3.5(a)

MILLION: 0.00207234

MORE: 0.00170855

MR.: 0.00144132

MOST: 0.000787758

MARKET: 0.000780213

MAY: 0.000729749

M.: 0.000703264

MANY: 0.000696588

MADE: 0.000559748

MUCH: 0.000514493

MAKE: 0.000514358

MONTH: 0.000444819

MONEY: 0.000437018

MONTHS: 0.000405679

MY: 0.000400237

MONDAY: 0.000381908

MAJOR: 0.000370817

MILITARY: 0.000351974

MEMBERS: 0.000335993

MIGHT: 0.000273534

MEETING: 0.000265684

MUST: 0.000266454

ME: 0.000263519

MARCH: 0.000259741

MAN: 0.000252832

MS.: 0.000238942

MINISTER: 0.000239724

MAKING: 0.000211662

MOVE: 0.000209913

MILES: 0.000205927

3.5(b)

1st most likely word: <UNK> with bigram probability: 0.61502

2st most likely word: U. with bigram probability: 0.0133725

3st most likely word: FIRST with bigram probability: 0.0117203

4st most likely word: COMPANY with bigram probability: 0.0116588

5st most likely word: NEW with bigram probability: 0.00945148

6st most likely word: UNITED with bigram probability: 0.00867231

7st most likely word: GOVERNMENT with bigram probability: 0.00680349

8st most likely word: NINETEEN with bigram probability: 0.00665071

9st most likely word: SAME with bigram probability: 0.00628707

10st most likely word: TWO with bigram probability: 0.00616075

3.5(c)

log likelihood under unigram: -64.5115

log likelihood under bigram: -40.7006

The bigram model generates the highest log-likelihood.

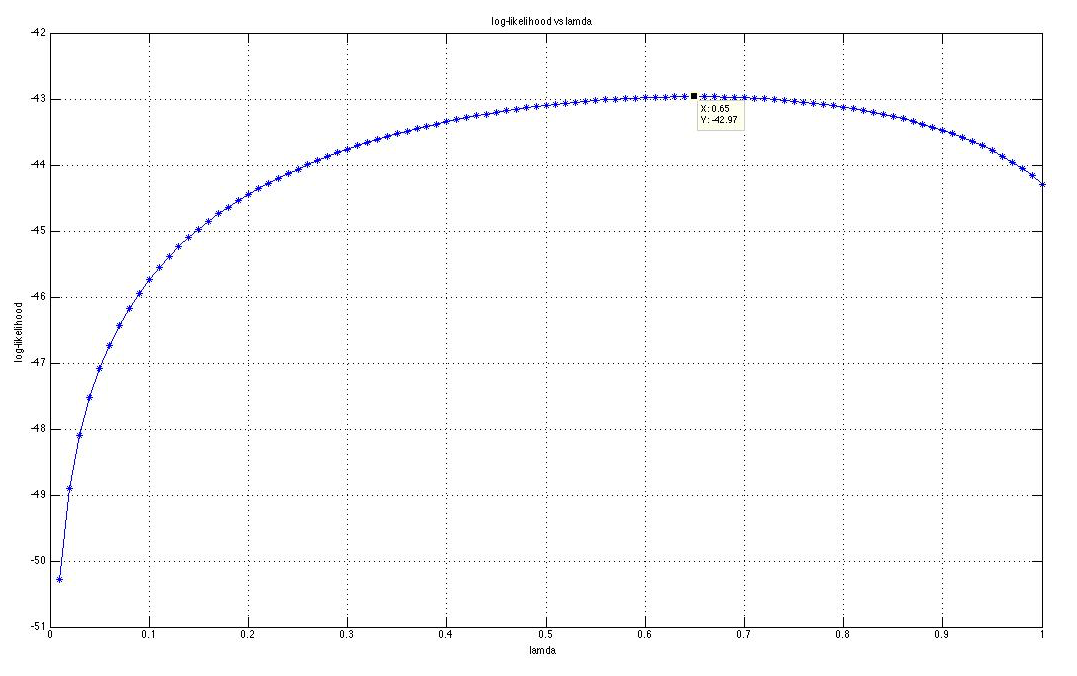
3.5(d)

log likelihood under unigram: -44.2932

log likelihood under bigram: neg-infinity

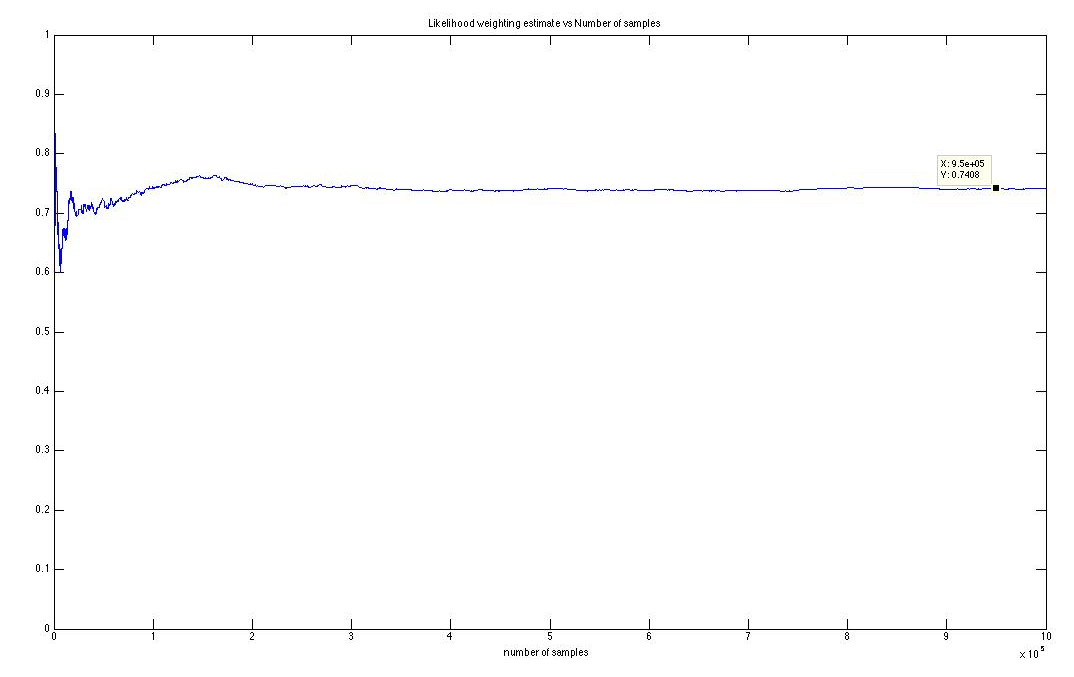
The adjacent word pair (SIXTEEN, OFFICIALS) is not observed in the training corpus. This will cause the multiplication of the probabilities to be 0. Consequently, the log-likelihood will be negative infinity.

3.5(e)



The lamda value that gives the biggest log-likelihood is 0.65.

3.2 (c)



From the plot we can see that the estimation converges to the correct value. The difference is less than 0.01.

(d)Code:

// Code Created for CSE 250A Homework 3 Question 2

// Creator: Jingyuan Li with All rights reserved.

// 10-18-2015 At UCSD

#include <iostream>

#include <cmath>

#include <vector>

using namespace std;

int main(){

srand(time(NULL));

int z = 64, n = 10, m = 1000000;

vector<double> res;

double alpha = 0.35, sum = 0, sum\_b7 = 0, p\_zb;

for (int i = 0; i < m; i++){

int num = rand() % int(pow(2,n));

p\_zb = (1-alpha)/(1+alpha)\*pow(alpha, abs(z-num));

sum += p\_zb;

if ((num >> 6) & 1) sum\_b7 += p\_zb;

res.push\_back(sum\_b7/sum);

}

for (int i = 0; i < res.size(); i++){

cout << res[i] << endl;

}

cout << endl;

}

3.3

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Y1 | Y2 | Y3 | Y | P(Y|X = 0) | P(Y|X = 1) | P(Z1 = 1|Y) | P(Z2 = 1|Y) |
| 0 | 0 | 0 | 1 | 0.09375 | 0.09375 | 0.9 | 0.1 |
| 1 | 0 | 0 | 2 | 0.28125 | 0.09375 | 0.8 | 0.2 |
| 0 | 1 | 0 | 3 | 0.09375 | 0.03125 | 0.7 | 0.3 |
| 0 | 0 | 1 | 4 | 0.03125 | 0.28125 | 0.6 | 0.4 |
| 1 | 1 | 0 | 5 | 0.28125 | 0.03125 | 0.5 | 0.5 |
| 1 | 0 | 1 | 6 | 0.09375 | 0.28125 | 0.4 | 0.6 |
| 0 | 1 | 1 | 7 | 0.03125 | 0.09375 | 0.3 | 0.7 |
| 1 | 1 | 1 | 8 | 0.09375 | 0.09375 | 0.2 | 0.8 |