Geol 300 Lab 8: Modeling the Theis equation
Write-up due April 4

Overview. Practice with matlab in operations and symbolic manipulation. Creation of scripts to solve the theis equation.

Turn in:

- Write-up with problem statement, method, code, results, and discussion.
- Powerpoint (hardcopy with several slides per page)

Goals:

A) Calculate the effect of two wells.
B) Generate a time animation over 24 hours (sampled every hour) with pumping from hour 3 to hour 6.

Practice with matlab

Try the following commands:

```
x = 180.0*pi/45
x = 2^8
x = 4^(0.5)
x = 6
x = 6;
```

create a row vector

```
x = [1 2 2 3 1 5]
```

create a column vector

```
y = [1;2;2;3;1;5]
```

create a matrix

```
k = [1 2 3;6 9 10; 18 17 16]
```

extract first column of the new matrix

```
fc = k[:,1]
```

extract the first row

```
fr = k[1,:]
```

another was to create a matrix is by assigning the elements

```
k(1,1) = 1;
k(1,2) = 2;
k(1,3) = 3;
k(2,1) = 6;
```

and so on.

Matlab variable are case sensitive; upper-case or lower-case makes a difference.

Matlab does a lot more; try help demos and try a few of the demo programs.

Now try “symintro” to see some of the symbolic functions. “diff(x)” calculates the derivative of x. Note that x must be declared as a symbolic variable for this to work – see symintro (and if you want to use x as a non-symbolic variable later, use the “clear” command. I think).
What is the derivative of x^6 + 0.5*x^4 + 256*x^18?
What is the integral of x^6 + 0.5*x^4 + 256*x^18?

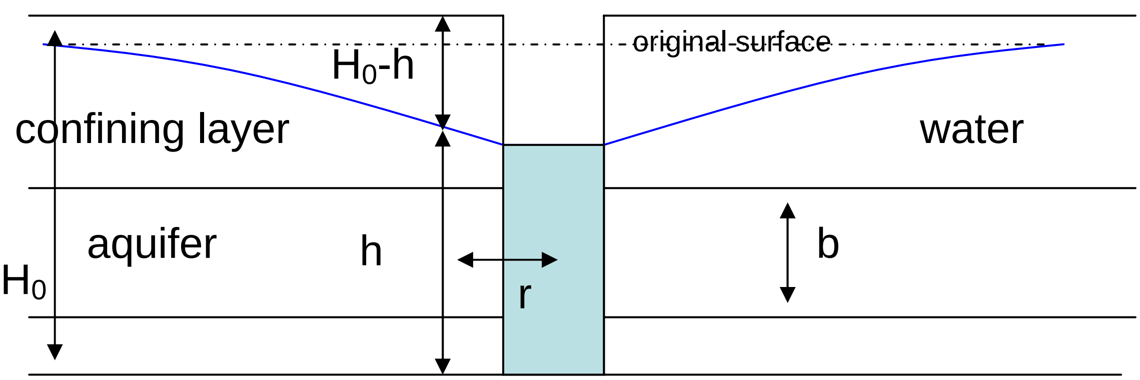
Using the solve command, solve for x in the quadratic equation
$Ax^2 + Bx + C$
Remember that A, B, C and x all have to declared as symbolic.

Theis equation. Now we will create some scripts to solve a set of equations numerically. Extracting water from a well will lower the aquifer surface, which will form a cone of depression around the well. The depth and radius of this depressed area depends on the rate of extraction and the properties of the aquifer.

Q	pumping rate
H	hydraulic head
H0	initial hydraulic head
H0-h	drawdown
T	aquifer transmissivity
t	time since pumping began
r	radial distance from well
S	aquifer storativity

Equations:

H0-h = (Q/rpT)W(u)
U = (r2S)/(4Tt)
W(u) = -0.5772-*ln*(u) + u – u2/2*2+u3/3*3-u4/4*4…



These equations can written as matlab scripts. A script is a file (mfile, usually named *.m) that can be run as a command in Matlab. For example, a script named theis.m when saved in the correct directory, can be run as the command “>> theis” – in effect, you are creating your own Matlab commands. There are two types of scripts: ones that require some sort of input which is surrounded by parentheses and one that don't require input. Scripts that require input are called functions. In the following section, well.m and theis.m are functions, gridtheis.m is not.

Mfiles are simply a way to put Matlab commands in a file. Anything that can be typed into Matlab can be put in a mfile (and vice versa). The big advantage is that it saves typing the same thing over and over again. The usual procedure in creating a script is to do in steps. First, start simple and make sure it works. Then gradually add more complicated parts.

One way to create an mfile is to go under "File, new, mfile". Once, created, remember to save it (and what directory it is saved in). For it to work as a command in Matlab, either Matlab must be running ("Current Directory" window) in the directory where the script was saved or the directory must be in the search path of Matlab. For more details on the search path, try "help path".

The following code fragments should be saved as matlab scripts.
Existing code: The well function (save as well.m). Percent signs denote comments in matlab. Use only the text in Courier font.

```
%-----------------------------------
% the well function
function [w] = well(u)
% the well function
% calculate out to 8
% log is the natural log
w = -0.5772 - log(u) + u - (u^2)/(2^2) + (u^3)/(3^3) -
(u^4)/(4^4) + (u^5)/(5^5) - (u^6)/(6^6) + (u^7)/(7^7) -
(u^8)/(8^8);
%------------------------------------
```

Test data:
>> **well(10^(-9))**

ans = 20.1461

>>

Save as theis.m

```
%------------------------------
% The Theis function
% Q pumping rate
% drawdown is drawdown (h0-h)
% r radial distance from well
% S storativity
% trans transmissivity
% time time since pumping began
```

```
function [u,drawdown] = theis(r,Q,S,trans,time)
u = (r^2)*S/(4*trans*time);
drawdown = (Q/(4.0*3.14159*trans))*well(u);
%----------------------------------
```

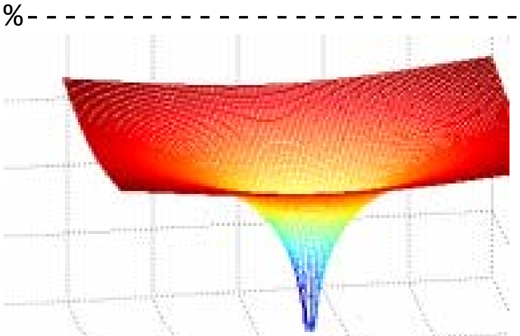
```
>> r = 7
>> Q = 2725
>> S = 0.0051
>> trans = 299
>> time = 1
>> [u,drawdown] = theis(r,Q,S,trans,time)
```

u = 2.0895e-004
drawdown =5.7269

```
%--------------------------------------------------------
% calculate over a 200 ft grid with 2 foot spacing
% well is at grid point (100,100)

r = 7
Q = 2725
S = 0.0051
trans = 299
time = 1

for i = 1:101
    for j = 1:101
        x = 2*i;
        y = 2*j;
        r = sqrt((100-x)^2 + (100-y)^2);
        [u,drawdown] = theis(r,Q,S,trans,time);
        ddgrid(i,j) = -1*drawdown;
    end
end
%--------------------------------------------------
```



Type in these equations and save as M files named well.m, theis.m, and gridtheis.m. Verify that the functions are performing as expected. After saving the results in your powerpoint file, modify the script to do one of the following problems:

Solving for two wells. The Theis equation is linear, so the solution for two wells can be determined by adding the drawdown from two wells in different locations in one grid. Figure out how to make and combine the results from two wells located 75 feet apart. Then expand it to 5 wells. Plot the results.

Adding a time loop and making a movie. Suppose we want to know what the drawdown will look as a function of time. How would we add another loop so that the water level over the grid was calculated every hour for 24 hours? Remember that the entire grid must be calculated at each time step. With everything held constant, the water level should remain the same.
Now change the rate as a function of time. Simulate a test where the pumping rate was zero for the first 2 hours, then was 2725 for the next 6 hours, then was stopped again.
The following code fragment shows how to make a movie. Your time loop should serves as the loop here.

```
for j=1:n
   plot something here
   M(j) = getframe;
end
movie(M)
```