MatLab for The Human Sciences

Importing Data

Neil Carter

March 30, 2016

Importing Data

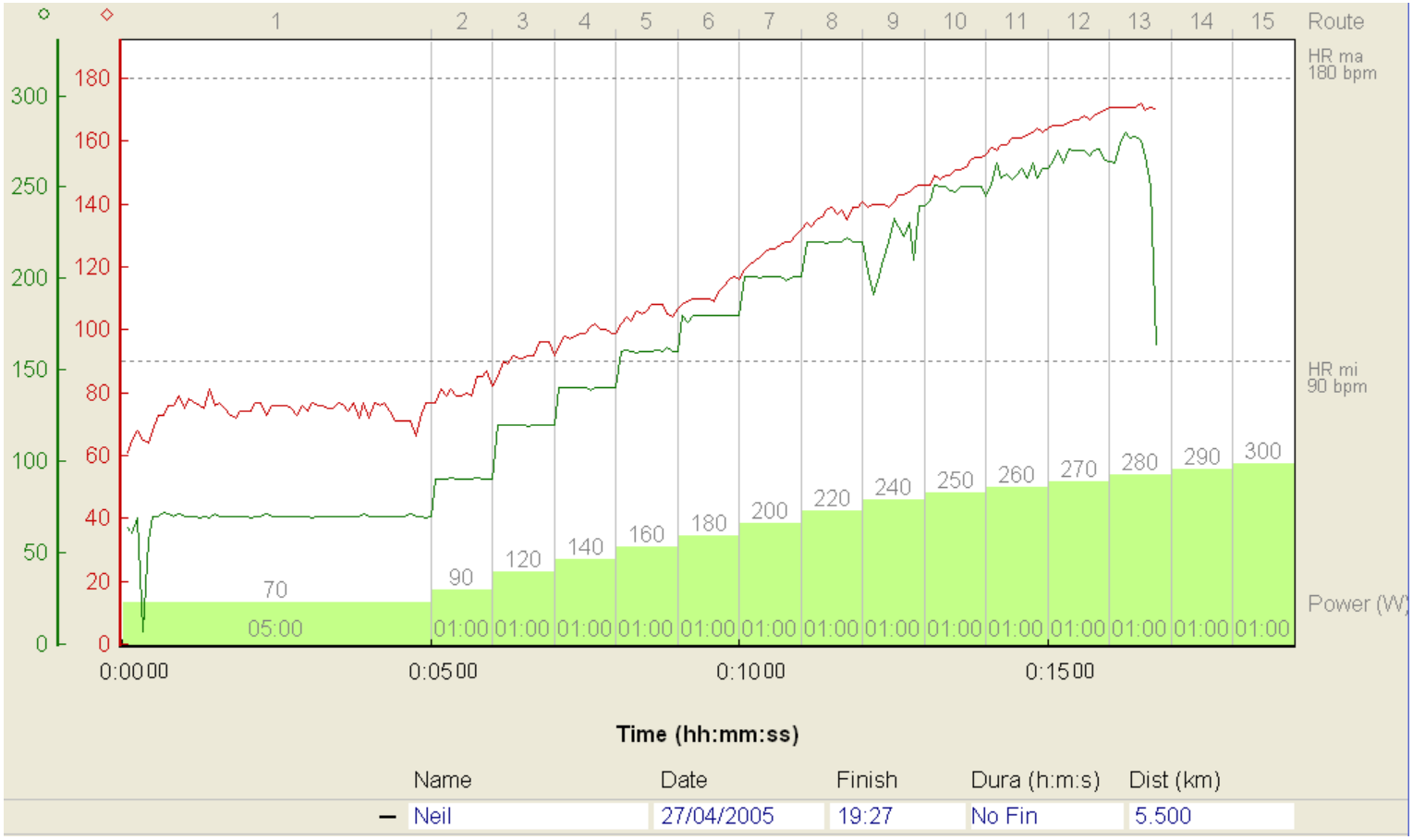
- Sometimes, the import tool is not sophisticated enough to cope with a file's format.
- In these cases, we can use the file handling commands such as `fopen()`, `textscan()`, and `fclose()`.

Step-by-step

1. Describe the format
2. Open the file
3. Read the data
4. Close the file

The Conconi Test

- This is data recorded during a sports-science graded exercise test where a person pedals an electronic bicycle at increasing workloads until exhaustion (red is heart rate, green is power):



Importing Conconi Data

- The data recorded during the exercise test is stored in a text file thus:

```
A.N.Other  27/04/2009  15:30

Time        Power      Cad   Pulse
(h:m:s)       (W)    (rpm)   (bpm)

00:00:05      64       20      61
00:00:10      61       27      65
00:00:15      69       40      68
00:00:20       7       40      65
00:00:25      57        6      64
00:00:30      70       45      68
00:00:35      70       55      73
00:00:40      72       68      73
00:00:45      71       69      76
00:00:50      70       68      76
00:00:55      71       71      79
00:01:00      70       66      75
00:01:05      70       72      78
00:01:10      70       76      77
00:01:15      69       75      76
```

The textscan() function

- The Time (e.g. “00:00:05”), Power (e.g. “70 W”), and Duration (e.g. “05:00”) columns are not in a simple numerical format.
- The `textscan()` function allows us to handle the non-standard formats in the previous slide:

```
C = textscan(fid, '%s%f%f%f', ...
    'Delimiter', '\t', ...
    'HeaderLines', 5 );
```

- We have to use a cell array because the data in the matrix has different types (is *heterogeneous*).
- The cell array returned by `textscan()` contains all the lines of data in a single row.

The textscan() function

- The string `'%s%f%f%f'` is called the format string; it tells MatLab the format of the data.
- `'` is the string delimiter.
- `%f` means floating point.
- `%s` means string (of text).
- It is implied that the format string refers to a single line.

textscan() Format String

00:00:05	64	20	61
%s	%f	%f	%f

The textscan() function

```
C = textscan(fid, '%s%f%f%f', ...
    'Delimiter', '\t', ...
    'HeaderLines', 5 );
```

- `Delimiter` indicates that the fields are separated by a tab character (`\t`).
- `HeaderLines` indicates that the first five lines contain only text that can be ignored.

Time Data

- Matlab stores time as serial data numbers.
- A serial date number is a floating-point value representing the number of days (including the fraction of the current day) since 00:00 on January the 1^{st} 0000.
- Matlab provides a variety of functions to convert between the various time formats that it understands.

The Time Field

- To change the time string to a format suitable for our purposes, we convert it into a serial date value using the `datenum()` function:

```
t = datenum( C{2}, 'HH:MM:SS' )
```

- Note that the index for `C` is surrounded by curly braces; this is because it is a cell array as opposed to a matrix.
- Recall that all the data is stored in a single row of the cell array (but each column goes into a different variable).

Step-by-step

Time (h:m:s)	Power (W)	Cad (rpm)	Pulse (bpm)
00:00:05	64	20	61
00:00:10	61	27	65
00:00:15	69	40	68

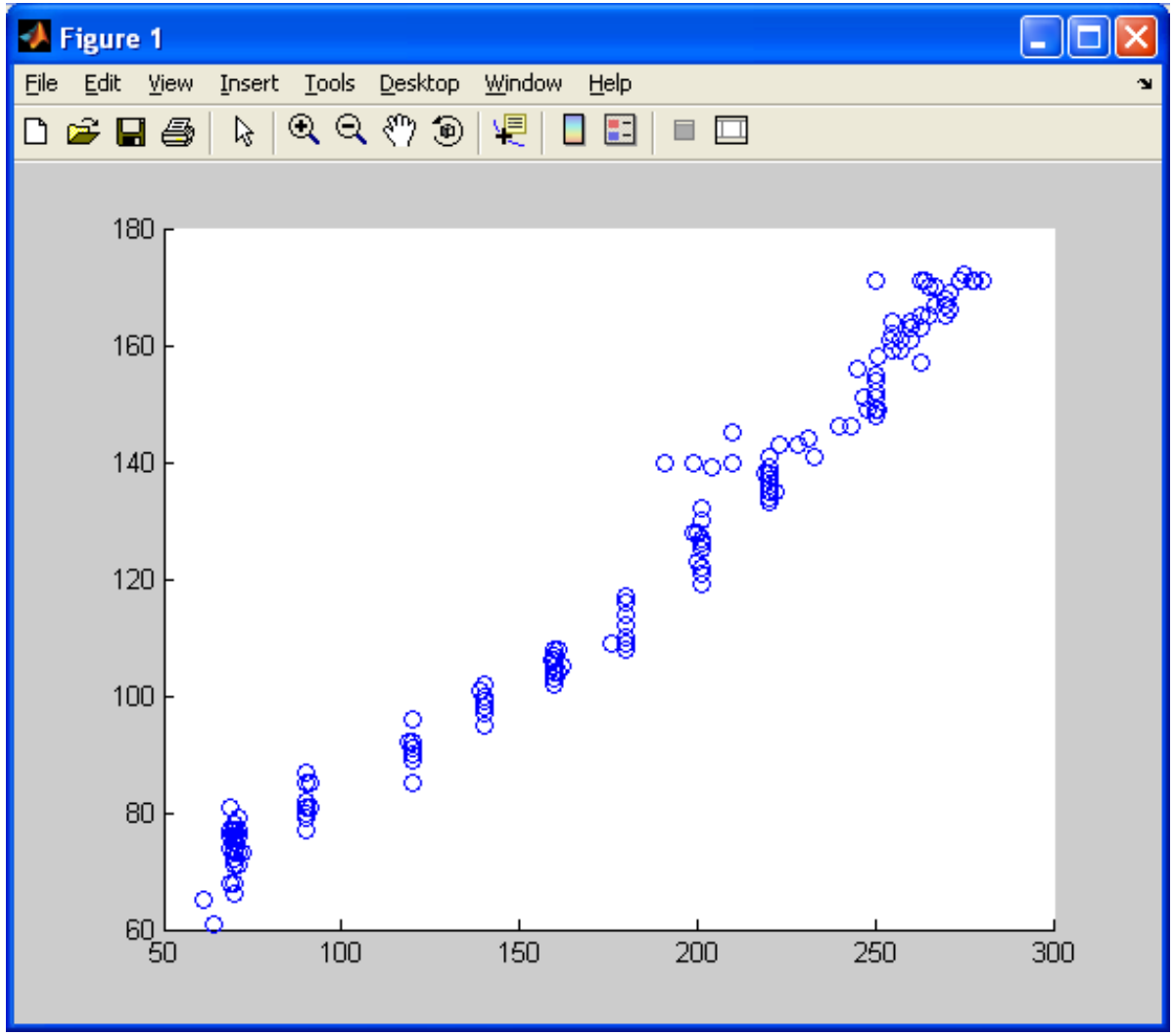
- The following line builds a matrix `M` from the cell array `C`, using only the columns of interest.
  `M=[ t, C{4}, C{5}, C{6}, C{7}, C{8} ]`
- `M`'s columns: 1=Time, 2=Power, 3=Cadence, 4=Pulse.

Proceed as Usual

- We now have our data in the rows and columns of a matrix.
- And all the data are in a proper numeric format.
- So we could plot HR against power using the command

```
scatter( M(:,3), M(:,5) )
```

Power vs Heart Rate



Useful Resources

- Useful guide on importing and plotting data with MatLab

  `http://web.cecs.pdx.edu/~gerry/MATLAB/plotting/loadingPlotData.html`