Physics through Computational Thinking

Data Analysis - Curve Fitting

Auditya Sharma and Ambar Jain
Dept. of Physics, IISER Bhopal

Outline

In this lecture you will

1. learn some curve fitting.
2. apply to concrete examples.

Galileo's *Mathematica* Experiment!

Let us generate some data from a thought experiment, which we will then analyze by the basic statistical tools we have described above. This is a recurring set of techniques that would come in handy for analyzing a variety of experimental/simulation data.

```
g = 9.8;

data = Prepend[Table[ Table[Round[1/2 g i^2 (1 + 0.15 RandomReal[]), 0.01], {i, 1, 10}], {5}], Range[1, 10]] // Transpose;

data // TableForm;

meandata = Table[
 set = data[[n, 2 ;;]];
 {n, Mean[set],  StandardDeviation[set] / Sqrt[Length[set] - 1]}
 , {n, 1, 10}
 ];

TableForm[meandata];
meandata[[ ;; , 2 ;; 3]];

Needs["ErrorBarPlots`"];

ErrorListPlot[meandata[[ ;; , 2 ;; 3]], Joined → True];
```

Basic Fitting Theory (Least Squares Minimization Criterion)

Suppose we have N data points $y_1(x_1)$, $y_2(x_2)$, $y_3(x_3)$, ..., $y_N(x_N)$ with error-bars σ_1, σ_2, σ_3, ..., σ_N respectively. If we have reason to believe that these data points must correspond to a linear curve, our task would then be to extract the best straight line which would correspond to the available data. In other words we wish to find the best parameters a, b such that the linear curve $y = a\,x + b$ fits the available data as closely as possible. One standard approach for this problem is to consider the quantity

$$\chi^2(a, b) = \frac{(y_1 - a\,x_1 - b)^2}{\sigma_1{}^2} + \frac{(y_2 - a\,x_2 - b)^2}{\sigma_2{}^2} + \ldots + \frac{(y_N - a\,x_N - b)^2}{\sigma_N{}^2}, \tag{1}$$

and minimize it over a, b.

The above scheme can of course be generalized to more complicated functions than linear. Also several more complicated functions can be recast into a form such that linear regression holds, and this is a standard trick. For example if one is interested in fitting to a quadratic form, simply taking the logarithm of both the dependent and independent variables then makes it a problem of linear regression of the 'logged' quantities.

An enormous amount of theory exists on this topic, and it is definitely not the objective of this course to get into the details of all of this. We simply point out this as an example of the type of philosophy that goes into the fitting routines. Here, we wish to make use of the fitting functions that are in-built in *Mathematica* to be able to carry out some quick analyses of our data.

Extracting *g* from fits

```
Manipulate[Show[Plot[a x^2, {x, 0, 10}, PlotLabel → a], ErrorListPlot[ meandata⟦ ;; , 2 ;; 3⟧]], {a, 1, 10}];
```

```
testfunc = Table[5 x^2, {x, 1, 10}];
(meandata[[1, 2]] - testfunc[[1]])^2 / meandata[[1, 3]]^2;
temparray = (meandata⟦ ;; , 2⟧ - testfunc)^2 / meandata[[ ;; , 3]]^2;
```

```
chisq = Total[temparray];
```

```
chisqdata = Table[
   f[x_] = a x^2;
   chisq = Total[(meandata⟦ ;; , 2⟧ - Table[f[t], {t, 1, 10}])^2 / meandata[[ ;; , 3]]^2];
 {a, chisq}, {a, 4, 6, 0.01}];
```

```
ListPlot[chisqdata, Joined → True];
```

```
parabola[x_] = Fit[meandata[[ ;; , 2]], {1, x, x^2}, x]
```

$-0.350067 - 0.2453\,x + 5.42889\,x^2$

```
Show[Plot[parabola[x], {x, 0, 10}], ErrorListPlot[ meandata⟦ ;; , 2 ;; 3⟧, Joined → True]];
```

$\frac{5.03 - 9.8/2}{9.8};$

Nonlinear Fits

```
fn[x_] = a + b x + c x^2;

fit = FindFit[meandata[[ ;; , 2]], {a + b x + c x^2}, {a, b, c}, {x}] ;

fn[x] /. fit;

Show[Plot[fn[x] /. fit, {x, 0, 10}], ErrorListPlot[ meandata[[ ;; , 2 ;; 3]], Joined → True]];
```