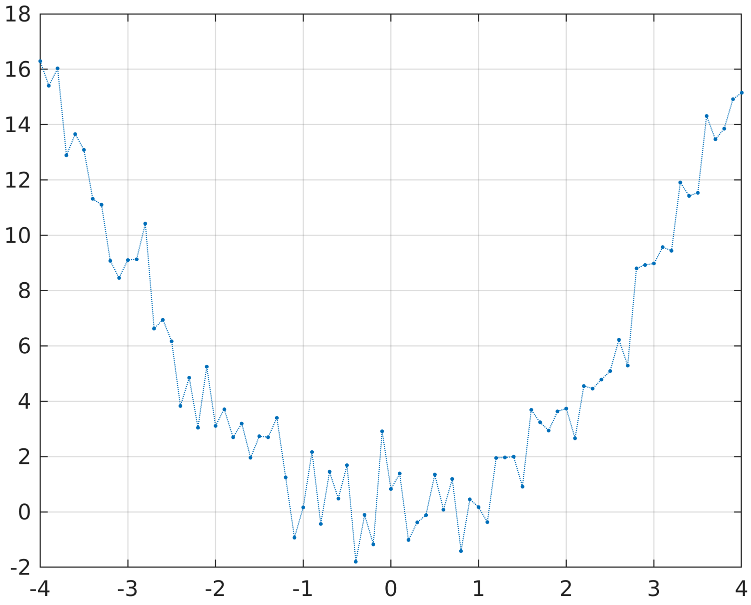
Problem 2: Polyfit with noisy data

HA 4.12.2018 File: Exercise2_polyfit.m

- Evaluate $y = x^2$ for `x=-4:0.1:4`.
- Add random noise to these samples. Use `randn`.
- Plot the noisy signal with (.)- markers
- Fit a 2^{nd} degree polynomial to the noisy data
- Plot the fitted polynomial on the same plot, using the same x values and a red curve.

```
close all
```

```
x=-4:.1:4;
y=x.^2;
noise=randn(size(y)); % mean:0, std:1
ynoisy=y+noise;
plot(x,ynoisy,'.:');grid on
```

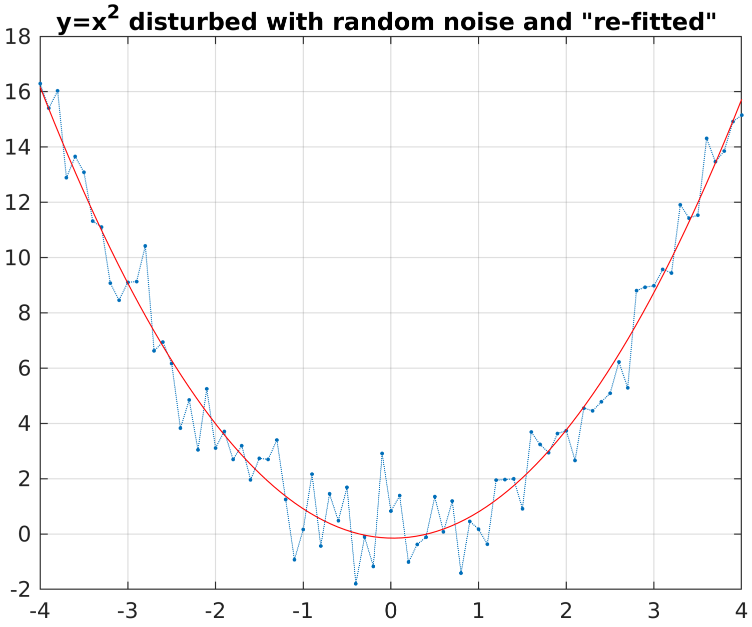


Fit polynomial:

```
c=polyfit(x,ynoisy,2)
```

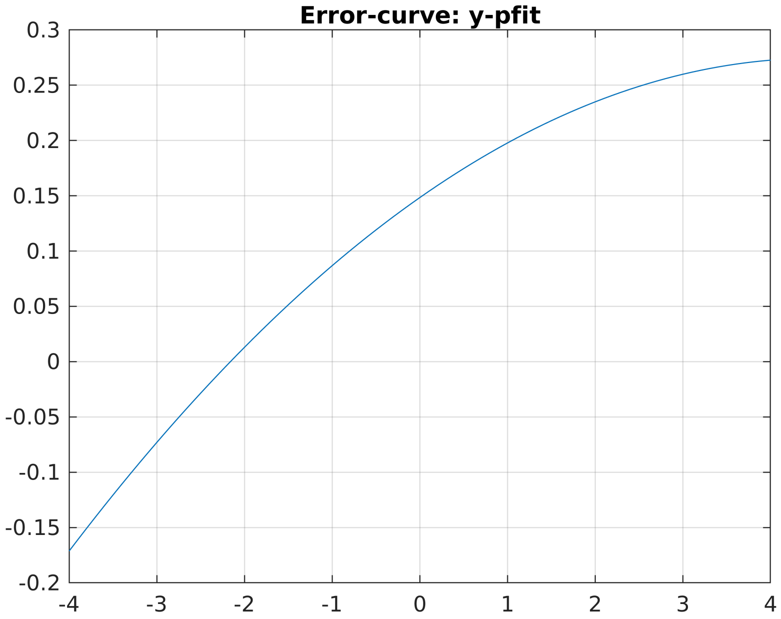
```
c = 1×3
    1.0061   -0.0555   -0.1484
```

```
xev=linspace(-4,4); % Points of evaluation
% In this case (Nr. of datapoints >> deg. of polynomial)
% xev can be taken the same as x (everybody did).
% Especially in case of interpolation (small data) this would
% only show the datapoints connected with line segments.
pval=polyval(c,xev);
hold on
plot(xev,pval,'r')
title('y=x^2 disturbed with random noise and "re-fitted" ')
```



Error: (Not required)

```
figure
plot(x,y-polyval(c,x));grid on;
title('Error-curve: y-pfit')
```



```
shg
maxerr=max(abs(y-polyval(c,x)))
```

```
maxerr = 0.2727
```

Note that different runs produce different error-curves.