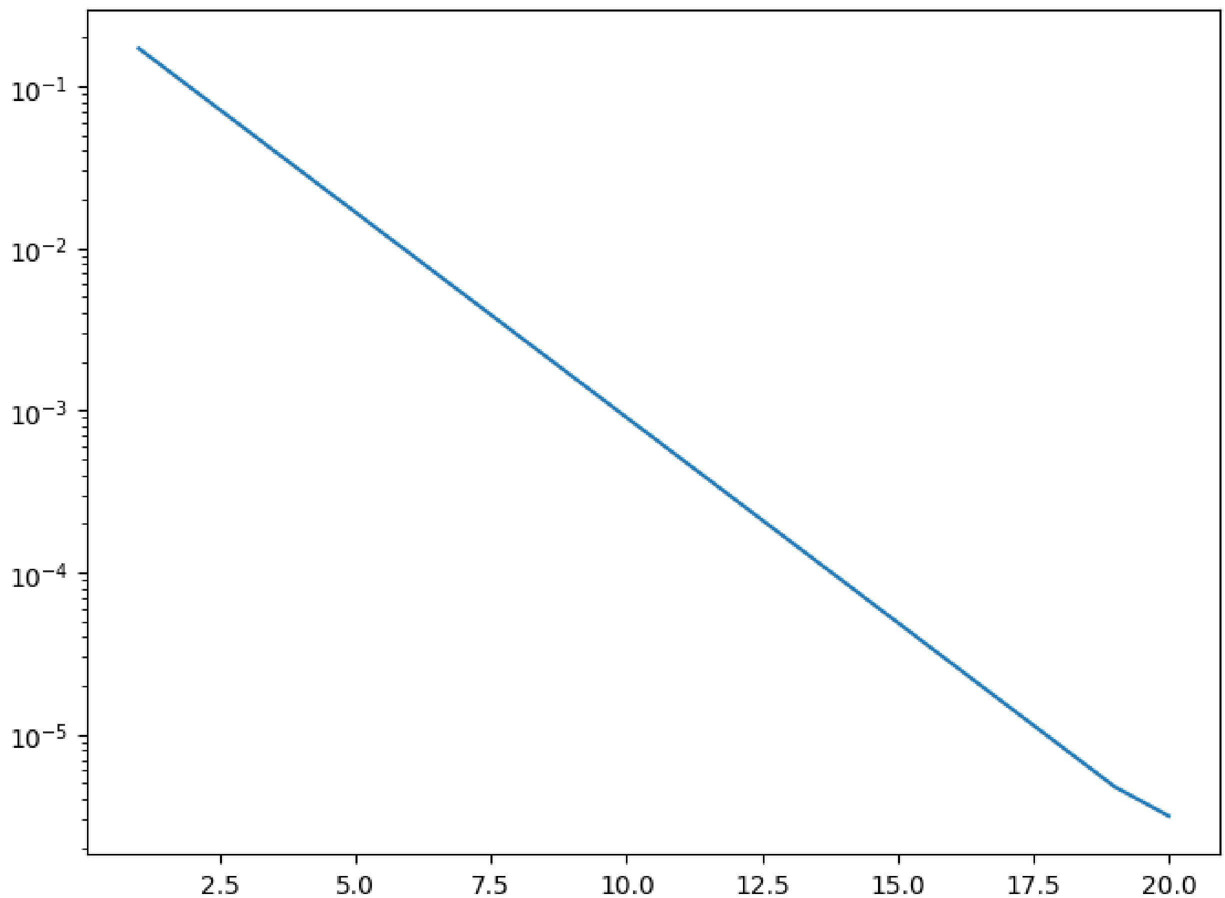
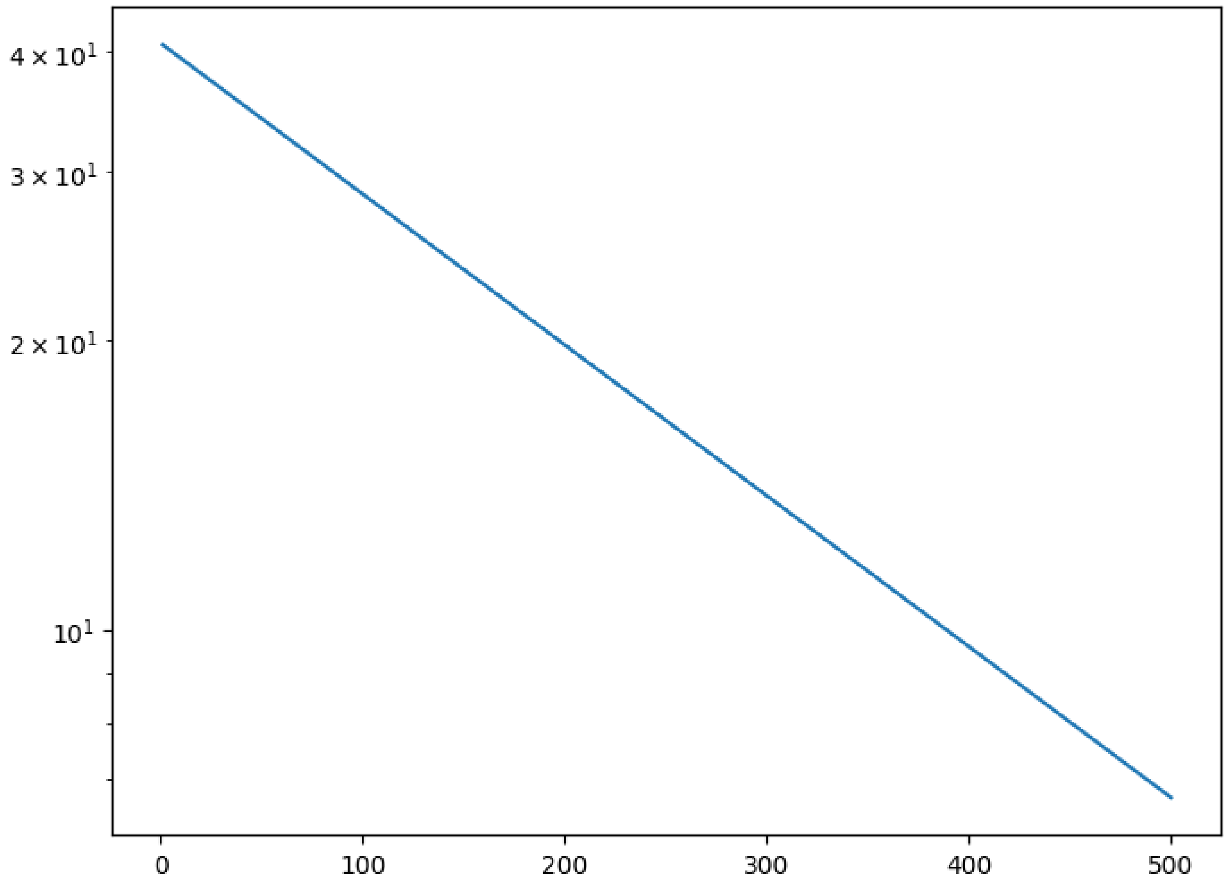
1)

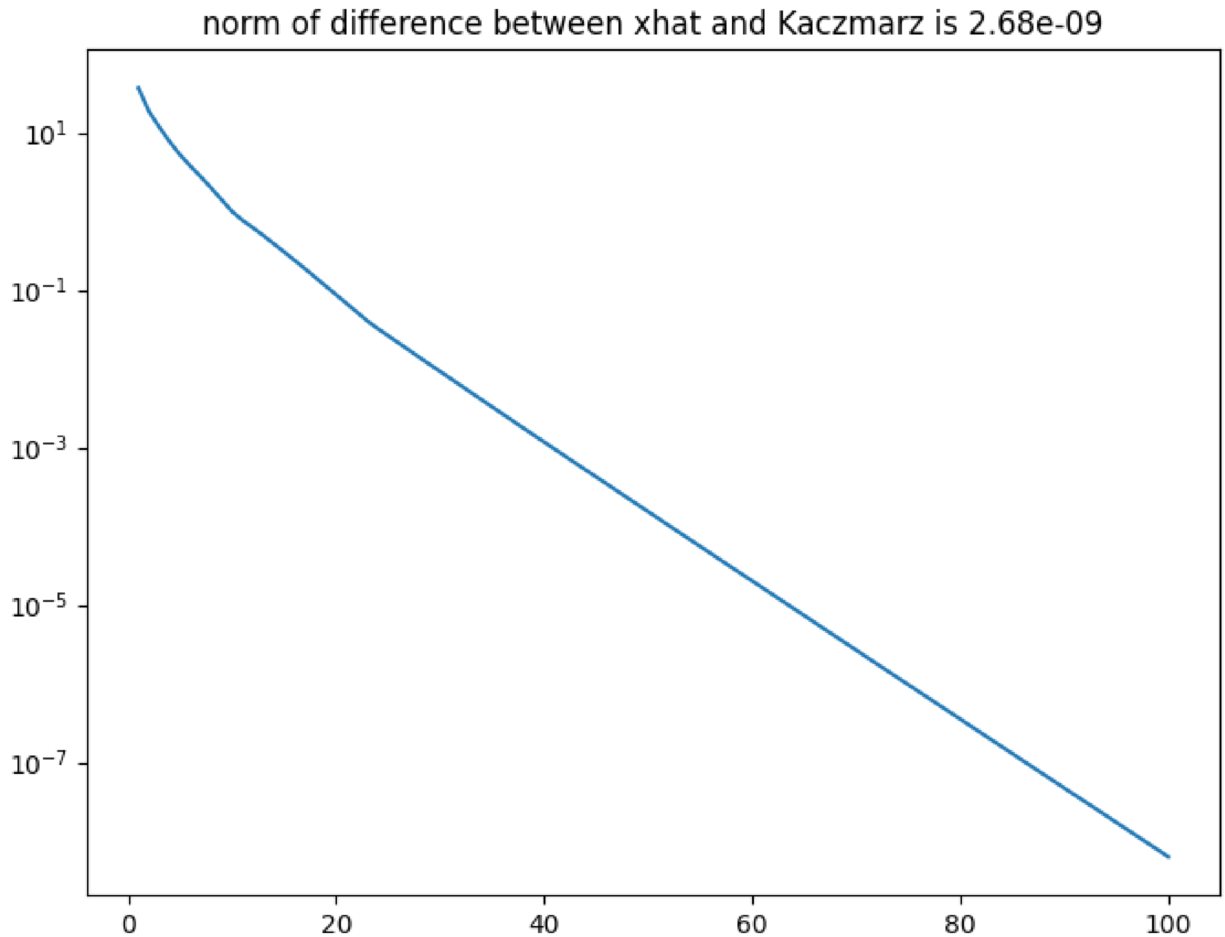
$n = 14$ is enough for the projection to be correct to 0.00



2)

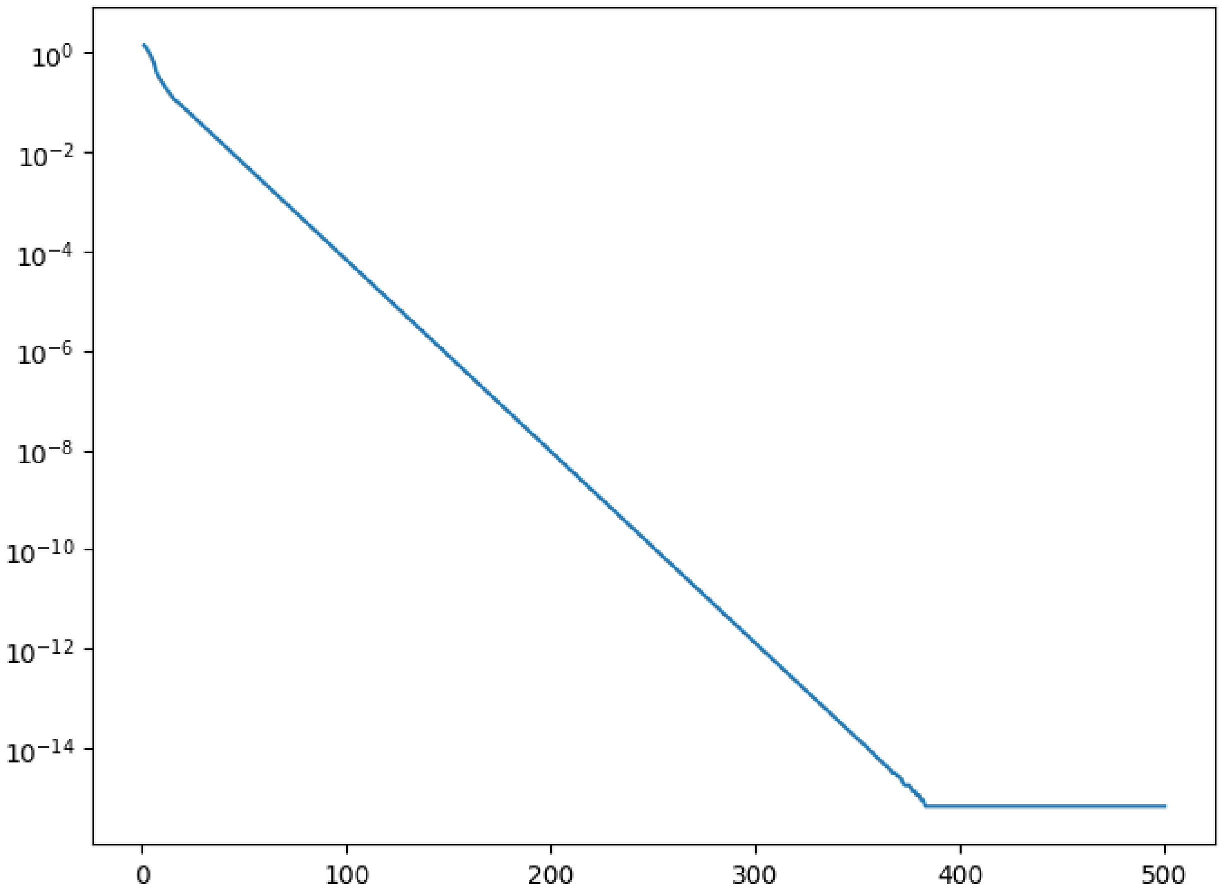


3)



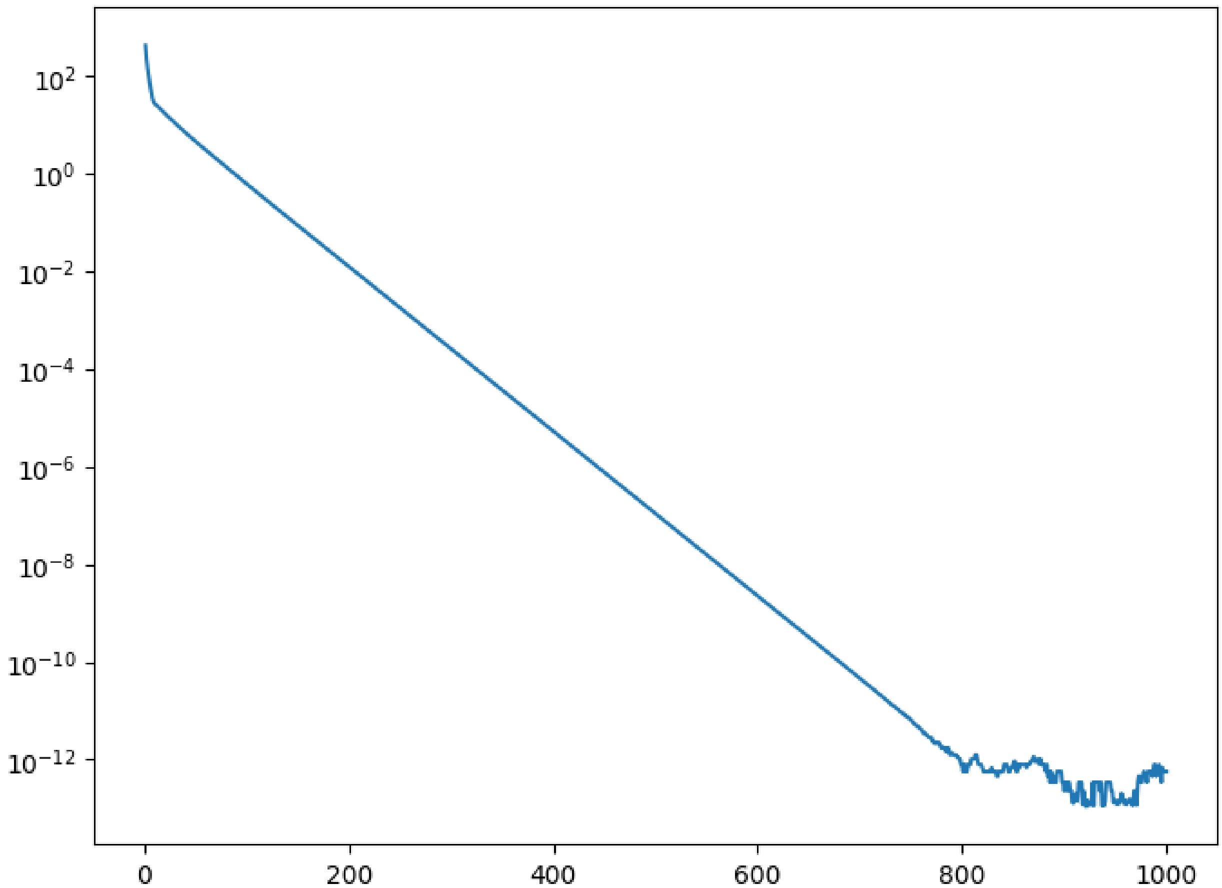
4)

$n = 96$ is enough iterations for the feasibility gap to be at most 0.0001



10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-14}
0
100
200
300
400
500

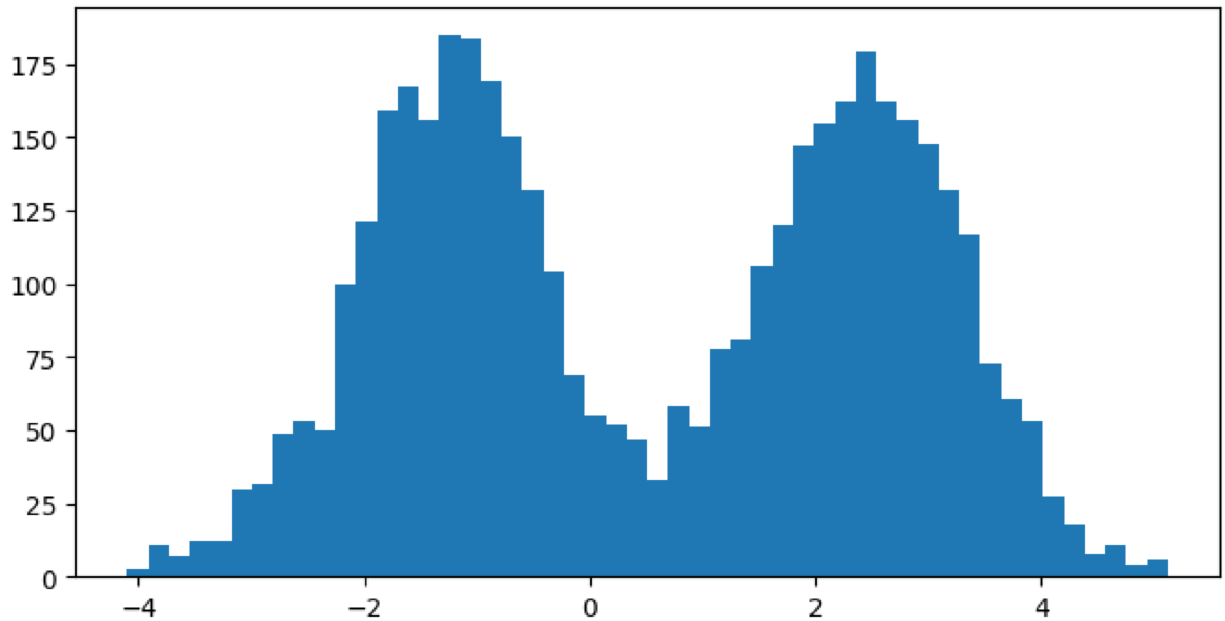
5)



6)

There is a benefit to setting $s < 1$. When this is the case, the training and testing error is lower. This is likely because when s is closer to 1, it overshoots the correct x. I found when $s = 1$ training and testing error were around 25%. When $s = 0.01$ the training and testing error was around 3%.

Training error = 3.26%
Testing error = 4.06%
Training Confusion matrix:
[[1971 117]
[22 2153]]
Testing Confusion matrix:
[[1945 144]
[29 2147]]



When the error rate converges to zero, the histogram of the two function outputs is bimodal.

7)

There less benefit to setting $s < 1$. When this is the case, the training and testing error is lower. This is likely because when s is closer to 1, it overshoots the correct x. I found when $s = 1$ training and testing error were around 8.77% and 13.2% respectively. When $s = 0.01$ the training and testing error were around 7.24% and 9.12% respectively.

Training error = 7.24%
Testing error = 9.12%

Training Confusion matrix:

```
[[2004    0    6    2    4   27   15    1    7    2]
 [   0 2283    7    7    2    4    1    6   16    3]
 [   8   22 1826   32   30   11   24   22   60   10]
 [   4    5   45 1951    1   75    6   17   41   15]
 [   2    3   15    2 1922    6   16    5    9   66]
 [  23    6   14   53   21 1695   28    9   35   17]
 [  13    3   19    1   12   31 1975    0    6    0]
 [   7    8   20    6   10    4    0 2071    6   78]
 [  18   33   16   62    9   81   10   10 1815   18]
 [   6    9    9   25   47    8    0   52   16 1937]]
```

Testing Confusion matrix:

```
[[1981    0   15    5    2   21   25    2   12    1]
 [   0 2297    7    6    4    7    7    4   21    2]
 [  15   22 1871   43   41   18   29   28   58    7]
 [   6   10   46 1925    2   92    8   27   50   25]
 [   9   11   12    1 1868    5   29    4   12   75]
 [  28   12   11   80   14 1606   38    9   72   24]
 [  18    1   28    0   28   28 1970    0    4    0]
 [  14    7   29    9   18    6    2 2023    7   76]
 [  14   43   23   58    7   99   15    7 1688   37]
 [  18   10    6   21   63   17    1   69   19 1855]]
```