Geol 335.3

Lab 3 – Interpretation of refraction seismic data using the PLUS-MINUS method

The data for this lab represents a shallow seismic investigation for groundwater exploration. The problem is to find the deepest point of a buried valley that might contain a gravel aquifer.

24 geophones were placed at 12-m intervals in a fixed spread. Four shots were fired at locations along the spread. Each shot was fired at the depth of 1 m. Readings of first-arrival travel times from each shot and at each geophone position are shown in the table below. This table is also provided in Excel format from which you can extract columns or create a comma-separated file for loading into other software.

Distance (m)	Elevation (m)	Shot A	Shot B	Shot C	Shot D
0	101.0	Shot position	.068	.144	.196
12	101.0	.015	.059	.135	.187
24	100.9	.024	.051	.127	.179
36	100.9	.033	.042	.118	.170
48	100.9	.042	.034	.110	.162
60	100.8	.050	.025	.102	.153
72	100.7	.059	.015	.093	.145
84	100.6	.068	Shot position	.085	.136
96	100.4	.077	.015	.076	.128
108	100.2	.086	.026	.068	.120
120	100.2	.095	.035	.060	.112
132	100.5	.104	.045	.052	.104
144	101.2	.114	.054	.045	.097
156	102.0	.124	.064	.030	.090
168	102.5	.134	.074	.015	.083
180	103.0	.144	.085	Shot position	.075
192	103.3	.154	.095	.015	.068
204	103.6	.163	.104	.030	.062
216	103.9	.168	.109	.045	.053
228	104.2	.173	.114	.053	.044
240	104.5	.179	.119	.058	.035
252	104.7	.184	.124	.063	.026
264	104.8	.190	.130	.069	.015
276	105.0	.196	.137	.076	Shot position

You may use Matlab or Octave, Excel, or do all plotting and calculations by hand. If using Matlab (Octave) follow the fairly detailed template `lab3.m` available from the lab web page.

> The following data plotting and analysis will consist of a (relatively) long series of steps. You can conduct the whole sequence interactively in a Matlab command window, but it is a better idea to put these steps in a program file, for example named `lab3.m` (like the template). Then, edit this file and execute all commands at once by typing
>
> ```
> lab3
> ```
>
> in the command window. In Matlab, you can also use the "Play" icon (and many other!) to execute the program.
>
> Note that if using this method, you need to start plotting sequences with command `clf` ("clear current figure"). Otherwise, plot elements will be continuously added to the currently open figure.

The table above can be loaded into Matlab by extracting a text file and using command `data = load(…)`. As in the preceding lab, coordinates of picks can then be extracted in column matrices:

```
x = data(:,1);       % all coordinates
z = data(:,2);       % elevations
tA = data(:,3);      % all times from shot A
tB = data(:,4);      % all times from shot B
```

and so on.

In the Excel data file, the times at shot positions are set equal zero for convenient loading. These times cannot be picked from the data of course.

> A more sophisticated way for loading Excel data table is implemented by function `readtable(…)` in Matlab. The entire spreadsheet can be loaded using `data = readtable(filename`, but the resulting data will be a structure also containing other information like spreadsheet column names. The contents of the structure can be examined by command `fieldnames(data)`, which returns a list of column names and other attributes (try it!). Data columns can be extracted from the respective fields of structure `data`. In particular, with our table, they are obtained like this:
>
> ```
> x = data.Distance;        % all coordinates
> z = data.Elevation;       % elevations
> tA = data.Shot_A;         % all times from shot A
> ```

and so on.

In Octave, there is a similar `.xlsx` spreadsheet-reading function called `xlsread(…)` (in package '`io`'). This function extracts only numerical data (without column names) and outputs a simple matrix like matrix `data` above.

1) [5%] On the same sheet of paper (Matlab figure), **plot refraction time/distance graphs and surface elevations**. Use lines and symbols for travel times and one line for elevation. In Matlab or Octave, use command `subplot(…)` to produce two plots nicely aligned vertically. Use commands `xlabel(…)`,`ylabel(…)`, and `title(…)` to annotate the subplots.

2) [5%] Using the plots, identify the travel-time branches (segments) of direct waves and head waves from each shot. **Plot these segments by lines** using different colors for direct waves and head waves.

   <u>Note the change in the head-wave moveouts</u> from shots A and B near x = 200–220 m. Consider two ways to interpret these changes: (i) as caused by a change in refractor dip (in which direction?), and (ii) as an indication of another, deeper refractor. <u>For the moment, stay with option (i).</u>

In Matlab, data segments can be conveniently represented by defining "Boolean" arrays, i.e. arrays containing values 0 or 1 for each location `x`. A value of 1 (which means 'true'), would mean that this location is included in the travel-time segment, and value = 0 (meaning 'false') would mean that the location is not included. For example, if you see from your plots that direct waves from shot C are seen in the interval of distances from 140 to 220 m, then expression

```
dwC = x > 140 & x < 220;          %    (**)
```

gives such a Boolean selector for this interval. Using this selector, you can obtain indices of the corresponding points using expression

```
ind_dwC = find(dwC);
```

and using these indices, you obtain the coordinates of these points `x(ind_dwC)`, elevations `z(ind_dwC)`, direct-wave times `tC(ind_dwC)`, etc. Using these indices, all direct-wave travel times can be plotted like this:

```
plot(x(ind_dwC),tC(ind_dwC),'r-*')
```

This command will draw red asterisks at the interpreted direct-wave picks and connect them with a solid line.

3) [5%] Compare the near-offset (direct wave) moveouts from all four shots. **Estimate V_1** from shot C. Note that in other shots, direct-wave branches are poorly sampled by geophones. What does this mean in terms of the thickness of the first layer?

4) [5%] Identify pairs of head wave travel-time segments that you will use in the plus-minus inversion. These segments should be recorded in opposite directions and cover the same range of distances.

> In Matlab, the reversed head-wave segments can be obtained from Boolean selector arrays in expression (**) above. For example, if `hwfA` is a such a selector array for the forward headwave from source A and array `hwbD` is a selector for the backward headwave from source D, then the selector for the overlap is their <u>Boolean product</u>:
>
> ```
> hw_AD = hwfA & hwbD;
> ```
>
> Indices of these points can again be found using function `find(…)`:
>
> ```
> ind_hw_AD = find(hw_AD);
> ```

5) [25%] Calculate the MINUS times for all pairs of reversed head-wave segments:

$$t_{MINUS} = T_{Ax} - T_{Bx}.$$

where T_{Ax} is the time from shot A to receiver x. Use only the head-wave segments that have opposite slopes on the time/distance graphs.

Plot the MINUS times vs. offset for ALL shot pairs on the same graph. From the slope of this graph, determine the refractor velocity (V_2). In Matlab, the slope can be obtained by using function `polyfit(…)` for the appropriate range of points. The slope of the MINUS graph is $2/V_2$.

Determine whether (and where) V_2 changes significantly within the profile.

6) [25%] Calculate the PLUS times for all pairs of shots. Use only the head-wave segments that have opposite slopes:

$$t_{PLUS} = T_{Ax} + T_{Bx} - T_{AB}.$$

The theory for PLUS times is valid for shots on the surface. If the shots are buried, travel times are smaller than they would be if the shot were on the surface, and so a correction should be made. The correction is the time required for the wave from the source to reach the surface. This time should be added to all recorded times. In this exercise, we will use 1 ms for all shot delay times:

$$t_{PLUS} = T_{Ax} + T_{Bx} - T_{AB} + 0.001 \text{ s}.$$

7) [15%] Determine the depth below the shot points to the first interface using the following equation:

$$t_{PLUS} = 2T_{delay} = \frac{2Z_1 \cos i_c}{V_1},$$

where Z_1 is the depth to the first interface, V_1 is the first layer velocity at offset x, and i_c is the critical angle (recall the relation $\sin i_c = \frac{V_1}{V_2}$). Therefore:

$$Z_1 = \frac{t_{Delay} V_1}{\cos i_c}. \qquad (***)$$

Plot the vertical position of the refractor $(z_{\text{elevation}} - Z_1)$ **on your elevation plot**. This should be your final model. **Comment on the agreement** between depth and velocity V_2 estimates from the different head-wave pairs.

> When evaluating eq. (***), you will first need to interpolate values V_2 at all points within the profile. In Matlab, this can be easily done by function `interp1`. If you measure values of V_2 at several points, for example `x1`, `x2`, `x3`, `x4`, and these values are `V21`, `V22`, `V23`, `V24`, then expression
>
> ```
> V2 = interp1([x1,x2,x3,x4],[V21,V22,V23,V24],…
>                   x,'linear','extrap');
> ```
>
> will linearly interpolate these values for all values of `x` and extrapolate them outside of the ends of the interval [`x1,x4`].
>
> After interpolating the values of V_2, you can obtain the critical angle by expression (recall the relation $\sin i_c = \frac{V_1}{V_2}$):
>
> ```
> ic = asin(V1*V2.^(-1));
> ```
>
> and then you can similarly evaluate equation (***).

8) [10%] Now return to the uncertainty of the change in refractor slopes in question 2). How would the model change if we now assume that the travel-time branches at x > 200–220 m from shots A and B come from a deeper refractor? Can you offer further arguments in favor of or against such a model?
9) [5%] Rounding the measured times to 1 ms implies a time error of $\delta t \approx 0.5$ ms. Using eq. (***), how large depth error would this δt lead to? For a comparison, how much depth uncertainty would be caused by a 0.5-ms error in <u>normal-incidence reflection</u> time?

Hand in:

Plots and write-ups on paper in a binder or in a Word, Excel, or PDF file electronically.