# Fast K-NN classification (extra pruning tricks)

*K-nearest-neighbor classification without finding the K-NN*

The Auton Lab

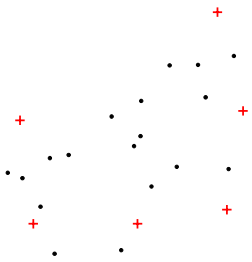
Carnegie Mellon University

www.autonlab.org

## KNS2

- Assume binary output
- Assume positive class is much less frequent than negative class
- Assume we want more than a "positive/negative" prediction: we want to know exactly how many of the K-NN are from the +ve class

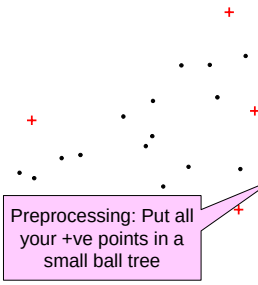
KNS2 does this without finding the K-NN



Assume we have a set of data points.

Some are +ve points (denoted +)

The large majority are –ve points (denoted •)



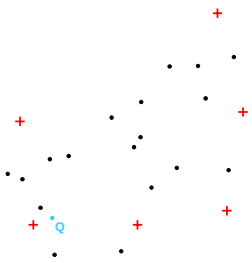
Assume we have a set of data points.

Some are +ve points (denoted +)

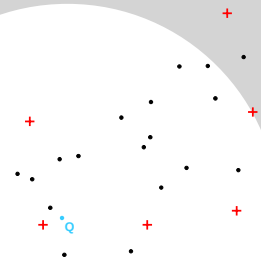
The vast majority are –ve points (denoted •)

Preprocessing: Put all your +ve points in a small ball tree

Preprocessing: Put all your -ve points in a separate large ball tree

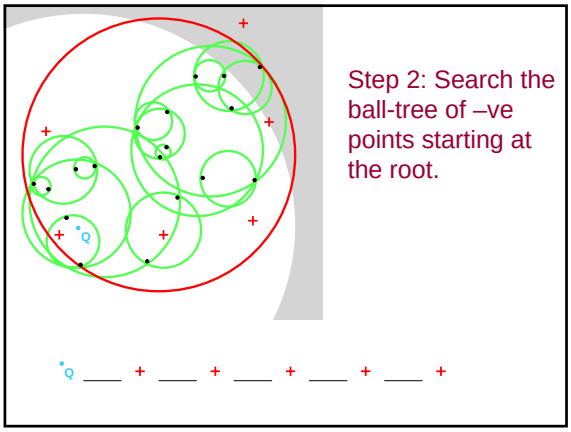


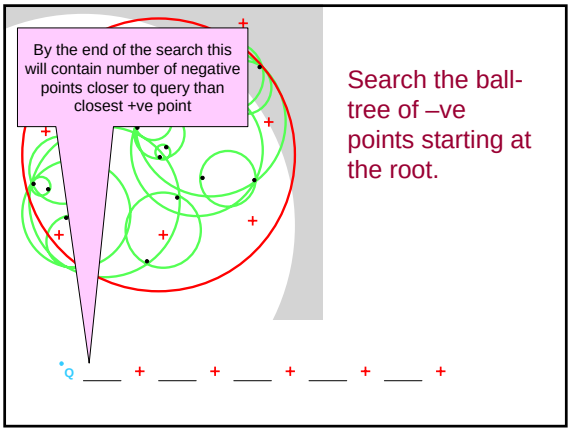

Goal: Find out how many of the 5-nearest neighbors of Q are positive.

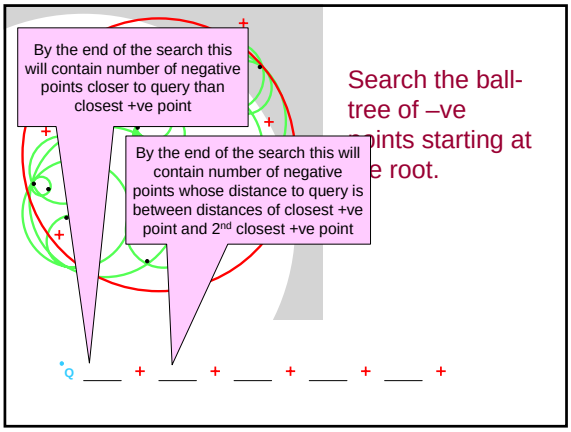


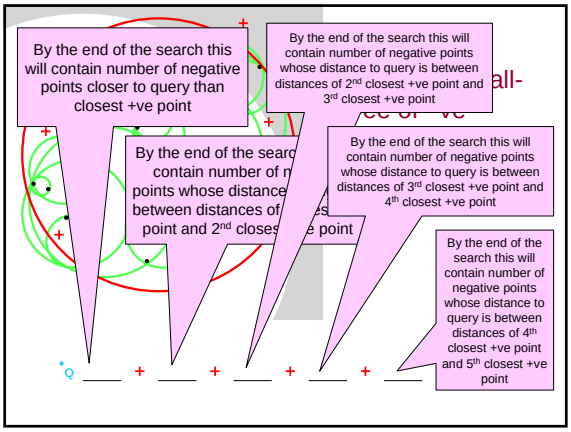

Step One: Find the five nearest +ve points using KNS1.

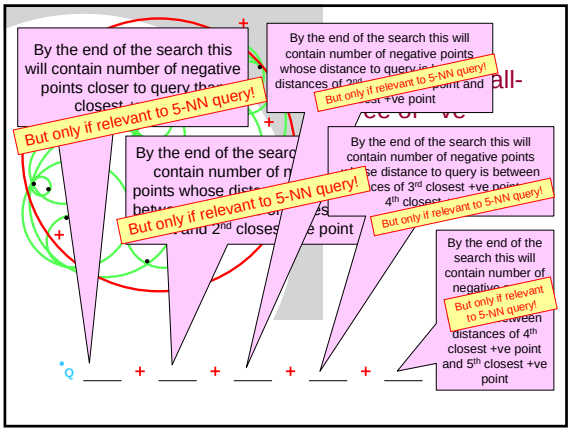
We're assuming there are far fewer +ves than –ves so this is not the dominant cost.

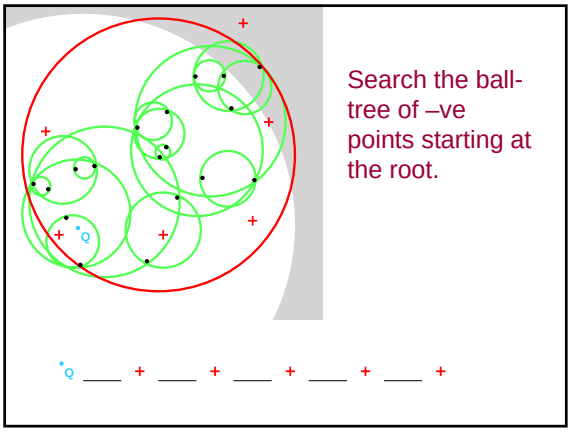

Step 2: Search the ball-tree of –ve points starting at the root.

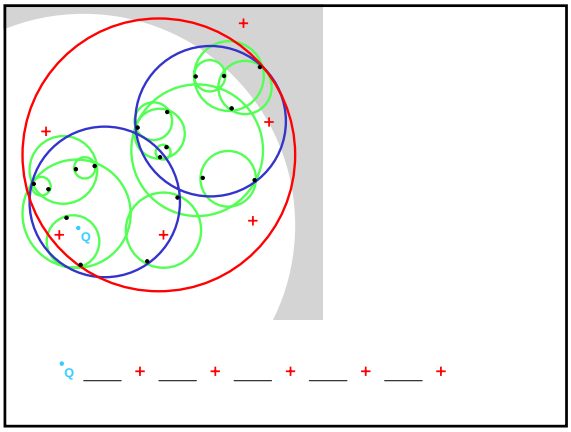

By the end of the search this will contain number of negative points closer to query than closest +ve point
Search the ball-tree of –ve points starting at the root.

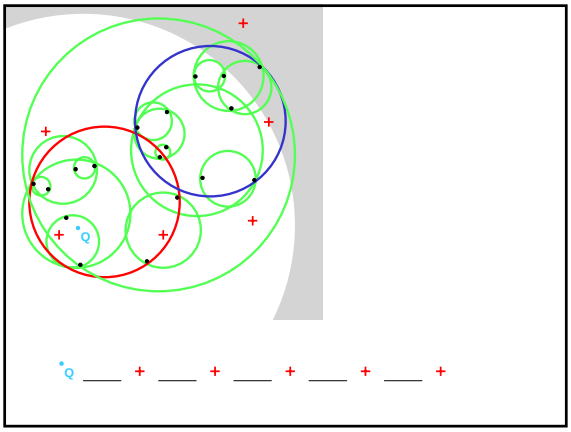

By the end of the search this will contain number of negative points closer to query than closest +ve point
By the end of the search this will contain number of negative points whose distance to query is between distances of closest +ve point and 2nd closest +ve point
Search the ball-tree of –ve points starting at the root.

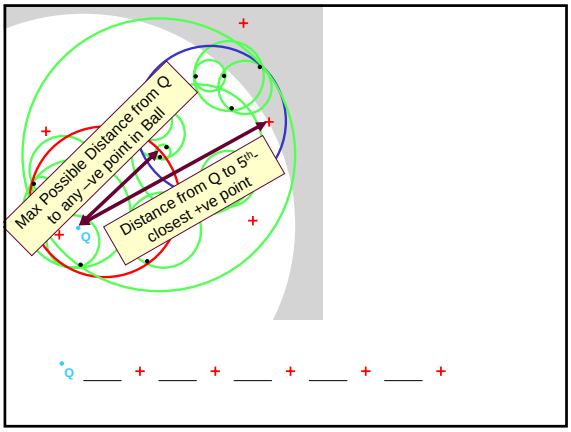

By the end of the search this will contain number of negative points closer to query than closest +ve point
By the end of the search this will contain number of negative points whose distance to query is between distances of 2nd closest +ve point and 3rd closest +ve point
By the end of the search this will contain number of negative points whose distance to query is between distances of 3rd closest +ve point and 4th closest +ve point
By the end of the search this will contain number of negative points whose distance to query is between distances of 4th closest +ve point and 5th closest +ve point

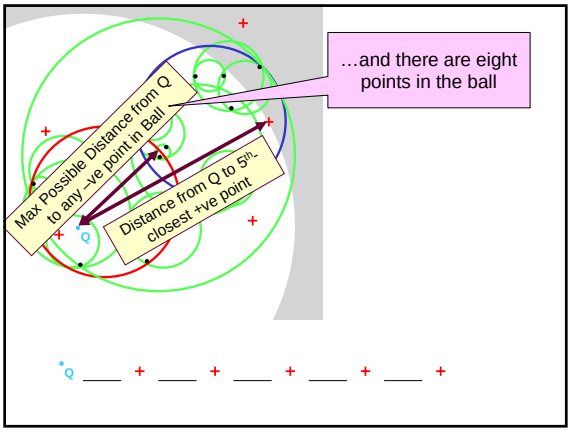

By the end of the search this will contain number of negative points closer to query than closest +ve point
But only if relevant to 5-NN query!
By the end of the search this will contain number of negative points whose distance to query is between distances of 2nd closest +ve point and 3rd closest +ve point
But only if relevant to 5-NN query!
By the end of the search this will contain number of negative points whose distance to query is between distances of 3rd closest +ve point and 4th closest +ve point
But only if relevant to 5-NN query!
By the end of the search this will contain number of negative points whose distance to query is between distances of 4th closest +ve point and 5th closest +ve point
But only if relevant to 5-NN query!

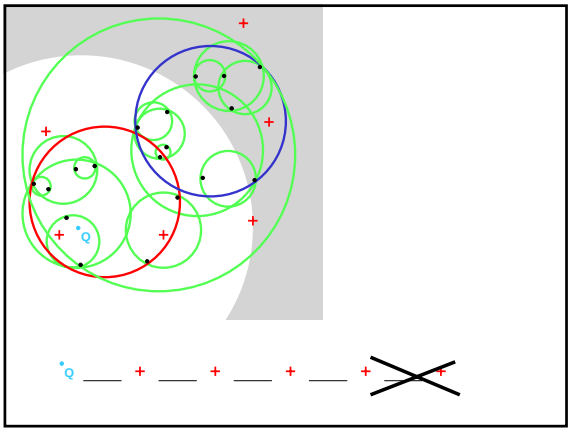

Search the ball-tree of –ve points starting at the root.

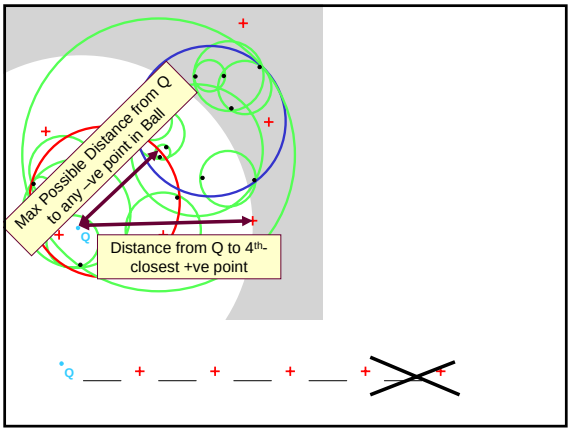

Q

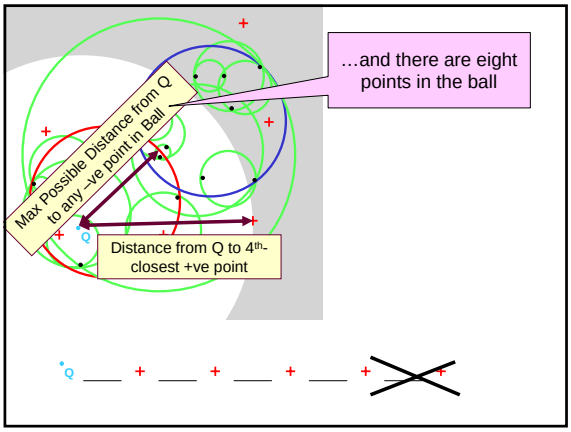

Q

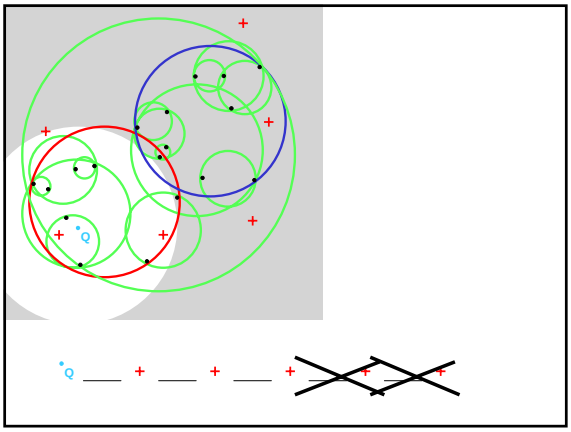

Max Possible Distance from Q to any –ve point in Ball
Distance from Q to 5th- closest +ve point
Q

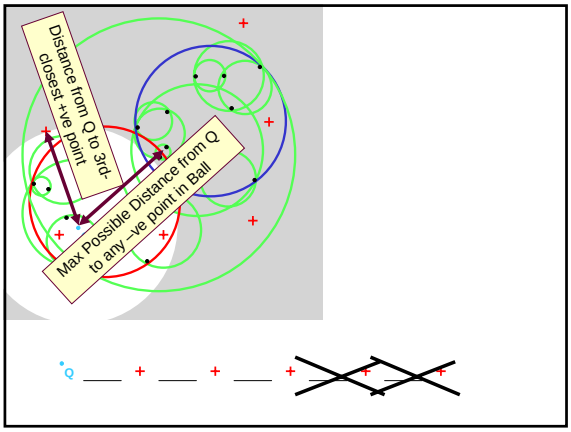

…and there are eight points in the ball
Max Possible Distance from Q to any –ve point in Ball
Distance from Q to 5th- closest +ve point
Q

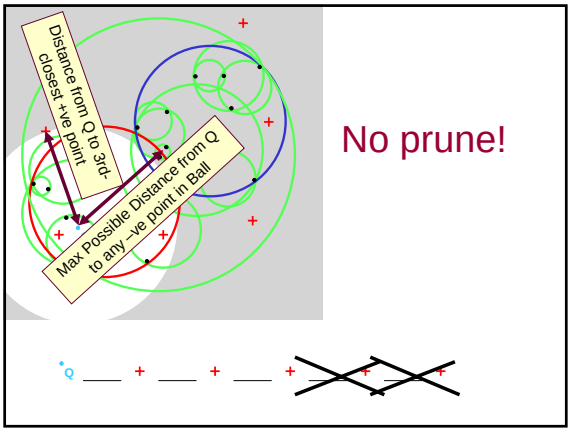

Q

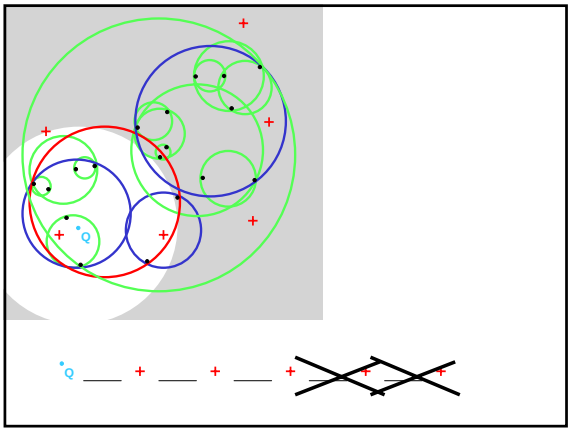

Max Possible Distance from Q to any –ve point in Ball
Distance from Q to 4th- closest +ve point
Q

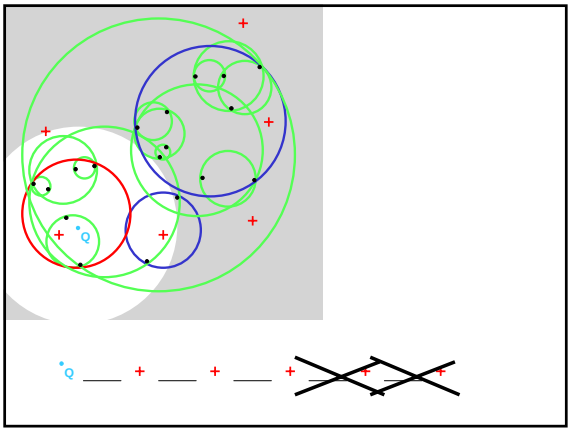

…and there are eight points in the ball
Max Possible Distance from Q to any –ve point in Ball
Distance from Q to 4th-closest +ve point
Q

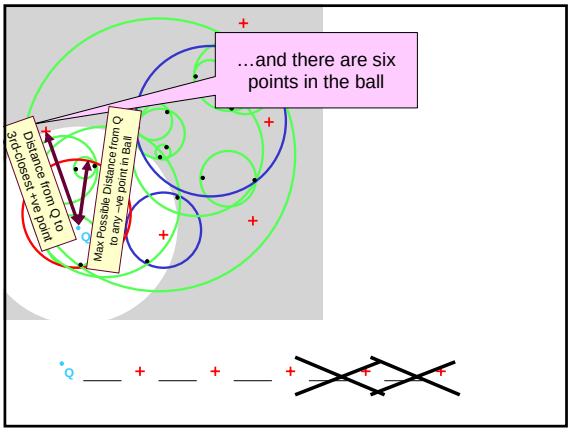

Q

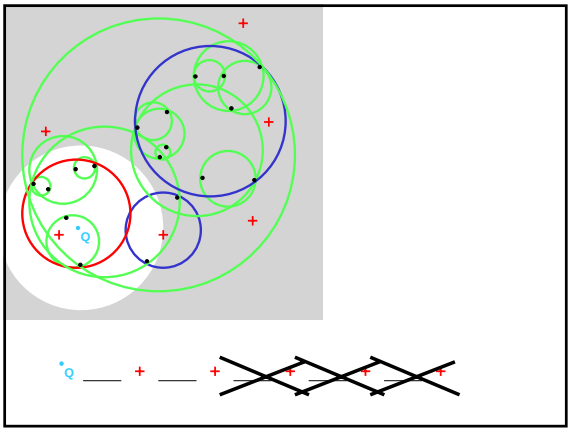

Distance from Q to 3rd-closest +ve point
Max Possible Distance from Q to any –ve point in Ball
Q

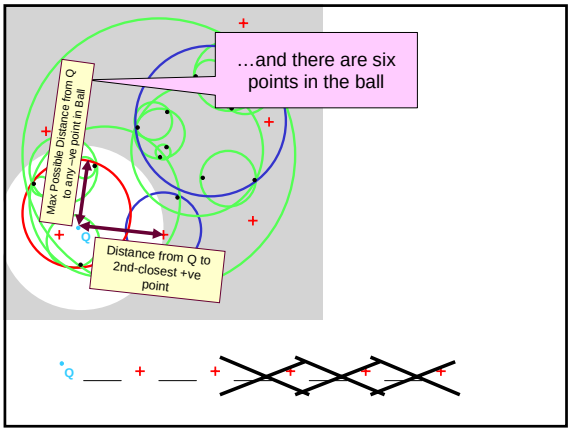

Distance from Q to 3rd-closest +ve point
Max Possible Distance from Q to any –ve point in Ball
No prune!
Q

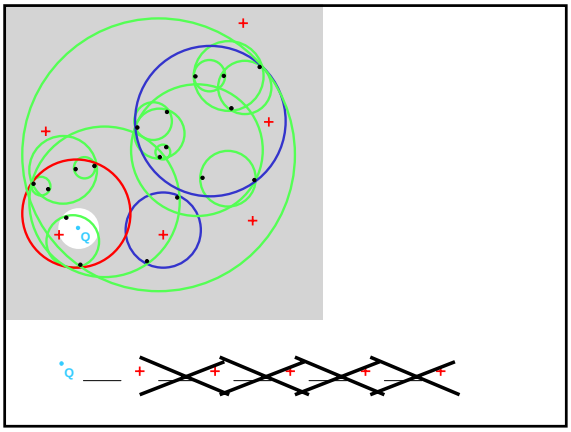

Q

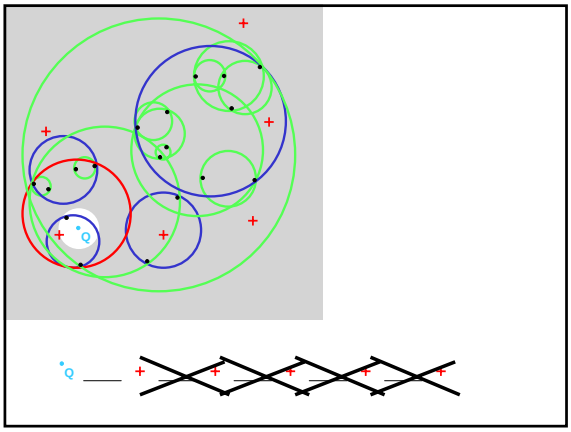

Q

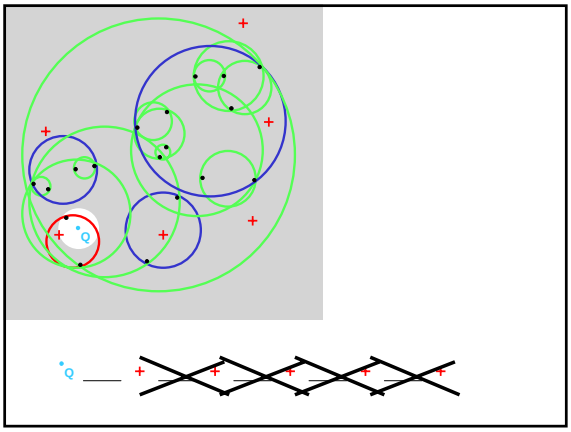

…and there are six points in the ball
Distance from Q to 3rd-closest +ve point
Max Possible Distance from Q to any –ve point in Ball
Q

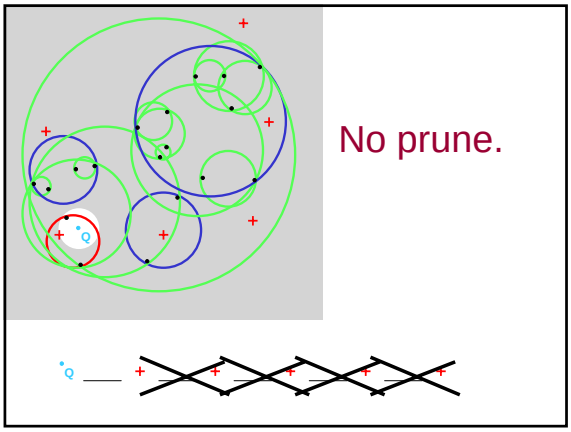

Q

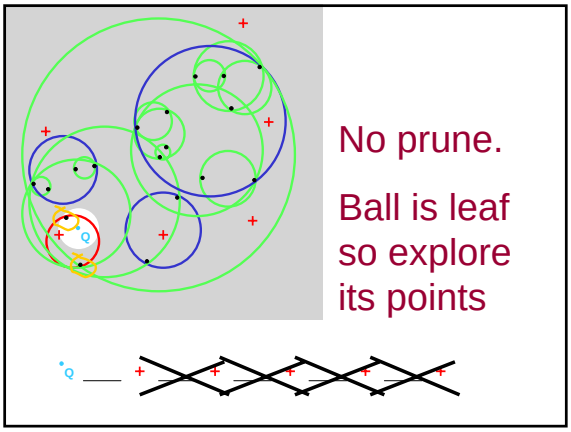

…and there are six points in the ball
Max Possible Distance from Q to any –ve point in Ball
Distance from Q to 2nd-closest +ve point
Q

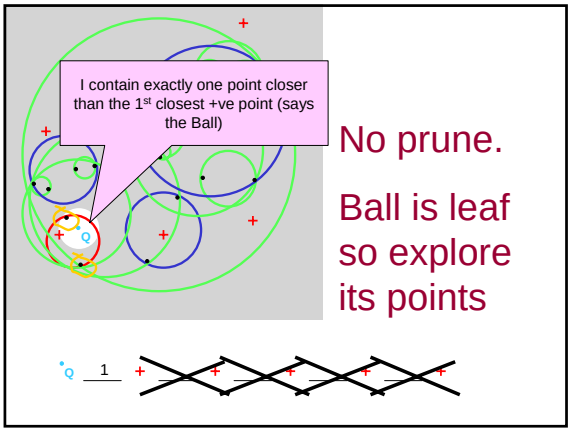

Q

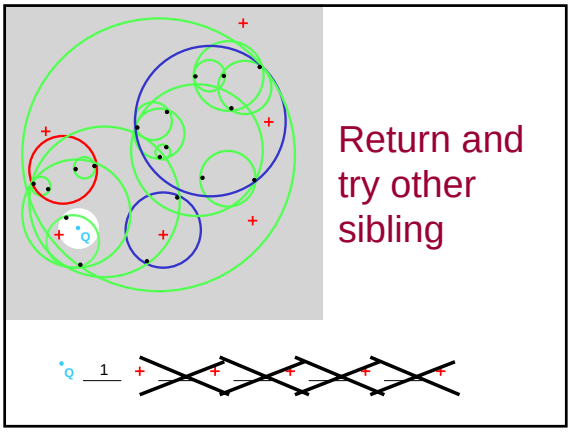

Q

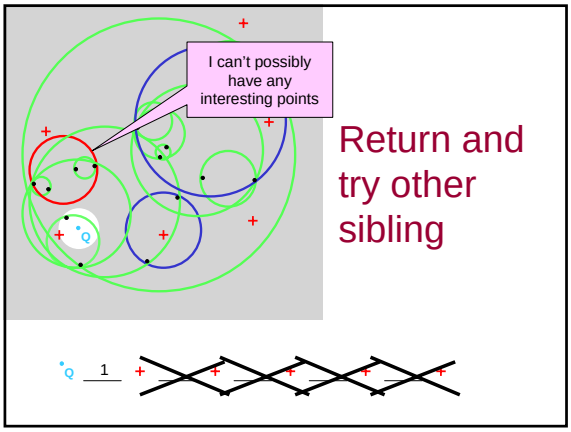

Q

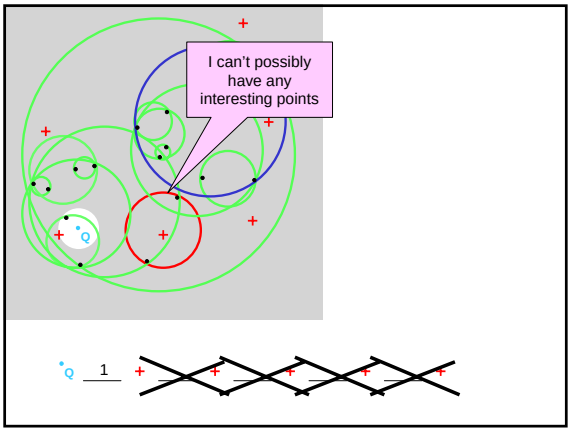

No prune.

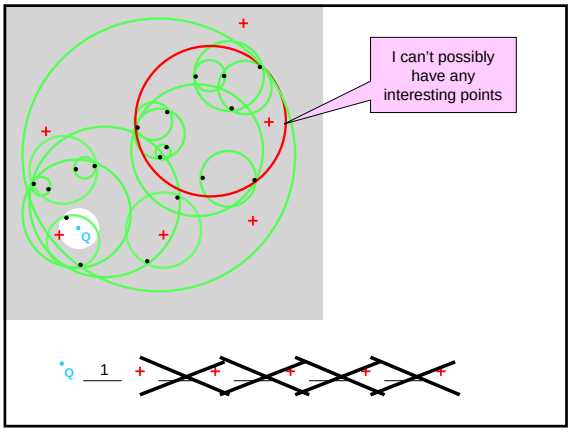

No prune.
Ball is leaf so explore its points

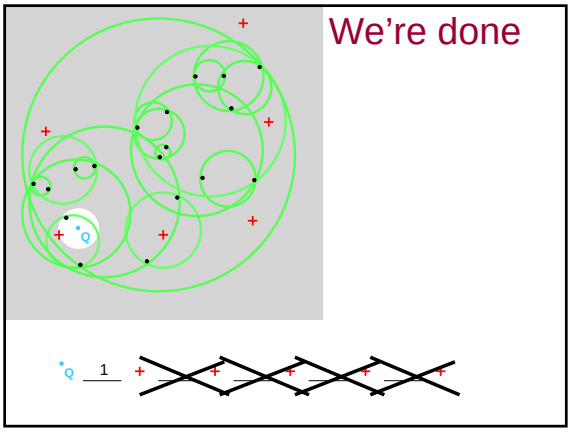

I contain exactly one point closer than the 1st closest +ve point (says the Ball)
No prune.
Ball is leaf so explore its points

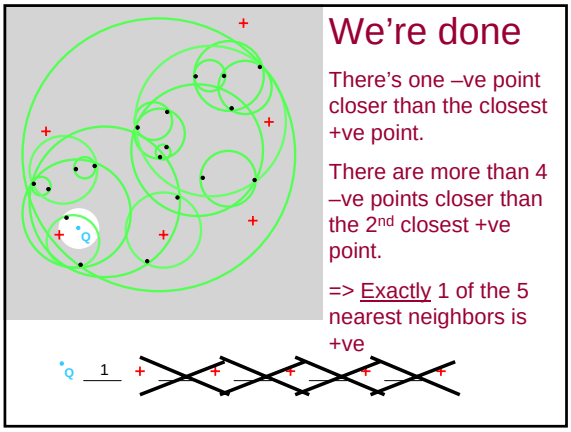

Return and try other sibling

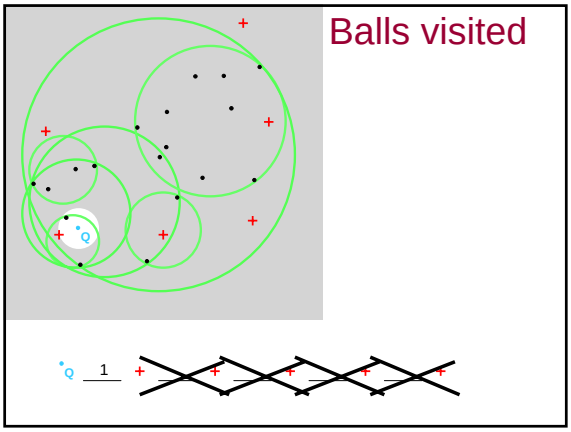

I can't possibly have any interesting points
Return and try other sibling

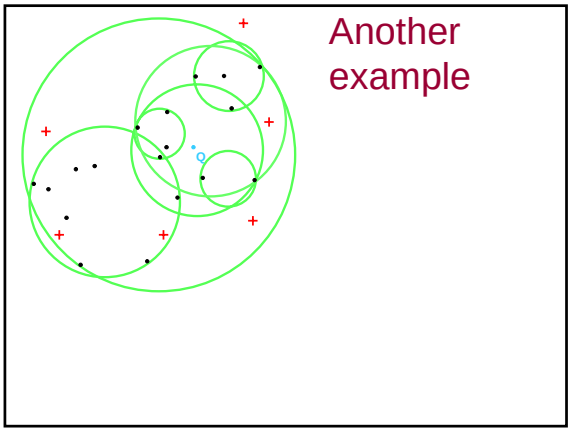

I can't possibly have any interesting points




## Conclusion

- It is often possible to exploit specific features of your statistical problem to give more pruning opportunities than were in the familiar geometrical problem.
- Here we worked with the special case of binary skewed classification: other cases have also been developed.