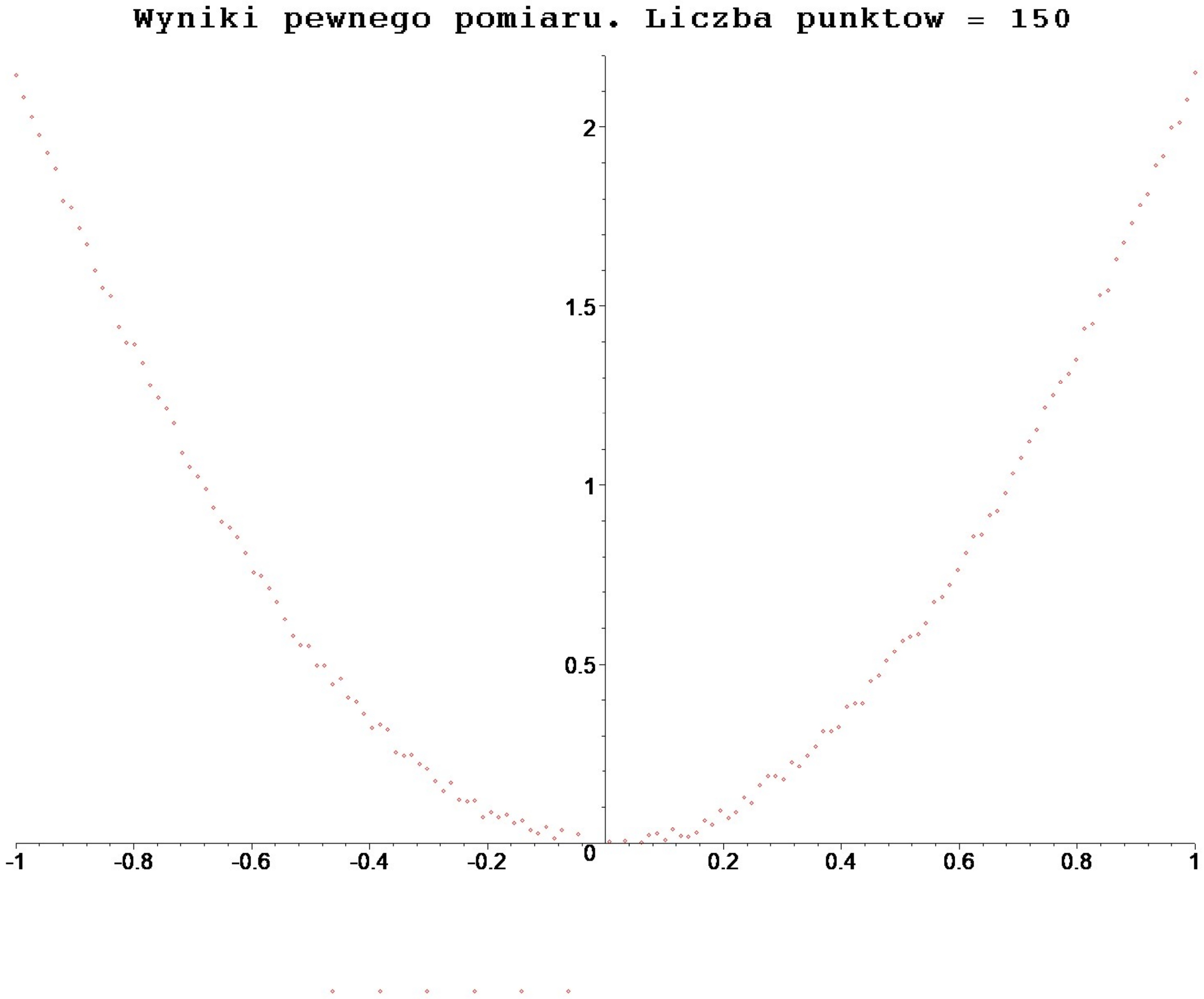



```
>
```

3. Przyklad: model w(x)=a x^2 i „odstające" obserwacje

```
>
> restart:
>
> Digits:=16:
>
>
> N:=149:
>
> wezly:=array(1..N+1,[seq(evalf(-1+2*i/N),i=0..N)]):
>
> f_wezly:=array(1..N+1,[seq(2.15*(-1+2*i/N)^2+stats[random,uniform[-0.025,0.025]](1),i=0..N)]):
```

```
>
> p:=[seq([wezly[i],f_wezly[i]],i=1..N+1)]:
>
> tytul:="Wyniki pewnego pomiaru. Liczba punktow = "||(N+1):
>
> pomiar:=plot(p,style=POINT,title=tytul,titlefont=[COURIER,BOLD,15],
             legend="Wyniki pomiaru",scaling=UNCONSTRAINED,view=[-1..1,0..2.2]):
>
> plots[display](pomiar);
>
```

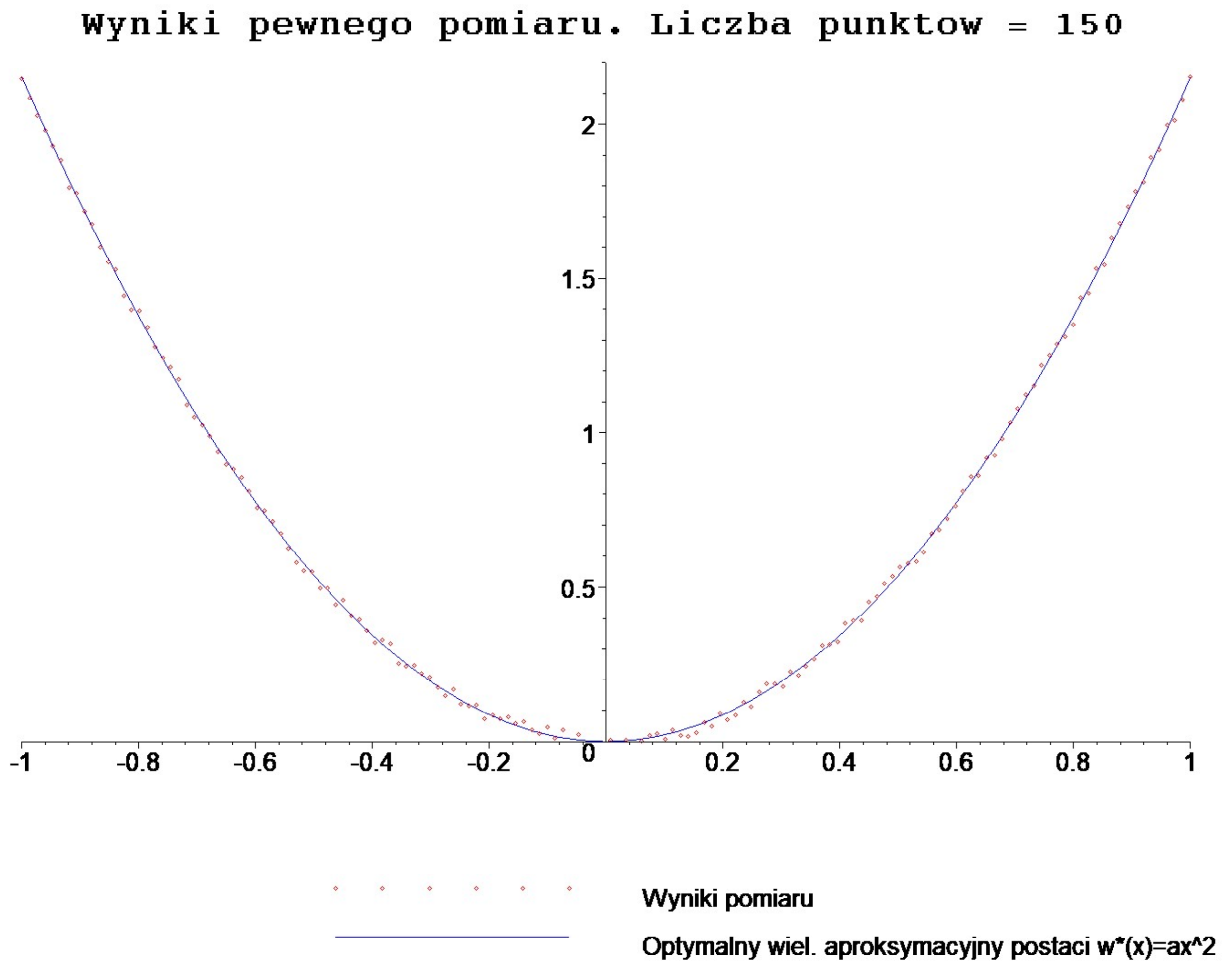



```
>
>
> w:=x->sum(f_wezly[i]*wezly[i]^2,i=1..N+1)/sum(wezly[i]^4,i=1..N+1)*x^2:
>
> printf("\n\n");
> 'w(x)'=w(x);
> Blad_sredniokwadratowy:=sqrt(sum((f_wezly[i]-w(wezly[i]))^2,i=1..N+1));
> printf("\n");
>
> aproksymacja:=plot(w(x),x=-1..1,y=0..2.2,
                   legend="Optymalny wiel. aproksymacyjny postaci w*(x)=ax^2",color=blue):
>
> plots[display](pomiar,aproksymacja);
```

>

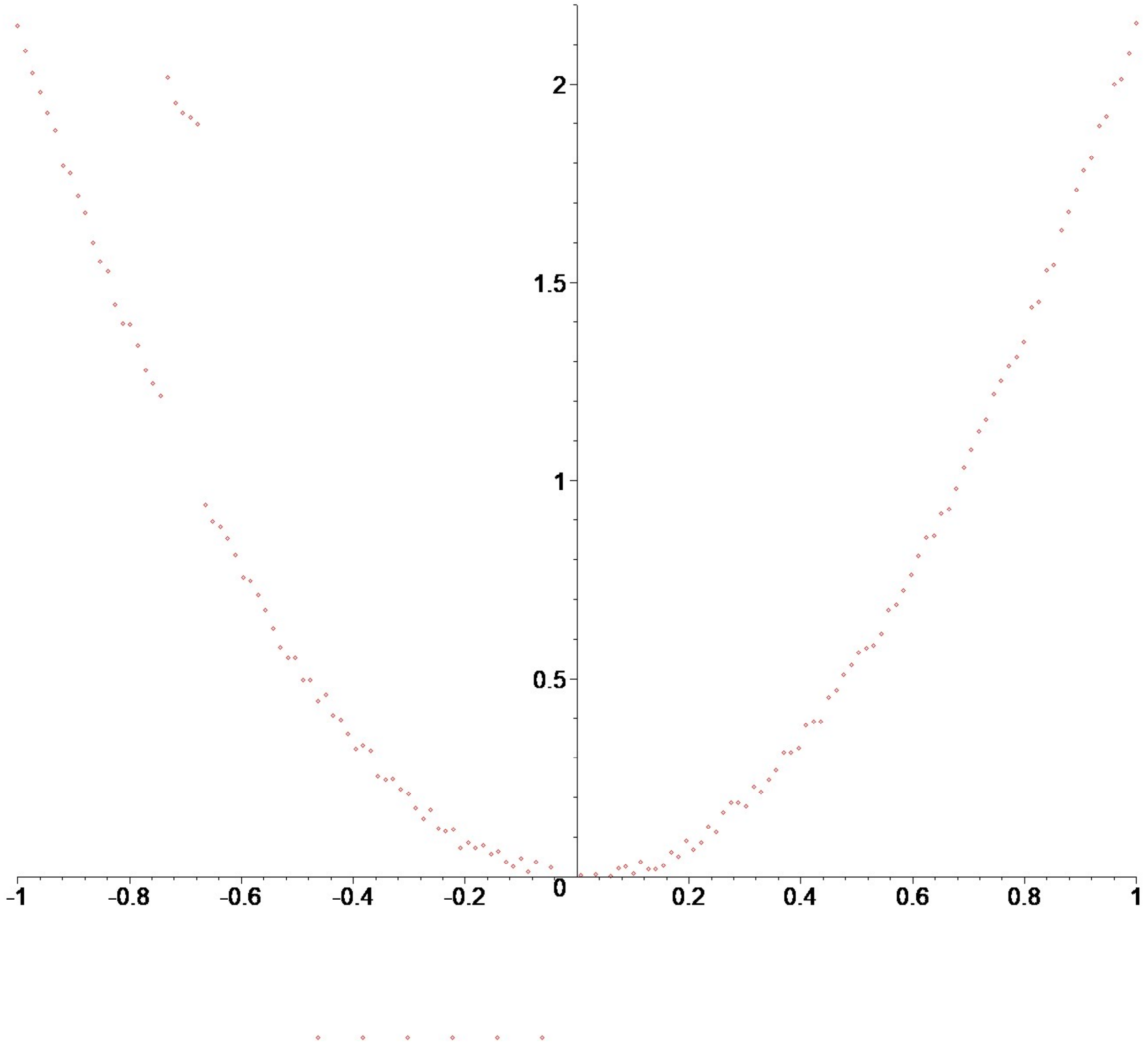
$w(x) = 2.151931131293775\ x^2$

$Blad\_sredniokwadratowy := 0.1843961864115166$




```
>
>
> for ii from 21 to 25
    do
      f_wezly[ii]:=f_wezly[ii]+(0.5+ii/61)
    od:
>
> p2:=[seq([wezly[i],f_wezly[i]],i=1..N+1)]:
>
> tytul2:="Wyniki pewnego pomiaru z ,,odstającymi'' obserwacjami. Liczba punktow = "||(N+1):
>
> pomiar2:=plot(p2,style=POINT,title=tytul2,titlefont=[COURIER,BOLD,15],
             legend="Wyniki pomiaru",scaling=UNCONSTRAINED,view=[-1..1,0..2.2]):
>
> plots[display](pomiar2);
>
>
```

Wyniki pewnego pomiaru z ,,odstającymi'' obserwacjami. Liczba p

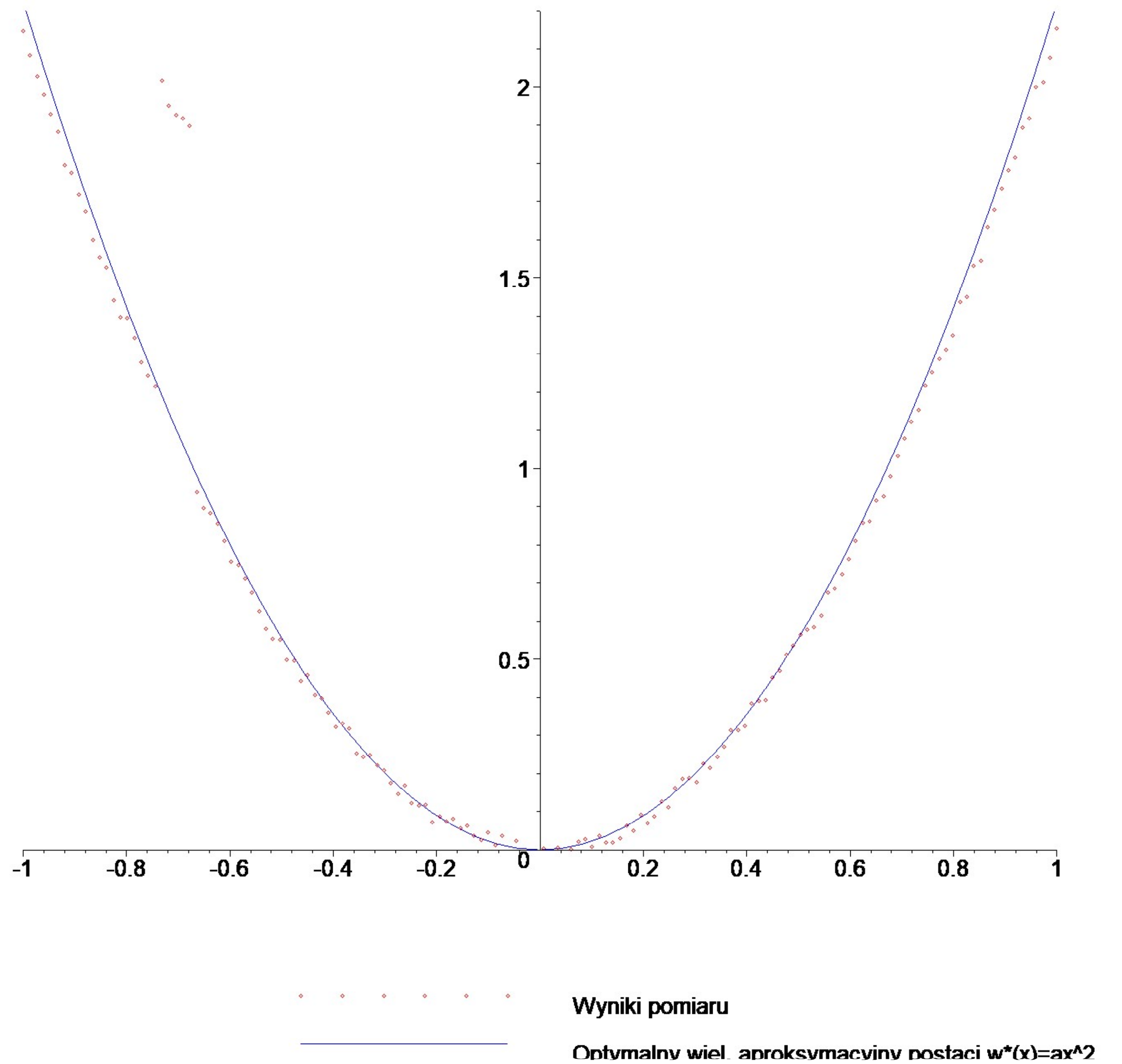



```
>
>
> w2:=x->sum(f_wezly[i]*wezly[i]^2,i=1..N+1)/sum(wezly[i]^4,i=1..N+1)*x^2:
>
> printf("\n\n");
> 'w(x)'=w(x);
> Blad_sredniokwadratowy:=sqrt(sum((f_wezly[i]-w2(wezly[i]))^2,i=1..N+1));
> printf("\n");
>
> aproksymacja2:=plot(w2(x),x=-1..1,y=0..2.2,
                    legend="Optymalny wiel. aproksymacyjny postaci w*(x)=ax^2",color=blue):
>
> plots[display](pomiar2,aproksymacja2);
>
>
```

$$w(x) = 2.222566087212106\, x^2$$

$$Blad\_sredniokwadratowy := 1.922004148062266$$

Wyniki pewnego pomiaru z ,,odstającymi'' obserwacjami. Liczba pu

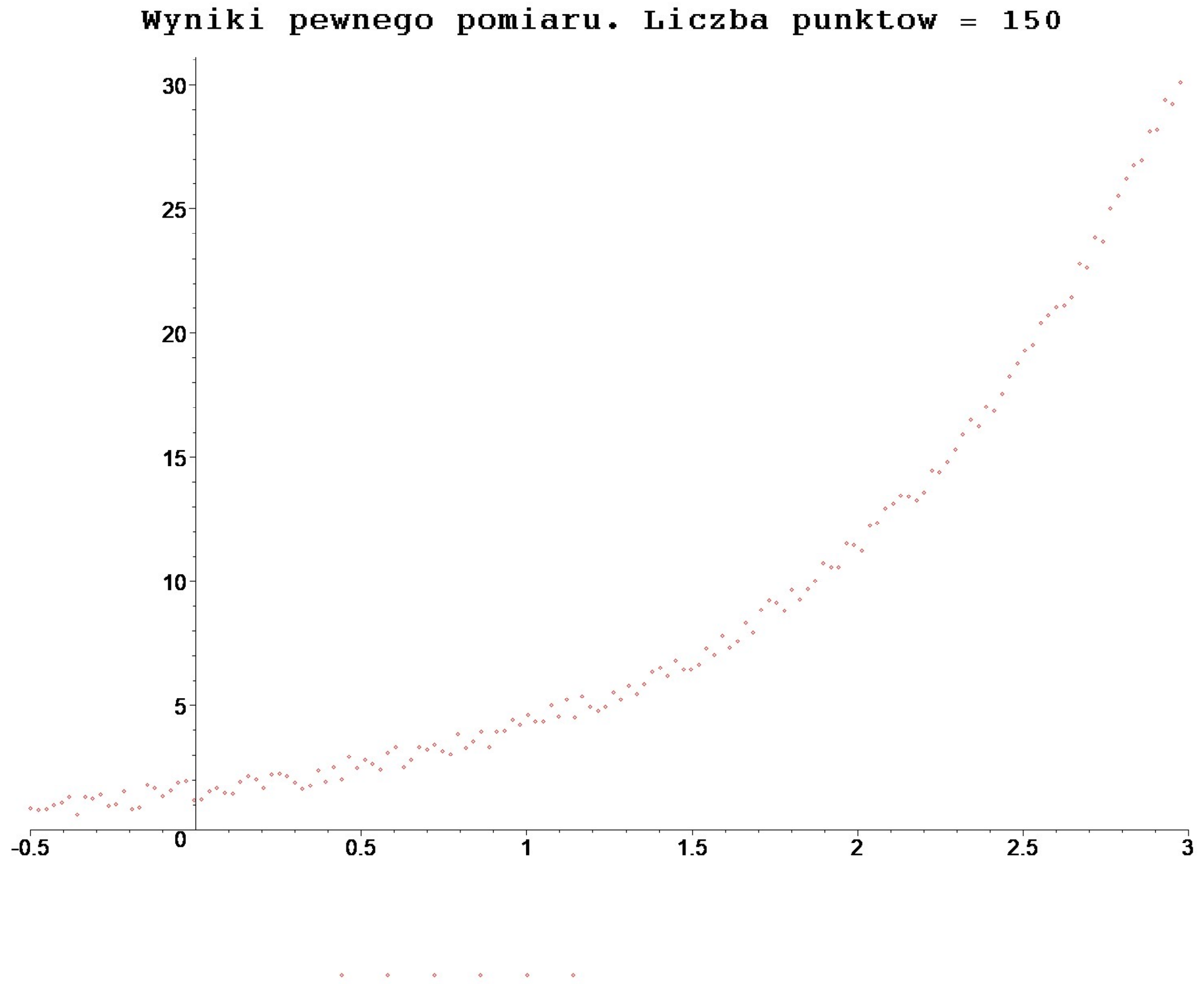


>

4. Przyklad: model w(x)=a exp(x)

```
>
> restart:
>
> Digits:=16:
>
>
> N:=149:
>
> wezly:=[seq(evalf(-0.5+3.5*i/N),i=0..N)]:
>
> f_wezly:=[seq(1.55*exp(-0.5+3.5*i/N)+stats[random,uniform[-0.5,0.5]](1),i=0..N)]:
>
> p:=[seq([wezly[i],f_wezly[i]],i=1..N+1)]:
>
> tytul:="Wyniki pewnego pomiaru. Liczba punktow = "||(N+1):
```

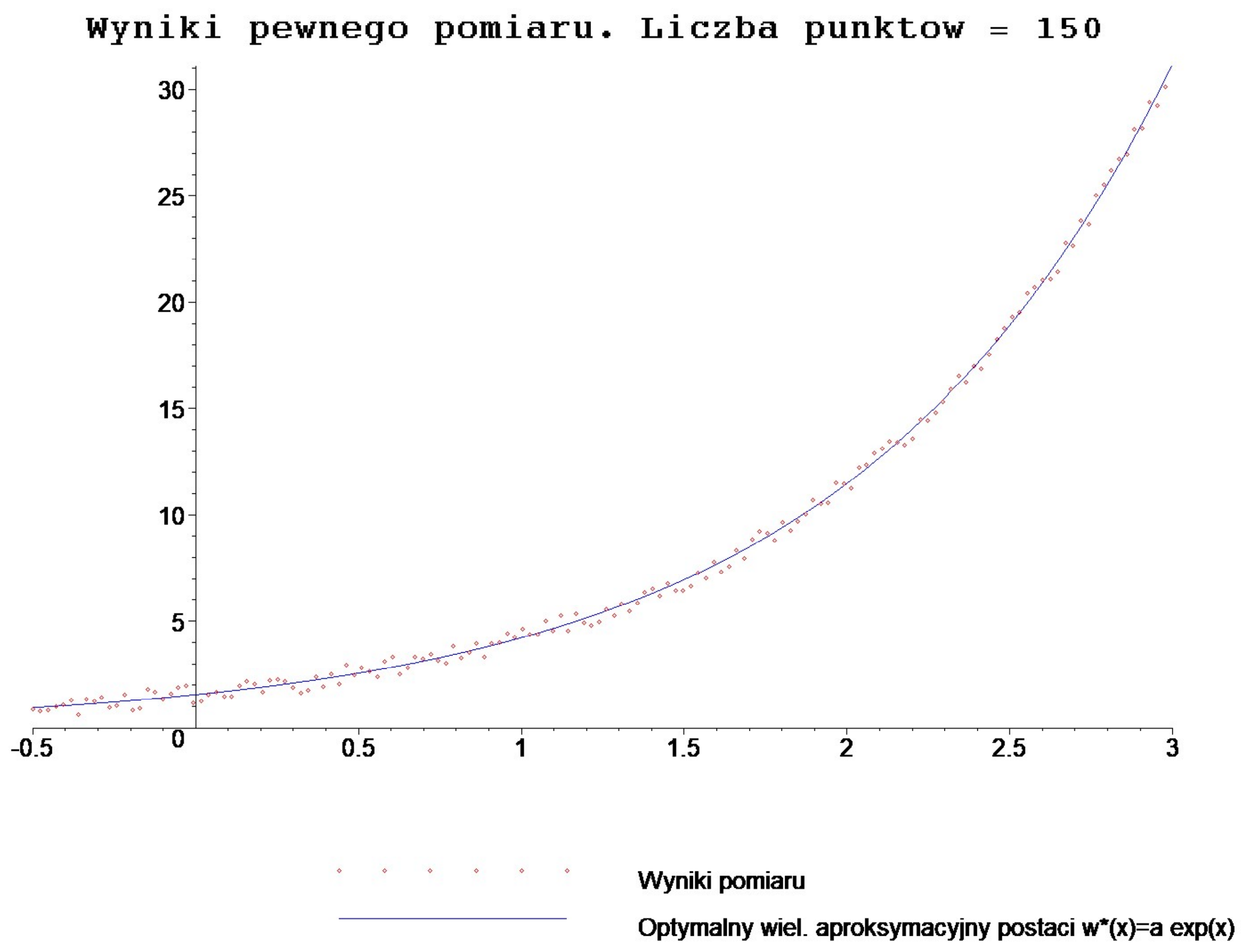
```
>
> pomiar:=plot(p,style=POINT,title=tytul,titlefont=[COURIER,BOLD,15],
             legend="Wyniki pomiaru",scaling=UNCONSTRAINED,view=[-0.5..3,0..1.55*exp(3)]):
>
> plots[display](pomiar);
>
```




```
>
>
> w:=x->sum(f_wezly[i]*exp(wezly[i]),i=1..N+1)/sum(exp(2*wezly[i]),i=1..N+1)*exp(x):
>
> printf("\n\n");
> 'w(x)'=w(x);
> Blad_sredniokwadratowy:=sqrt(sum((f_wezly[i]-w(wezly[i]))^2,i=1..N+1));
> printf("\n");
>
> aproksymacja:=plot(w(x),x=-0.5..3,y=0..1.55*exp(3),
              legend="Optymalny wiel. aproksymacyjny postaci w*(x)=a exp(x)",color=blue):
>
> plots[display](pomiar,aproksymacja);
>
```

$$w(x) = 1.552839299269584\ e^{x}$$

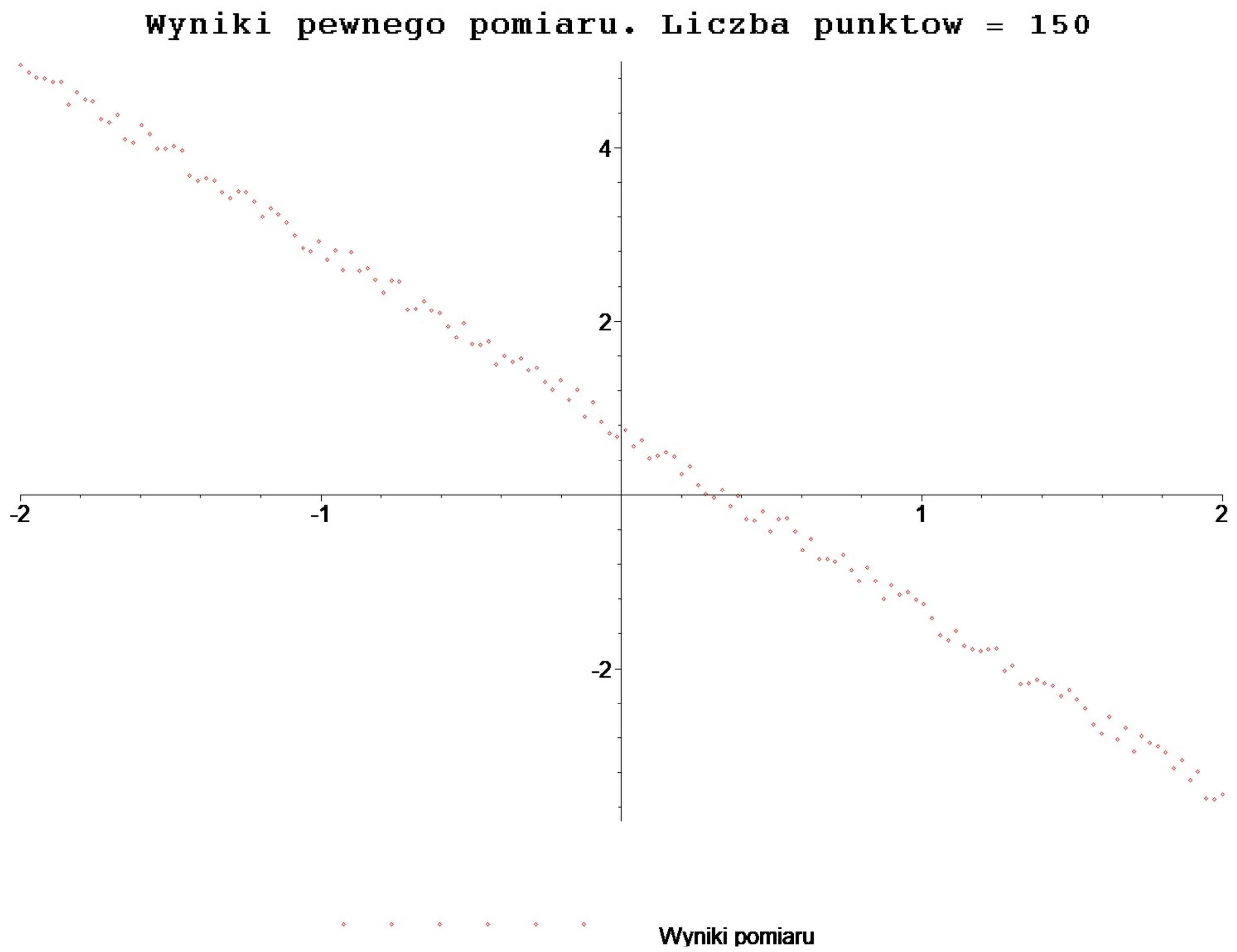
$$Blad\_sredniokwadratowy := 3.684553544773371$$




```
>
```

5. Przyklad: model w(x)=a x + b

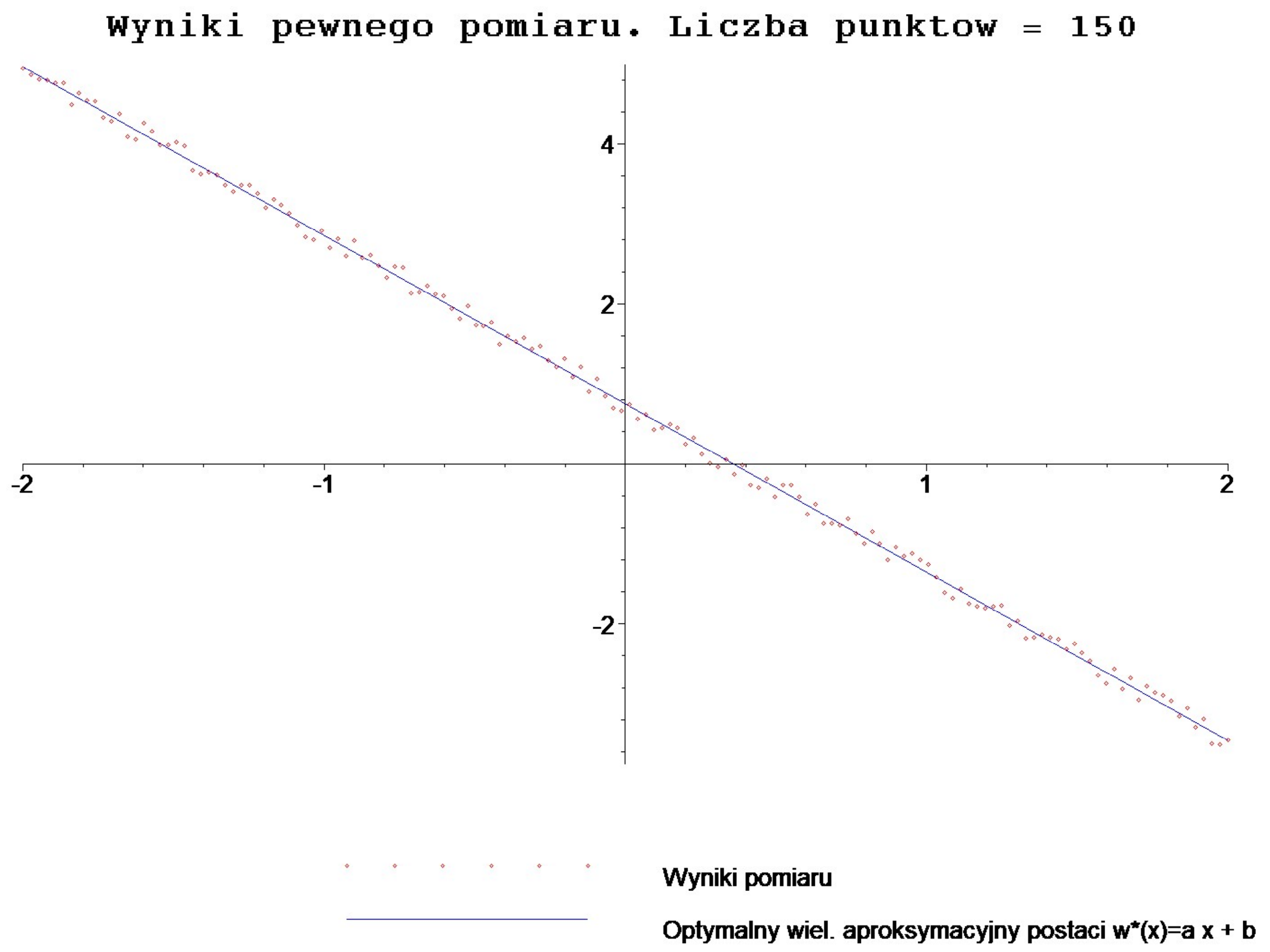
```
>
> restart:
>
> Digits:=16:
>
>
> N:=149:
>
> wezly:=[seq(evalf(-2+4*i/N),i=0..N)]:
>
> f_wezly:=[seq(-2.11*(-2+4*i/N)+0.75+stats[random,uniform[-0.15,0.15]](1),i=0..N)]:
>
> p:=[seq([wezly[i],f_wezly[i]],i=1..N+1)]:
>
> tytul:="Wyniki pewnego pomiaru. Liczba punktow = "||(N+1):
>
> pomiar:=plot(p,style=POINT,title=tytul,titlefont=[COURIER,BOLD,15],
             legend="Wyniki pomiaru",scaling=UNCONSTRAINED,view=[-2..2,-3.75..5]):
>
> plots[display](pomiar);
>
```




```
>
>
> s[1]:=sum(wezly[k],k=1..N+1):
> s[2]:=sum(wezly[k]^2,k=1..N+1):
> s[3]:=sum(f_wezly[k],k=1..N+1):
> s[4]:=sum(wezly[k]*f_wezly[k],k=1..N+1):
>
> a:=((N+1)*s[4]-s[1]*s[3])/((N+1)*s[2]-s[1]^2):
> b:=(s[2]*s[3]-s[1]*s[4])/((N+1)*s[2]-s[1]^2):
>
> w:=x->a*x+b:
>
> printf("\n\n");
> 'w(x)'=w(x);
> Blad_sredniokwadratowy:=sqrt(sum((f_wezly[i]-w(wezly[i]))^2,i=1..N+1));
> printf("\n");
>
> aproksymacja:=plot(w(x),x=-2..2,y=-3.75..5,
               legend="Optymalny wiel. aproksymacyjny postaci w*(x)=a x + b",color=blue):
>
> plots[display](pomiar,aproksymacja);
>
```

$$\mathrm{w}(x) = -2.105780895473499\, x + 0.7526267071121448$$

$$Blad\_sredniokwadratowy := 1.106148112789793$$


Wyniki pewnego pomiaru. Liczba punktow = 150
4
2
-2
-1
1
2
Wyniki pomiaru
Optymalny wiel. aproksymacyjny postaci w*(x)=a x + b

>
>