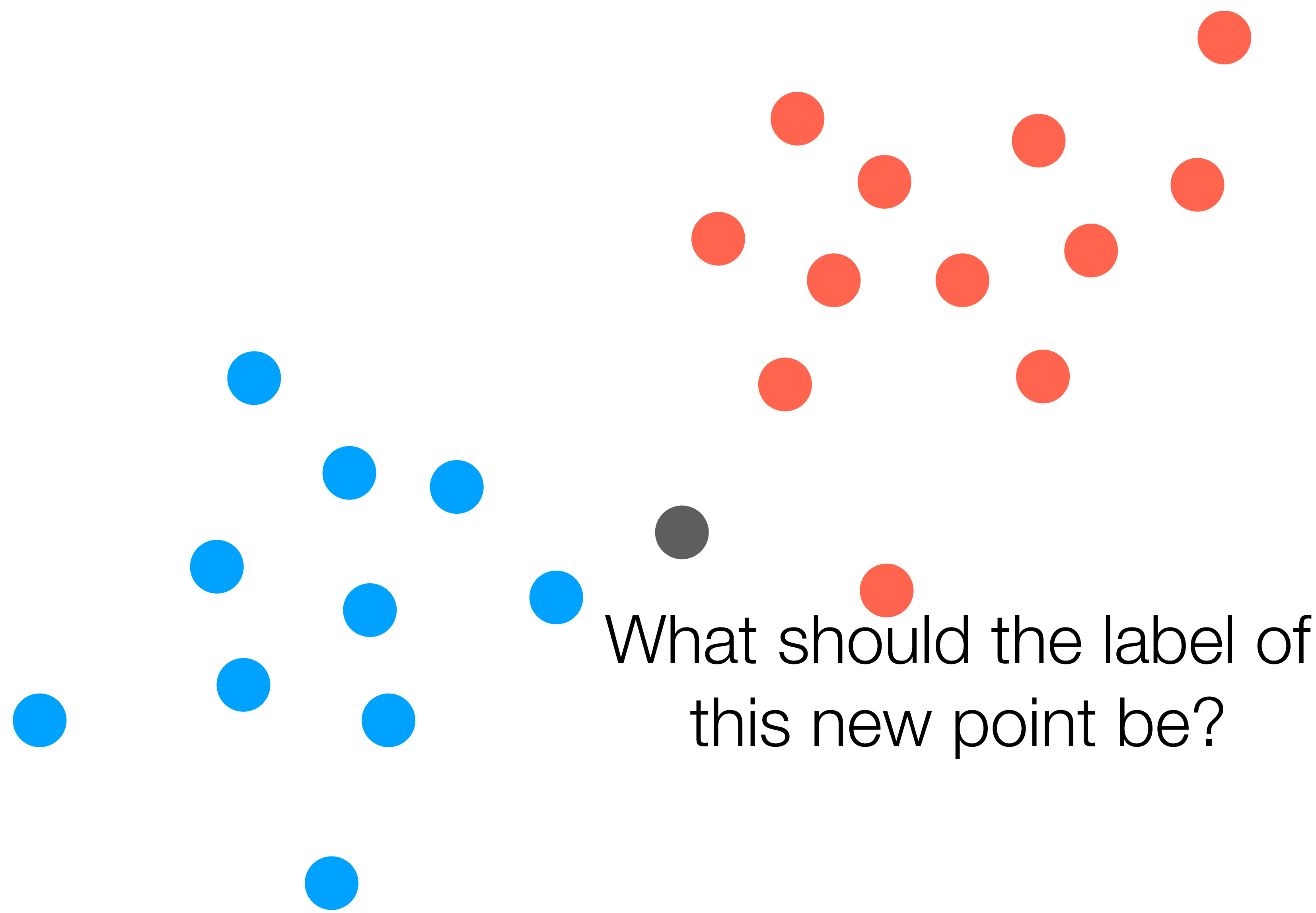
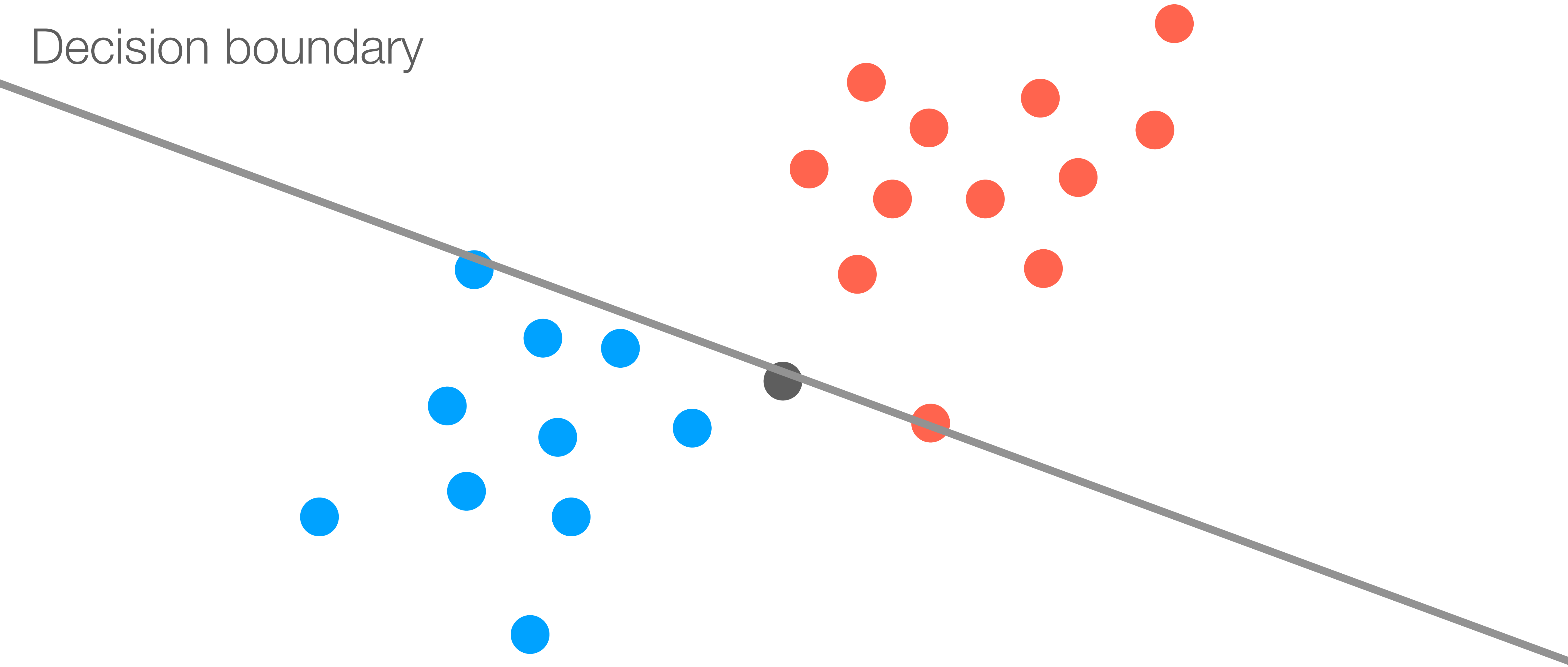
Carnegie Mellon University
Heinz College

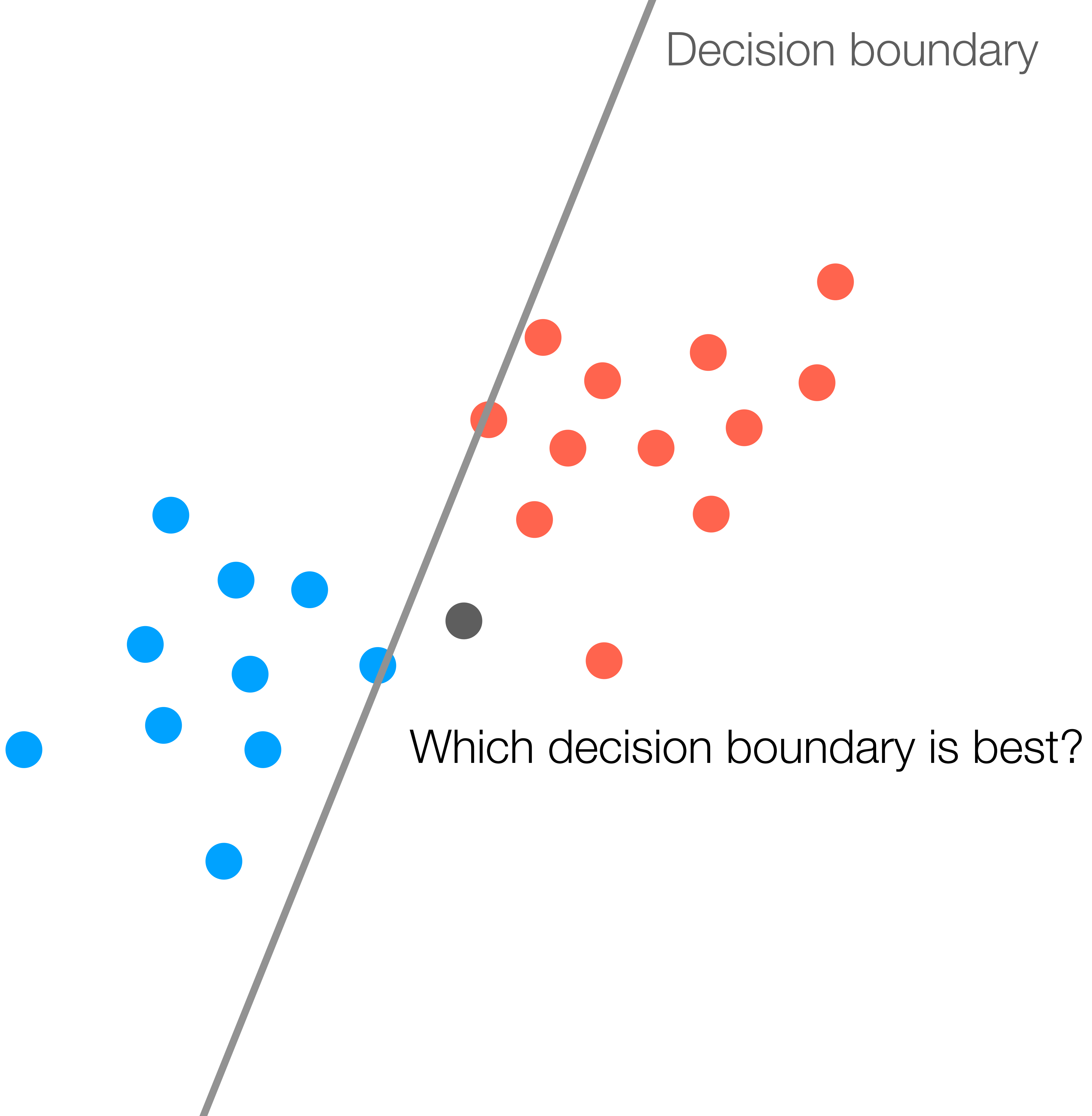
95-865: Support Vector Machines, Decision Trees and Forests

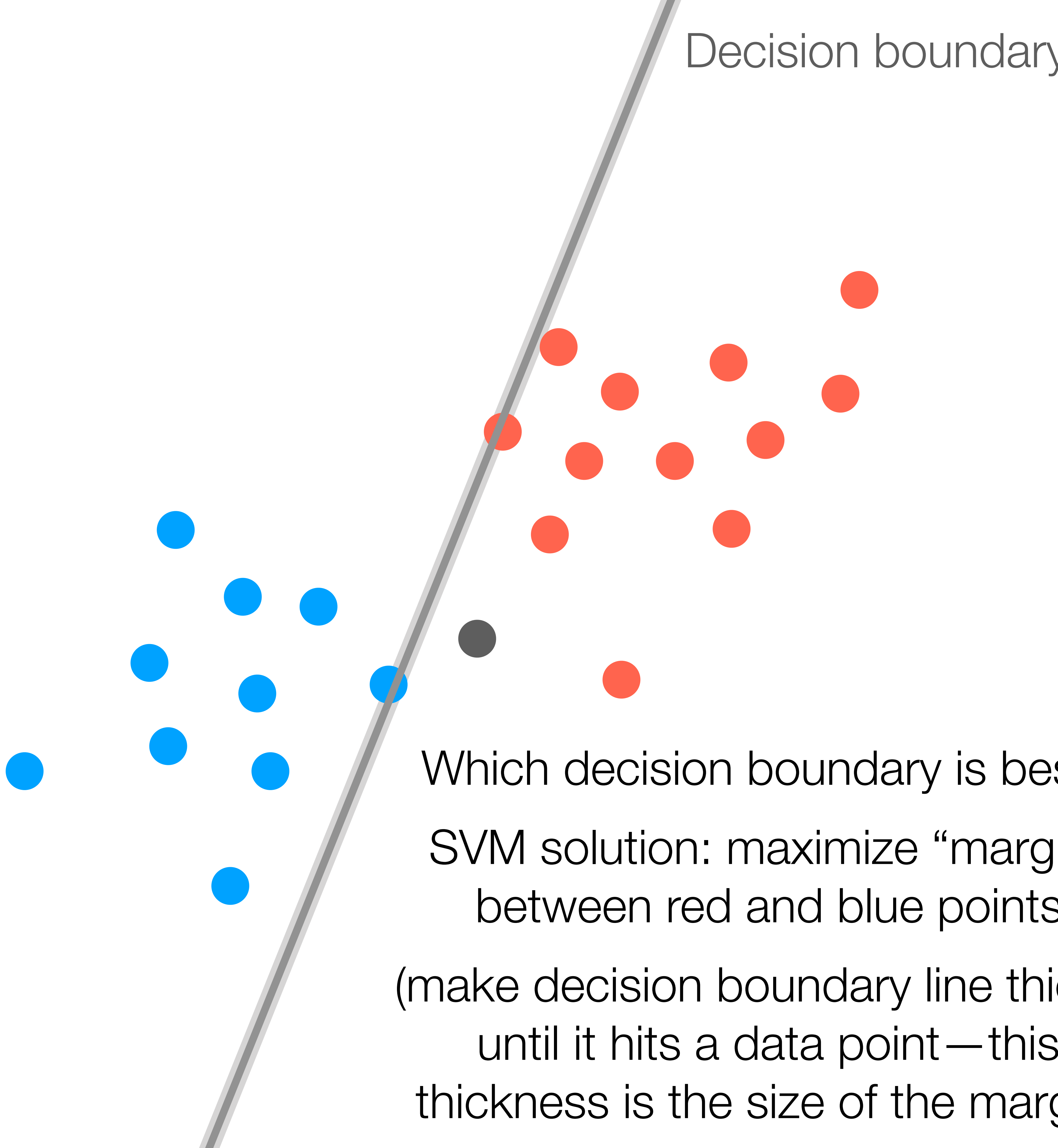
George Chen

Support Vector Machines


What should the label of this new point be?


Decision boundary

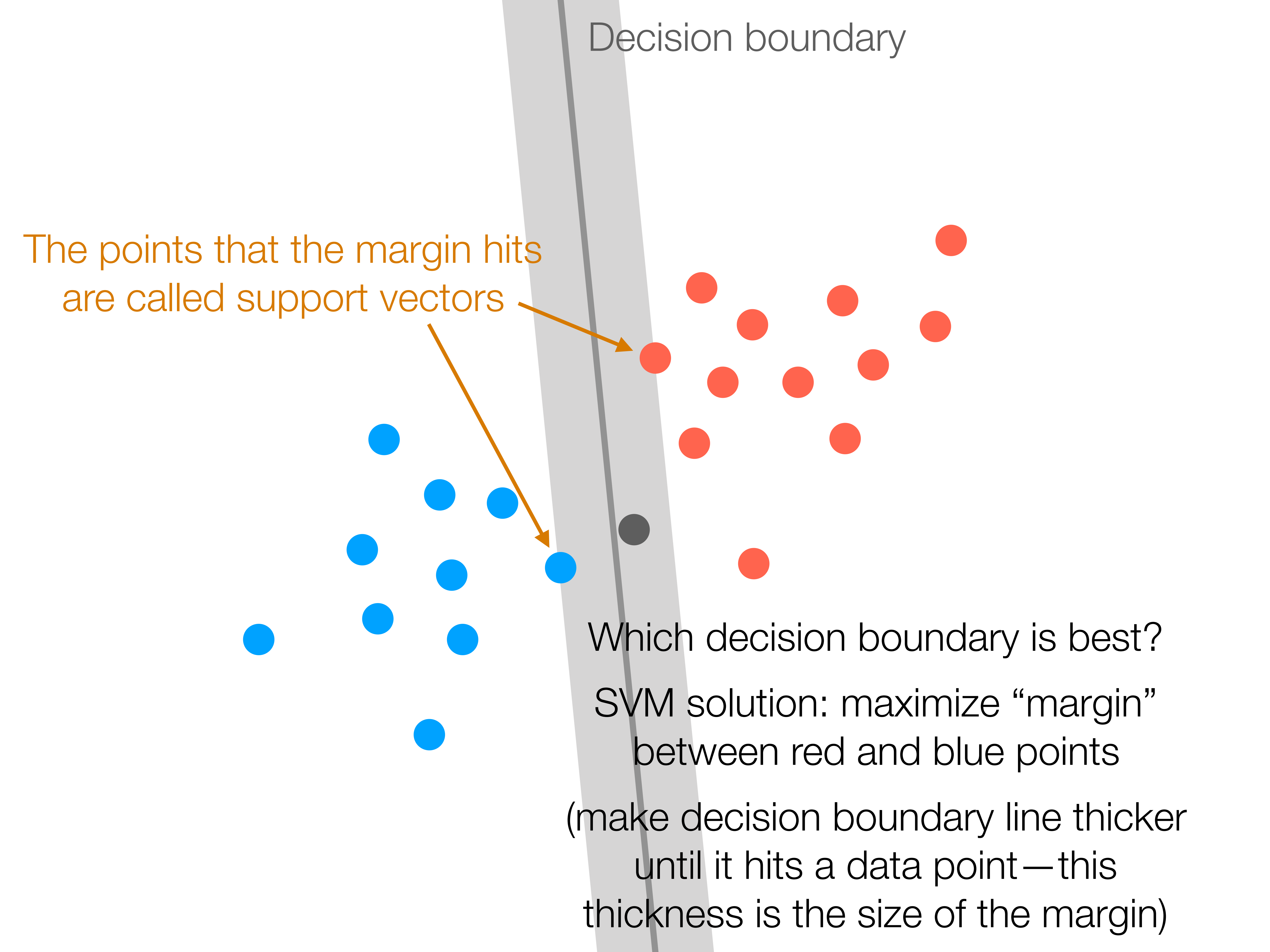

Decision boundary
Which decision boundary is best?

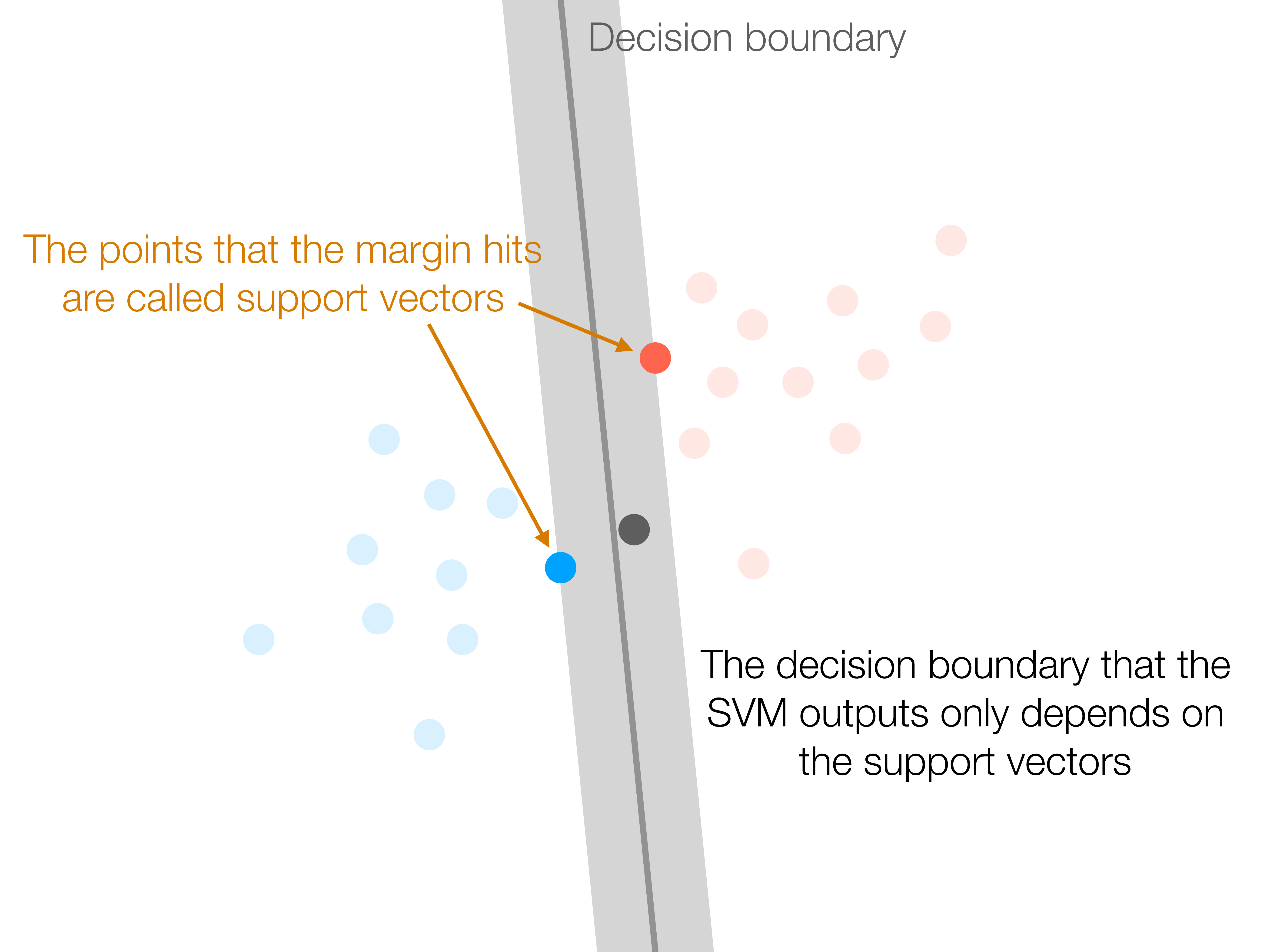


Which decision boundary is best?

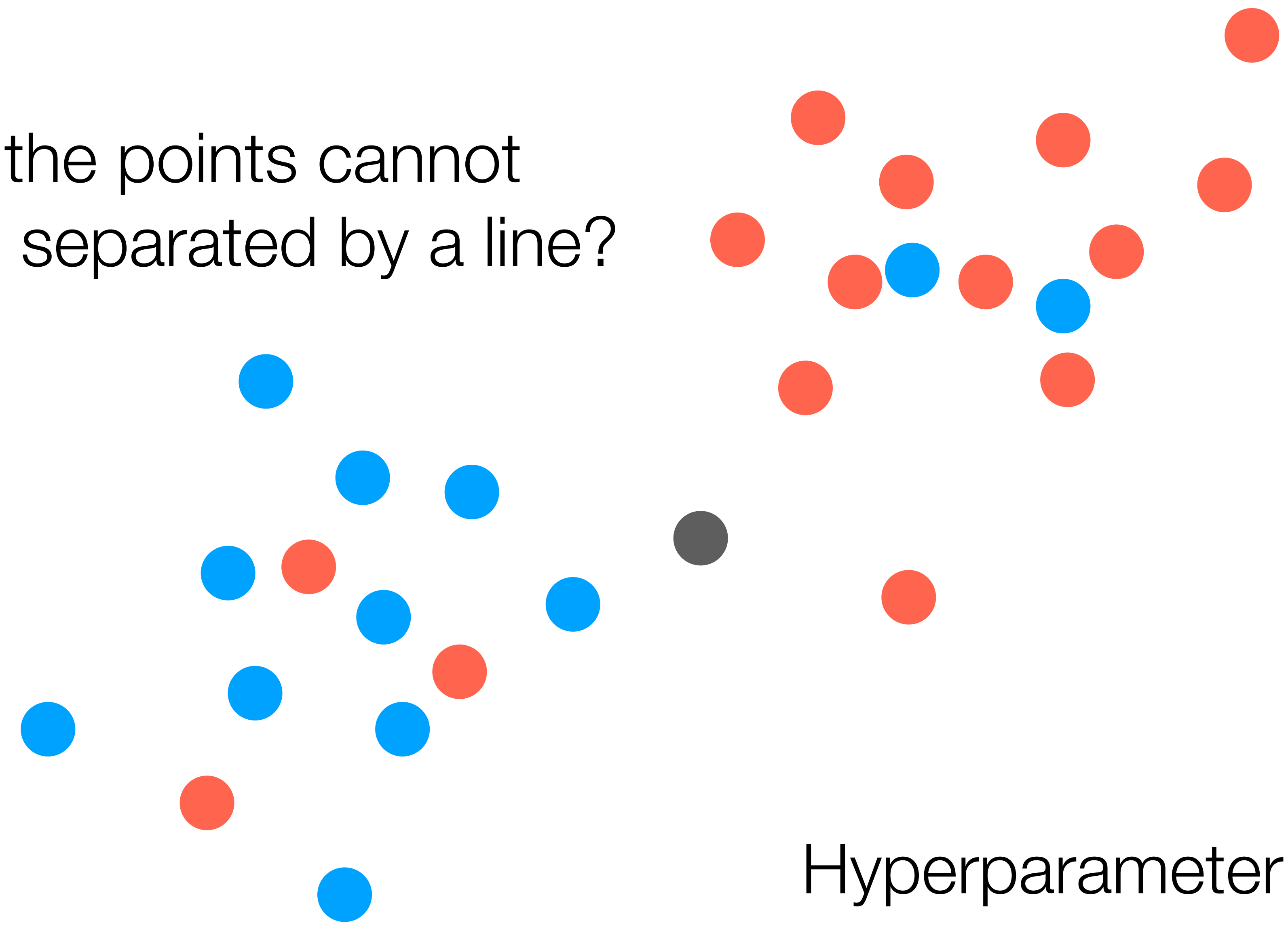
SVM solution: maximize “margin” between red and blue points

(make decision boundary line thicker until it hits a data point—this thickness is the size of the margin)


Decision boundary
The points that the margin hits
are called support vectors
Which decision boundary is best?
SVM solution: maximize “margin”
between red and blue points
(make decision boundary line thicker
until it hits a data point—this
thickness is the size of the margin)


Decision boundary
The points that the margin hits
are called support vectors
The decision boundary that the
SVM outputs only depends on
the support vectors

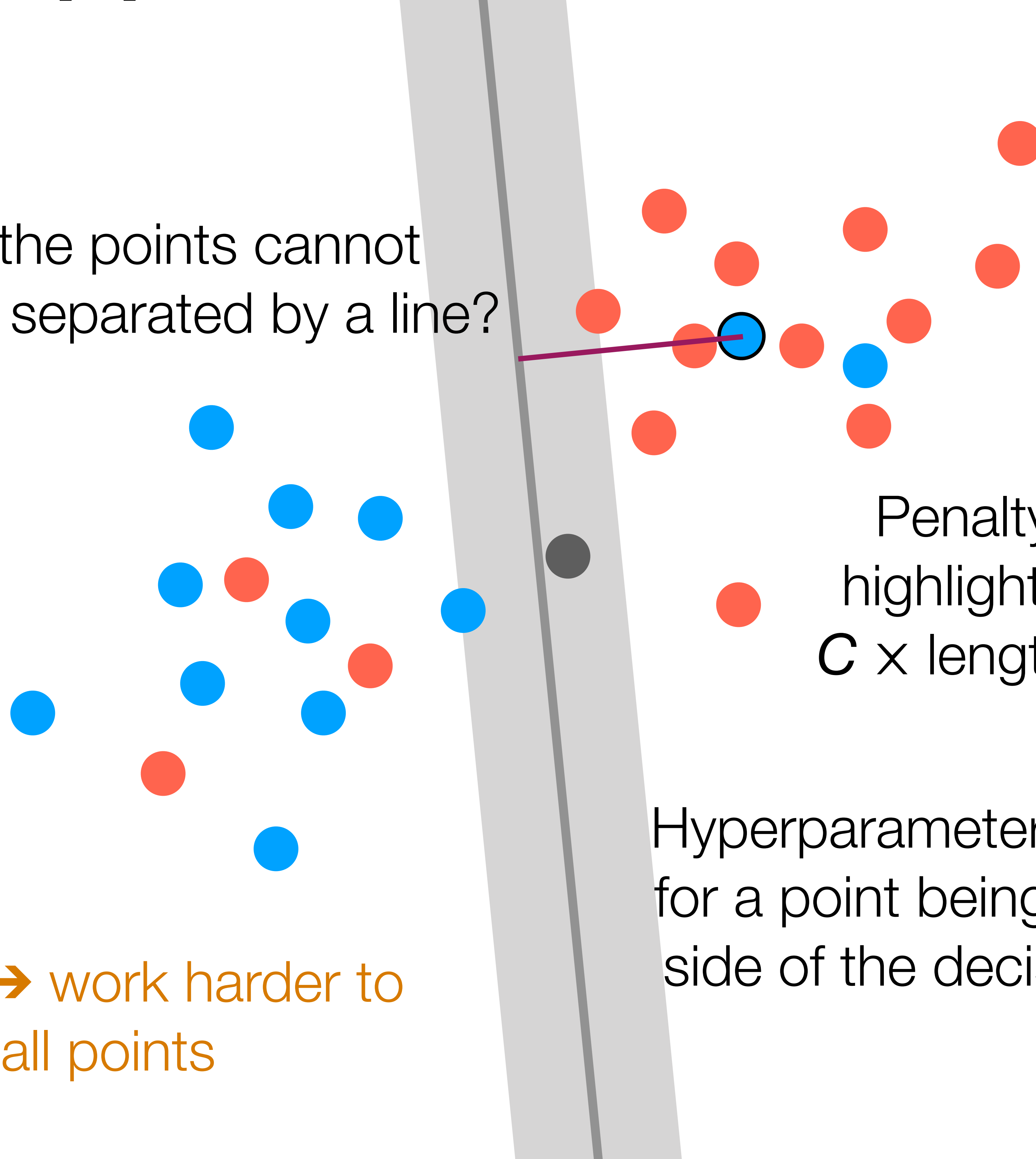
What if the points cannot
actually be separated by a line?



Hyperparameter ***C*** is a penalty
for a point being on the wrong
side of the decision boundary

C-Support Vector Classification

What if the points cannot actually be separated by a line?



Penalty incurred for highlighted blue point: *C* × length of purple line

Hyperparameter ***C*** is a penalty for a point being on the wrong side of the decision boundary

Larger ***C*** ➔ work harder to fit all points

C-Support Vector Classification

- Basic version measures distance using Euclidean distance
  - Turns out to correspond to measuring similarity between two points by taking their dot product

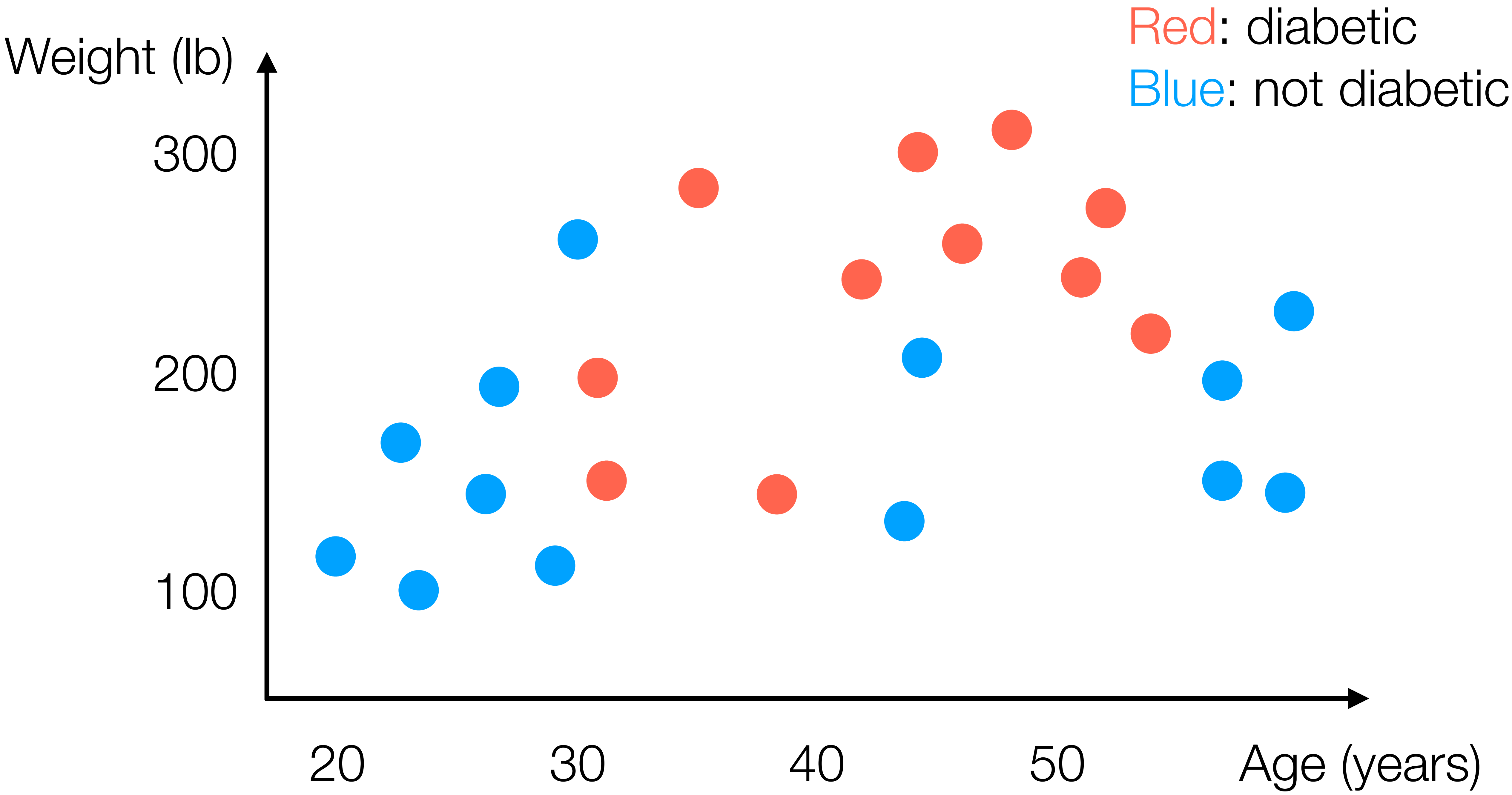
- Can instead use a different similarity function (“kernel” function) instead (popular choice: Gaussian kernel, also called “radial basis function” kernel)

C-Support Vector Classification

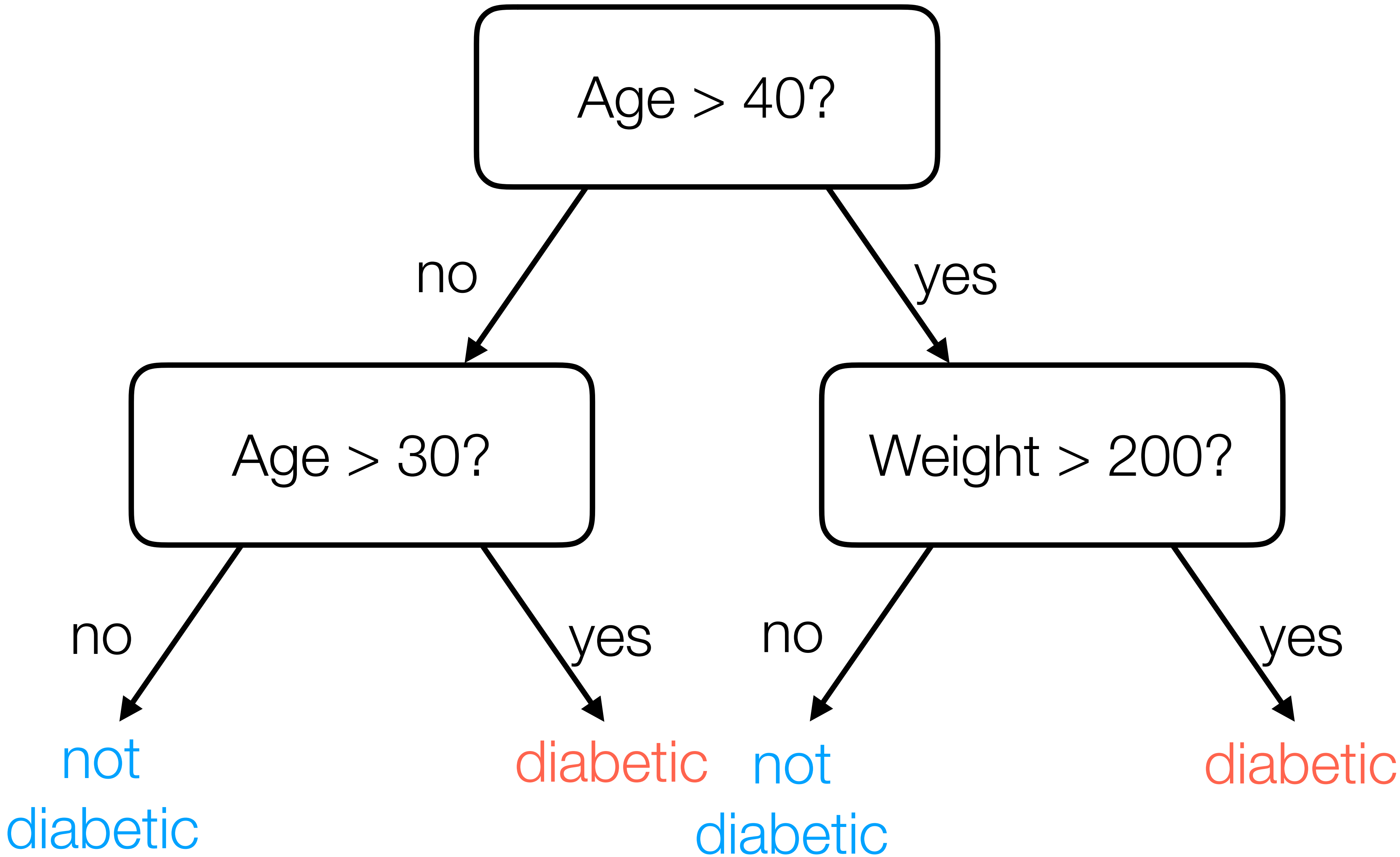
Demo

Decision Trees

Example Made-Up Data



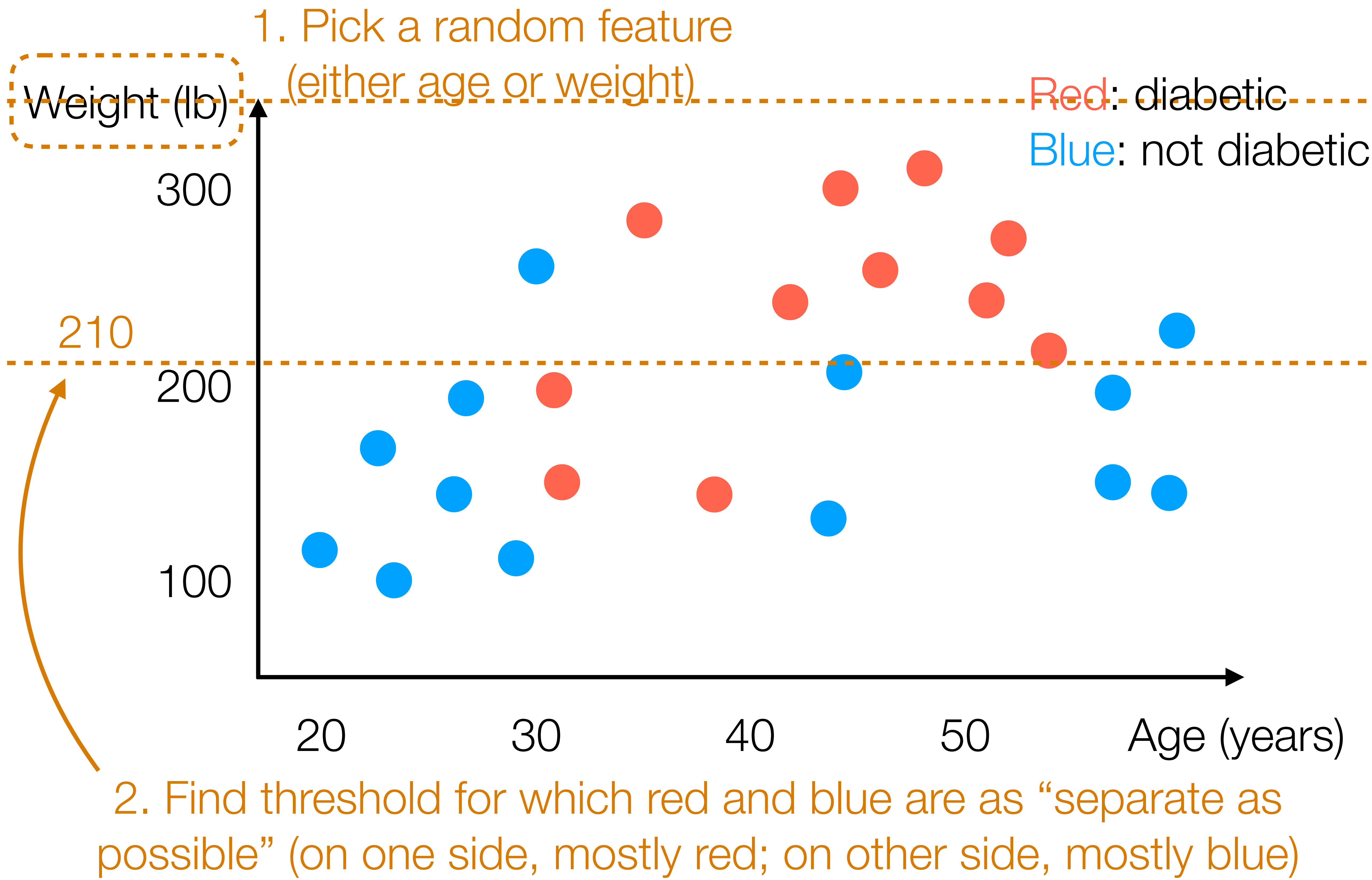
Example Decision Tree



Learning a Decision Tree

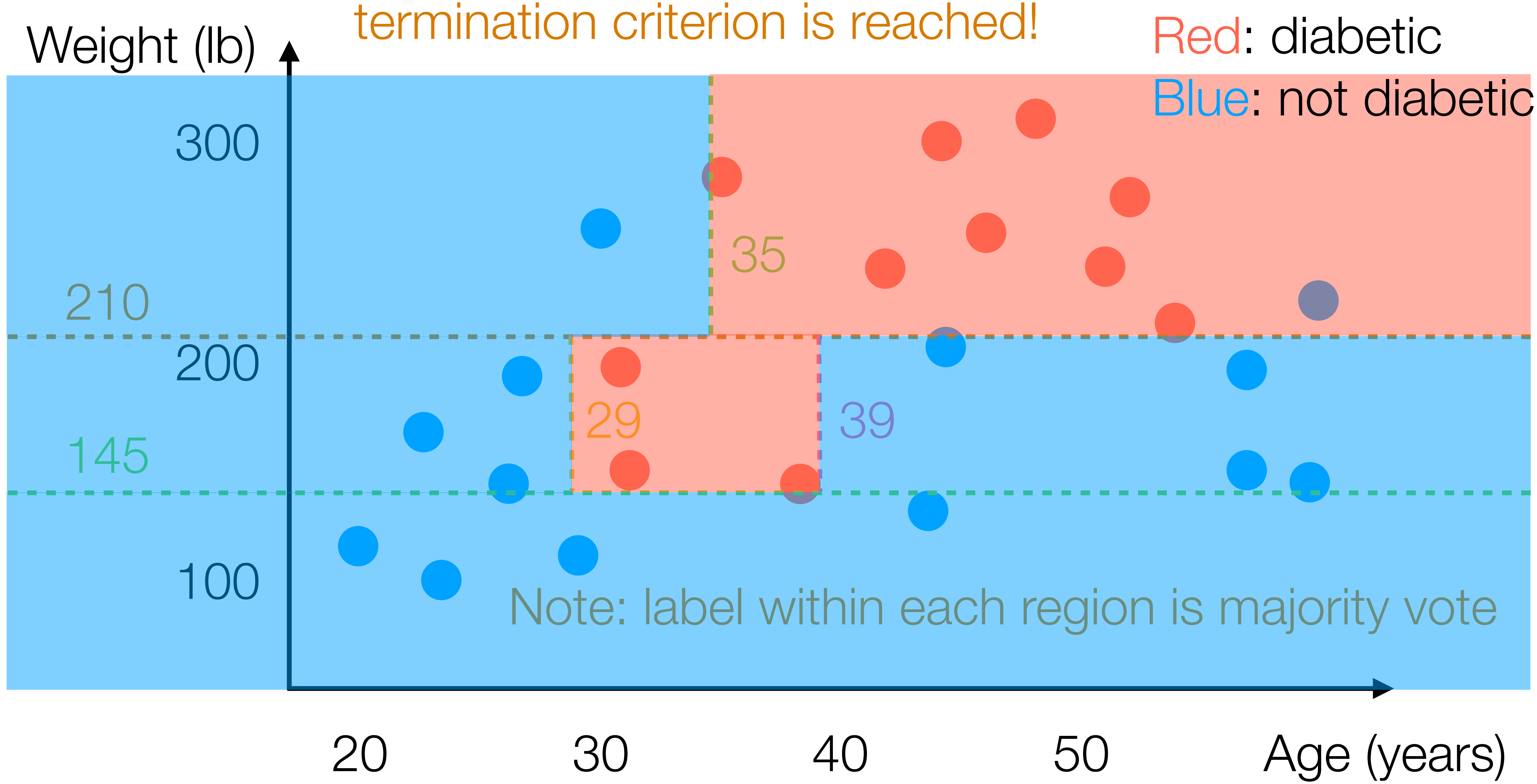
- Many ways: general approach actually looks a lot like divisive clustering *but accounts for label information*
- I'll show one way (that nobody actually uses in practice) but it's easy to explain

Learning a Decision Tree



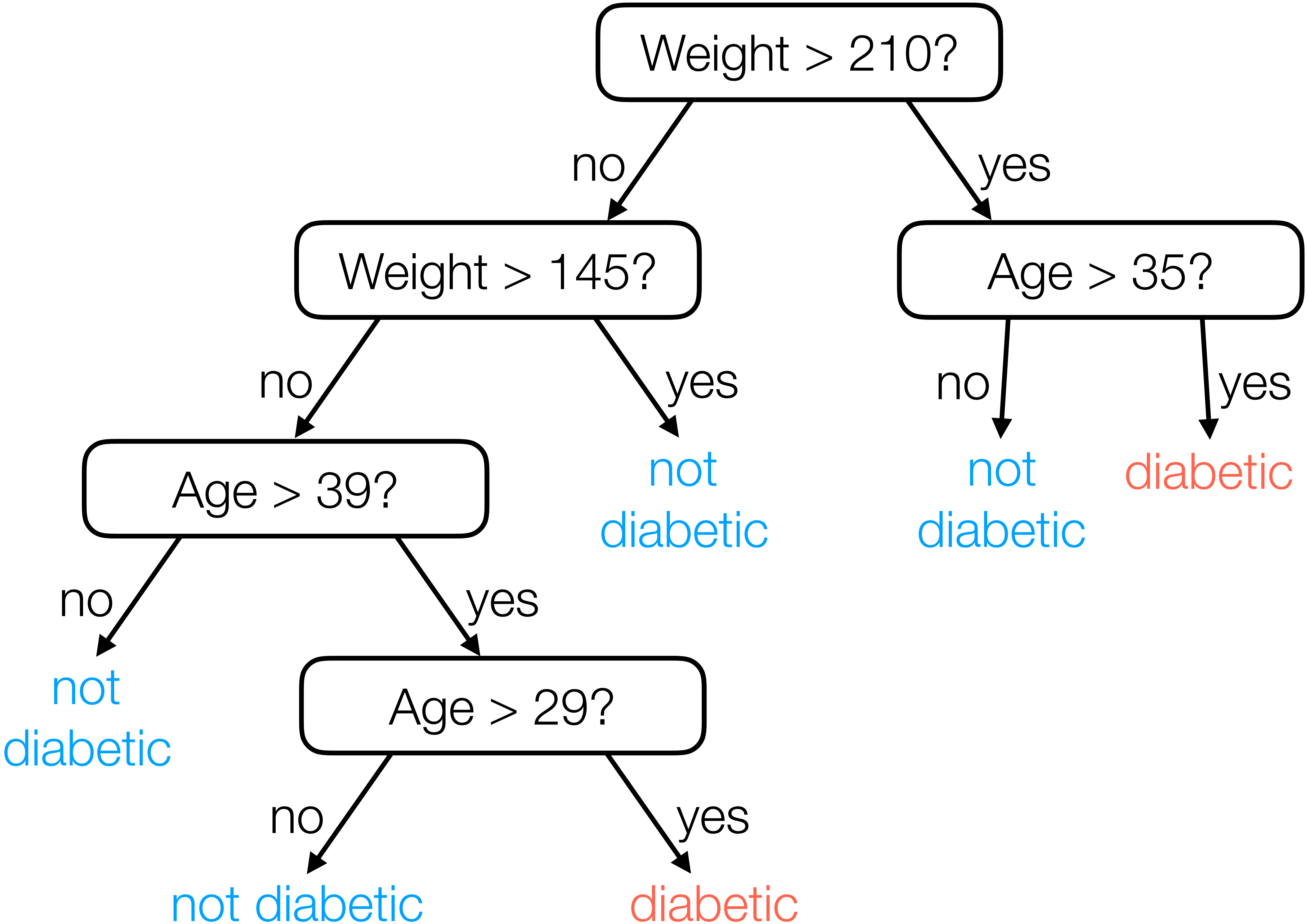
Learning a Decision Tree

Within each side, recurse until a termination criterion is reached!



Example termination criteria: ≥90% points within region has same label, number of points within region is <5

Decision Tree Learned



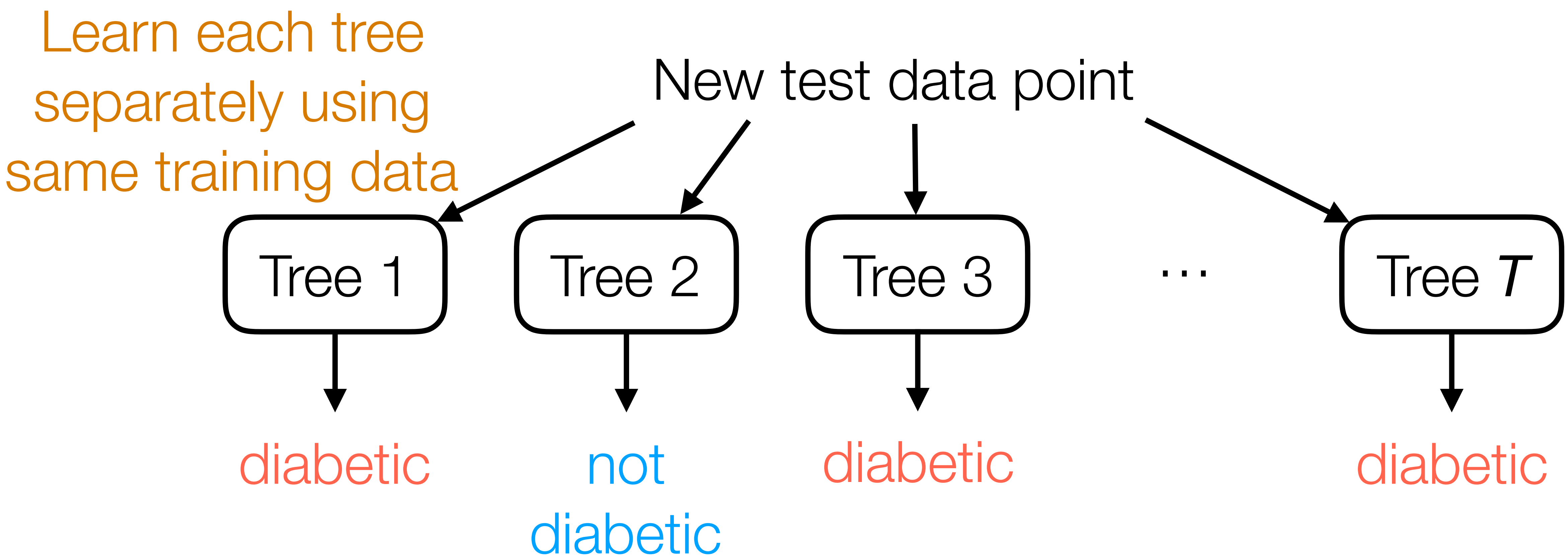
For a new person with feature vector (age, weight), easy to predict!

Decision Forest for Classification

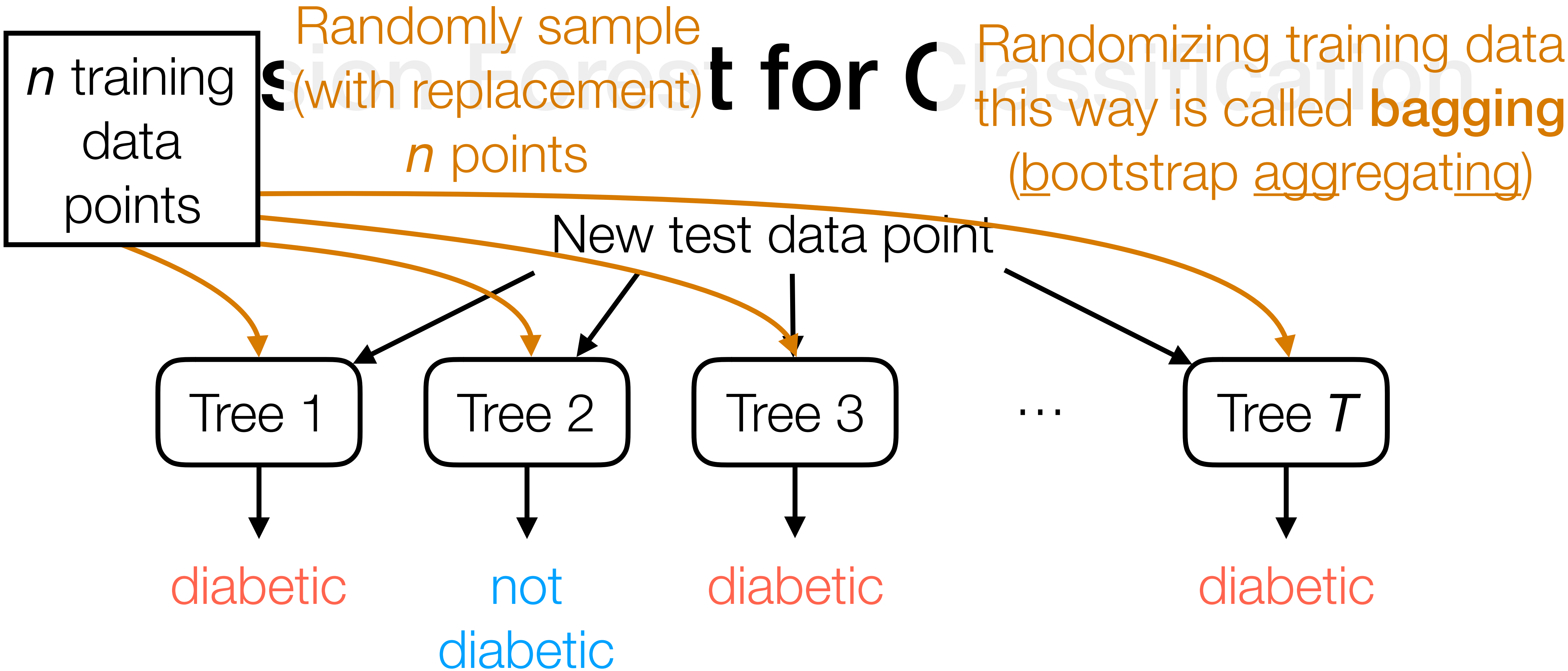
- Typically, a decision tree is learned with randomness (e.g., we randomly chose which feature to threshold)

  ➔ by re-running the same learning procedure, we can get different decision trees that make different predictions!

- For a more stable prediction, use many decision trees



Final prediction: majority vote of the different trees' predictions



Question: What happens if all the trees are the same?

Adding randomness can make trees more different!

- **Random Forest:** in addition to randomly choosing features to threshold, also randomize training data used for each tree
- **Extremely randomized trees:** further randomize thresholds rather than trying to pick clever thresholds