- Bill Knowlton MSE 310-ECE 340

## Using LinearFitModel[ ] and NonlinearFitModel commands to fit data

- **`LinearModelFit`** returns a symbolic **`FittedModel`** object to represent the linear model it constructs. The properties and diagnostics of the model can be obtained from *model*`["`*property*`"]`.

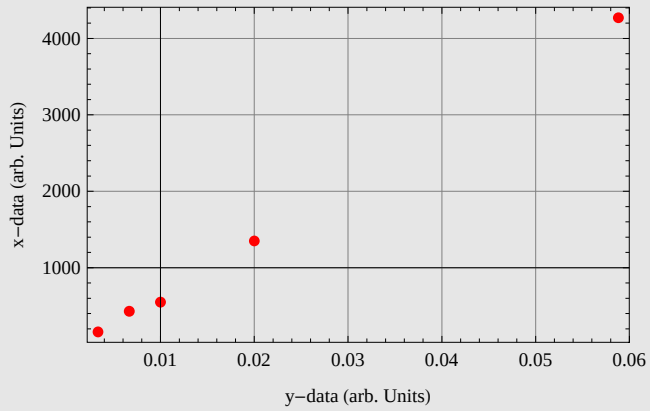
  **`NonlinearModelFit`** returns a symbolic **`FittedModel`** object to represent the linear model it constructs. The properties and diagnostics of the model can be obtained from *model*`["`*property*`"]`.

  Using this approach to fit data allows one to use the execute additional commands

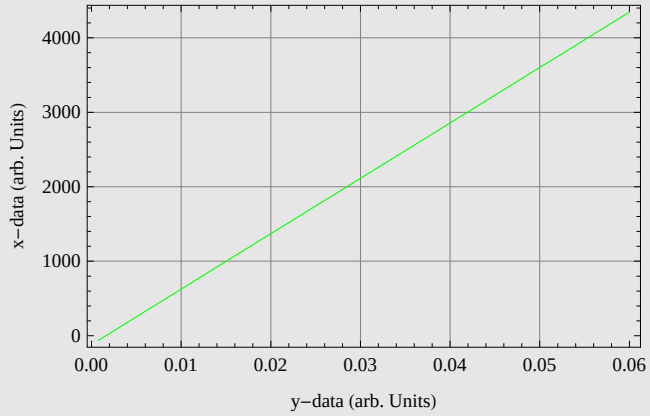
- ### Here are some Examples

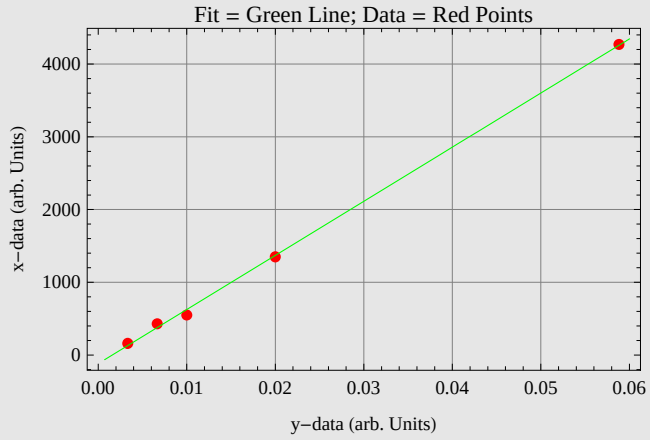
```
Clear[DifData, DifDataplot, DifDatafit, plotDifFit, diffit]
(*Place data in list*)

DifData =
 {{1 / 17, 4270}, {1/50, 1350}, {1/100, 550}, {1/150, 430}, {1/300, 160}}

(*Create scatter plot of data*)
DifDataplot = ListPlot[DifData, Frame → True, GridLines → Automatic,
  PlotStyle → {RGBColor[1, 0, 0], PointSize[0.02`]},
  FrameLabel → {"y-data (arb. Units)", "x-data (arb. Units)"}]

(*Perform fit and define the fitting function*)
Print["y = ", DifDatafit = Fit[DifData, {1, x}, x]]

plotDifFit = Plot[DifDatafit, {x, 0.00075, .06}, Frame → True,
  GridLines → Automatic, PlotStyle → {RGBColor[0, 1, 0]},
  FrameLabel → {"y-data (arb. Units)", "x-data (arb. Units)"}]
(*showing the plots pplot and plotpfit on the same graph*)
Show[DifDataplot, plotDifFit,
 PlotLabel → "Fit = Green Line; Data = Red Points"]

(* Use LinearModelFit command that is new to Mathematica 7*)
diffit = LinearModelFit[DifData, x, x]
Print["R^2 = ", diffit["RSquared"]] (* Provides the R^2 value *)
Print["Adjusted R^2 = ", diffit["AdjustedRSquared"]]
(* Provides the adjusted R^2 value *)
Print["Percent Error = ", (-24246.5 - (-24137))/(-24246.5) 100, " %"]
(* Provides the percent error *)
```

```
{{1/17, 4270}, {1/50, 1350}, {1/100, 550}, {1/150, 430}, {1/300, 160}}
```




$y = -118.625 + 74\,406.6\,x$




FittedModel[ $-118.625 + 74\,406.6\,x$ ]

$R^2 = 0.999133$

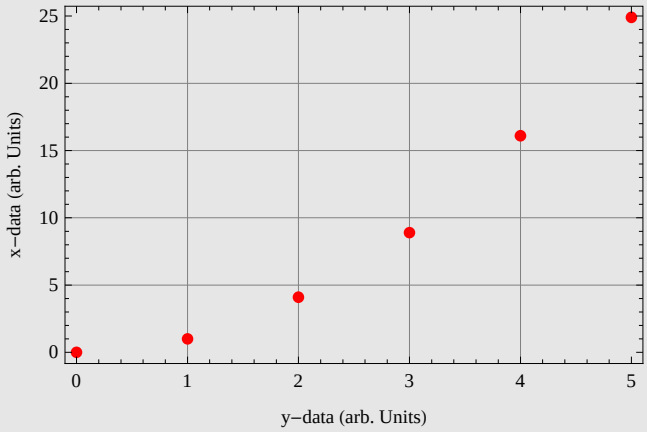
Adjusted $R^2 = 0.998844$

Percent Error = 0.451612 %

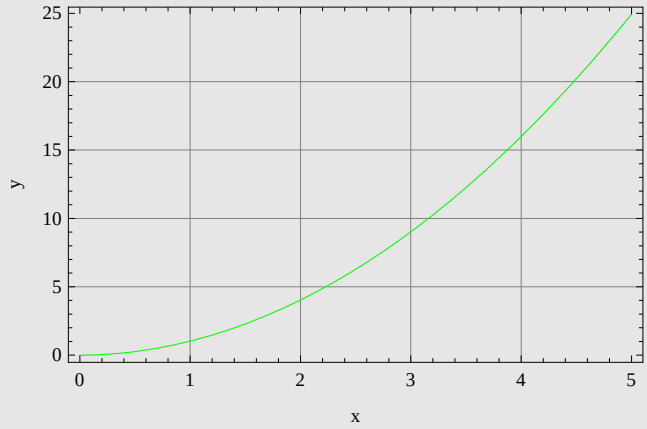
For Non-linear curve fitting and output of $R^2$ , one can use the NonlinearModelFit[ ] function.

```
Clear[SomeData, SomeDataplot, SomeDatafit, plotFit]
(*Place data in list*)

SomeData = {{0, 0}, {1, 1}, {2, 4.1}, {3, 8.9}, {4, 16.1}, {5, 24.9}}

(*Create scatter plot of data*)
SomeDataplot = ListPlot[SomeData, Frame → True, GridLines → Automatic,
  PlotStyle → {RGBColor[1, 0, 0], PointSize[0.02`]},
  FrameLabel → {"y-data (arb. Units)", "x-data (arb. Units)"}]

(*Perform fit and define the fitting function*)
Print["y = ", SomeDatafit =
  NonlinearModelFit[SomeData, a x^2 + b x + c, {a, b, c}, x]]

(*Plotting the NonlinearModelFit data. Note that SomeDatafit has
 to show that it is a function of x; I.e., SomeDatafit[x]*)
plotFit = Plot[SomeDatafit[x], {x, 0, 5},
  Frame → True, GridLines → Automatic,
  PlotStyle → {RGBColor[0, 1, 0]}, FrameLabel → {"x", "y"}]

(*showing the plots pplot and plotpfit on the same graph*)
Show[SomeDataplot, plotFit,
 PlotLabel → "Fit = Green Line; Data = Red Points"]

(* Use LinearModelFit command that is new to Mathematica 7*)
Print["R^2 = ", SomeDatafit["RSquared"]]
(* Provides the R^2 value *)
Print["Adjusted R^2 = ", SomeDatafit["AdjustedRSquared"]]
(* Provides the adjusted R^2 value *)
```

```
{{0, 0}, {1, 1}, {2, 4.1}, {3, 8.9}, {4, 16.1}, {5, 24.9}}
```




y = FittedModel[ $-0.00714286 + 0.0421429\,x + 0.989286\,x^2$ ]





$R^2$ = 0.999966

Adjusted $R^2$ = 0.999932