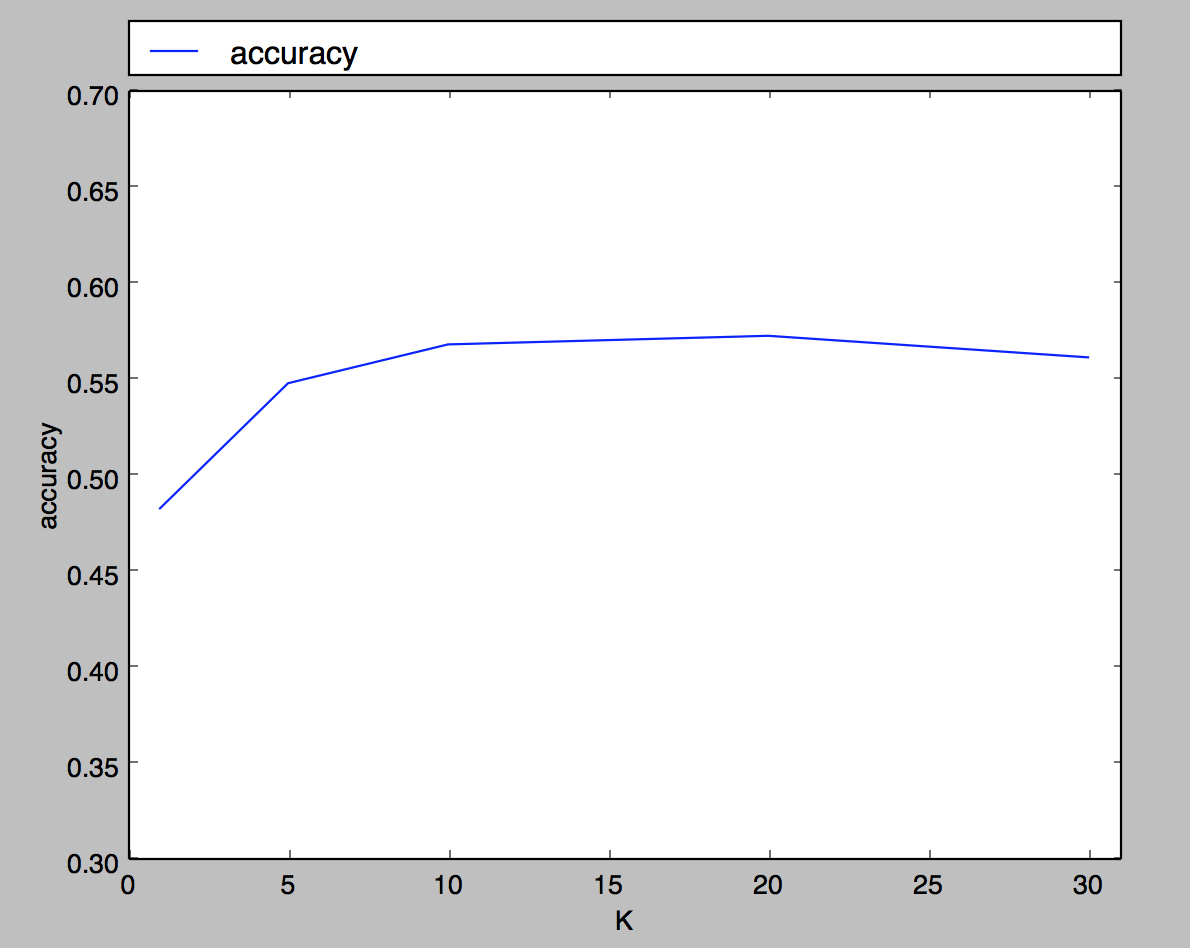
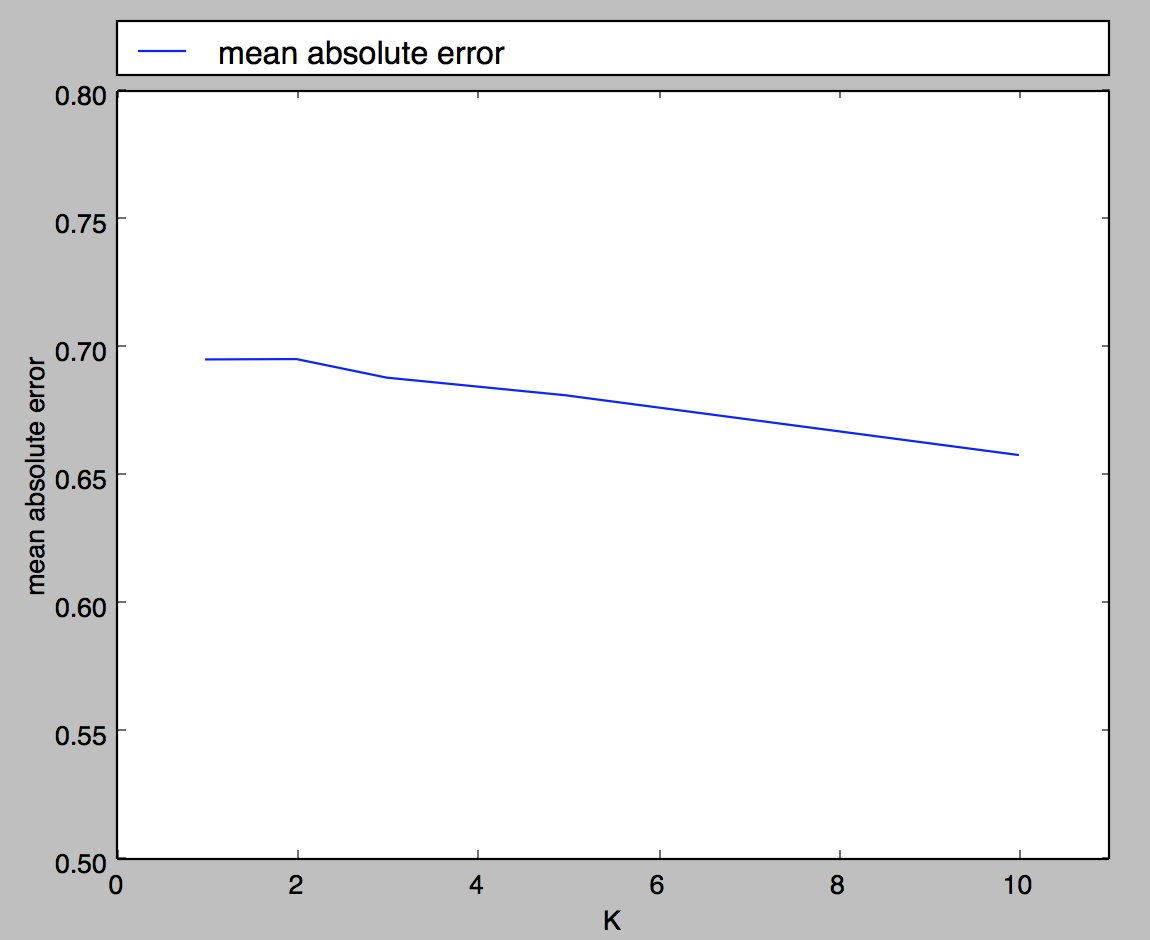
Hao Fu

Please see part2-1, part2-2, part2-3, and part3 below.

Part 2-1: For yeast data set, the accuracy plot:



Part 2-2: For wine data set, the mean absolute error plot.



Part 2-3

K = 1

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | CYT | MIT | NUC | ME3 | ME2 | ME1 | EXC | VAC | POX | ERL |
| CYT | 69 | 39 | 21 | 4 | 1 | 0 | 1 | 2 | 1 | 0 |
| MIT | 46 | 57 | 13 | 6 | 0 | 0 | 0 | 1 | 0 | 0 |
| NUC | 26 | 12 | 34 | 2 | 3 | 0 | 2 | 0 | 2 | 0 |
| ME3 | 3 | 9 | 2 | 32 | 1 | 0 | 0 | 0 | 0 | 0 |
| ME2 | 3 | 0 | 2 | 3 | 6 | 1 | 2 | 0 | 1 | 0 |
| ME1 | 0 | 0 | 1 | 0 | 3 | 6 | 3 | 0 | 1 | 0 |
| EXC | 0 | 0 | 1 | 0 | 1 | 3 | 5 | 0 | 0 | 0 |
| VAC | 3 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| POX | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 5 | 0 |
| ERL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

K = 30

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | CYT | MIT | NUC | ME3 | ME2 | ME1 | EXC | VAC | POX | ERL |
| CYT | 89 | 36 | 11 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| MIT | 48 | 59 | 13 | 3 | 0 | 0 | 0 | 0 | 0 | 0 |
| NUC | 25 | 5 | 45 | 1 | 1 | 3 | 1 | 0 | 0 | 0 |
| ME3 | 3 | 2 | 2 | 40 | 0 | 0 | 0 | 0 | 0 | 0 |
| ME2 | 4 | 2 | 3 | 3 | 3 | 3 | 0 | 0 | 0 | 0 |
| ME1 | 0 | 0 | 1 | 0 | 1 | 9 | 3 | 0 | 0 | 0 |
| EXC | 0 | 0 | 1 | 0 | 0 | 4 | 5 | 0 | 0 | 0 |
| VAC | 2 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| POX | 6 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ERL | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

We can see that values in diagonal of the matrices increase as k value increase, suggesting that a higher K better the kNN classification accuracy.

Part 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pop | Current Distance | Best Distance | Best Node | Priority Queue |
|  |  |  |  | (f, 0) |
| f (6, 3) |  |  | f | (h, 0) (c, 1) |
| h (12, 5) |  |  | f | (i, 0) (c, 1) (g, 5) |
| i (10, 10) | 3 | 3 | i | (c, 1) (j, 3) (g, 5) |
| c (5, 10) | 2 | 2 | c | (e, 0) (b, 0) (j, 3) (g, 5) |
| e (2, 4) |  | 2 | c | (d, 0) (b, 0) (j, 3) (g, 5) |
| d (2, 8) |  | 2 | c | (b, 0) (j, 3) (g, 5) |
| b (3, 12) |  | 2 | c | (j, 3) (g, 5) |

Finally, we have the node j (13, 11.5), current bound is 3, and it is bigger than 2 then we

return result as below:

best node = c, return best distance = 2