CPSC-540 Machine Learning 2005

Homework # 3

Due Wednesday March 2 in class.

NAME:____

Signature:____

STD. NUM: ____

General guidelines for homeworks:

You are encouraged to discuss the problems with others in the class, but all write-ups are to be done on your own.

Homework grades will be based not only on getting the "correct answer," but also on good writing style and clear presentation of your solution. It is your responsibility to make sure that the graders can easily follow your line of reasoning.

Try every problem. Even if you can't solve the problem, you will receive partial credit for explaining why you got stuck on a promising line of attack. More importantly, you will get valuable feedback that will help you learn the material.

Please acknowledge the people with whom you discussed the problems and what sources you used to help you solve the problem (e.g. books from the library). This won't affect your grade but is important as academic honesty.

When dealing with Matlab exercises, please attach a printout with all your code and show your results clearly.

1. **Importance sampling for Logistic Regression**: The file `islogit.m`, consists of input-output i.i.d. data sets $x \triangleq x_{1:T} \triangleq \{x_0, x_1, \ldots, x_T\}$ and $y \triangleq y_{1:T} \triangleq \{y_0, y_1, \ldots, y_T\}$, where $x_t \in \mathbb{R}$ and $y_t \in \{0, 1\}$. The idea is to come up with a model that takes a new input x_{T+1} and produces as output $p(y_{T+1} = 1|x_{T+1})$ and $p(y_{T+1} = 0|x_{T+1})$. This classification problem arises in several areas of technology, including condition monitoring and binary decision systems. For example, when monitoring patients, we might wish to decide whether they require an increase in drug intake based on new evidence.

   For practical reasons, we parameterise our model. In particular, we introduce the following Bernoulli likelihood function:

$$p(y_t|x_t, \theta) = \left[\frac{1}{1 + \exp(-\theta x_t)}\right]^{y_t} \left[1 - \frac{1}{1 + \exp(-\theta x_t)}\right]^{1-y_t}$$

   where θ are the model parameters. The logistic function $p(y_t = 1|x_t) = \frac{1}{1+\exp(-\theta x_t)}$ is conviniently bounded between 0 and 1.

   We also assume a Gaussian prior

$$p(\theta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(\theta - \mu)'(\theta - \mu)\right)$$

   The goal of the analysis is then to compute the posterior distribution $p(\theta|x_{1:T}, y_{1:T})$. This distribution will enable us to classify new data as follows

$$p(y_{T+1}|x_{1:T+1}) = \int_\Theta p(y_{T+1}|x_{T+1}, \theta) p(\theta|x_{1:T}, y_{1:T}) d\theta$$

   Bayes' rule gives us the following expression for the posterior

$$\begin{aligned} p(\theta|x_{1:T}, y_{1:T}) \quad &\propto \quad \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(\theta - \mu)'(\theta - \mu)\right) \\ &\quad \times \prod_{t=1}^{T} \left[\frac{1}{1 + \exp(-\theta' x)}\right]^{y_t} \left[1 - \frac{1}{1 + \exp(-\theta' x)}\right]^{1-y_t} \end{aligned}$$

   The problem is that in this case we can't solve the normalising integral analytically. So we have to use numerical methods — in this case importance sampling — to approximate $p(\theta|x_{1:T}, y_{1:T})$. Note that we cannot sample from $p(\theta|x_{1:T}, y_{1:T})$ directly because we don't know the normalising constant. So instead we sample from a proposal distribution $q(\theta)$ (say a Gaussian) and weight the samples using importance sampling. After obtaining N samples of θ from the posterior, we can classify new data as follows

$$p(y_{T+1}|x_{1:T+1}) = \frac{1}{N} \sum_{i=1}^{N} p(y_{T+1}|x_{T+1}, \theta^{(i)})$$

   Load the data in the file `islogit.m`. Assume the following prior and proposal

$$p(\theta) = \mathcal{N}(1, 1.5)$$

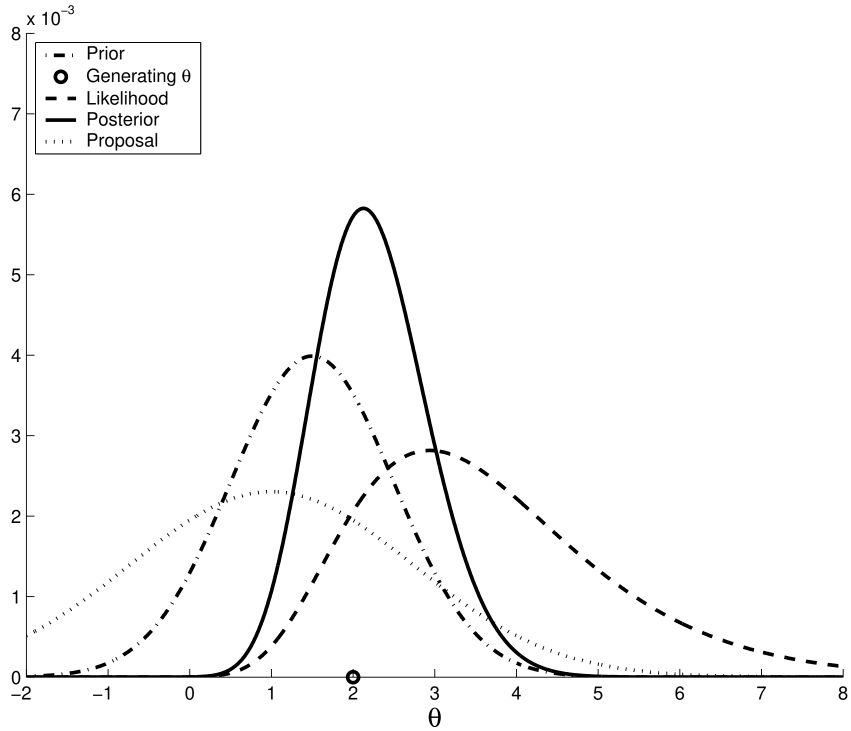


Figure 1: Plots of the distributions used in problem 1

.

$$q(\theta) = \mathcal{N}(1, 3)$$

The prior, likelihood, posterior and proposal are shown in Figure 1. I used numerical algorithms to compute them.

For this question, you need to implement an importance sampler to approximate $p(\theta|x, y)$. So you need to sample from it and then you need to use the `hist` command to produce the plot shown in Figure 2. The top histogram represents the approximation of the posterior, while the bottom one is the predictive distributions (probability of outcome being zero or one). Repeat the experiment for $N = 1000$ and $N = 10000$ samples.

To help you a bit, here's the structure of the program. Fill in the missing code where the sign ??? appears.

```
clear;

% SIMULATION PARAMETERS:
% ====================
N = 10000;              % Number of samples..
N_bins = 20;            % Number of bins in the histogram.
theta = zeros(N,1);     % Samples.
```

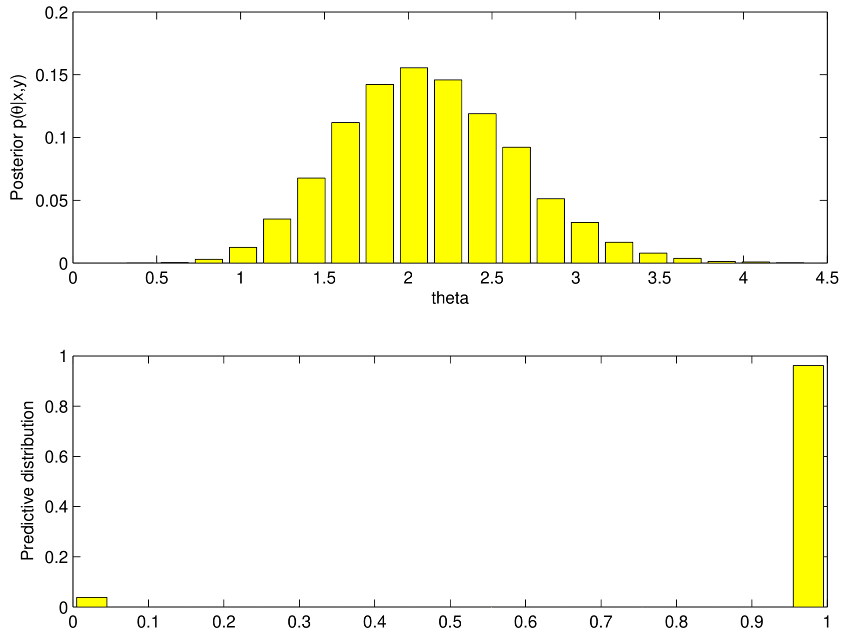


Figure 2: Solution to problem 1

.

```
% PROPOSAL SPECIFICATION:
% ======================
s_q = 3;              % Proposal variance.
m_q = 1;              % Proposal mean.

% PRIOR SPECIFICATION:
% ===================
s_p = 1;                % Prior variance.
m_p = 1.5;              % Prior mean.

% LOAD THE DATA:
% =============
islogit;
[T,arb] = size(x);

% IMPORTANCE SAMPLING:
% ===================
for i=1:N,
  z = ???
  logLikelihood = ???
  target =  ???
  proposal = ???
  w(i) = target/(proposal+1e-99);
  theta(i,:) = ???;
end;
```

```
w = w./sum(w);           % Normalise the weights.

% RESAMPLING:
% ===========
% This is a black box routine.
resampled_index = deterministicR(1:N,w');
theta(:,:) = theta(resampled_index,:);


% PLOTS THE TRUE POSTERIOR AND THE MONTE CARLO APPROXIMATION:
% ===========================================================
figure(1)
clf;
subplot(211)
[b,a] = hist(theta,N_bins);
bar(a,b/sum(b),'y')
ylabel('Posterior p(\theta|x,y)')
xlabel('theta')

% TEST DATA:
% =========
logit_pred = zeros(N,1);
y_pred = zeros(N,1);
x = randn; % Generate a random test point.
for i=1:N,
  logit_pred(i) = (1+exp(-x*theta(i))).^(-1);
  y_pred(i) = rand < logit_pred(i); % i.e. y_pred is Bernoulli(logit_pred).
end;
subplot(212)
[b,a] = hist(y_pred,N_bins);
bar(a,b/sum(b),'y')
ylabel('Predictive distribution')
```

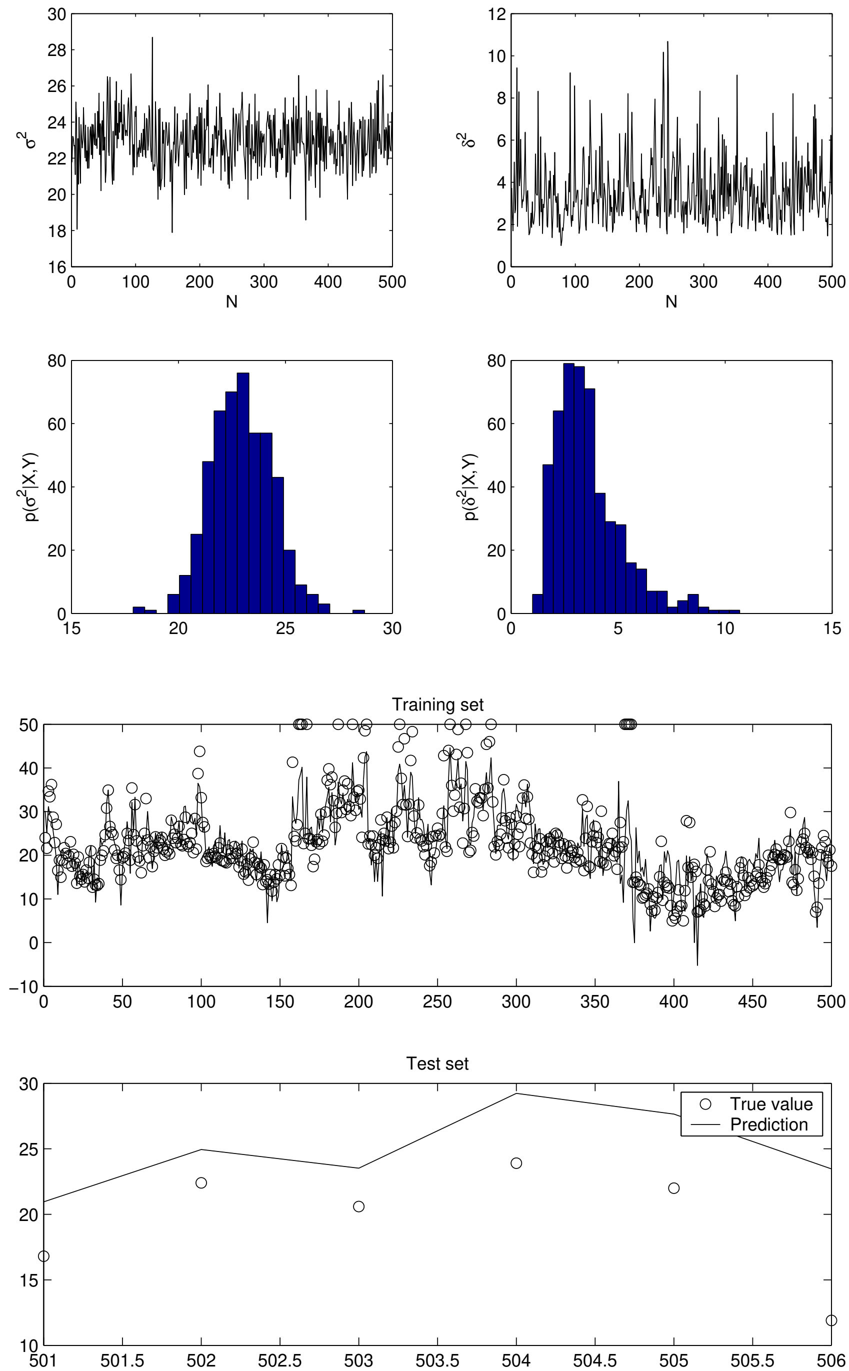
2. **Gibbs Sampling for Linear Regression**:

   Implement the Gibbs sampler for linear regression from page 93 of the lecture notes. Apply it to the problem of predicting house prices in Boston. The code should start with the following lines:

```
echo off; clear;

% LOAD THE DATA AND EDIT IT:
% =========================

data = load('housing.data');      % Load the data.
x = data(:,[1:3 5:13]);            % Input data.
[nn,dold] = size(x);
x = [ones(nn,1) x];               % Add 1 for bias term.
[nn,d] = size(x);                 % d is the dimension of theta
                                  % nn is the total number of data
y = data(:,14);                   % Output data.
```

```
% PRIOR PARAMETERS
% ================
a = 0;
b = 0;
alpha = 2;
beta = 10;


% CREATE THE TRAIN AND TEST SETS:
% ==============================
n = 500;                % Number of training examples.
xTrain = x(1:n,:);      % Training input data.
yTrain = y(1:n,:);      % Training ouput data.
xTest = x(n+1:nn,:);    % Test input data.
yTest = y(n+1:nn,:);    % Test output data.
[n2,tmp] = size(xTest);

% INITIALIZATION:
% ===============
XY = xTrain'*yTrain;
XX = xTrain'*xTrain;
X = xTrain;
Y = yTrain;
N = 500;                % Number of samples.
theta = zeros(d,N);
mu = zeros(d,N);
sigma2 = zeros(N,1);
delta2 = zeros(N,1);
delta2(1) = inv(gengamma(alpha,beta));

for i=1:N,

???
```

Your code should be able to produce the following plots



Compare your training and test errors with the least squares solution.

3. **Multivariate Gaussian-Wishart Model**

   Give the following likelihood:

$$p(Y|X, \theta, \Sigma) = |2\pi\Sigma|^{-\frac{1}{2}} \, e^{-\frac{1}{2}(Y - X\theta)^T \Sigma^{-1}(Y - X\theta)}$$

   and the Wishart prior:

$$p(\Sigma|\alpha, \Sigma^*) \propto |\Sigma|^{-(\alpha+d+1)/2} e^{-\frac{1}{2} trace(\alpha \Sigma^* \Sigma^{-1})},$$

   derive an expression for the posterior distribution of Σ.