P318 Computational Modeling
Spring 2014
Week 11 Homework
Due March 31st

For this homework assignment, I want you to implement some of the code necessary to fit the diffusion model to observed data.

I have provided a great deal of code for you to get started in Week11_Homework.zip.

Homework11.m opens with code to simulate the diffusion model using these parameters:

```
a   = 0.100;       % upper boundary (lower boundary is 0)
z   = 0.050;       % starting point
sz  = 0.010;       % variability in starting point from trial to trial
eta = 0.050;       % variability in drift rate from trial to trial
Ter = 0.200;       % non-decisional time
st  = 0.05;        % variabilit in TR from trial to trial
s2  = .01;         % diffusion coeffient is the amount of within-trial noise
drifts = [.1 .2 .3];
```

This simulates three "conditions" where each condition has its own unique drift rate. You will use this as simulated "data" and then fit the diffusion model to the data using the approach discussed in class.

Your assignment is embedded in the file mymodel_diff_cdf.m.

You will find several sections of code that note INSERT YOUR CODE HERE. That's your assignment. Fill in the missing pieces of code. To do this, you'll need to decipher the code I gave you and fill in the missing pieces based on the material from class.

Recall that CDFDif.m basically implements the complicated equations behind the diffusion model. So instead of simulating 500 or 5000 times per condition, you only need to make one call to CDFDif.m per percentile of the RT distribution.

Recall that the calling convention for CDFDif is

```
     y = CDFDif(t,x,Par)
```

where t is the time at which to evaluate the CDF, x=0 returns the CDF for hitting the 0 (lower) boundary (correct), and x=1 returns the CDF for hitting the a (upper) boundary (error)
Par is an array:

```
%   Par = [ a  Ter  eta  z  sZ  st  nu  ]
%    a = boundary separation
%    Ter = mean non-decisional component time
%    eta = standard deviation of normal drift distribution
%    z = mean starting point
%    sZ = spread of starting point distribution
%    st = spread of non-decisional component time distribution
%    nu = mean drift rate
```

Recall that CDFDif returns what is known as a degenerate CDF. Whereas a regular CDF goes between 0 and 1, the degenerate CDF goes between 0 and the probability of hitting that boundary, which would be P(Correct) or P(Error) if that's what the boundaries correspond to. So pa = CDFDif(999,1,Par) would return the probability of hitting upper boundary a.