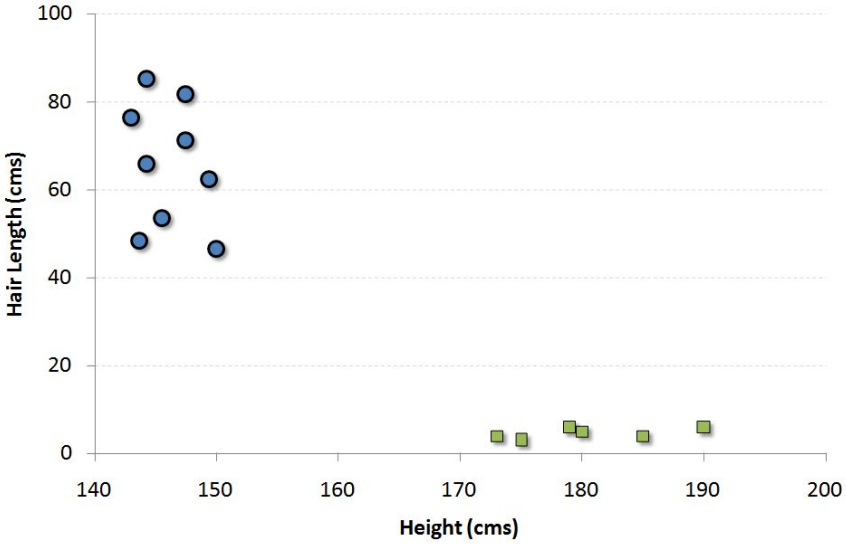
Support Vector Machines

By [redacted]

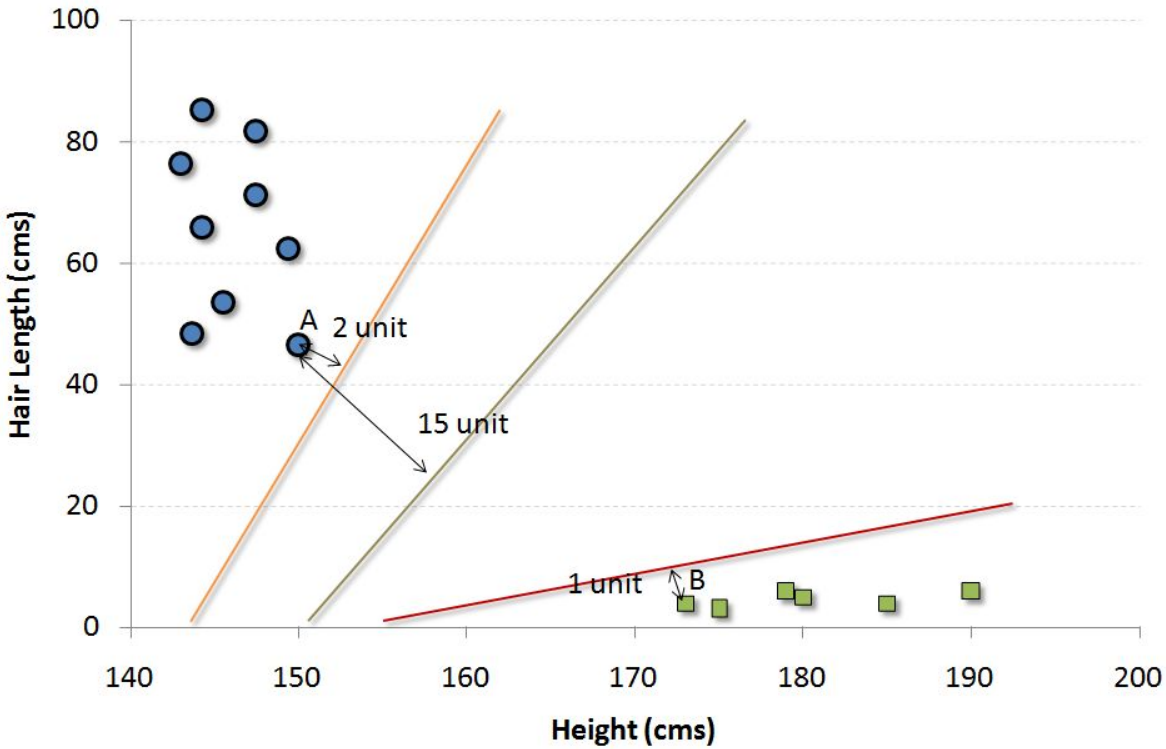
what is a Support Vector Machine?

- supervised ML Algorithm
- mostly used for classification
- finds a hyperplane to separate datapoints in n-dimensional space
  - n is the number of features you have
- Support Vectors: another word for coordinates of each datapoint
- Support Vector Machine: a frontier that best separates data into groups



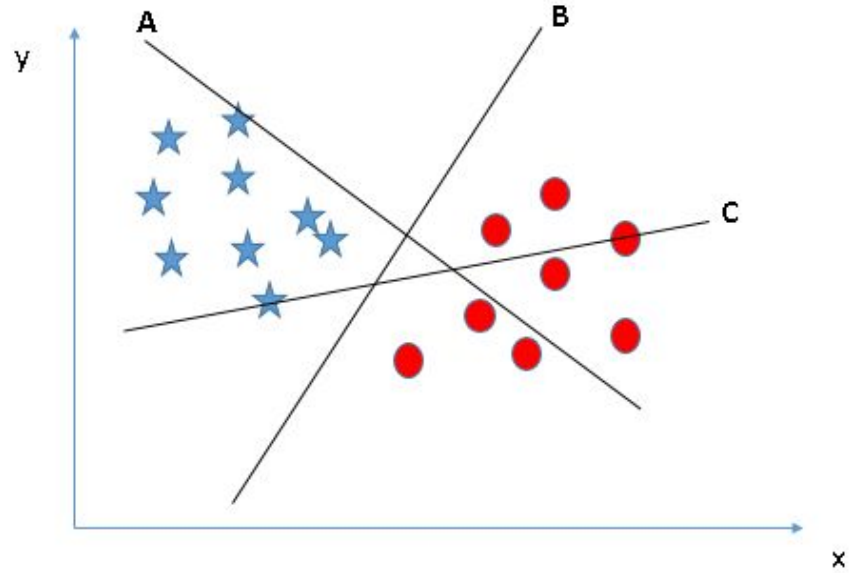
what does it mean to “best separate”?

- these all separate the datapoints into their respective categories
- what makes one better than another?



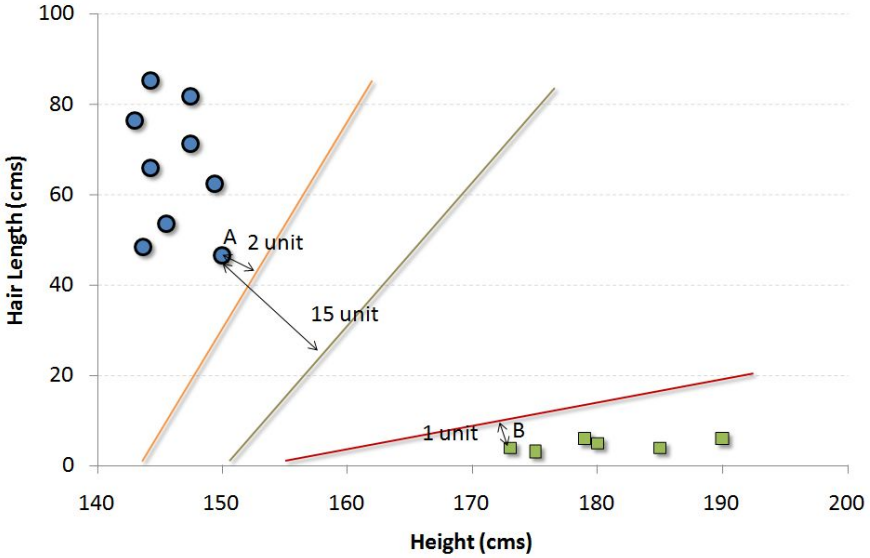
rules for creating good frontiers

- top priority is accuracy:
  dividing the datapoints into their respective categories
- best line here is line B



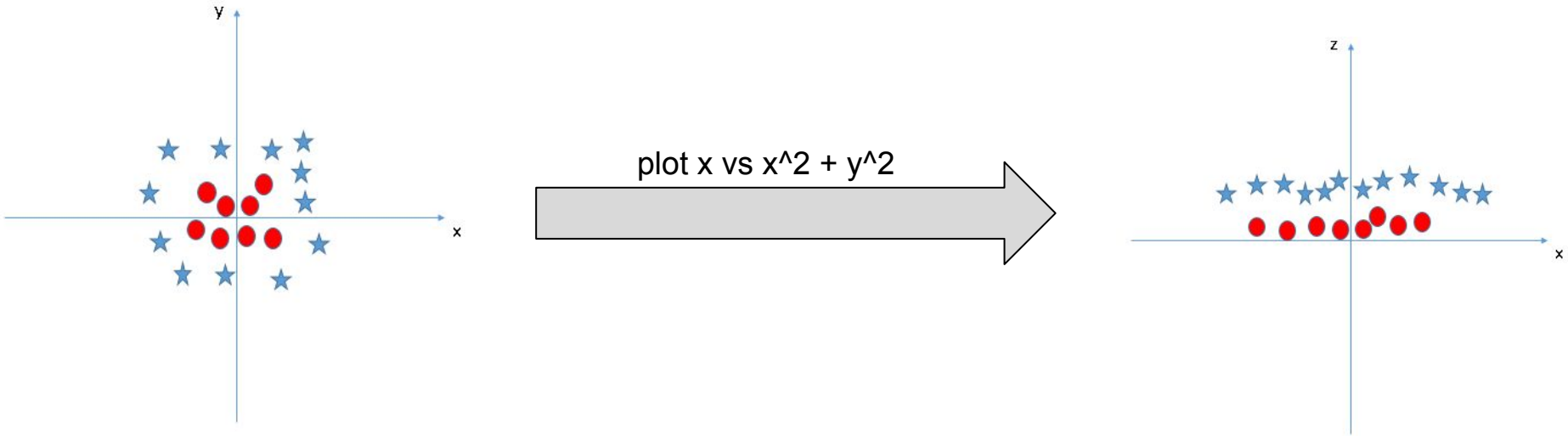
rules for creating good frontiers

- the second priority is maximizing margin: the distance between any given point and the frontier
- the reduces the chance we will misclassify data
- in this graph, the green line is the best choice to divide the data



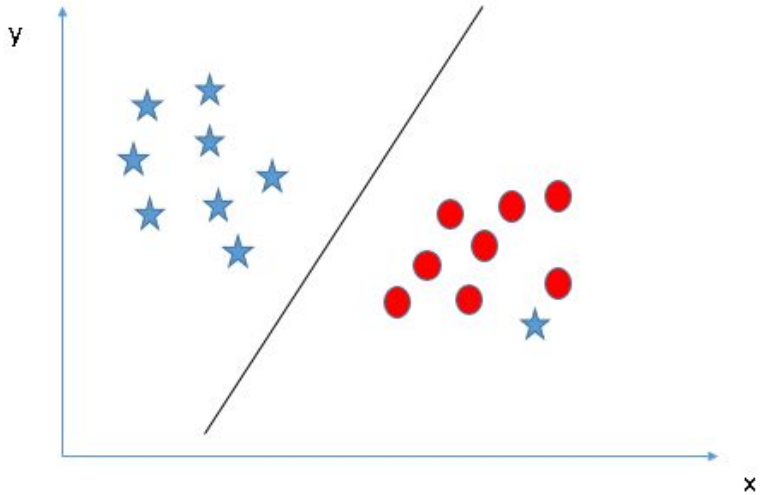
the kernel trick

- what if we don't have an easy linear division between the categories?
- SVMs use something called the kernel trick to embed the data into a higher dimension, allowing it to exhibit more linear separation



hyperplanes, outliers, etc.

- the notion of a ‘hyperplane’ is just a higher dimensional frontier to separate the datapoints
  - this allows for more possibilities than simple lines, as shown previously
  - SVMs contain algorithms to automatically do the kernel trick for you, so you don’t have to worry about it
- SVMs also have some tolerance for outliers, so that more simple solutions can be found to datasets with a few bad outliers
- all of these parameters can be manipulated in most implementations of SVMs



pros and cons

Pros:

- It works really well with clear margin of separation
- It is effective in high dimensional spaces.
- It is effective in cases where number of dimensions is greater than the number of samples.
- It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.

Cons:

- It doesn’t perform well, when we have large data set because the required training time is higher
- It also doesn’t perform very well, when the data set has more noise i.e. target classes are overlapping
- SVM doesn’t directly provide probability estimates, these are calculated using an expensive five-fold cross-validation. It is related SVC method of Python scikit-learn library.