

FSTY-405 — Lab 3

Site index, gnuplot

Name:

Objectives

Understanding site index. Using **gnuplot**.

Answer the questions in the spaces provided, attach printouts, and hand in at the end of the lab for marks.

1 Gnuplot, site index equations

Gnuplot is a flexible, free graphing program. It can be downloaded from <http://www.gnuplot.info>, where there are also pointers to documentation, etc. (see the Syllabus for a recommended short tutorial). It is already installed on the lab computers. There is another good tutorial at www.me.umn.edu/~langlais/UMN/gnuplot-shortcourse/gnuplot.pdf.

Start **wgnuplot**, resize the window to an appropriate size. Gnuplot is command-based; this MS Windows version provides also menus to generate the commands. Explore the menus and see what is available there. Try pressing the right mouse button. We shall stick to the simplest 2-D graphics facilities for now.

The following site index equation for interior trembling aspen (Alberta Forest Service, 1985) is taken from the BC MOF *Land Management Handbook, Field Guide Insert 6*:

$$\begin{aligned}
H &= 1.3 + (S - 1.3) b_1/b_2 \\
b_1 &= 1 + \exp [6.879 - 1.32 \ln 50 - 0.618 \ln (S - 1.3)] \\
b_2 &= 1 + \exp [6.879 - 1.32 \ln A - 0.618 \ln (S - 1.3)] \\
y_{tb} &= 1.33 + 38.56 / S
\end{aligned}$$

where:

H = top height (m)
S = site index (height at 50 years breast-height age)
A = breast-height age (years)
y_{tb} = number of years to reach breast height (1.3 m)
ln = natural logarithm
exp = exponential function to the base e

Notes:

The height-age (site index) curves were developed from stem analysis of 207 dominant and co-dominant trembling aspen trees located throughout Alberta. Site index ranged from about 9 to 24 m at 50 years breast-height age and included trees up to 90 years old.

We want to plot H for various values of A and S . Instead of writing the equations every time, it is easier to *define a function* first, using a command like `H(A,S) = ...`. Here the function is given in three parts, so we have to define also b_1 and b_2 .

Note that b_1 is a special case of b_2 , so we can define just one expression for b :

$$b(A,S) = 1 + \exp(6.879 - 1.32 * \log(A) - 0.618 * \log(S - 1.3))$$

(I am using `this` font for computer input). Observe the slightly different syntax required, typical of computer code. Type it in gnuplot, and press **Enter**. Now we can enter the expression for H :

$$H(A,S) = 1.3 + (S - 1.3) * (b(50,S)/b(A,S))$$

Think about it! Let's check, displaying the height at 50 years for site index 20: `print H(50,20)`.

Height =..... Why?
.....
Approx. asymptote for site index 18? :..... (Hint: try large ages)

Notes: The up and down arrows (or the *Prev* and *Next* buttons) allow moving through previous commands, which can be edited and re-submitted. Cut and paste can be done by marking dragging the mouse, and using the right-click pop-up menu. Pitfall **warning**: gnuplot uses integer division if both arguments are integers; compare `print 1/3` and `print 1./3`. By the way, commands can be abbreviated: `pr exp(1)`.

Now plot it: `plot H(x,20)`. Not really what we wanted? Set the range of x to be between 0 and 150 years, and try again:

```
set xrange [0:150]
replot
(or use the Axes menu and Replot button).
```

OK, let's now plot a family of site index curves, for $S = 10, 15, \dots, 25$. First, just for looks, make A the dummy (x -axis) variable, instead of the default x : `set dummy A` (or use the *Functions* menu). Now do it:

```
p H(A,10), H(A,15), H(A,20), H(A,25)
```

Not bad! A few additional touches (think about what you are doing, enter “`help whatever`” or use the *Help* menu for details):

```
set xlabel 'Breast-height age (years)'
set ylabel 'Top height (m)'
set title 'Interior Trembling Aspen (AFS 1985)'
rep

unset key
set grid
rep
```

Now label the curves with the respective site indices. We give the label's text, and its location coordinates. Can use the *Chart* menu, or faster, type in the commands:

```
set label '10' at 152,17
se la '15' at 152,24
```

```
...
se la '25' at 152,38
rep
```

Labels not quite in the right place? Fix them. E.g.:

```
show label
set label 4 at 152,37
rep
```

(you may find useful the coordinates display at the bottom of the graph window).

Finally, we can use the same line type/colour for all the curves (line type 1) Try the *Plot* menu (and/or *Help*) when you forget the commands:

```
plot H(A,10) with line 1, H(A,15) w l 1, H(A,20) w l 1,
H(A,25) w l 1
```

Plotting data

Do not exit gnuplot, so that we can continue using our defined functions (or use *Save/Open* to save and restore the session). Copy the data file `At750gp.dat` from the `Fsty405` folder to a directory in your disk. Use the *ChDir* button to use that directory by default. The data was extracted from the VDYP overlay files in the MOF G&Y web site.

The data file is plain text, looking like this:

```
#PLOTID AGE HEIGHT SI CC DBH BA VOLUME TSA STK_CL

4501205 90 23.40 17.70 70.00 29.00 1.96 15.60 11 2
4501205 100 25.70 18.80 70.00 15.60 0.94 6.70 11 2
4501205 110 26.10 18.40 70.00 12.90 4.25 20.60 11 2

5600604 43 19.90 22.80 80.00 15.80 6.79 45.30 26 0

...
```

Here we are interested only in columns 2 (age), and 3 (top height). Gnuplot ignores any lines starting with #. Each of the other lines is a PSP measurement, and different plots are separated by blank lines. Get a graph of height over age:

```
plot 'At750gp.dat' using 2:3
```

What do the different parts of this command mean?

The default is to plot data as points. Change it for joining the consecutive measurements with lines:

```
set style data linespoints  
rep
```

(or use the *Styles* menu). The lines are interrupted by the blank lines in the file.

All together now! (you may use *Prev* or up-arrow to recall the old command, and add to it):

```
p H(A,10) with line 1, H(A,15)w l 1,H(A,20)w l 1,H(A,25)w l 1,  
'At750gp.dat' using 2:3
```

Print the graph and attach to the handout. **Make sure that you include your name in the title of every attachment.** You can use the *newline* character \n, e.g.:

```
set title "Aspen Curves\nJohn Doe"
```

(must use double quotes for the newline to work).

The easiest way of printing is to click on the upper-left corner icon of the graph window, and select *Options>Print...* Better quality printouts and image files can be generated by selecting appropriate devices (investigate the *File* menu and the *Help* sometime).

Do-it-yourself

We will use a guide curve method with the Schumacher equation. A *guide curve* is:

.....
.....
.....
.....

The equation: $\text{Schum}(t)=a*\exp(-b/t)$. Guess values for the parameters a and b ¹:

```
a=30.0
b=40.0
plot Schum(A), 'At750gp.dat'u 2:3
```

Not too good? Gnuplot has a built-in nonlinear regression command, `fit`. It finds the parameter values that minimize the sum of squares of the residuals (differences between observed and predicted values). The parameters are specified through “via”:

```
fit Schum(A)'At750gp.dat'u 2:3 via a,b
print a,b
rep
```

Better? No linearization transformations needed (non-linear, not linear regression). But be aware that the iterative procedure used by `fit` can fail, or converge to the wrong solution, if the initial guess is not good enough.

Now we want proportional site index curves $H = k \times \text{Schum}(A)$, such that for $A = 50$ their height is S . Convince yourself that k must equal $S/\text{Schum}(50)$. Check with `print`.

Define our new Schumacher site index curve function $H(A,S)=(S/ \dots$. Produce a graph with these curves and the data, **print and attach**. Compare to the previous one.

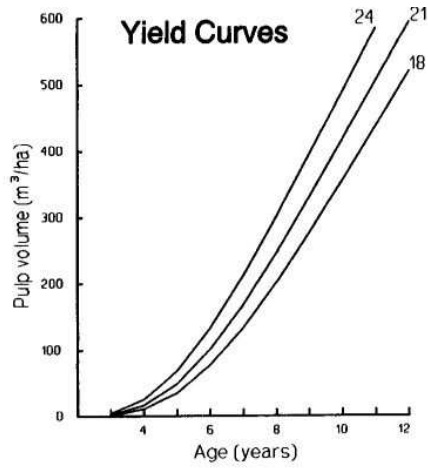
On your own

Exit and restart gnuplot to get rid of the previous settings (or enter `reset`). Here is a graph, equation and data, from the textbook:

¹The adequacy of the model could be assessed, and initial parameter estimates be obtained (from the observed slope and intercept), by graphing $\ln H$ over $1/t$. This is done with “`using (1./$2):(log($3))`” in the `plot` command. You would need to go back to automatic axis scaling: `set auto`.

Yield Table

Age (y)	Site Index (m)		
	18	21	24
3	1	2	4
4	11	17	25
5	36	50	69
6	78	102	133
7	134	169	212
8	201	246	301
9	276	330	395
10	356	418	491
11	438	507	586
12	520	594	679



Yield Equation: $\log(V+1) = 3.534 - 14.02/t + 0.2314 S/t$

From the equation, re-create the graph using gnuplot (curves, labels, etc.) Then, plot the table data points on the same graph.

Hints:

(a) Use *Notepad* (Start → Programs → Accessories → Notepad) or your favourite text editor to create a data file with the data from the textbook figure. Free format, numbers may be separated by spaces or tabs.

(b) $\log_a y = x \Leftrightarrow y = a^x$

(c) It is not clear from context if the log is a natural log or base 10. Use the one that makes sense.