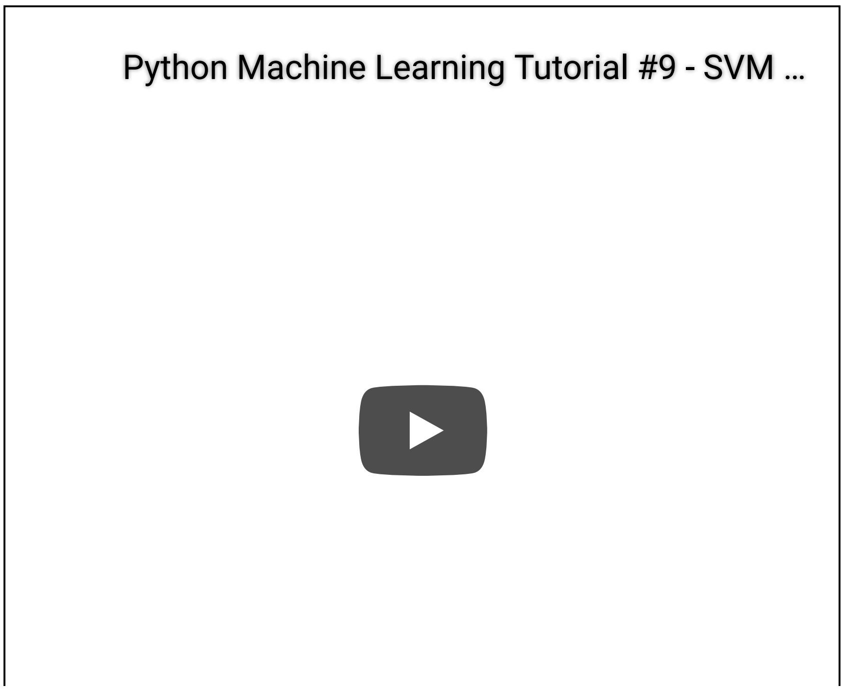


Subscribe to Tech With Tim! YouTube 1M

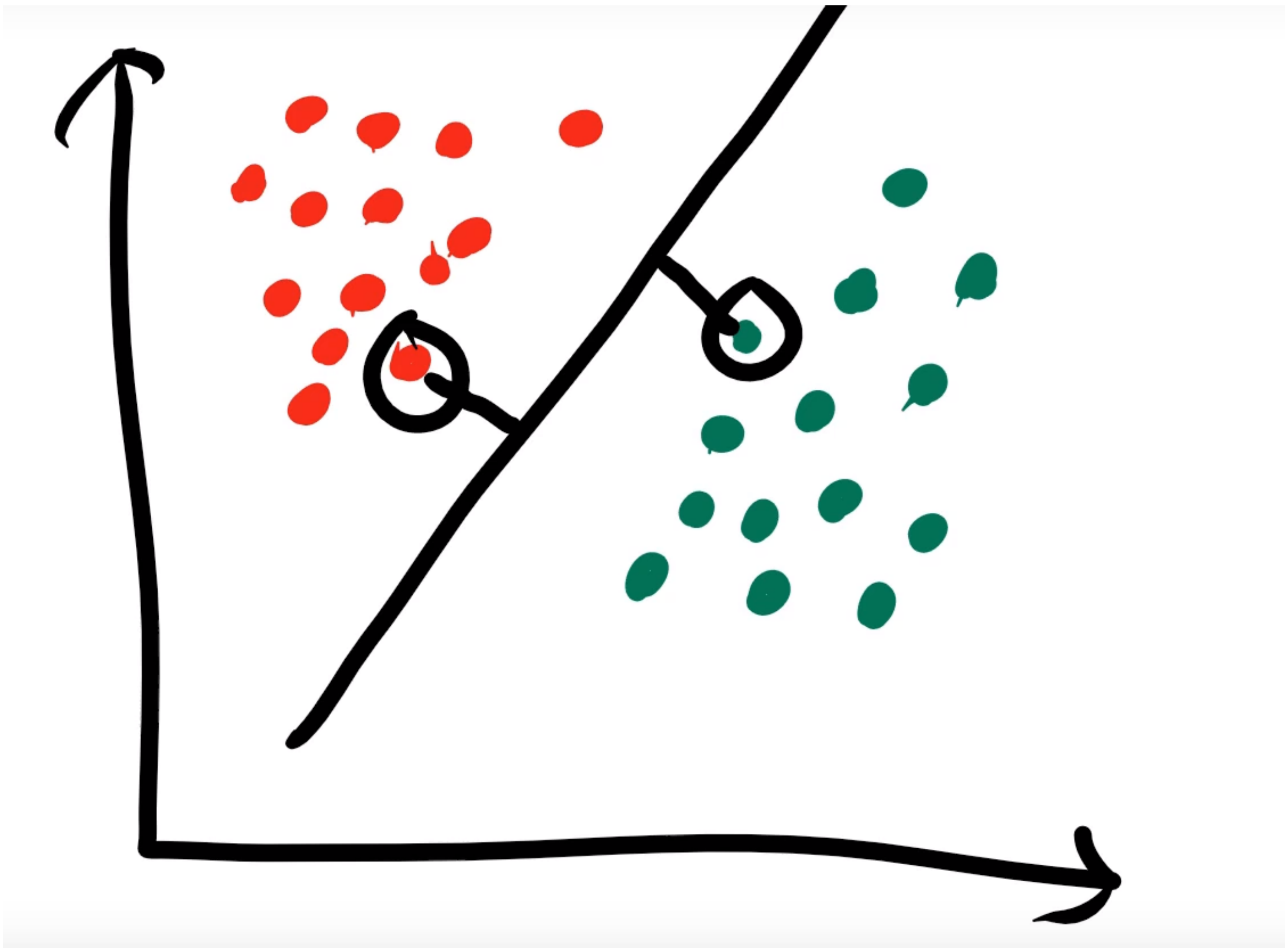
What a SVM Does?

A SVM has a large list of applicable uses. However, in machine learning it is typically used for classification. It is a powerful tool that is a good choice for classifying complicated data with a high degree of dimensions(features). Note that K-Nearest Neighbors does not perform well on high-dimensional data.

How A Support Vector Machine Works

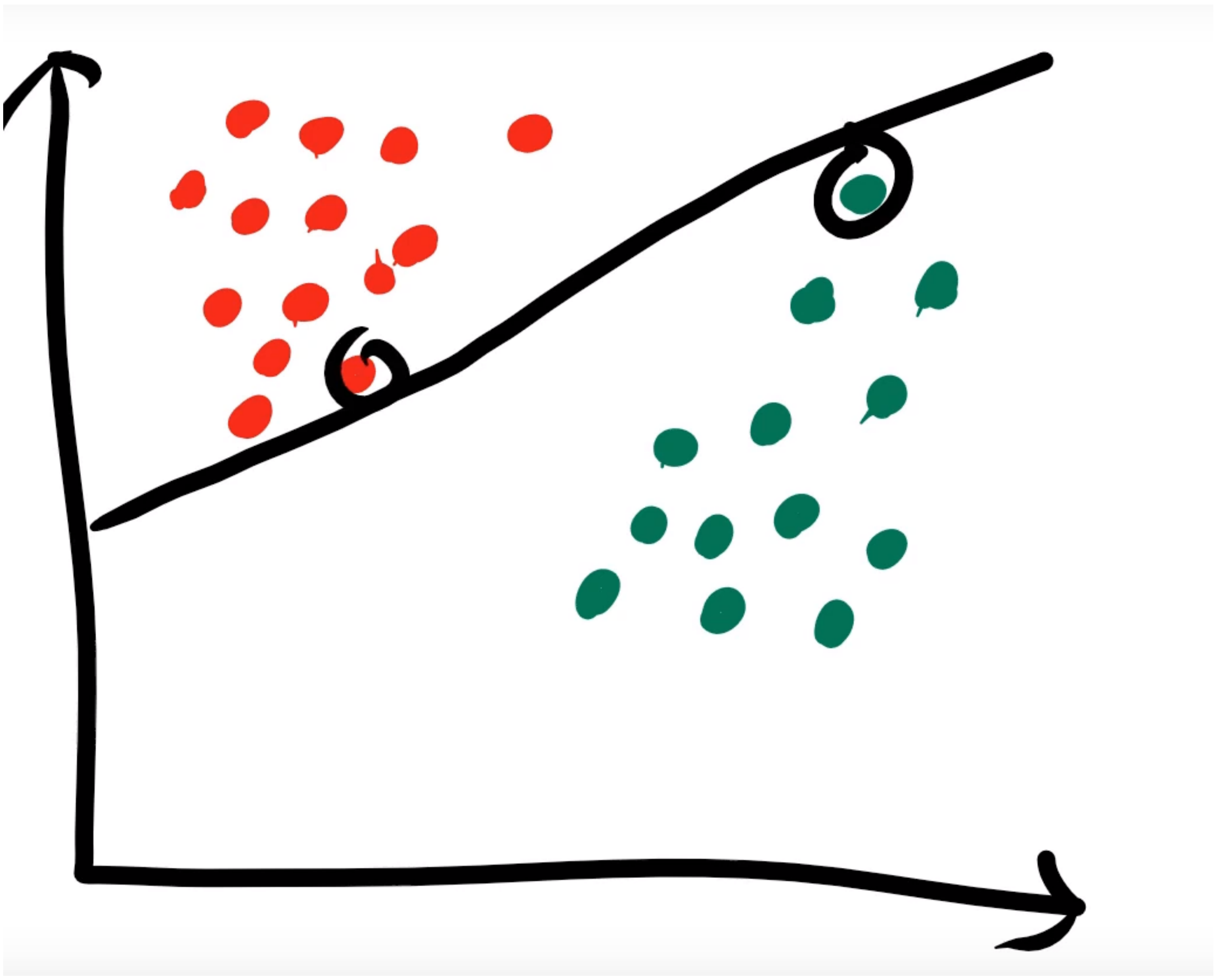
In short a support vector machine works by dividing data into multiple classes using something called a **hyper-plane**. A hyper plane is a fancy word for something that is straight that can divide data points. In 2D space a hyper-plane is simply a line, in 3D space a hyper-plane is a plane. In any space higher than 3D it is simply called a hyper-plane.

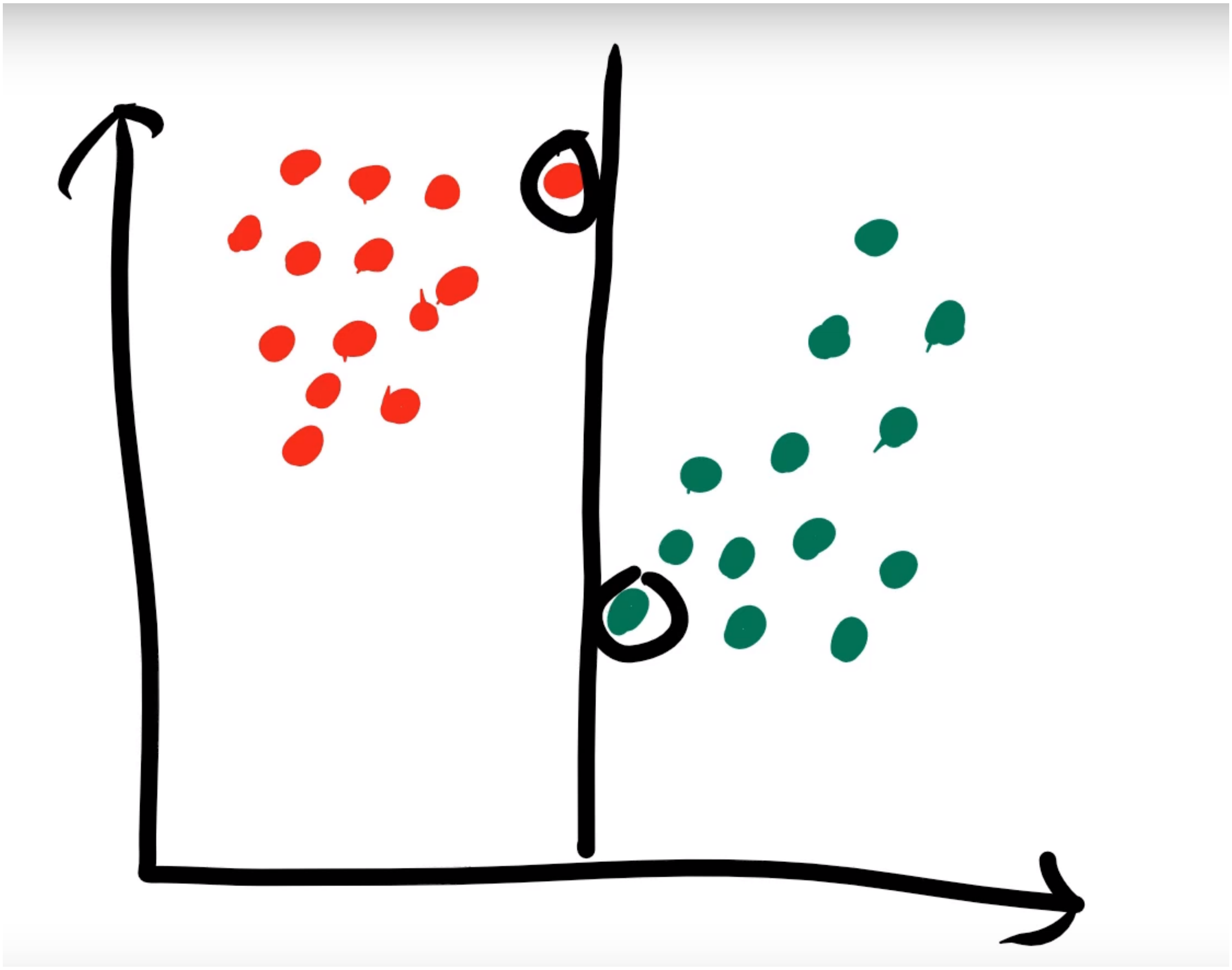
Here's an example of a hyper-plane for the data points on the 2D graph below.



Hyper-Planes

When we create a hyper-plane we need to do the following. We must pick two points that are known as our **support vectors**. These points must be the two closest points to the hyper-plane and their distance from the hyper-plane must be identical. In the example above we can see that the two circled points are our support vectors and their distance to the hyper-plane is the same, they are also the closest points. With this rule we can actually create an infinite amount of hyper planes (see below).




All of the images above are valid hyper-planes.

Picking a Hyper Plane

Once we create a hyper-plane we are going to use it to classify our data. If a test point is on the left side of the plane we would classify it as red (in our examples above) and if it is on the right we would classify it as green. So how can we pick a hyper-plane that will give us the best classification predictions?

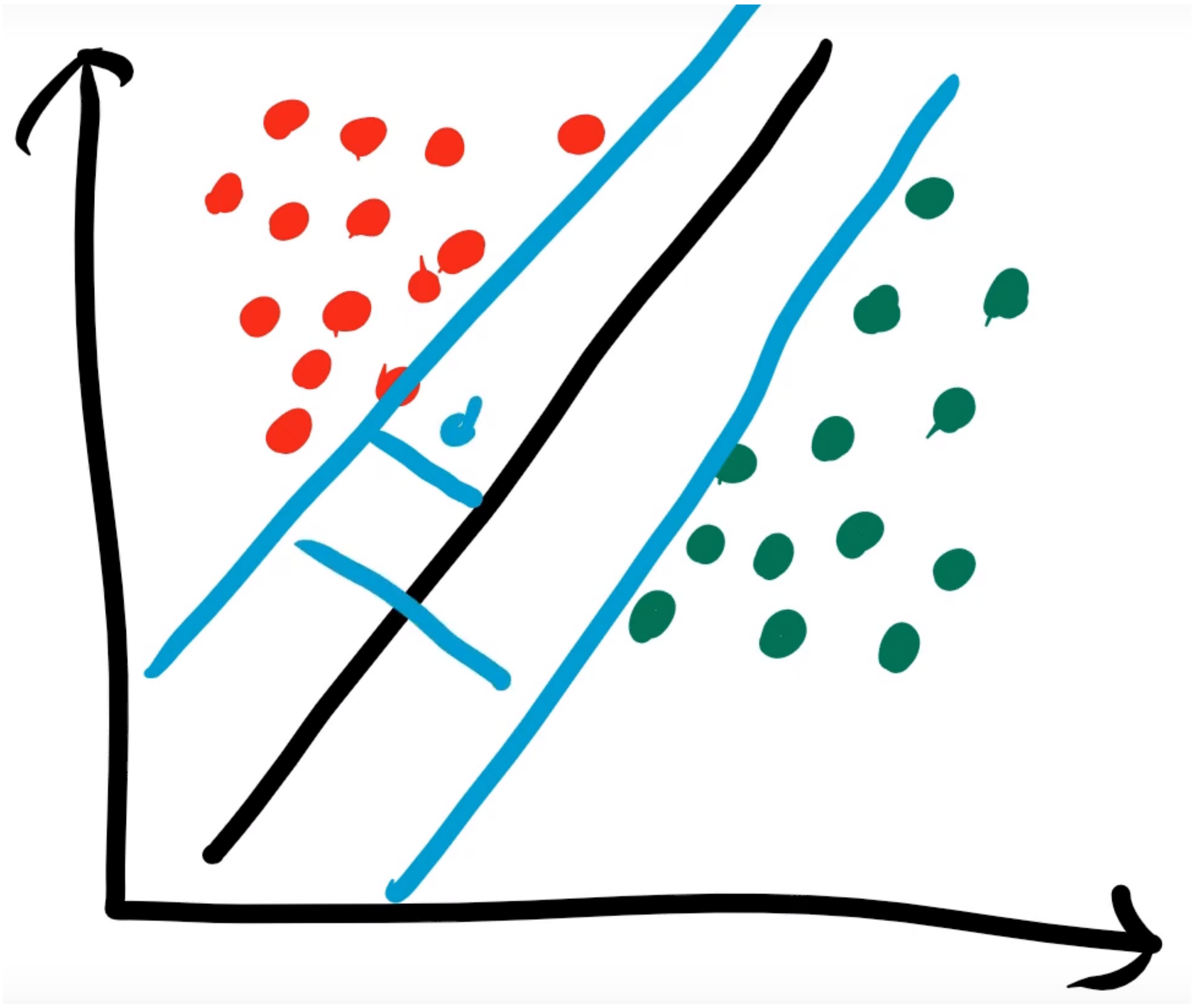
Have a look at the hyper-planes above and determine which you think would give the best classification for a mystery test point. What do you notice about that hyper-plane?

Well the best possible hyper-plane would be the first image on this page. Notice the distance between the support vectors and the hyper-plane is far greater than the other generated hyper-planes.

When we pick a hyper-plane we want to pick one that has the greatest possible **margin**.

Margin

The margin is the distance that separates all of the points in our test data. The blue lines below show you the margin for this particular data and hyper-plane. Typically the greater our margin the better our classification will be.

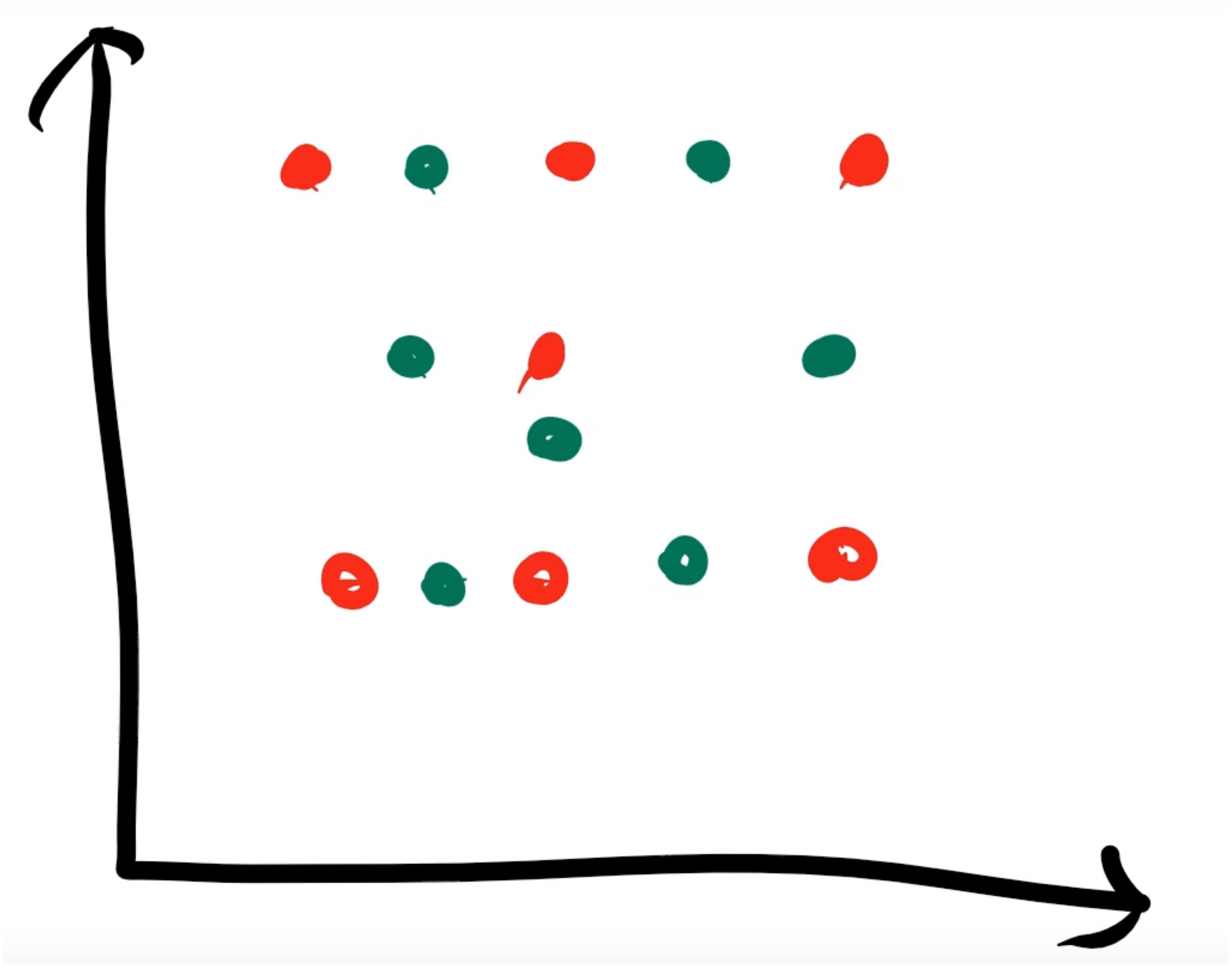


Note: Imagine the blue lines are parallel to the black...

Kernels

So you now have a very basic understanding of how a SVM works. Seems pretty simple in theory, but in practice we can run into a lot of issues.

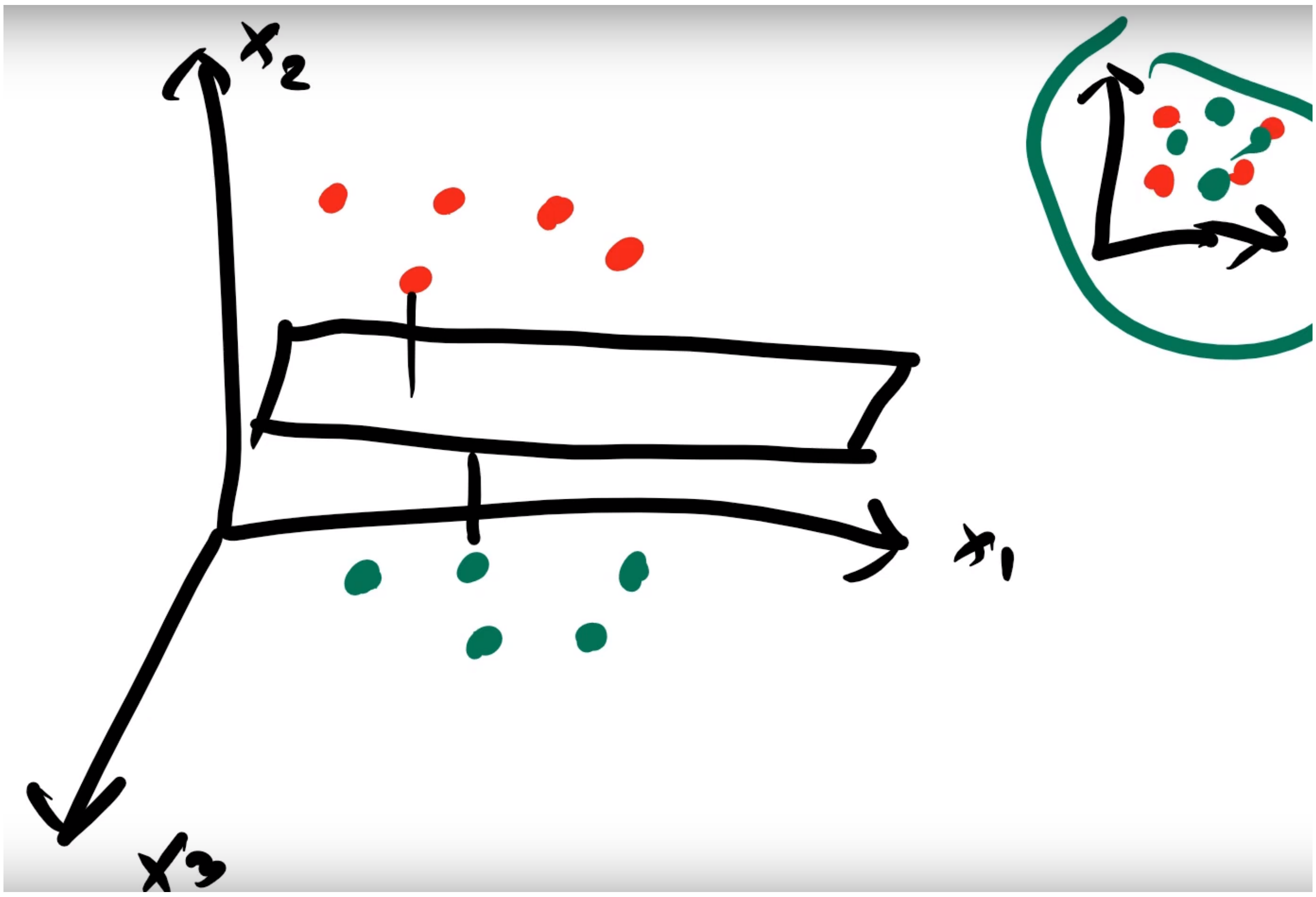
Let’s say our data isn’t as pretty and we have some points that look like this:



Can you determine which **hyper-plane** would be the best for this data? Even if you could it would make a horrible classifier. This is where we introduce something called **kernels**.

Kernels provide a way for us to create a hyper-plane for data like seen above. We use a kernel to bring our data up to a higher dimension (in this case from 2D->3D). We hope that by doing this we will have our points plotted in a way that we can divide them using a **hyper-plane**.

By applying a kernel to our data above we hope to get something that looks like the following:

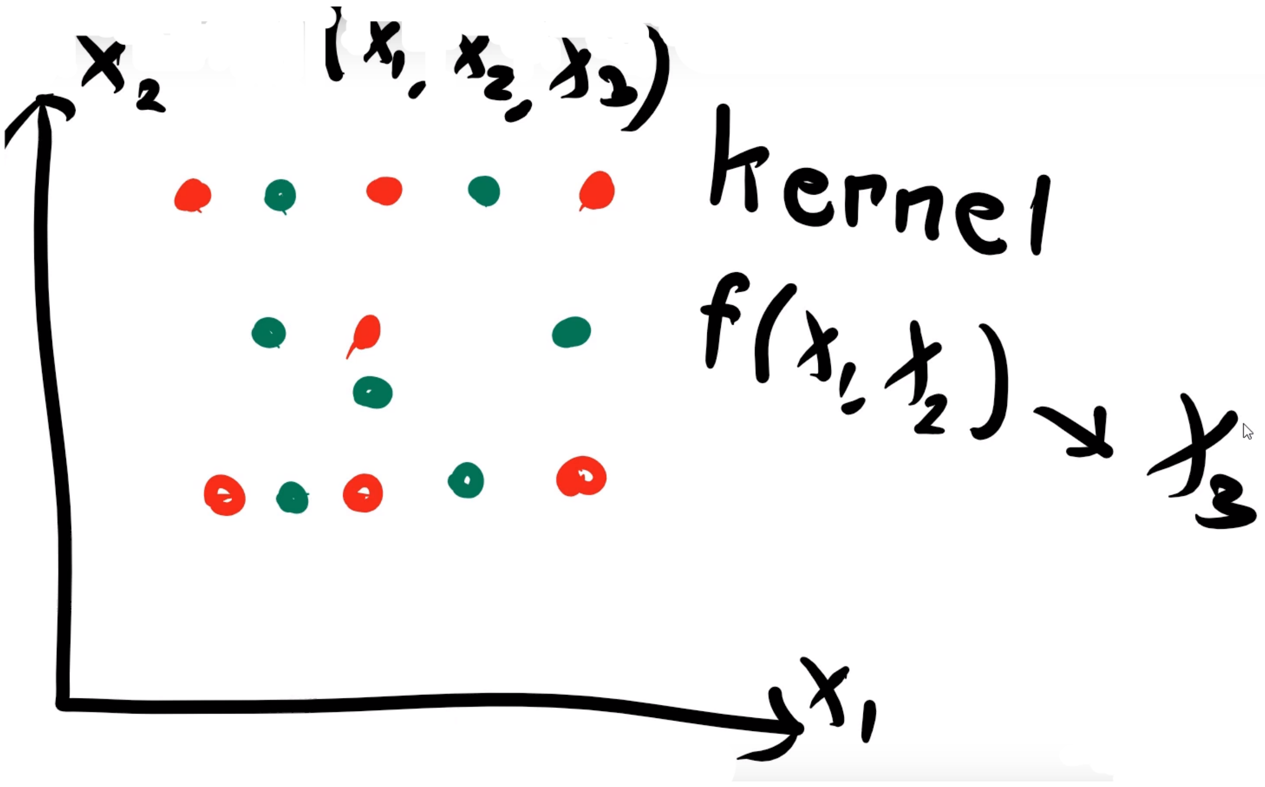


You can see that we can now divide our points with a plane in 3D. By applying the kernel our data has become separable.

What Is A Kernel?

A kernel is simply a function that takes as input our features (x1, x2 in our example) and returns a value equal to the third dimensional coordinate (x3). An example of a kernel copuld be the equation:

(x1)^2 + (x2)^2 = x3



\

Typically when we use a kernel we use a pre-existing one. There is much debate about which kernel is the best but here are some examples of popular kernels.

– **Linear**

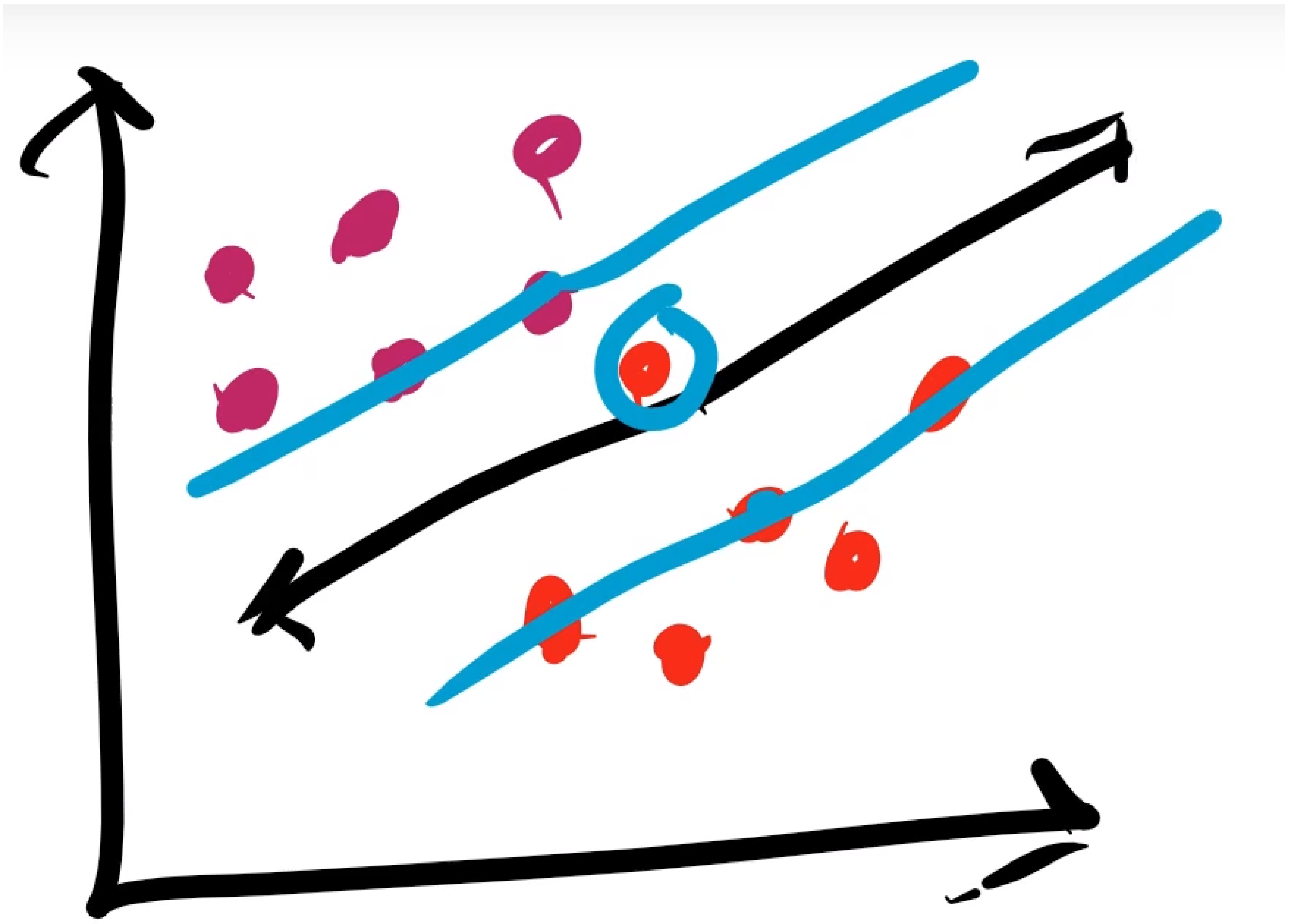
– **Polynomial**

– **Circular**

– **Hyperbolic Tangent (Sigmoid)**

Soft & Hard Margin

The last topic to touch on is soft and hard margins. A **hard margin** is precisely what you've learned already, no points may exist inside the margin. However, sometimes if we have outlier points we want to allow them to exist inside the margin and use points that are not the closest to the hyper-plane to be our support vectors. Doing this is called using a **soft margin**.



You can see in the example above that there is a point that exists inside the margin. If we had not allowed this not only would it be difficult to create a hyper-plane but our classification would perform poorly.

The amount of points you allow to exists inside the margin is something we can define as **hyper-parameter.**

In this next tutorial we will implement an SVM and do some hyper-parameter tuning using kernels and soft margins.

Search …

TOPICS

- ✔ Introduction & Setup (https://www.techwithtim.net/tutorials/machine-learning-python/introduction/)

WORKOUT PLAN
ACCORDING TO THE GOAL
MONDAY
TUESDAY
WEDNESDAY
THURSDAY
FRIDAY
OTTERMODE
WORKOUT PLAN ACCORDING TO TH
ATHLETE
WORKOUT PLA
WORKOUT PLA