ISyE 6416 – HMM Model in Economic Recession Identification

Team Member Names: Ting Sun, Shiyu Zheng, Zhe Liu

Problem Statement

Financial markets provide daily analysis and interpretation on a large set of new microeconomic and macroeconomic information leading them to update their expectations on the future growth path. Theoretically, these alterations must be reflected by the dynamics of asset prices or monetary aggregates. We want to infer economic movements of the United States from those data, especially, the turning point of the economy detected by our analysis. Four "macroeconomic" time series have been selected in this case: the Dow Jones Industrial Average, the Consumer Price Index (CPI), the unemployment rate, and GDP growth rate.

In this project, we develop an extension of the Hidden Markov Model (HMM) to find the hidden economic states of the US behind the data. Specifically, the HMM has a Gaussian mixture at each state as the forecast generator. Gaussian mixtures have desirable features in that they can model series that does not fall into the single Gaussian model. Then the parameters of the HMM are updated in each iteration with an add-drop Expectation-Maximization (EM) algorithm. At each timing point, the parameters of the Gaussian mixtures, the weight of each Gaussian component in the output, and the transformation matrix are all updated dynamically to cater to the non-stationary financial time series data.

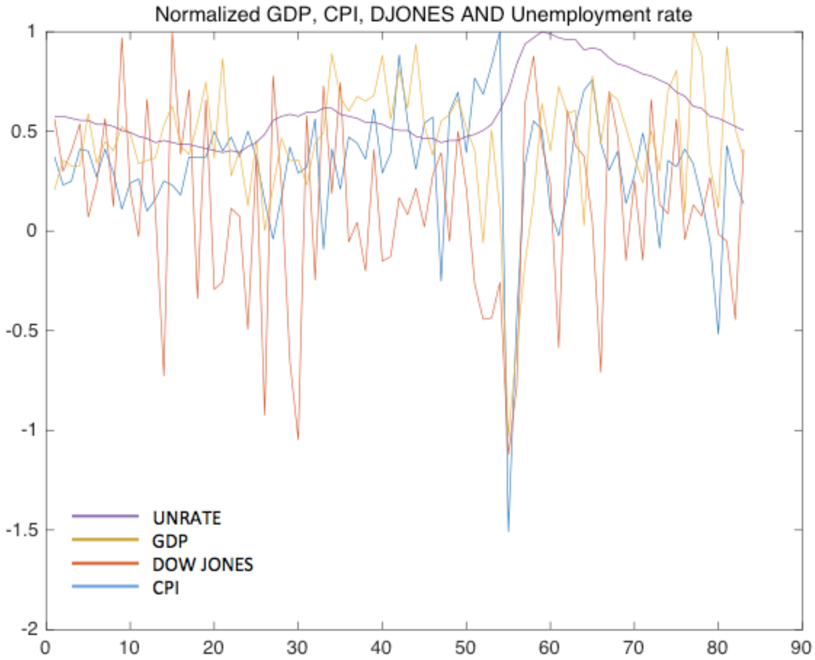
Using the techniques from the EM Theorem, we can prove that the convergence of the proposed algorithm is guaranteed. Therefore we are able to identify the "hidden" economic states, recession and no recession, from the analysis of these macroeconomic time series with the Hidden Markov Model. Then we can validate the result given by our HMM by comparing it with the true recession data posted by the National Bureau of Economic Research (NBER).

Finally, in order to further compare HMM with alternative methods to identify the recession with the macroeconomic variables, we also run a logistic regression with the response variable to be the recession data posted by NBER, and conduct an in sample test on the logistics model.

Data Source

Most economists use the NBER (National Bureau of Economic Research) recession data as a reference. We also compare our outcome with NBER recession data in this project. The four "macroeconomic" time series: the Dow Jones Industrial Average, the Consumer Price Index (CPI), the unemployment rate, and GDP growth rate are downloaded from the Federal Reserve Economic Data. For the Dow Jones Industrial Average and the Consumer Price Index, these two variables are expressed as percentage changes over time in order for them to be stationary. See figure 1 for time series plot of four macroeconomic variables.

In history there a two recessions during our data period: The early 2000s recession, from Mar 2001-Nov 2001, and the Great Recession, Dec 2007-June 2009. Our time period of interest is from Jan 1995 to Oct 2015, which cover the most two recessions. Data frequency is quarter.



Plot 1

Methodology

The general HMM approach framework is an unsupervised learning technique, which allows new patterns to be found. It can handle variable lengths of input sequences, without training a model.

We use HMM because the underlying macroeconomic system being modeled is assumed to be a Markov process with unobserved states. We are trying to identify hidden recession state with four groups of observable data: CPI change, Dow Jones Industrial Average, GDP growth and unemployment rate. These four indexes are chosen as representative of economic status.

We then implied HMM under assumption that observable states are following a Gaussian Mixture distribution. These four indexes can be thought of generated by underlying recession or boom state. We use Gaussian Mixture instead of Gaussian because it is more realistic. The estimation result also performs well when we assume the distribution is a mixture of three Gaussian. As a result, the transition probabilities as well as the observation generation probability density function are both adjustable, which gives us more power to fit the model. Given plenty of data that are generated by some hidden power, we can create an HMM architecture and use EM algorithm to find out the best model parameters that account for the observed data.

In statistics, an expectation–maximization (EM) algorithm is an iterative method for finding maximum likelihood or maximum a posteriori estimates of parameters. Because EM gives us local maximum, every time we will get a different estimation of Gaussian Mixture parameters. To lower the error, we take average of 500 trials.

After getting parameters, we use the Viterbi algorithm to find the most likely hidden state sequence, $\mathrm{S}' = (\mathrm{s}_1, \mathrm{s}_2 \dots \mathrm{s}_\mathrm{T})$, given three observed data sets (here we take one as example) $\mathrm{O} = (\mathrm{o}_1, \mathrm{o}_2 \dots \mathrm{o}_\mathrm{T})$, from $\mathrm{argmax}_{\mathrm{S}'} \mathrm{P}(\mathrm{S}', \mathrm{O}|\mu)$. Denote he following variable $\delta_\mathrm{j}(\mathrm{x}) = \mathrm{max}_{\mathrm{s}_1,\mathrm{s}_2 \dots \mathrm{s}_{\mathrm{t}-1}} \mathrm{P}(\mathrm{s}_1, \mathrm{s}_2 \dots \mathrm{s}_{\mathrm{t}-1}, \mathrm{o}_1, \mathrm{o}_2 \dots \mathrm{o}_{\mathrm{t}-1}, \mathrm{s}_\mathrm{t} = \mathrm{j}|\mu)$. This variable stores the probability of observing $\mathrm{o}_1, \mathrm{o}_2 \dots \mathrm{o}_\mathrm{t}$ using the most likely path that ends in state i at time t given the model μ. The corresponding variable $\psi_\mathrm{j}(\mathrm{t})$ stores the node of the incoming arc that leads to this most probable path.

The calculation is done by induction, similar to the forward-backward algorithm, except that the forward-backward algorithm uses summing over previous states while the Viterbi algorithm uses maximization. The complete Viterbi algorithm is as follows ($b_i(\mathrm{o}), a_{ij}\ as\ in\ class$):

1. Initialization: $\delta_j(1) = \pi_i b_i(o_1), \psi_j(1) = 0, j = 1 \dots N$.

2. Induction: $\delta_j(t) = b_{ijo_t} \mathrm{max}_{1\le \mathrm{i} \le \mathrm{N}}\, \delta_i(t-1)\, a_{ij}, \psi_j(t) = arg \max_{1\le \mathrm{i} \le \mathrm{N}} \delta_i(t-1)\, a_{ij}$.

3. Update time: $t = t + 1$ while $t \le T$ or turn to step 4.

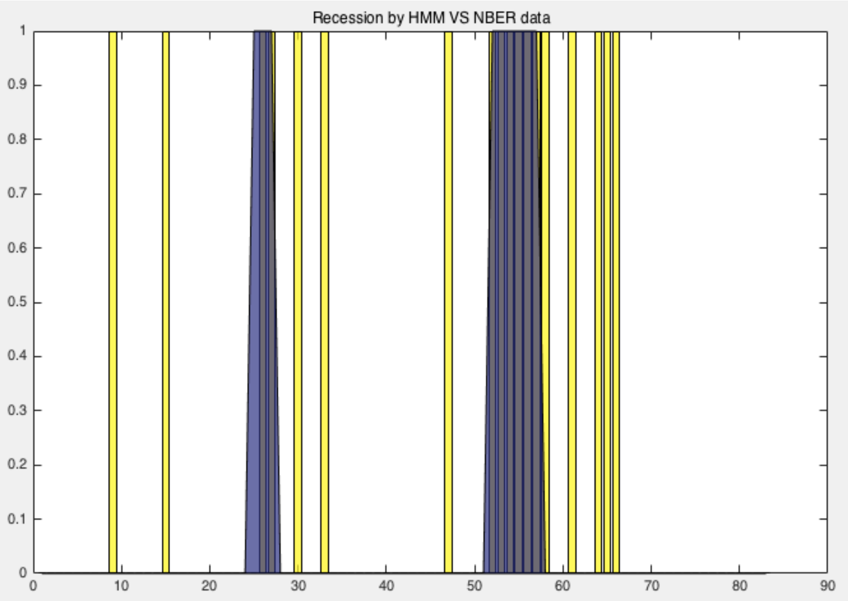
4. Termination: find optimal path $\mathrm{s}_T^* = arg \max_{1\le \mathrm{i} \le \mathrm{N}} \delta_i(T)$. Read out the path.

Then there are multiple paths generated by the Viterbi algorithm. Take the average of path (hidden state series) as our final estimated recession Markov chain. When state greater than 0.5, approximate it to 1(recession); when it is less than 0.5, approximate it to 0 (non-recession).

Finally, compare our result with NBER data.

Results

As we have applied a multivariate qualitative hidden Markov model to four data sets, it provides a reliable framework to track the growth cycle and predict economic downturns. See result in figure 2 for the estimated economic recession with real recession data. HMM accurately identified the recession state, but seems like over estimating recession period, probably because the economic was still not that good enough. The state of recovery from a recession can be mistakenly read as recession state, since we still have a rather low employment rate and GDP growth rate, comparing to economy state outside a recession. This recovery state may not be recognized as recession as defined by NBER, but it can still be a labeled recession time in our model. However, as we can see in plot 1, there are still some mislabeled recession states that cannot be explained in this way. Three years are labeled as recession before the recession actually happened, and there are several gaps between those years and the actual recession denoted by NBER. This might be generated by the business cycle of the economy, or inaccurate choice of predicting variables.

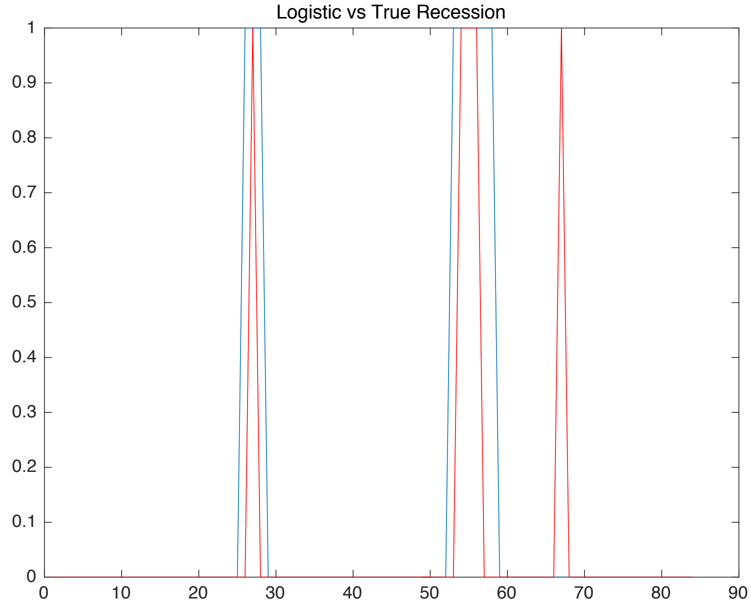


Plot 2

We may blame our unsatisfying results due to these reasons: not perfectly chosen Gaussian Mixture model as observation generator, other hidden forces behind the data that disrupt our judgment, or flaw of the EM algorithm that sometimes got trapped in local maximum. We are trying to eliminate the influence of local maximum instead of global maximum by running our path generation function for many times. Each time we choose prior distribution of states and transmission probability matrix randomly, and regenerate EM algorithm parameters for 500 times. For each of those random generations, we set the maximum iteration number of EM algorithm to 5 in order to shorten our running time.

For comparison, we also performed a simple logistic regression to identify economic downturns. It turns out that logistic regression generally underestimates the length of true recession period. Interestingly, both HMM and logistic regression state that there are signs of economic downturn even in 2010, just as we can observe in plot 3 below. Also, this simple logistic regression can provide further evidence of the significance of our variable choice. For those four variables, we got a rather satisfying significant level. Statistically, we can say that the four variables chosen are closely related to the hidden state of US economy.

In a short summary, we have built a rather reliable frame for recession identification in the US. We can run our program for several times and our success rate (defined as the successfully marked label numbers divided by total label numbers) is between 85 percent to 90 percent each time.



Plot 3

Further Discussion

First we will discuss the advantages and disadvantages of HMM model in this specific problem.

HMM model can handle variations in a pre-specified structure, that is, we can also use our model to analyze data in the future with minor changing of our program. To validate our model, we can also test newly generated macroeconomic data as back testing.

However, HMM model cannot give us the most efficient number of hidden states to use or the observations to choose. We can only estimate the parameters using those pre-specified data. This can easily lead to unsatisfying outcome if we do not do cross-validation or back testing of HMM model. If we change the scope of time in our model, it is possible that the original variables we chose are not performing as efficiently. In this case, we need to constantly update our observation variables if we want to further elaborate on this model. We have compared our outcome with a simple logistic regression model. We can also run AIC and BIC in choosing the most related variables.

Appendix

List of recessions in the United States, Wikipedia,
https://en.wikipedia.org/wiki/List_of_recessions_in_the_United_States

Hidden Markov Model (HMM) Toolbox for Matlab,
http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html

Matlab Code:

```
clc
close all
clear all
load 'cpi.mat';
load 'djones.mat';
load 'gdp.mat';
load 'unrate.mat';
load 'recession.mat';

addpath(genpath('/Users/tsun/Desktop/HMMall'))

% Process the difference of CPI, DJONES and GDP
% Normalize 4 variables
CPI = CPI(2:end) - CPI(1:end-1);
CPI = CPI./max(CPI);
DJONES = DJONES(2:end);
DJONES = DJONES./max(DJONES);
GDP = GDP(2:end) - GDP(1:end-1);
GDP = GDP./max(GDP);
UNRATE = UNRATE(2:end);
UNRATE = UNRATE./max(UNRATE);
X=(1:1:83);
figure()
plot(X,CPI,X,DJONES,X,GDP,X,UNRATE)

title('Normalized GDP, CPI, DJONES AND Unemployment rate')

% Initialization
data = transpose([CPI,DJONES,GDP,UNRATE]);
nex = 1;
O = 4;
num = 500;
flag = 0;

% For the purpose of accuracy, we iteratively run our algorithm and get
% total path of recession
path_total = zeros(1,83);
for i = 1:num
    path = f(data);
    if path == ones(1,83);
        flag = flag;
    else
        flag = flag+1;
    end
    path = path-1;
    err = 0;
    for j = 1:83
        err = err + abs(RECESSION(j+1)-path(j));
    end
    if err/83>0.5
        path = 1-path;
    end
    path_total = path_total + path;
end

% Process the err based on total path
path_total = path_total./flag;
path = (path_total > 0.50);

err = 0;
for i = 1:83
    err = err + abs(RECESSION(i+1)-path(i));
end
err = err/83
figure()
bar((1:1:83),path,'y')
hold on
area((1:1:83),transpose(RECESSION(2:end)))
hold off
alpha(.7)
title('Recession by HMM VS NBER data')



function [ path ] = f( data )
%Now let use fit a mixture of M=2 Gaussians for each of the Q=2 states using
K-means.
O = 4;
M = 3;
Q = 2;
left_right = 0;
prior0 = normalise(rand(Q,1));
transmat0 = mk_stochastic(rand(Q,Q));
[mu0, Sigma0] = mixgauss_init(Q*M, data,'full');
mu0 = reshape(mu0, [O Q M]);
Sigma0 = reshape(Sigma0, [O O Q M]);
mixmat0 = mk_stochastic(rand(Q,M));

%Finally, let us improve these parameter estimates using EM. Then we can
%estimate hidden state path
[LL, prior1, transmat1, mu1, Sigma1, mixmat1] = ...
    mhmm_em(data, prior0, transmat0, mu0, Sigma0, mixmat0, 'max_iter', 5);
loglik = mhmm_logprob(data, prior1, transmat1, mu1, Sigma1, mixmat1);
B = mixgauss_prob(data, mu1, Sigma1, mixmat1);

% If loglikelihood function ends in positive number, we set path to default
% and we will not use this outcome
if LL(end)<0
    path = viterbi_path(prior1, transmat1, B);
else
    path = ones(1,83);
end

end
```

```
clc
clear all
close all

load('cpi.mat')
load('djones.mat')
load('gdp')
load('unrate.mat')
load('recession')

Independent=[CPI DJONES GDP UNRATE];

B = glmfit(Independent,RECESSION,'binomial');
pihat=mnrval(B,Independent);

FitRecession=1*(pihat(:,1)>0.4);
Rate=sum(FitRecession==RECESSION)/length(RECESSION)

plot(RECESSION)
hold on
plot(FitRecession,'r')
% legend('True','Logistic','Location','northwest')
hold off
title('Logistic vs True Recession')
```