

# ML Homework 3

Due Date : 2019/10/28 (MON) 23:59

TAs' email address: [jhhlab.tw@gmail.com](mailto:jhhlab.tw@gmail.com)

## Description:

### 1. Random Data Generator

- a. Univariate gaussian data generator
  - Input
    - Expectation value or mean:  $m$
    - Variance:  $s$
  - Output: A data point from  $N(m, s)$
  - HINT
    - [Generating values from normal distribution](#)
    - You have to handcraft your generator based on one of the approaches given in the hyperlink.
    - You can use uniform distribution function (Numpy)
- b. Polynomial basis linear model data generator
  - $y = W^T \phi(x) + e$ 
    - $W$  is a  $n \times 1$  vector
    - $e \sim N(0, a)$
  - Input:  $n$  (basis number),  $a, w$ 
    - e.g.  $n = 2 \rightarrow y = w_0 x^0 + w_1 x^1$ ,
  - Output:  $y$  (a number)
  - Internal constraint
    - $-1.0 < x < 1.0$
    - $x$  is uniformly distributed.

### 2. Sequential Estimator

- Sequential estimate the mean and variance
  - Data is given from the univariate gaussian data generator (1.a).
- Input:  $m, s$  as in (1.a)
- Function:
  - Call (1.a) to get a new data point from  $N(m, s)$
  - Use sequential estimation to find the current estimates to  $m$  and  $s$

- Repeat steps above until the estimates converge.
- Output: Print the new data point and the current estimates of  $m$  and  $s$  in each iteration.
- Notes
  - You should derive the recursive function of mean and variance based on the sequential estimation.
  - Hint: [Online algorithm](#)
- Sample input & output (⚠ *for reference only* ⚠)

```

1 Data point source function: N(3.0, 5.0)
2
3 Add data point: 3.234685454257290
4 Mean = 3.408993960833291    Variance = 0.030383455464755956
5 Add data point: 0.519242879651157
6 Mean = 2.445743600439247    Variance = 1.875958150575018
7 Add data point: 1.347113997201991
8 Mean = 2.171086199629932    Variance = 1.633278676389248
9 Add data point: 8.979491998496083
10 Mean = 3.532767359403163    Variance = 8.723325264636875
11 Add data point: 3.603448448693051
12 Mean = 3.544547540951477    Variance = 7.270131583917285
13 Add data point: 4.127197937610908
14 Mean = 3.627783311902824    Variance = 6.273110519038578
15 Add data point: 4.992735798186870
16 Mean = 3.798402372688330    Variance = 5.692747751482052
17
18 ...
19
20 Add data point: 4.233592159021013
21 Mean = 2.961576104513964    Variance = 5.045715437349161
22 Add data point: 3.529990930040463
23 Mean = 2.961883688294010    Variance = 5.043159812425648
24 Add data point: 1.125210345431449
25 Mean = 2.960890354955524    Variance = 5.042255747918937

```

### 3. Bayesian Linear regression

- Input
  - The precision (i.e.,  $b$ ) for initial prior  $w \sim N(0, b^{-1}I)$
  - All other required inputs for the polynomial basis linear model generator (1.b)
- Function
  - Call (1.b) to generate one data point
  - Update the prior, and calculate the parameters of predictive distribution
  - Repeat steps above until the posterior probability converges.
- Output
  - Print the new data point and the current parameters for posterior and predictive distribution.

- After probability converged, do the visualization
  - Ground truth function (from linear model generator)
  - Final predict result
  - At the time that have seen 10 data points
  - At the time that have seen 50 data points
  - Note
    - Except ground truth, you have to draw those data points which you have seen before
    - Draw a black line to represent the mean of function at each point
    - Draw two red lines to represent the variance of function at each point
      - In other words, distance between red line and mean is **ONE** variance
- Hint: Online learning
- Sample input & output (*for reference only*)

1.  $b = 1, n = 4, a = 1, w = [1, 2, 3, 4]$

```

1  Add data point (-0.64152, 0.19039):
2
3  Postirior mean:
4      0.0718294547
5      -0.0460797888
6      0.0295609502
7      -0.0189638408
8
9  Posterior variance:
10     0.6227289276,    0.2420256620,   -0.1552634839,    0.0996041049
11     0.2420256620,    0.8447365161,    0.0996041049,   -0.0638976884
12     -0.1552634839,    0.0996041049,    0.9361023116,    0.0409914289
13     0.0996041049,   -0.0638976884,    0.0409914289,    0.9737033172
14
15  Predictive distribution ~ N(0.00000, 2.65061)
16  -----
17  Add data point (0.07122, 1.63175):
18
19  Postirior mean:
20     0.6736864869
21     0.2388980107
22     -0.1054659080
23     0.0710615952
24
25  Posterior variance:
26     0.3765992302,    0.1254838660,   -0.1000441911,    0.0627881634
27     0.1254838660,    0.7895542671,    0.1257503020,   -0.0813299447
28     -0.1000441911,    0.1257503020,    0.9237138418,    0.0492510997
29     0.0627881634,   -0.0813299447,    0.0492510997,    0.9681964094

```

```

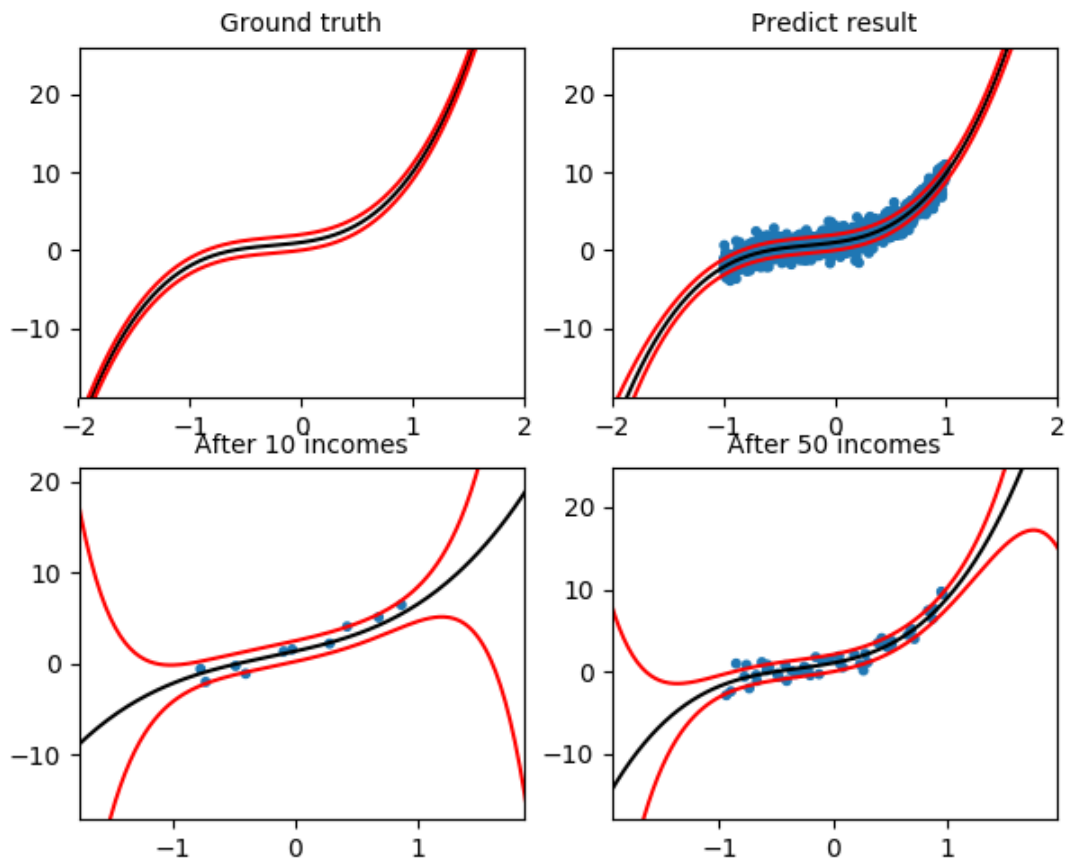
30
31 Predictive distribution ~ N(0.06869, 1.66008)
32 -----
33 Add data point (-0.19330, 0.24507):
34
35 Postirior mean:
36     0.5760972313
37     0.2450231522
38     -0.0801842453
39     0.0504992402
40
41 Posterior variance:
42     0.2867129751,    0.1311255325,    -0.0767580827,    0.0438488542
43     0.1311255325,    0.7892001707,    0.1242887609,    -0.0801412282
44     -0.0767580827,    0.1242887609,    0.9176812972,    0.0541575540
45     0.0438488542,    -0.0801412282,    0.0541575540,    0.9642058389
46
47 Predictive distribution ~ N(0.62305, 1.34848)
48 -----
49
50 ...
51
52 -----
53 Add data point (-0.76990, -0.34768):
54
55 Postirior mean:
56     0.9107496675
57     1.9265499885
58     3.1119297129
59     4.1312375189
60
61 Posterior variance:
62     0.0051883836,    -0.0004416700,    -0.0086000319,    0.0008247001
63     -0.0004416700,    0.0401966605,    0.0012708906,    -0.0554822477
64     -0.0086000319,    0.0012708906,    0.0265353911,    -0.0031205875
65     0.0008247001,    -0.0554822477,    -0.0031205875,    0.0937197255
66
67 Predictive distribution ~ N(-0.61566, 1.00921)
68 -----
69 Add data point (0.36500, 2.22705):
70
71 Postirior mean:
72     0.9107404583
73     1.9265225090
74     3.1119408740
75     4.1312734131
76
77 Posterior variance:
78     0.0051731092,    -0.0004872471,    -0.0085815201,    0.0008842340

```

```

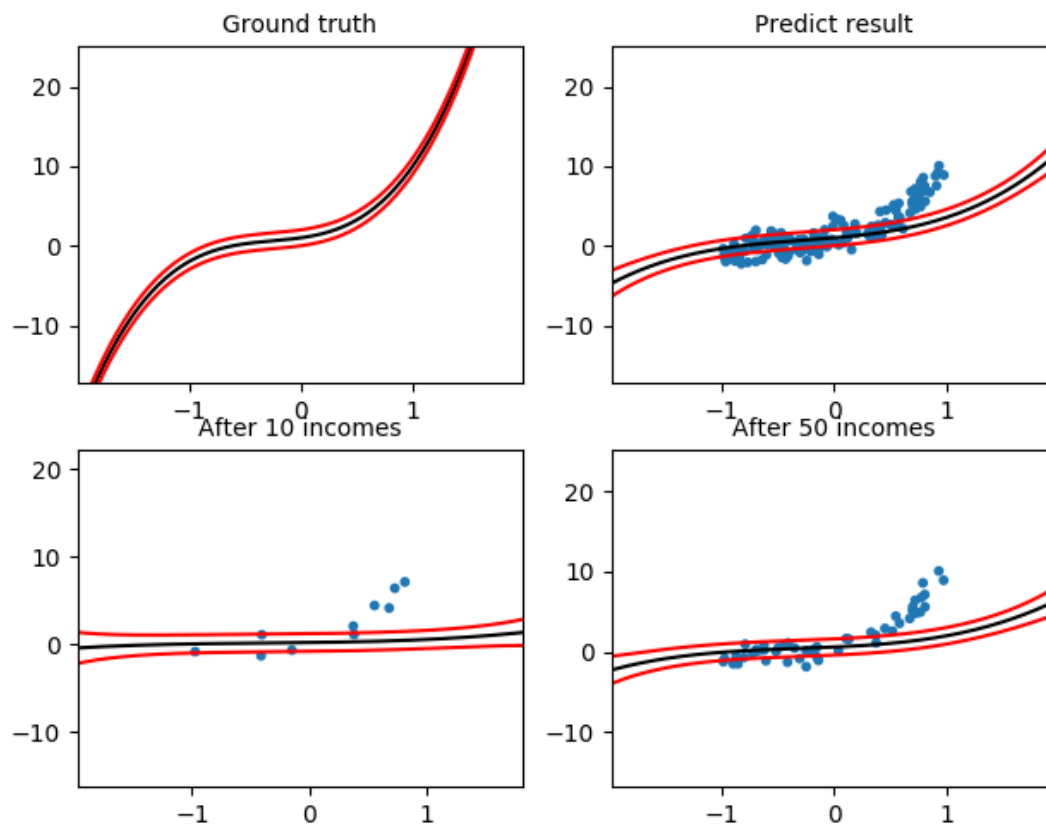
79 -0.0004872471, 0.0400606628, 0.0013261280, -0.0553046044
80 -0.0085815201, 0.0013261280, 0.0265129556, -0.0031927398
81 0.0008842340, -0.0553046044, -0.0031927398, 0.0934876838
82
83 Predictive distribution ~ N(2.22942, 1.00682)
84 -----

```



2.  $b = 100, n = 4, a = 1, w = [1, 2, 3, 4]$

1 | (Console output omitted)



3.  $b = 1, n = 3, a = 3, w = [1, 2, 3]$

1 | (Console output omitted)

