ELEN E4903 Homework 3

Kliment Mamykin UNI 2770

In [1]:

```
%load_ext autoreload
%autoreload 2
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
from hw3.hw3 import *
```

Problem 1 (a):

In [6]:

```
problem_1_part_a()
```

```
Test RMSE: 3.13
```

Problem 1 (b):

In [7]:

```
problem_1_part_b()
```

	σ^2=0.1	σ^2=0.2	σ^2=0.3	σ^2=0.4	σ^2=0.5	σ^2=0.6	σ^2=0.7	σ^2=0.8	σ^2=0.9	σ^2=1
b=5	1.968375	1.935695	1.926298	1.925341	1.928144	1.932793	1.938400	1.944518	1.950910	1.957447
b=7	1.921775	1.906854	1.910319	1.918335	1.927385	1.936397	1.945039	1.953237	1.961007	1.968399
b=9	1.898987	1.904196	1.919520	1.934500	1.947749	1.959318	1.969499	1.978587	1.986826	1.994411
b=11	1.891693	1.916448	1.940431	1.959555	1.974831	1.987353	1.997925	2.007108	2.015292	2.022752
b=13	1.896930	1.936870	1.965923	1.986806	2.002568	2.015070	2.025434	2.034362	2.042306	2.049567
b=15	1.910596	1.960666	1.991906	2.012957	2.028334	2.040346	2.050263	2.058827	2.066495	2.073558

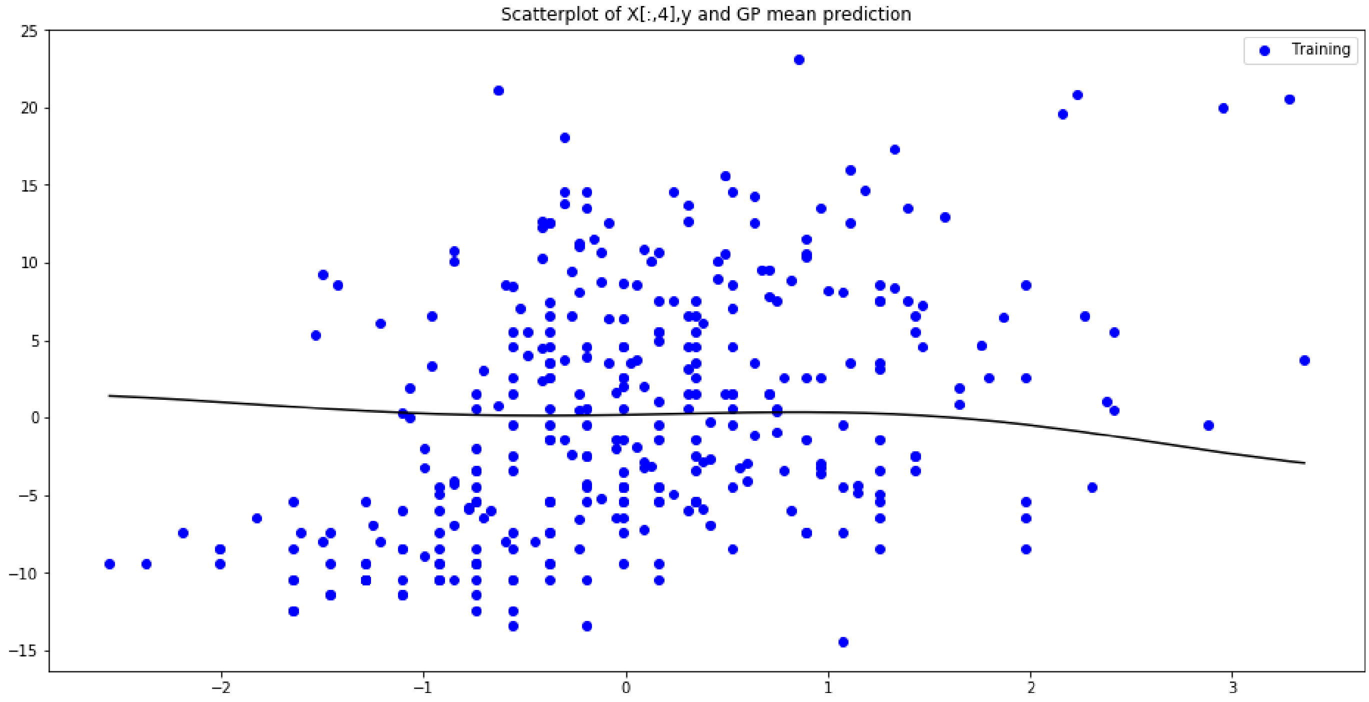
Problem 1 (c):

Best test RMSE value for GP is **1.891** for hyper-parameters b=11 and sigma^2=0.1. From HW1, the best RMSE for Least Squares regression was **2.634**, and best RMSE for Ridge Regressions (polynomial order 2 feature expansion) was **2.193**. On the test set GP clearly outperforms both LS and RR models.

The drawback in GP is the computational complexity. In both LS and RR linear models the computational complexity of matrix multiplication is $O(ND^2)$ and for matrix inverse $O(D^3)$. Since D << N, the overall complexity will be (ND^2). For Gaussian Process the matrix inversion is computed on the covariance matrix NxN and its complexity $O(N^3)$ will dominate other computations.

Problem 1 (d):

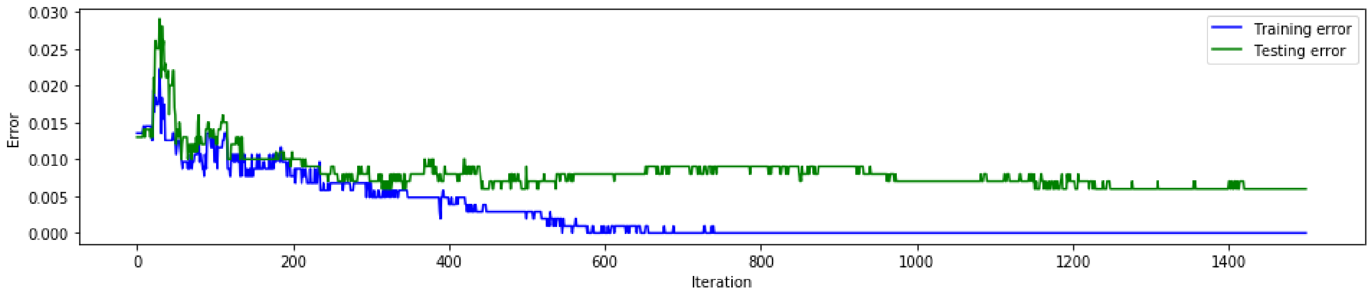
```
In [26]: problem_1_part_d()
```



Problem 2 (a):

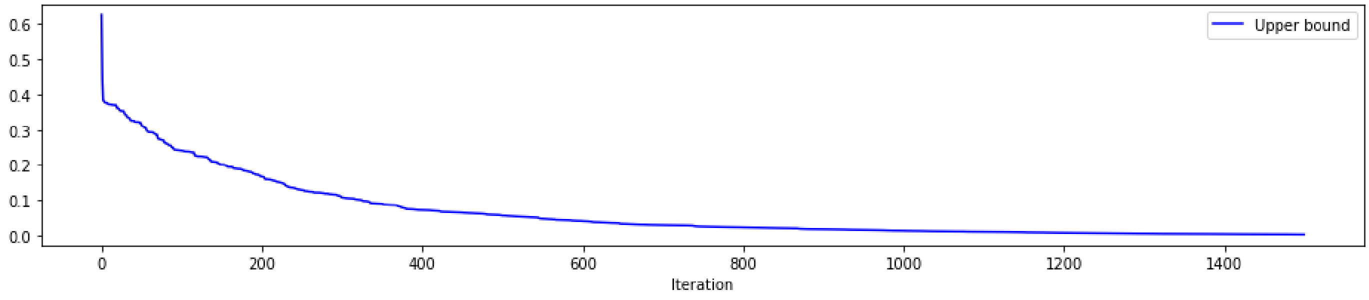
```
In [49]: results = train_boosted_classifier(1500)
```

```
In [50]: problem_2_part_a(*results)
```



Problem 2 (b):

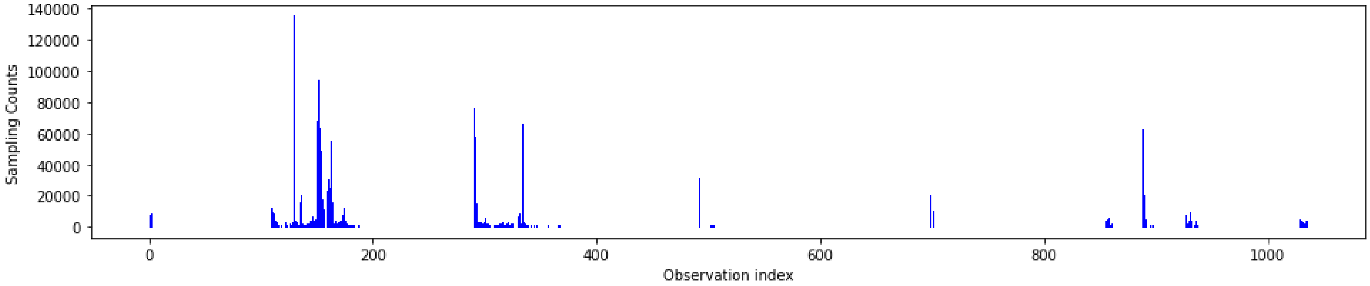
```
In [51]: problem_2_part_b(*results)
```



Problem 2 (c):

In [52]:

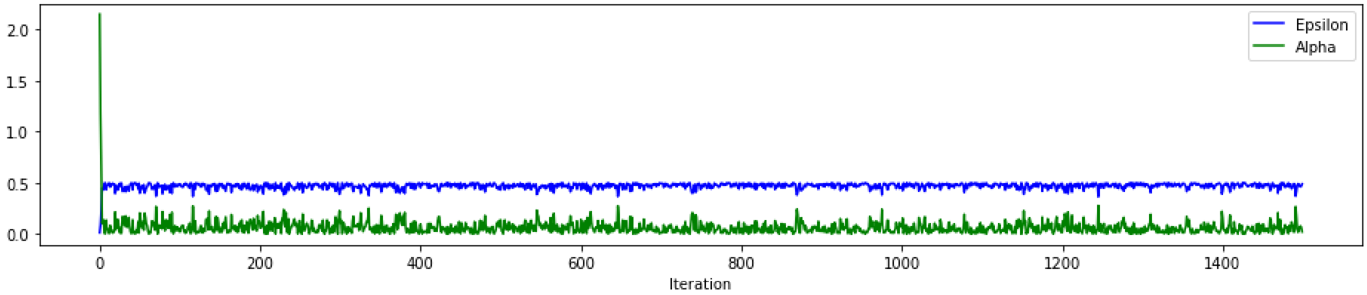
```
problem_2_part_c(*results)
```



Problem 2 (d):

In [53]:

```
problem_2_part_d(*results)
```



References:

- https://en.wikipedia.org/wiki/Computational_complexity_of_mathematical_operations (https://en.wikipedia.org/wiki/Computational_complexity_of_mathematical_operations)