

## 7.4

8 мая 2016 г.

### 1 Задача 4

```
In [18]: %matplotlib inline
import numpy as np
import math as mt
import matplotlib
import matplotlib.pyplot as plt
from pylab import *
from scipy.stats import *
```

```
In [19]: data = np.loadtxt('6.csv', delimiter=',', skiprows=3)
print(len(data))
```

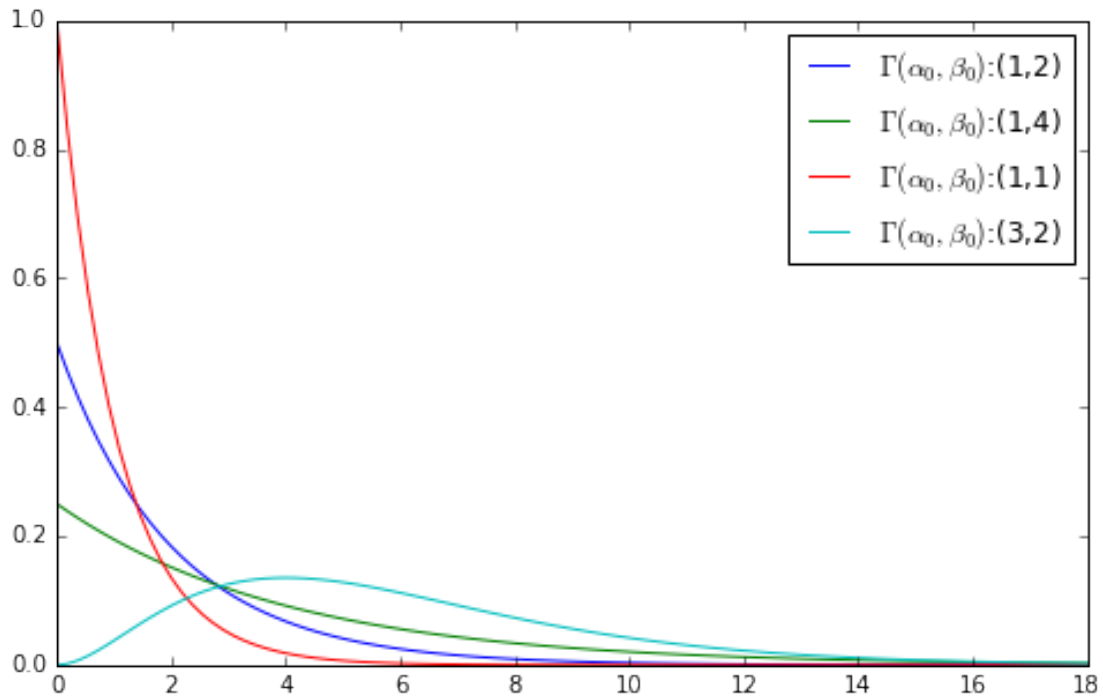
1000

Для  $Exp(\theta)$  априорное распределение  $\Gamma(\alpha_0, \beta_0)$  with mean =  $\frac{\alpha_0}{\beta_0}$ .  
Байесовская оценка для параметра  $\theta$ :  $\theta^* = \frac{\alpha_0 + n}{\beta_0 + \sum_{i=1}^n x_i}$ .

```
In [20]: def BayesEst(x, alpha, beta):
# Байесовская оценка для экспоненциального распределения
return (alpha + len(x))/(beta + sum(x))
```

Задача состоит в том, чтобы верно подобрать параметры  $\alpha_0$   $\beta_0$ .

```
In [21]: x = linspace(0,18,100)
params = [(1,2),(1,4),(1,1),(3,2)]
figure(figsize=(8,5))
for p in params:
    plot(x, gamma.pdf(x, p[0], scale=p[1]), \
         label='$\Gamma(\alpha_0, \beta_0)$'.format(p[0], p[1]))
legend()
show()
```



Поскольку в начальный момент времени мы ничего не знаем о “скорости выхода из строя” серверов, то нам не подходят параметры Гамма распределения, при которых образуется явный “горб” над каким-то конкретным числом. Нам следует этого избегать.

Значит, нужно брать параметр  $\alpha_0 = 1$ , а за параметр  $\beta_0$  можно взять число 2, при котором график плотности выглядит не очень крутым.

```
In [22]: # Истинное значение theta
lmd = 1./66.
t_0 = 200
t = 70000
param = (1,1)

# Сюда запишу время между i-м моментом выхода из строя сервера и (i+1)-м,
# то есть delta - график "разности времен"
delta = []
for i in range(len(data)-1):
    delta.append(data[i+1] - data[i])

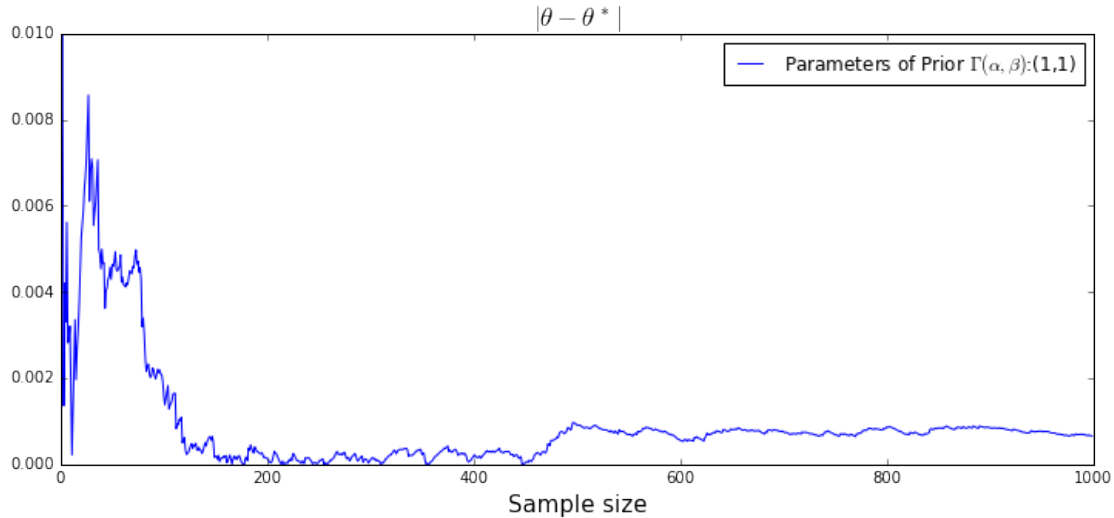
# Сюда запишу оценки для параметра theta
est = np.array([BayesEst(delta[(j+1)], param[0], param[1]) for j in range(len(data))])

# Строю график модуля разности истинного значения theta и байесовской оценки
figure(figsize=(12,5))
ylim((0,0.01))
xlabel("Sample size", fontsize=15)
```

```

plot(abs(lmd-est), label="Parameters of Prior  $\Gamma(\alpha, \beta):(\{ \}, \{ \})"$ \
      .format(param[0], param[1]))
title(" $|\theta - \theta^*|$ ", fontsize=17)
legend()
show()

```



Исправлена задача 6.1 под новые условия.

Теперь вместо  $\lambda$  будет стоять оценка  $\theta^*$  в выражении для предсказанного числа серверов, которые необходимо заменить.

```

In [23]: condition_expect = [] # Здесь будут предсказанные значения количества серверов
time = [] # Сюда записываю время с шагом t_0
quant = [] # Количество серверов, вышедших из строя, на данный момент

plt.figure(figsize=(18,7))
for i in range(t/t_0):
    time.append(t_0*(i+1))
    quant.append(shape(np.where(data <= time[-1]))[1]) # Нахожу это количество подсчетом
                                                         # числа записей времени, меньших
                                                         # текущего времени
    condition_expect.append(((t-time[-1])*est[quant[-1]-1]) + quant[-1]) # Предсказанное значение

plt.xlabel('Time', fontsize=15)
plt.ylabel('Number of Servers to Buy', fontsize=15)
plt.plot(time,condition_expect)
plt.grid()
plt.show()

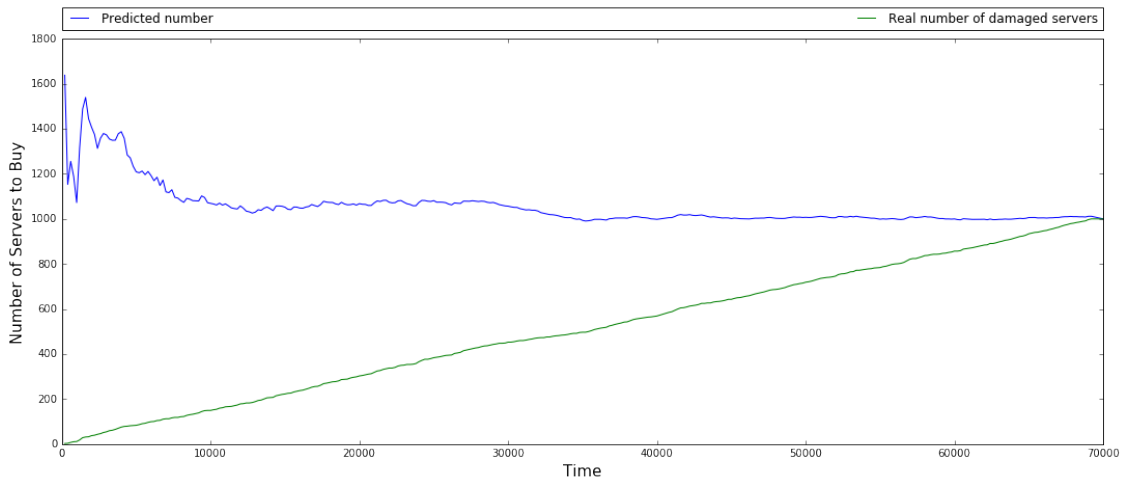
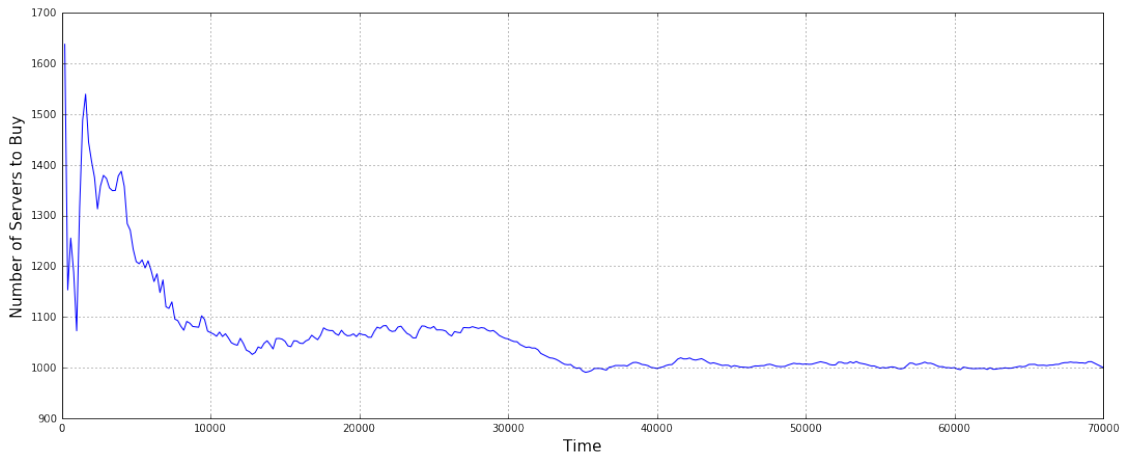
plt.figure(figsize=(18,7))

```

```

plt.xlabel('Time', fontsize=15)
plt.ylabel('Number of Servers to Buy', fontsize=15)
plt.plot(time, condition_expect, label='Predicted number')
plt.plot(time, quant, color='g', label='Real number of damaged servers')
plt.legend(bbox_to_anchor=(0., 1.02, 1., .102), loc=3, ncol=2, \
           mode="expand", borderaxespad=0.)
plt.show()

```



Каждые  $t_0 = 200$  секунд вывожу значение предсказания.  
В самом конце оно совпадает с действительным значением  $N = 1000$ .

```
In [24]: print('\n'.join(map(str, condition_expect)))
```

```
1637.53218541
1153.03741702
```

1255.27968819  
1187.74150019  
1072.48625837  
1315.534249  
1487.65375756  
1539.15326911  
1444.61244213  
1407.20322234  
1374.9692205  
1312.95279233  
1357.95933558  
1379.01055923  
1373.02480774  
1354.47769677  
1348.88199883  
1349.32507889  
1378.02031752  
1387.04126344  
1357.34317399  
1283.72360534  
1270.88084309  
1233.23047978  
1208.8747368  
1204.76539754  
1212.25979855  
1196.38333281  
1210.34100193  
1192.01605338  
1169.53196834  
1184.84471033  
1147.99999733  
1172.70369785  
1119.84208215  
1116.64575808  
1129.45549982  
1095.53430323  
1092.40438559  
1081.59282461  
1073.55991889  
1090.94357688  
1087.5419352  
1080.95431648  
1080.30969453  
1079.47476452  
1101.92897054  
1094.81908763  
1071.89473007  
1068.83195688

1066.08336161  
1061.71514547  
1069.86680594  
1061.18454279  
1066.61571659  
1058.24015512  
1048.86526347  
1045.80386846  
1043.98311233  
1057.58813311  
1047.52966041  
1034.36631384  
1031.41018081  
1025.80657021  
1029.63575163  
1040.5199748  
1037.98811107  
1047.77462691  
1052.75018909  
1045.41051675  
1036.62244471  
1056.95642979  
1057.43397135  
1056.07232912  
1051.92193683  
1042.69081891  
1041.13822362  
1052.74677311  
1052.25419942  
1048.0318878  
1047.292533  
1052.42608995  
1055.11188391  
1063.6500488  
1058.92335546  
1054.90349311  
1063.90408194  
1078.27333432  
1074.49258961  
1073.017345  
1072.87369082  
1066.6862453  
1063.75739496  
1073.46053908  
1066.23371487  
1062.58847134  
1063.21471907  
1066.39934783

1061.09239095  
1067.53436155  
1064.96232345  
1064.35180882  
1059.73265165  
1059.65736854  
1071.15640555  
1079.67589119  
1077.2102095  
1082.23363108  
1082.75918455  
1074.17558215  
1071.10818389  
1072.06253135  
1079.99694773  
1081.53308483  
1073.90288294  
1067.32000163  
1064.27162555  
1058.4840107  
1058.38634462  
1072.91793695  
1082.0340234  
1081.55143546  
1078.46129759  
1077.4356959  
1081.08275234  
1074.09248291  
1074.32211111  
1073.88438613  
1072.00123651  
1065.73496577  
1062.20861233  
1070.91562865  
1069.38721919  
1068.61223351  
1078.85393447  
1078.47369516  
1078.25626462  
1080.56144996  
1078.73387947  
1077.12051743  
1078.68259683  
1077.79547962  
1073.58600162  
1071.64756419  
1073.06141375  
1068.69597412

1063.28134213  
1060.25039956  
1057.60970988  
1056.42380273  
1053.40668372  
1051.24393463  
1050.83384043  
1045.40826049  
1042.42148365  
1039.65563289  
1040.52626917  
1038.43086627  
1038.41870275  
1035.03809996  
1027.75029711  
1024.81511036  
1021.97742263  
1019.05775727  
1018.40572978  
1016.38836954  
1013.34192055  
1009.48589474  
1006.36154095  
1005.50030516  
1005.93643581  
1001.2223301  
998.361530492  
999.481280192  
993.396252914  
990.559702897  
991.774072513  
993.988342525  
998.064014669  
998.236023331  
998.170020862  
996.422856063  
994.905741475  
1000.72244128  
1001.66844371  
1003.88084746  
1003.95107679  
1003.77045993  
1004.05637576  
1002.97214727  
1006.59787924  
1009.7919795  
1010.18932062  
1008.63144056



1005.99851357  
1005.28077036  
1003.6552708  
1000.26125649  
999.442289679  
998.009817468  
999.784775159  
1001.19179955  
1003.79319246  
1005.47482802  
1005.75758684  
1010.62697575  
1016.64906753  
1019.14562281  
1016.92819327  
1017.0246319  
1018.7980393  
1015.86865911  
1014.65891127  
1015.94158562  
1017.45200236  
1014.54495049  
1010.80946875  
1007.92368327  
1009.31736825  
1007.70475918  
1005.824623  
1004.05224038  
1004.65045195  
1004.38267271  
1001.51455626  
1003.67444624  
1002.61980423  
1000.9592655  
1000.76709406  
1000.25955171  
999.831831068  
1001.5877666  
1003.12128079  
1002.97057966  
1003.59331007  
1003.68542244  
1005.75260655  
1006.45990709  
1004.79410654  
1002.69396436  
1001.99615728  
1001.80604878

1002.22681339  
1004.78919392  
1006.52987126  
1008.43604664  
1007.36397634  
1007.58637826  
1006.3431899  
1007.06004439  
1006.31839394  
1006.54314271  
1008.20769178  
1009.86891271  
1011.39599099  
1010.04007255  
1008.7603907  
1006.04830341  
1004.92564485  
1005.20643382  
1010.57356421  
1010.43071122  
1008.24008025  
1008.33009951  
1011.27780097  
1008.8507814  
1011.78384689  
1008.89488488  
1007.68733027  
1006.63286746  
1004.99050015  
1003.14801414  
1003.0859268  
1000.7838727  
998.681277921  
1000.12780621  
999.051662943  
1000.41287069  
1001.34672491  
1000.37988767  
998.138038252  
997.268516275  
998.729728798  
1003.79762101  
1008.83424457  
1008.41052355  
1005.52910911  
1006.66876227  
1008.28819459  
1010.68431017

1008.38393299  
1008.45948833  
1006.58658146  
1003.71664143  
1001.56908378  
1001.73271144  
999.73263592  
999.761723351  
999.026550962  
999.885144872  
997.027441975  
995.941643463  
1000.78065978  
999.640266853  
998.557206944  
997.819097696  
997.768510279  
998.159132052  
998.084017588  
998.43736513  
996.262118312  
999.401752572  
996.549074873  
996.862283673  
998.060485952  
998.234719159  
999.405844607  
998.559507142  
998.678310619  
999.889022337  
1001.15099794  
1002.44455634  
1001.40807682  
1002.70554151  
1006.04968118  
1006.21740522  
1006.31939304  
1004.23362275  
1004.43592685  
1004.55464309  
1003.64933163  
1004.77452223  
1004.92738586  
1006.13397922  
1006.25871127  
1008.44815446  
1009.60944038  
1009.72732339

```
1010.8597944
1009.92706171
1010.00343886
1009.14078857
1009.23980826
1008.34065498
1011.4700524
1011.59248778
1008.69436584
1005.79624389
1002.89812195
1000.0
```

```
In [25]: for i in range(len(est)):
          print("lambda = {:.4f}, {:.4f} - bayes estimate, {}/{}".format(lmd, est[i], i+1, len(est)))
```

```
lambda = 0.0152, 0.2476 - bayes estimate, 1/1000
lambda = 0.0152, 0.0962 - bayes estimate, 2/1000
lambda = 0.0152, 0.0234 - bayes estimate, 3/1000
lambda = 0.0152, 0.0165 - bayes estimate, 4/1000
lambda = 0.0152, 0.0193 - bayes estimate, 5/1000
lambda = 0.0152, 0.0185 - bayes estimate, 6/1000
lambda = 0.0152, 0.0208 - bayes estimate, 7/1000
lambda = 0.0152, 0.0180 - bayes estimate, 8/1000
lambda = 0.0152, 0.0180 - bayes estimate, 9/1000
lambda = 0.0152, 0.0184 - bayes estimate, 10/1000
lambda = 0.0152, 0.0170 - bayes estimate, 11/1000
lambda = 0.0152, 0.0154 - bayes estimate, 12/1000
lambda = 0.0152, 0.0164 - bayes estimate, 13/1000
lambda = 0.0152, 0.0175 - bayes estimate, 14/1000
lambda = 0.0152, 0.0185 - bayes estimate, 15/1000
lambda = 0.0152, 0.0171 - bayes estimate, 16/1000
lambda = 0.0152, 0.0179 - bayes estimate, 17/1000
lambda = 0.0152, 0.0183 - bayes estimate, 18/1000
lambda = 0.0152, 0.0188 - bayes estimate, 19/1000
lambda = 0.0152, 0.0196 - bayes estimate, 20/1000
lambda = 0.0152, 0.0204 - bayes estimate, 21/1000
lambda = 0.0152, 0.0207 - bayes estimate, 22/1000
lambda = 0.0152, 0.0210 - bayes estimate, 23/1000
lambda = 0.0152, 0.0215 - bayes estimate, 24/1000
lambda = 0.0152, 0.0218 - bayes estimate, 25/1000
lambda = 0.0152, 0.0223 - bayes estimate, 26/1000
lambda = 0.0152, 0.0229 - bayes estimate, 27/1000
lambda = 0.0152, 0.0237 - bayes estimate, 28/1000
lambda = 0.0152, 0.0213 - bayes estimate, 29/1000
lambda = 0.0152, 0.0219 - bayes estimate, 30/1000
lambda = 0.0152, 0.0222 - bayes estimate, 31/1000
```

lambda = 0.0152, 0.0220 - bayes estimate, 32/1000  
 lambda = 0.0152, 0.0207 - bayes estimate, 33/1000  
 lambda = 0.0152, 0.0210 - bayes estimate, 34/1000  
 lambda = 0.0152, 0.0212 - bayes estimate, 35/1000  
 lambda = 0.0152, 0.0217 - bayes estimate, 36/1000  
 lambda = 0.0152, 0.0222 - bayes estimate, 37/1000  
 lambda = 0.0152, 0.0201 - bayes estimate, 38/1000  
 lambda = 0.0152, 0.0200 - bayes estimate, 39/1000  
 lambda = 0.0152, 0.0197 - bayes estimate, 40/1000  
 lambda = 0.0152, 0.0201 - bayes estimate, 41/1000  
 lambda = 0.0152, 0.0198 - bayes estimate, 42/1000  
 lambda = 0.0152, 0.0198 - bayes estimate, 43/1000  
 lambda = 0.0152, 0.0188 - bayes estimate, 44/1000  
 lambda = 0.0152, 0.0192 - bayes estimate, 45/1000  
 lambda = 0.0152, 0.0192 - bayes estimate, 46/1000  
 lambda = 0.0152, 0.0195 - bayes estimate, 47/1000  
 lambda = 0.0152, 0.0195 - bayes estimate, 48/1000  
 lambda = 0.0152, 0.0197 - bayes estimate, 49/1000  
 lambda = 0.0152, 0.0194 - bayes estimate, 50/1000  
 lambda = 0.0152, 0.0198 - bayes estimate, 51/1000  
 lambda = 0.0152, 0.0197 - bayes estimate, 52/1000  
 lambda = 0.0152, 0.0199 - bayes estimate, 53/1000  
 lambda = 0.0152, 0.0201 - bayes estimate, 54/1000  
 lambda = 0.0152, 0.0197 - bayes estimate, 55/1000  
 lambda = 0.0152, 0.0196 - bayes estimate, 56/1000  
 lambda = 0.0152, 0.0197 - bayes estimate, 57/1000  
 lambda = 0.0152, 0.0197 - bayes estimate, 58/1000  
 lambda = 0.0152, 0.0200 - bayes estimate, 59/1000  
 lambda = 0.0152, 0.0194 - bayes estimate, 60/1000  
 lambda = 0.0152, 0.0195 - bayes estimate, 61/1000  
 lambda = 0.0152, 0.0193 - bayes estimate, 62/1000  
 lambda = 0.0152, 0.0193 - bayes estimate, 63/1000  
 lambda = 0.0152, 0.0193 - bayes estimate, 64/1000  
 lambda = 0.0152, 0.0194 - bayes estimate, 65/1000  
 lambda = 0.0152, 0.0193 - bayes estimate, 66/1000  
 lambda = 0.0152, 0.0195 - bayes estimate, 67/1000  
 lambda = 0.0152, 0.0196 - bayes estimate, 68/1000  
 lambda = 0.0152, 0.0196 - bayes estimate, 69/1000  
 lambda = 0.0152, 0.0196 - bayes estimate, 70/1000  
 lambda = 0.0152, 0.0197 - bayes estimate, 71/1000  
 lambda = 0.0152, 0.0197 - bayes estimate, 72/1000  
 lambda = 0.0152, 0.0200 - bayes estimate, 73/1000  
 lambda = 0.0152, 0.0201 - bayes estimate, 74/1000  
 lambda = 0.0152, 0.0198 - bayes estimate, 75/1000  
 lambda = 0.0152, 0.0199 - bayes estimate, 76/1000  
 lambda = 0.0152, 0.0196 - bayes estimate, 77/1000  
 lambda = 0.0152, 0.0197 - bayes estimate, 78/1000  
 lambda = 0.0152, 0.0194 - bayes estimate, 79/1000

lambda = 0.0152, 0.0183 - bayes estimate, 80/1000  
 lambda = 0.0152, 0.0185 - bayes estimate, 81/1000  
 lambda = 0.0152, 0.0182 - bayes estimate, 82/1000  
 lambda = 0.0152, 0.0176 - bayes estimate, 83/1000  
 lambda = 0.0152, 0.0173 - bayes estimate, 84/1000  
 lambda = 0.0152, 0.0174 - bayes estimate, 85/1000  
 lambda = 0.0152, 0.0175 - bayes estimate, 86/1000  
 lambda = 0.0152, 0.0172 - bayes estimate, 87/1000  
 lambda = 0.0152, 0.0172 - bayes estimate, 88/1000  
 lambda = 0.0152, 0.0172 - bayes estimate, 89/1000  
 lambda = 0.0152, 0.0174 - bayes estimate, 90/1000  
 lambda = 0.0152, 0.0174 - bayes estimate, 91/1000  
 lambda = 0.0152, 0.0172 - bayes estimate, 92/1000  
 lambda = 0.0152, 0.0171 - bayes estimate, 93/1000  
 lambda = 0.0152, 0.0172 - bayes estimate, 94/1000  
 lambda = 0.0152, 0.0174 - bayes estimate, 95/1000  
 lambda = 0.0152, 0.0173 - bayes estimate, 96/1000  
 lambda = 0.0152, 0.0173 - bayes estimate, 97/1000  
 lambda = 0.0152, 0.0172 - bayes estimate, 98/1000  
 lambda = 0.0152, 0.0172 - bayes estimate, 99/1000  
 lambda = 0.0152, 0.0171 - bayes estimate, 100/1000  
 lambda = 0.0152, 0.0167 - bayes estimate, 101/1000  
 lambda = 0.0152, 0.0165 - bayes estimate, 102/1000  
 lambda = 0.0152, 0.0167 - bayes estimate, 103/1000  
 lambda = 0.0152, 0.0168 - bayes estimate, 104/1000  
 lambda = 0.0152, 0.0170 - bayes estimate, 105/1000  
 lambda = 0.0152, 0.0164 - bayes estimate, 106/1000  
 lambda = 0.0152, 0.0165 - bayes estimate, 107/1000  
 lambda = 0.0152, 0.0166 - bayes estimate, 108/1000  
 lambda = 0.0152, 0.0166 - bayes estimate, 109/1000  
 lambda = 0.0152, 0.0168 - bayes estimate, 110/1000  
 lambda = 0.0152, 0.0168 - bayes estimate, 111/1000  
 lambda = 0.0152, 0.0168 - bayes estimate, 112/1000  
 lambda = 0.0152, 0.0160 - bayes estimate, 113/1000  
 lambda = 0.0152, 0.0161 - bayes estimate, 114/1000  
 lambda = 0.0152, 0.0161 - bayes estimate, 115/1000  
 lambda = 0.0152, 0.0162 - bayes estimate, 116/1000  
 lambda = 0.0152, 0.0162 - bayes estimate, 117/1000  
 lambda = 0.0152, 0.0162 - bayes estimate, 118/1000  
 lambda = 0.0152, 0.0156 - bayes estimate, 119/1000  
 lambda = 0.0152, 0.0157 - bayes estimate, 120/1000  
 lambda = 0.0152, 0.0158 - bayes estimate, 121/1000  
 lambda = 0.0152, 0.0155 - bayes estimate, 122/1000  
 lambda = 0.0152, 0.0154 - bayes estimate, 123/1000  
 lambda = 0.0152, 0.0154 - bayes estimate, 124/1000  
 lambda = 0.0152, 0.0154 - bayes estimate, 125/1000  
 lambda = 0.0152, 0.0155 - bayes estimate, 126/1000  
 lambda = 0.0152, 0.0156 - bayes estimate, 127/1000











































lambda = 0.0152, 0.0145 - bayes estimate, 992/1000  
lambda = 0.0152, 0.0145 - bayes estimate, 993/1000  
lambda = 0.0152, 0.0145 - bayes estimate, 994/1000  
lambda = 0.0152, 0.0145 - bayes estimate, 995/1000  
lambda = 0.0152, 0.0145 - bayes estimate, 996/1000  
lambda = 0.0152, 0.0145 - bayes estimate, 997/1000  
lambda = 0.0152, 0.0145 - bayes estimate, 998/1000  
lambda = 0.0152, 0.0145 - bayes estimate, 999/1000  
lambda = 0.0152, 0.0145 - bayes estimate, 1000/1000