Homework 3
CS 259 Numerical Methods for Data Science
Prof. **David Bindel**
TA. **Yurong You, Xinran Zhu**

Hongyu Yan (516030910595)
ACM Class, Zhiyuan College, SJTU
Due Date: June 22nd, 2018
Submit Date: July 18, 2018

Problem 1

question1

The number of iterate steps is 1000. And the learning rate α is 0.0005.

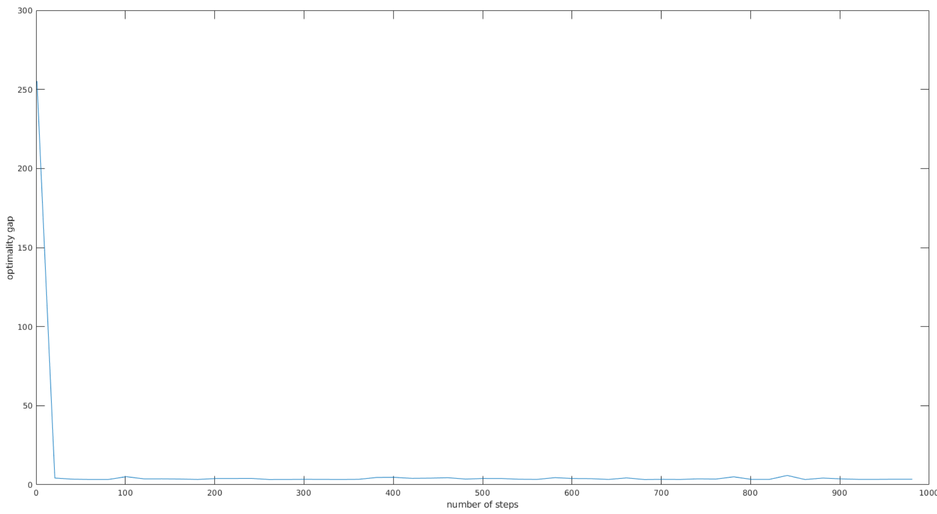

Figure 1: SGD method

question2

As the α increses, the optimality gap increses.

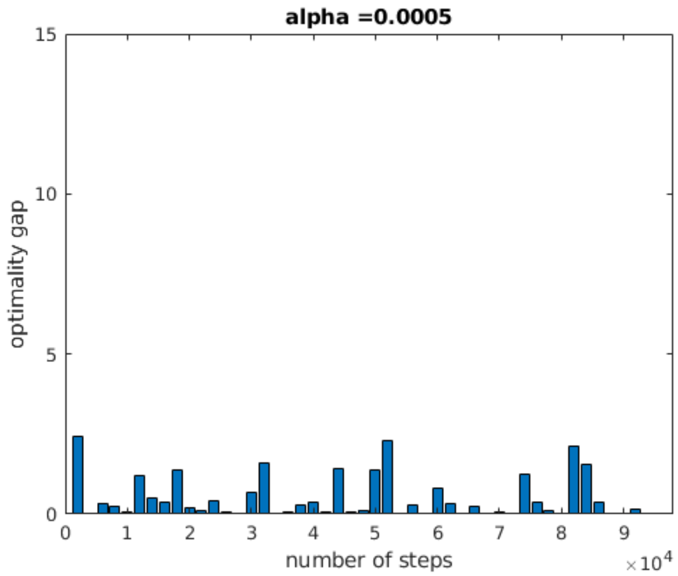

Figure 2: SGD method changing α

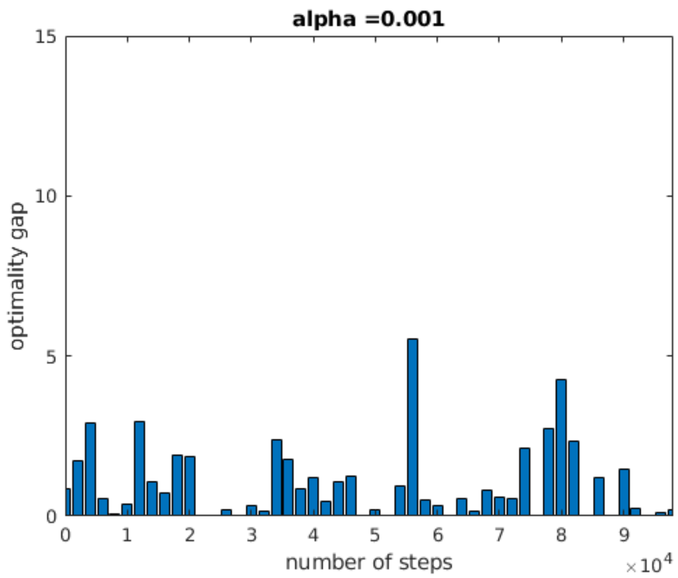

Figure 3: SGD method changing α

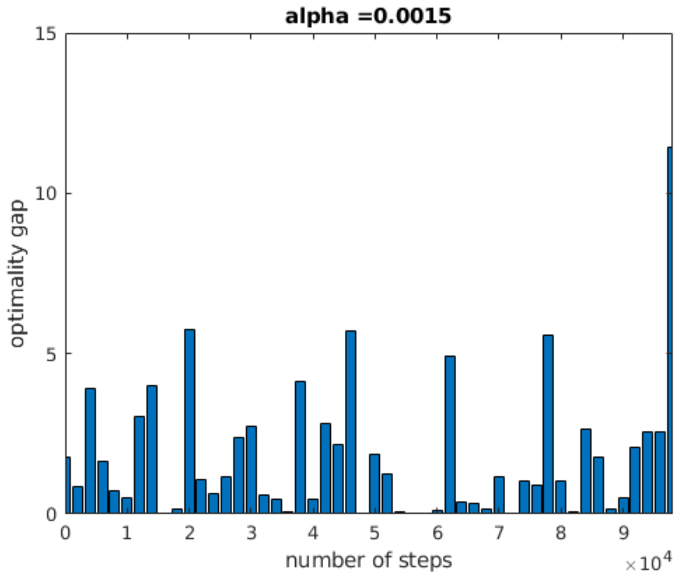

Figure 4: SGD method changing α

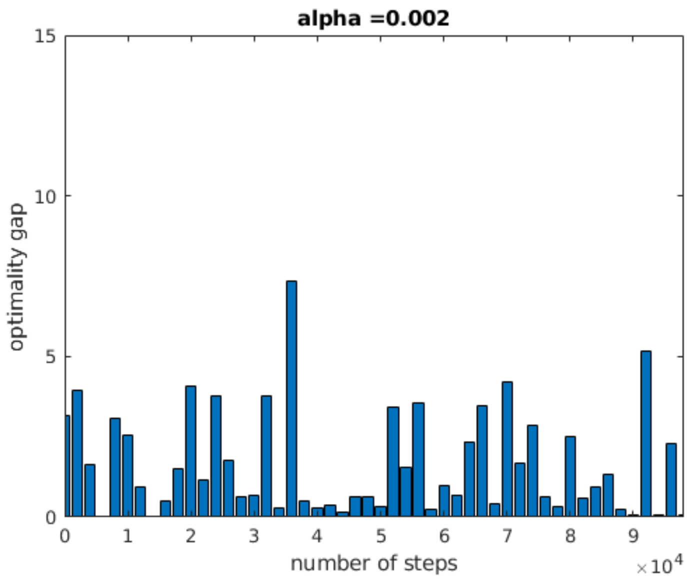

Figure 5: SGD method changing α

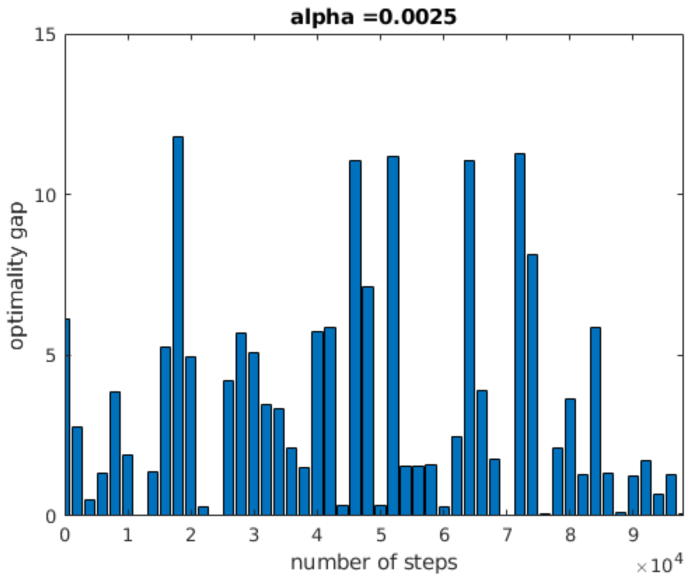

Figure 6: SGD method changing α

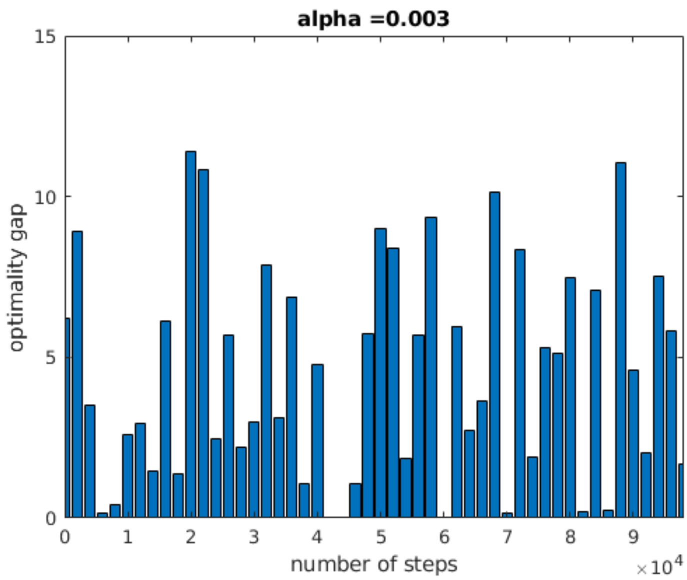

Figure 7: SGD method changing α

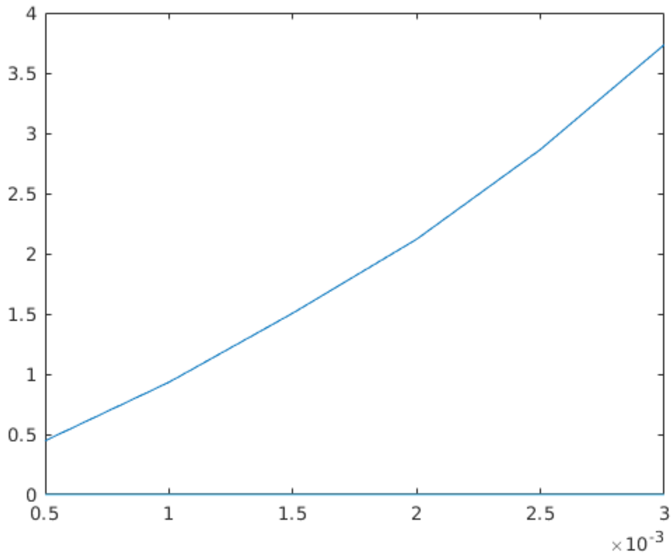

Figure 8: mean optimality gap v.s. α

Problem 2

question1

$log\rho(I) = 6.6613e - 16$

question2

With the sample sizes increses, the $log\rho(G)$ decreses.(almost linearly)

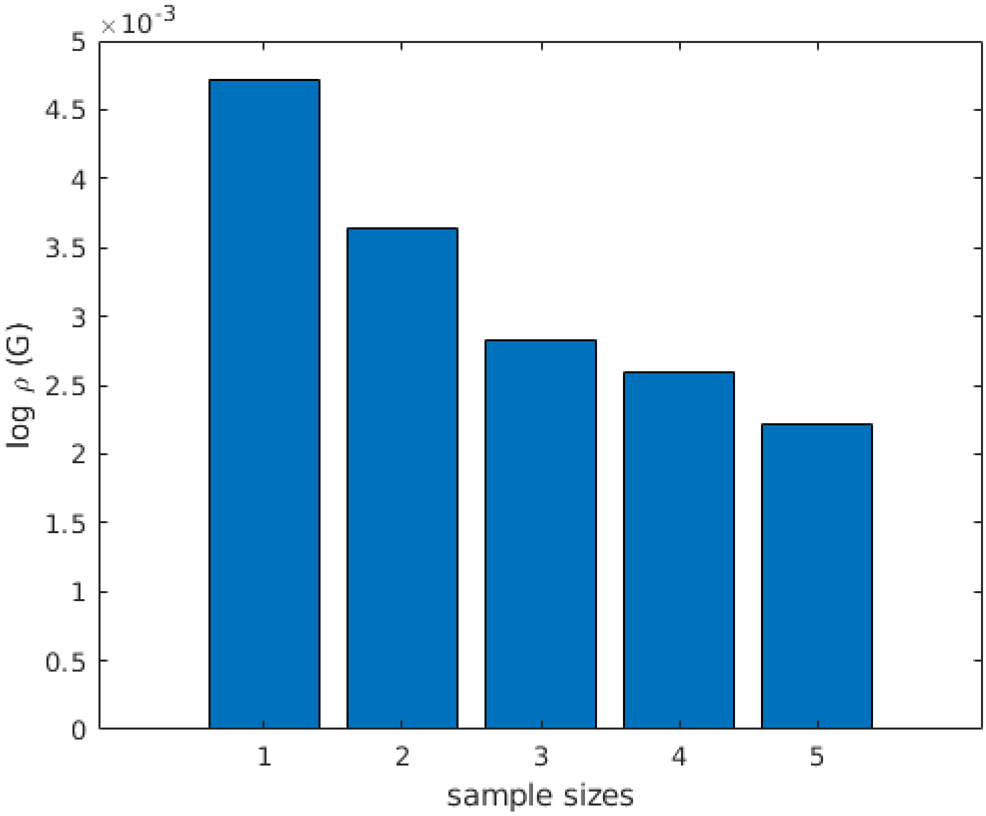

Figure 9: Gauss-Newton method

Problem 3

The figure shows that Gauss-Newton method converges quicker than SGD method.

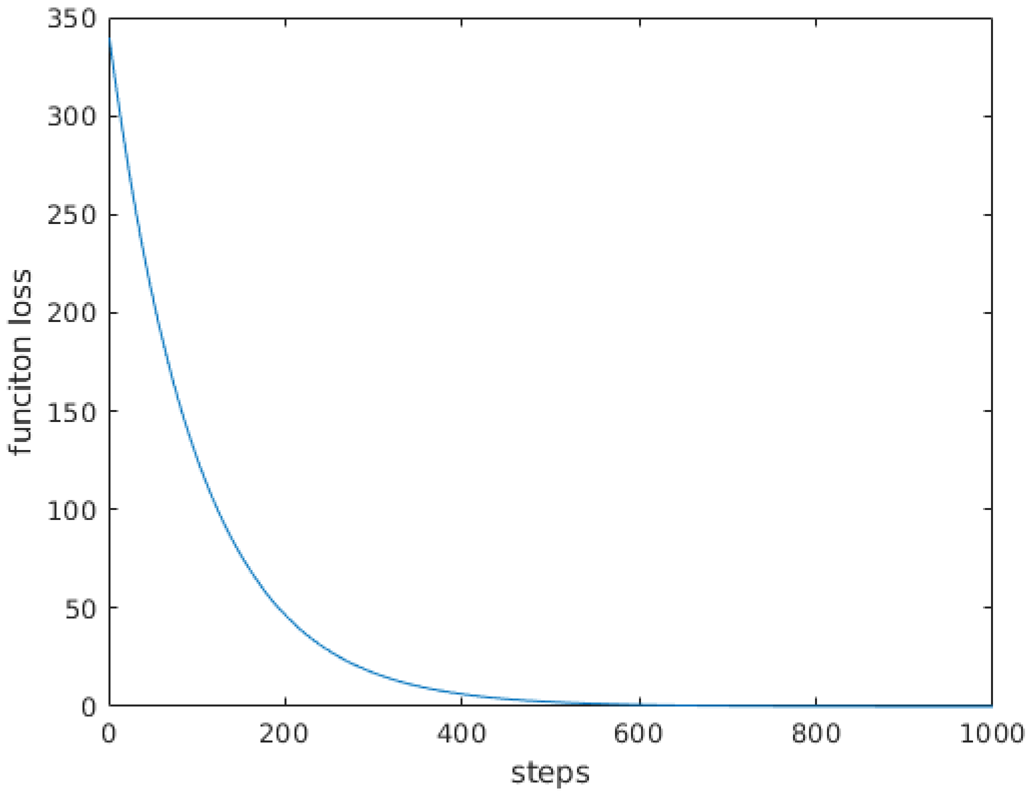

Figure 10: Gauss-Newton method

```
load('cricket.dat');
A = cricket(:,1);
b = cricket(:,2);

A = [A, ones(length(b), 1)];

% ===== initial ===== %

max_step = 1000;
alpha = 0.005;
c = 0.0003;
threshold = 1e-4;

len = length(b);
phi = 0;
tmp = A*x-b;
for i = 1:len
        phi = phi + exp(tmp(i)^2);
end
```

```
phi = phi - 1;
% ===== iterate ===== %

x = [0; 0];
errors = zeros(1, max_step);

for i = 1:max_step
        r = b - A*x;

        df = -2 * c * exp(c*r.^2).*r.*A;
        f = exp(c*r.^2) - 1;

        p = -(df'*df) \ (df'*f);
        x = x + alpha * p;

        errors(i) = 0.5 * (f' * f);

        if norm(df, 2) < threshold
                break;
        end
end

x_step = 1:max_step;
plot(x_step, errors);
xlabel('steps');
ylabel('funciton loss');
```