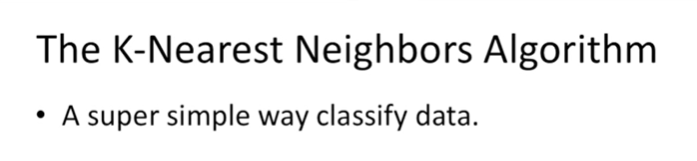
<https://www.youtube.com/watch?v=HVXime0nQeI&list=PLblh5JKOoLUICTaGLRoHQDuF_7q2GfuJF&index=35>



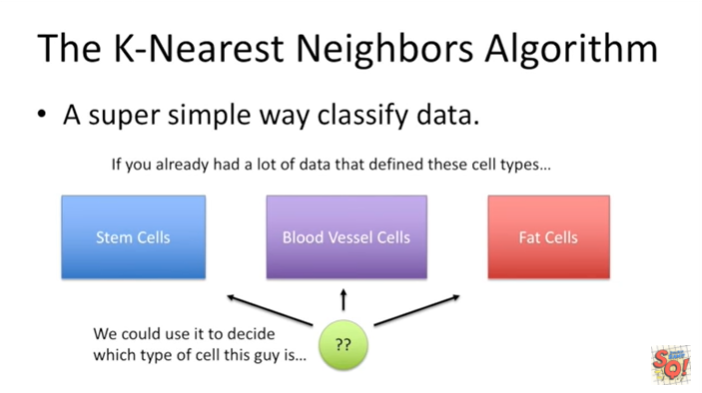
Today we're going to be talking about the K nearest neighbors algorithm.



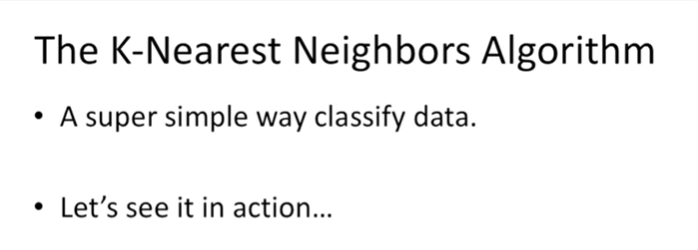
Which is a super simple way to classify data.



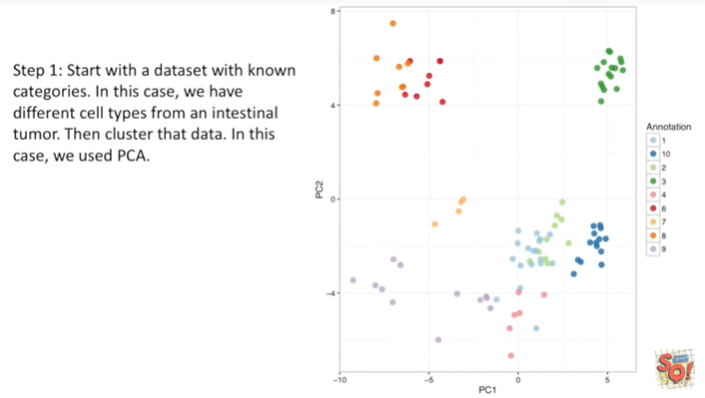
In a nutshell, if you already had a lot of data, that define these cell types…



We could use it to decide which type of cell this guy is.



Let's see it in action.

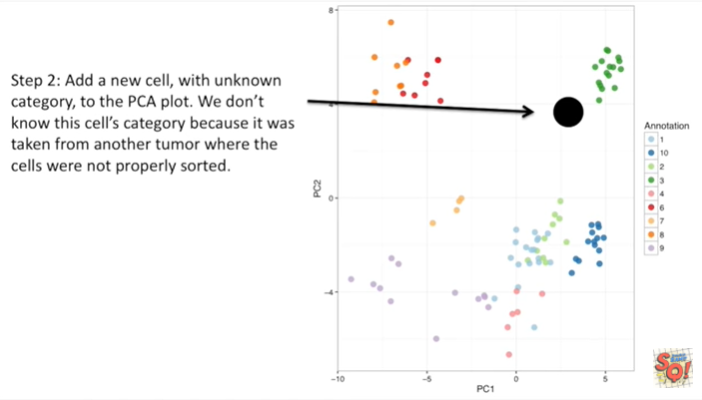


Step 1 : start with a data set with known categories.

In this case, we have different cell types from an intestinal tumor.

We then cluster that data.

In this case we use PCA.



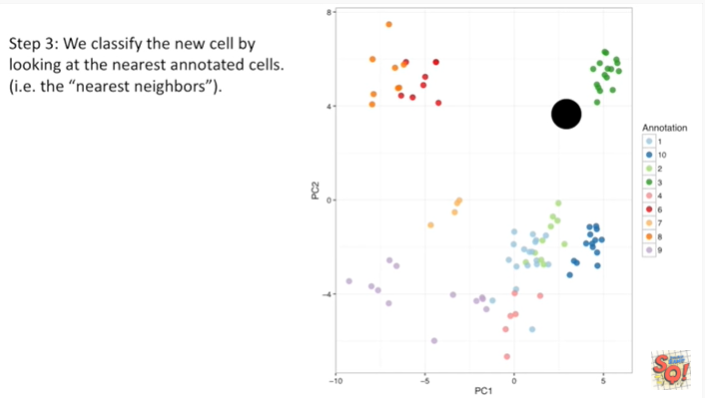
Step 2 : add a new cell with unknown category to the PCA plot.

We don't know this cells category because it was taken from another tumor where the cells were not properly sorted.

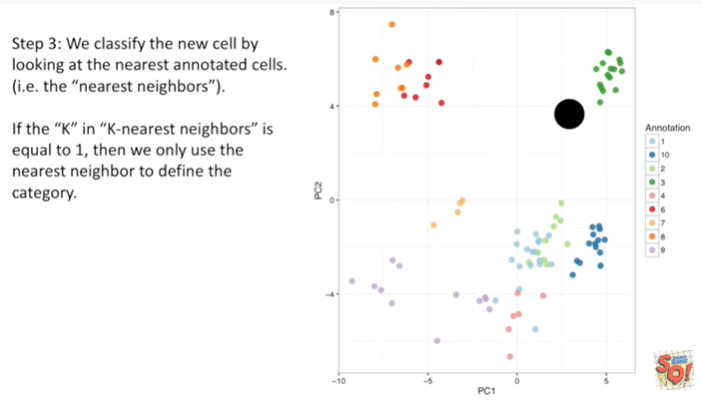
And so what we want to do is, we want to classify this new cell.

We want to figure out what cell it's most similar too.

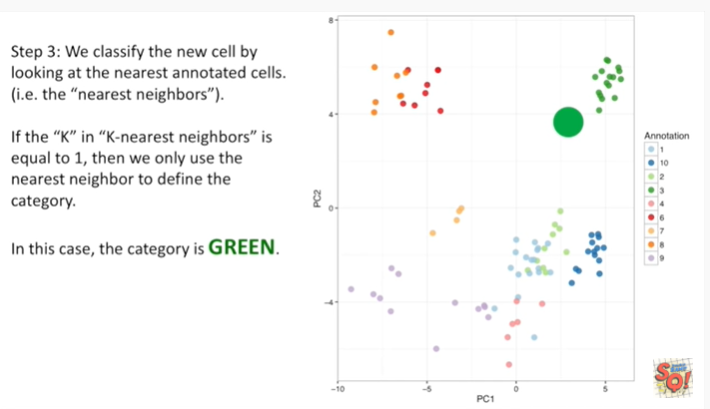
And then we're going to call it that type of cell.



Step 3 : we classify the new cell by looking at the nearest annotated cells (ie the nearest neighbors).



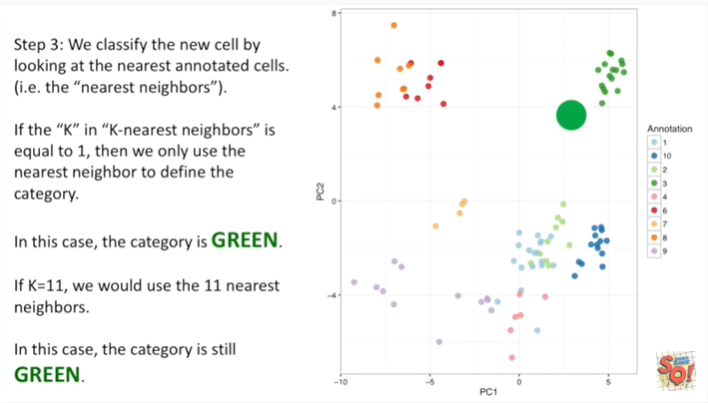
If the K, in K nearest neighbors, is equal to 1, then we will only use the nearest neighbor to define the category.



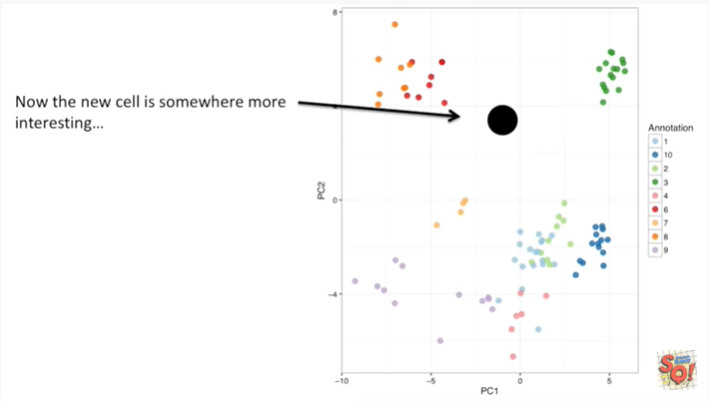
In this case the category is green because the nearest neighbor is already known to be the green cell type.



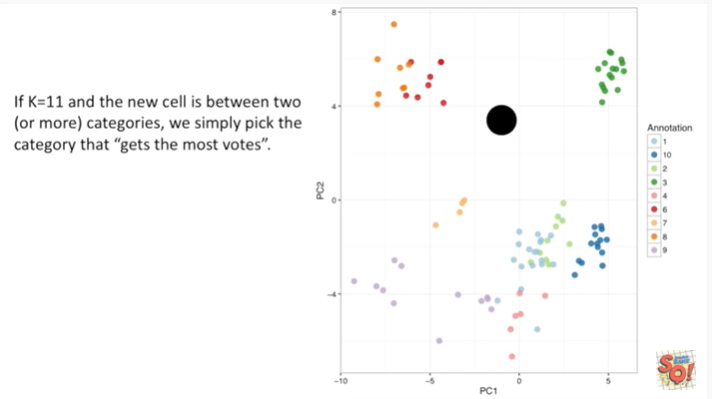
If K equals 11, we would use the 11 nearest neighbors.



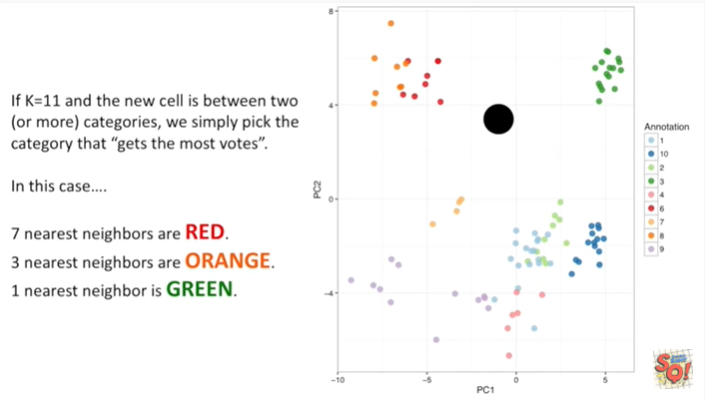
In this case the category is still green because the 11 cells that are closest to the unknown cell are already green.



Now the new cell is somewhere more interesting, it's about halfway between the green and the red cells.



If k equals 11 and the new cells between two or more categories, we simply pick the category that gets the most votes.



In this case :

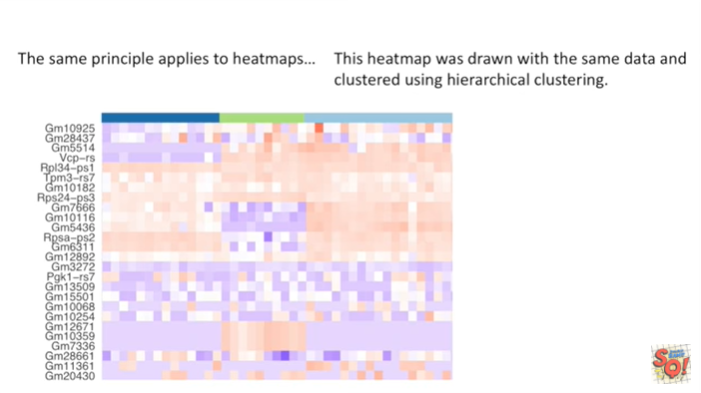
* seven nearest neighbors are red
* three nearest neighbors are orange
* one nearest neighbor is green.



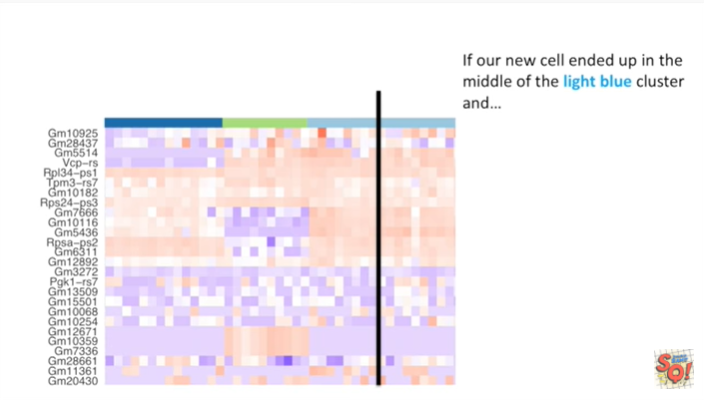
Since red got the most votes, the final assignment is red.



This same principle applies to heat maps.



This heat map was drawn with the same data and clustered using hierarchical clustering.

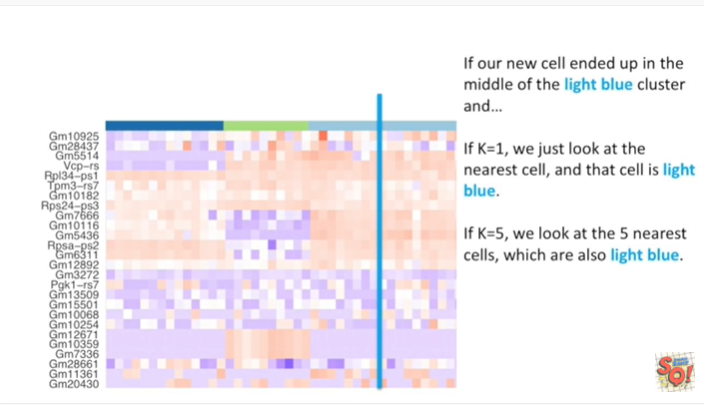


If our new cell ended up in the middle of the light blue cluster and



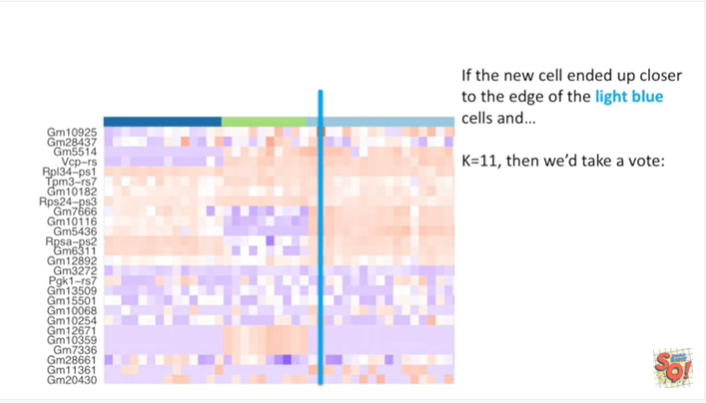
If K equals one, we just look at the nearest cell, and that cell is light blue.

So we classify that unknown cell as a light blue cell.

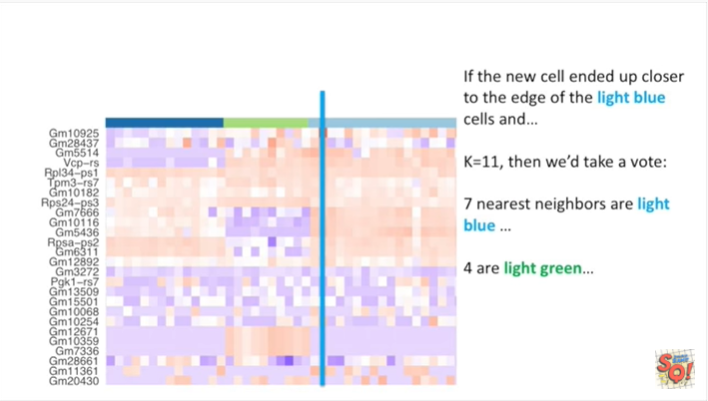


If K equals five, we'd look at the five nearest cells, which are also light blue.

So we'd still classify the unknown cell as light blue.



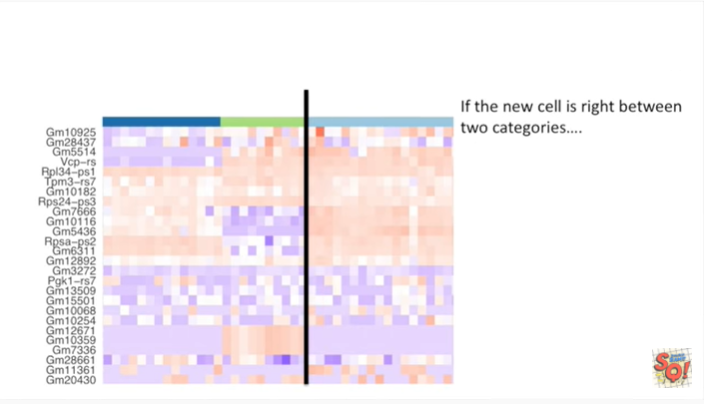
If the new cell ended up closer to the edge of the light blue cells and K equals 11 then we take a vote :



seven nearest neighbors are light blue

and four or light green.

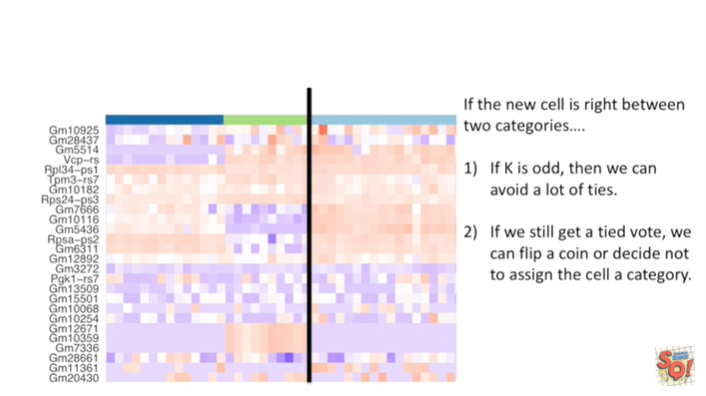
So we still go with light blue.



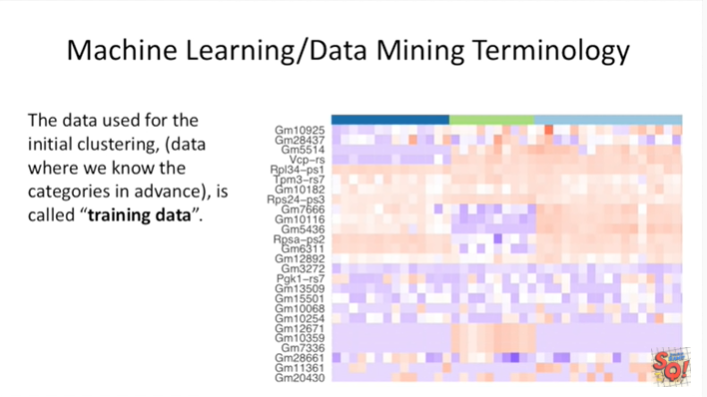
If the new cell is right between two categories :



1. well if K is odd then we can avoid a lot of ties.

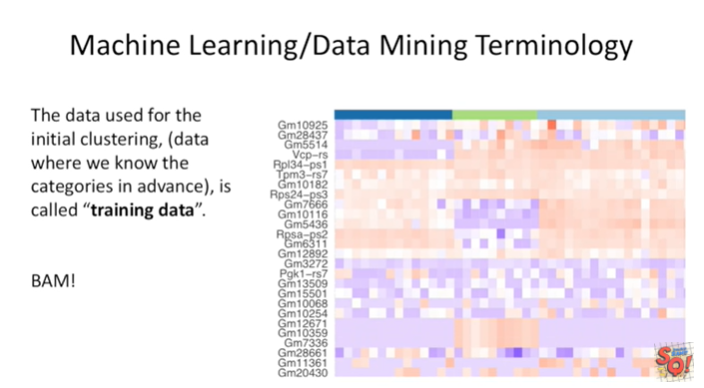


2. If we still get a tied vote we can flip a coin or decide not to assign the cell to a category.

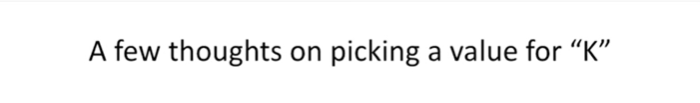


Before we go let's talk about a little machine learning / data mining terminology.

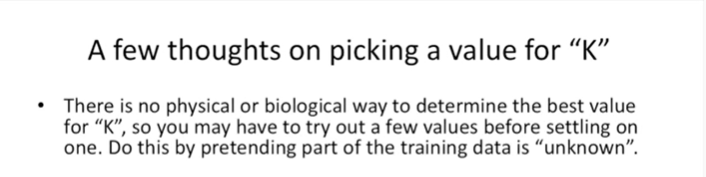
The data used for the initial clustering, the data where we know the categories in advance, is called training data.



BAM !



A few thoughts on picking a value for K.

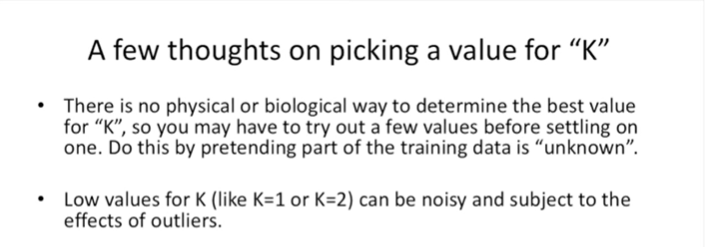


There is no physical or biological way to determine the best value for K.

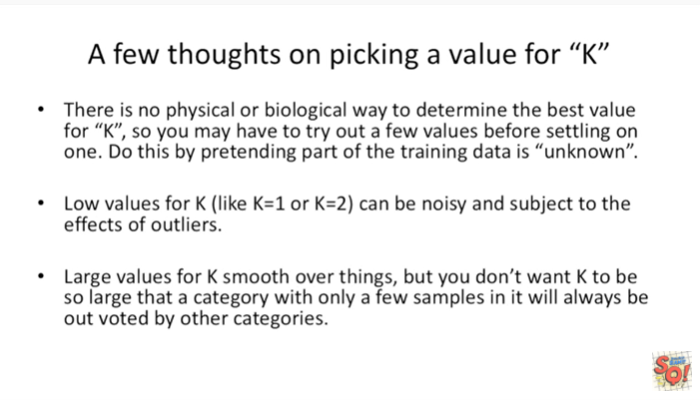
So you may have to try out a few values before settling on one.

Do this by pretending part of the training data is unknown.

And then what you do is you categorize that unknown data using the K nearest neighbor algorithm and you assess how good the new categories match what you know.



Already low values for K like k equals 1 or K equals 2 can be noisy and subject to the effects of outliers.



Large values for K smooth over things, but you don't want K to be so large that a category with only a few samples in it will always be outvoted by other categories.