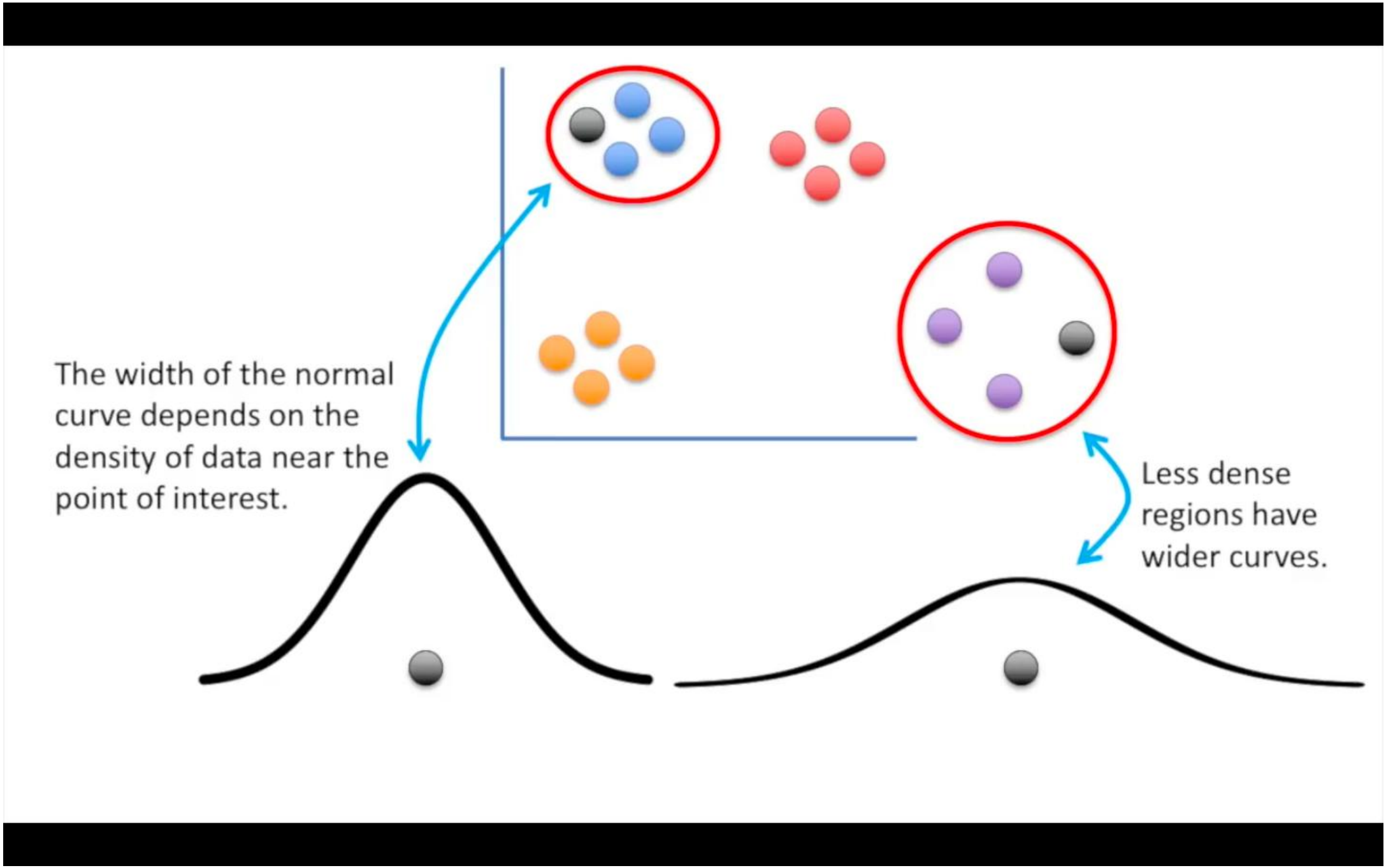
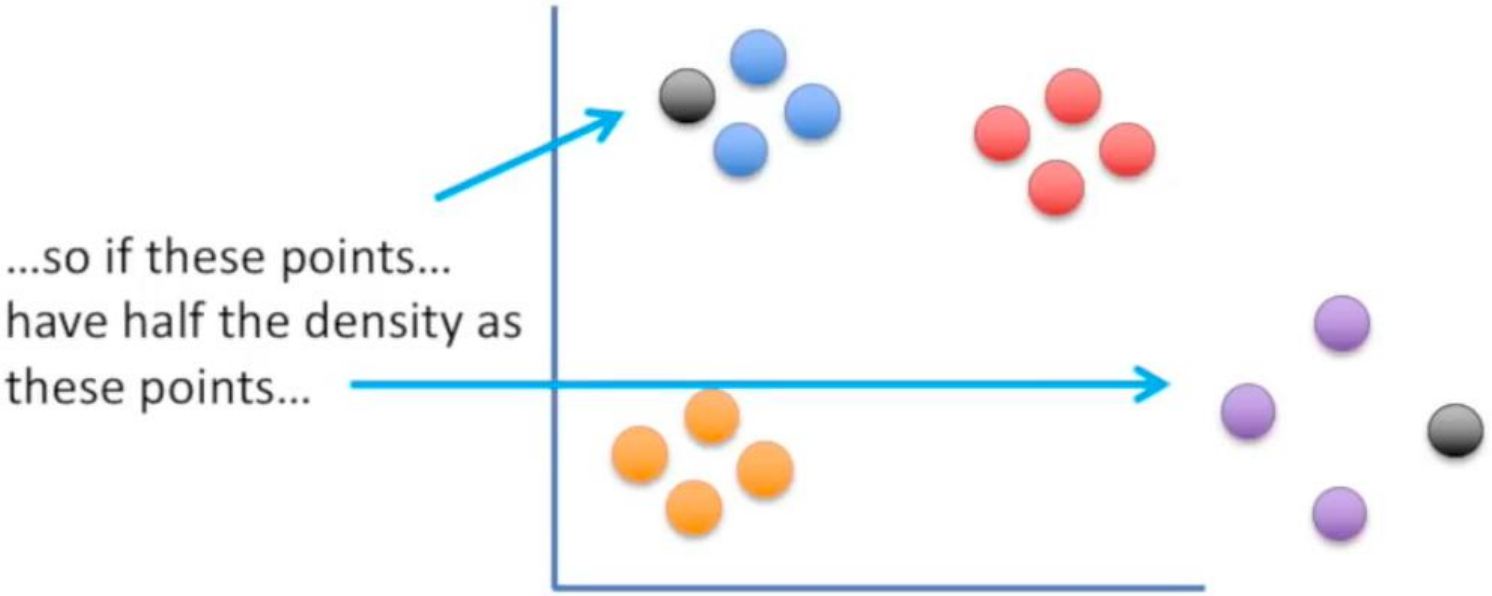
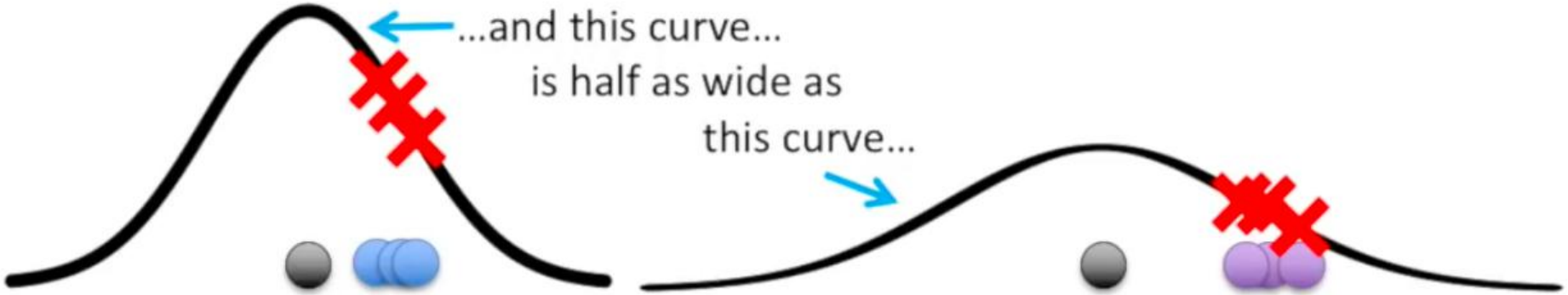

The width of the normal curve depends on the density of data near the point of interest.
Less dense regions have wider curves.

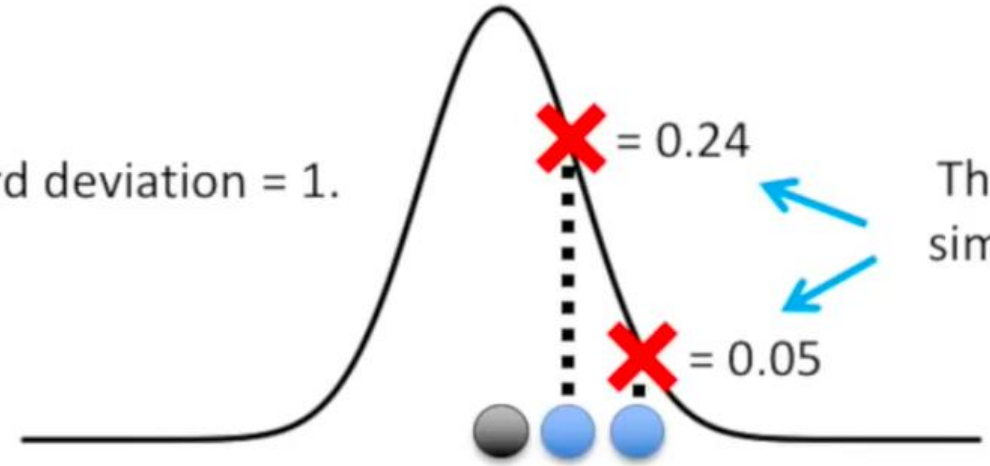




...then scaling the similarity scores will
make them the same for both clusters.

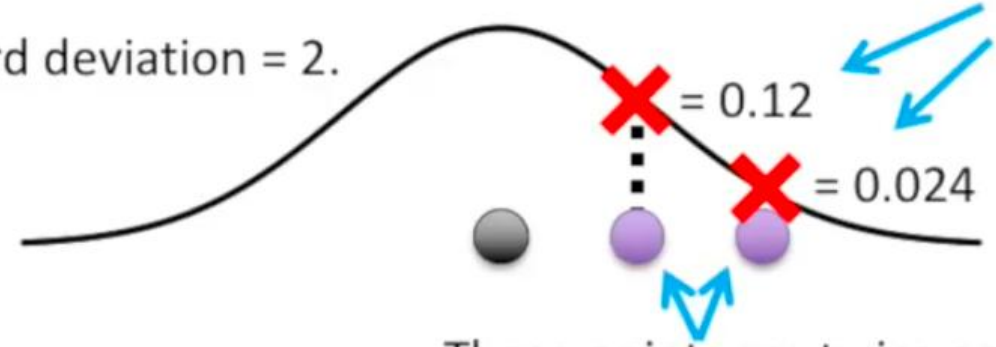
Here's an example...

This curve has a standard deviation = 1.



The "unscaled" similarity values

This curve has a standard deviation = 2.

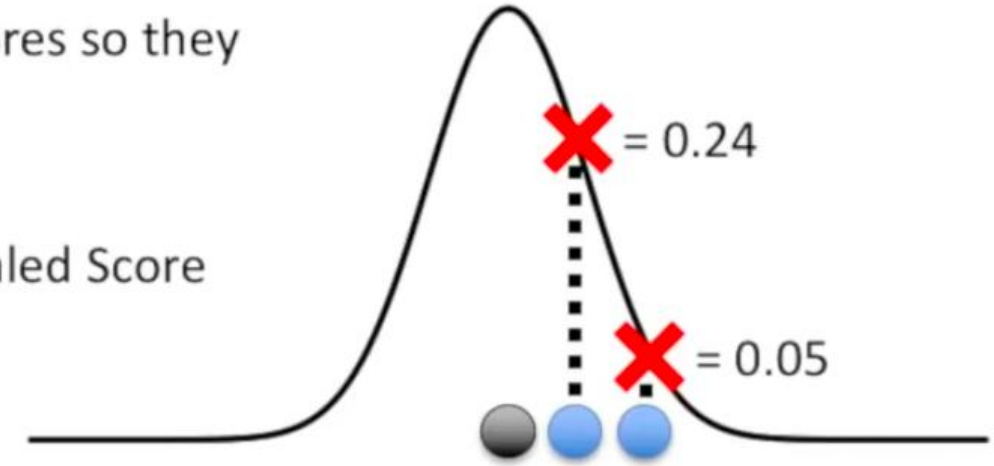


The "unscaled" similarity values are half of the other ones.

These points are twice as far from the middle.

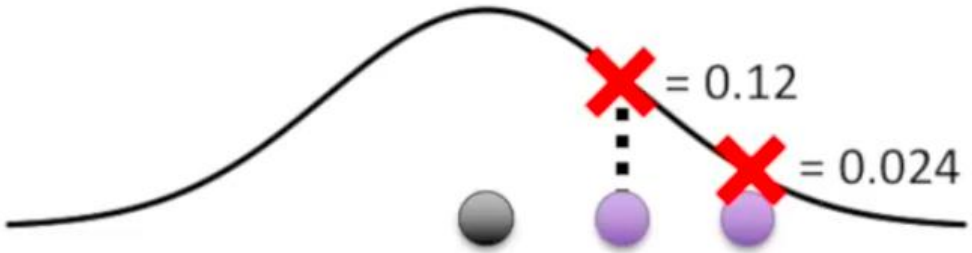
To scale the similarity scores so they sum to 1:

$$\frac{\text{Score}}{\text{Sum of all scores}} = \text{Scaled Score}$$



$$\frac{0.24}{0.24 + 0.5} = 0.82$$

$$\frac{0.05}{0.24 + 0.5} = 0.18$$

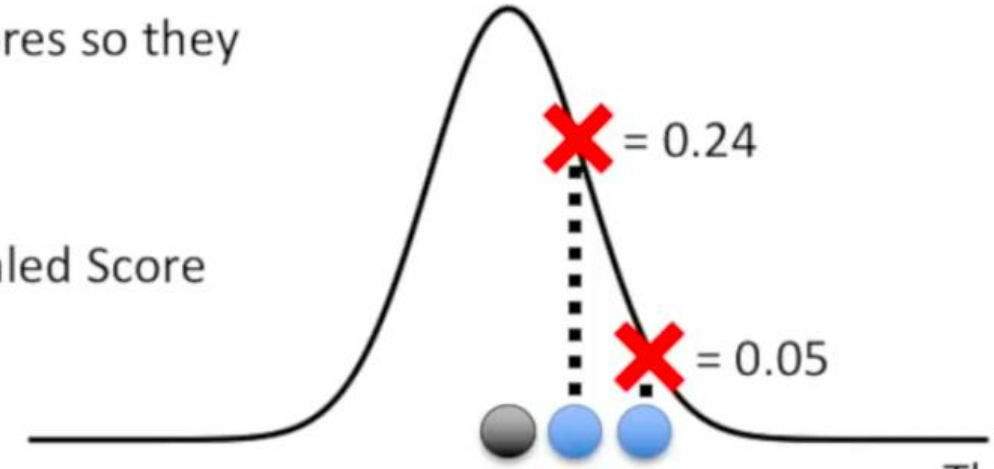


$$\frac{0.12}{0.12 + 0.024} = 0.82$$

$$\frac{0.024}{0.12 + 0.024} = 0.18$$

To scale the similarity scores so they sum to 1:

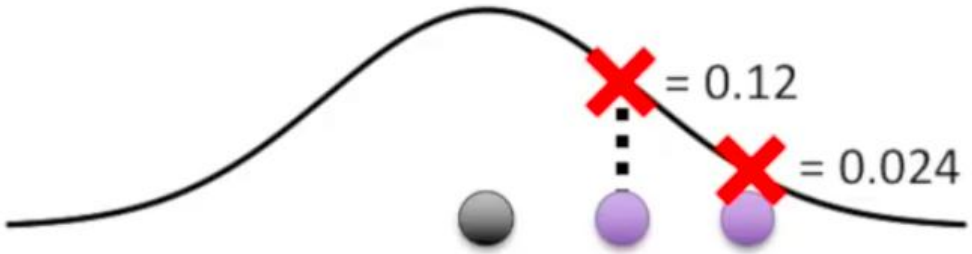
$$\frac{\text{Score}}{\text{Sum of all scores}} = \text{Scaled Score}$$



$$\frac{0.24}{0.24 + 0.5} = 0.82$$

$$\frac{0.05}{0.24 + 0.5} = 0.18$$

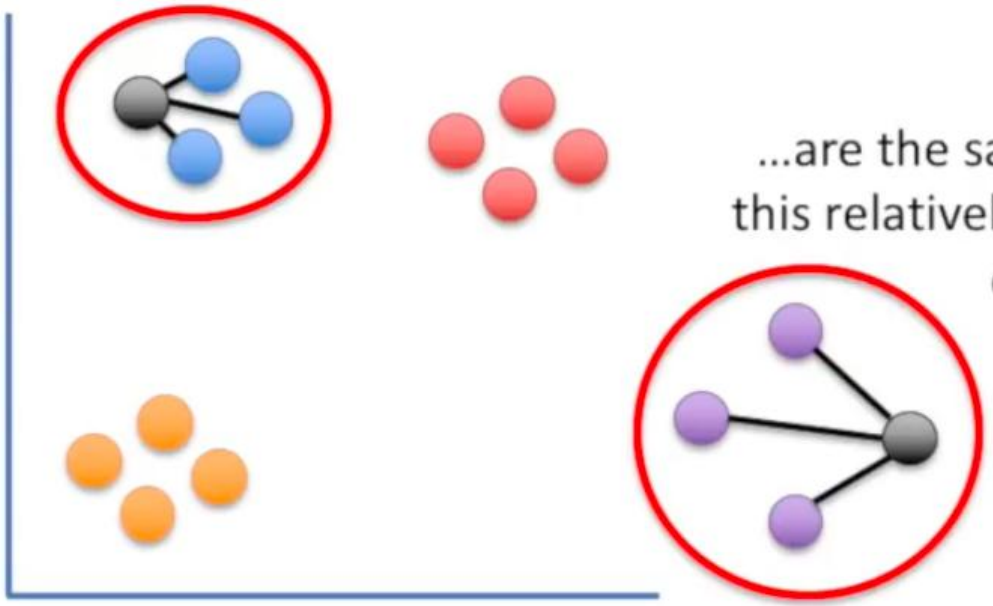
These are the same as these!



$$\frac{0.12}{0.12 + 0.024} = 0.82$$

$$\frac{0.024}{0.12 + 0.024} = 0.18$$

That implies that the scaled similarity scores for this relatively tight cluster...



...are the same for this relatively loose cluster!

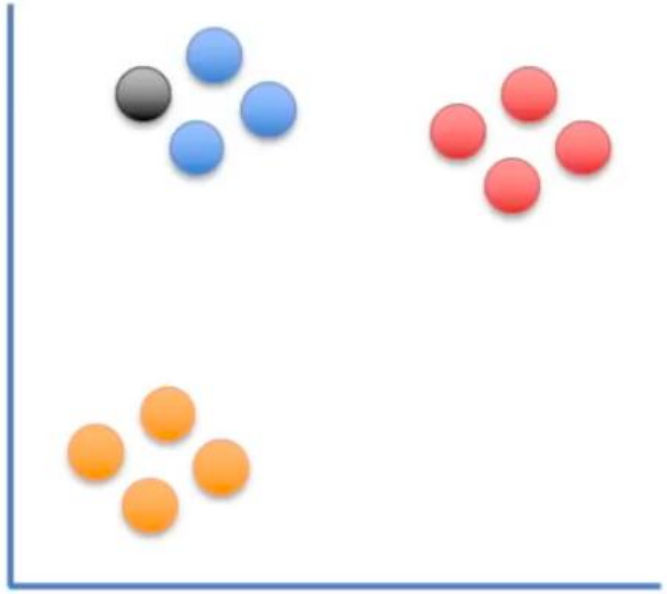
$$\frac{0.24}{0.24 + 0.5} = 0.82$$

$$\frac{0.05}{0.24 + 0.5} = 0.18$$

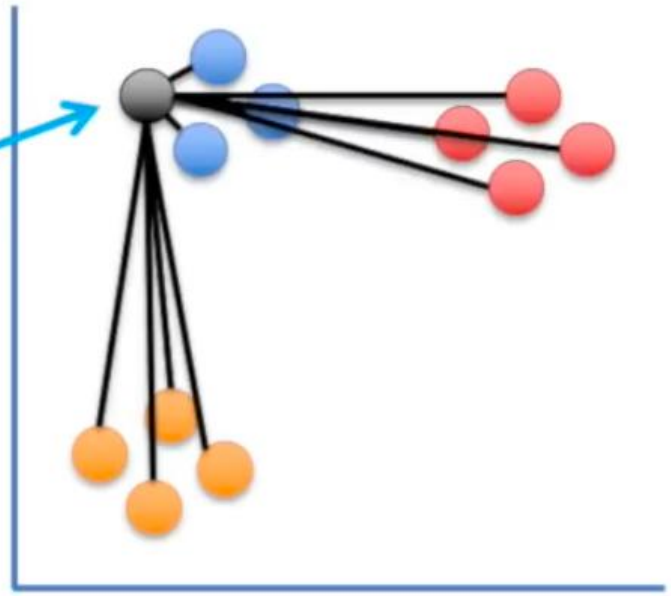
$$\frac{0.12}{0.12 + 0.024} = 0.82$$

$$\frac{0.024}{0.12 + 0.024} = 0.18$$

Now back to the original scatter plot…

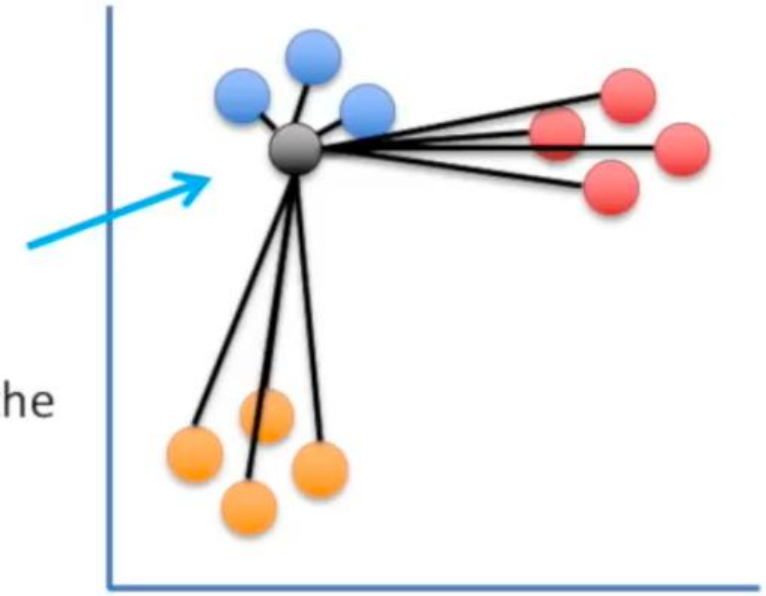


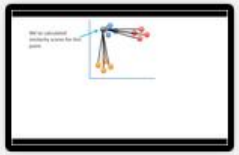
We've calculated similarity scores for this point.



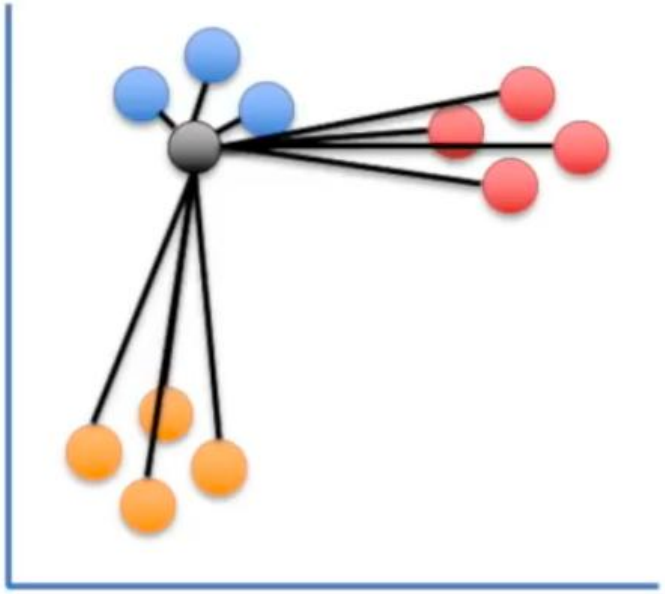
Now we do it for this point...

...and we do it for all the points.

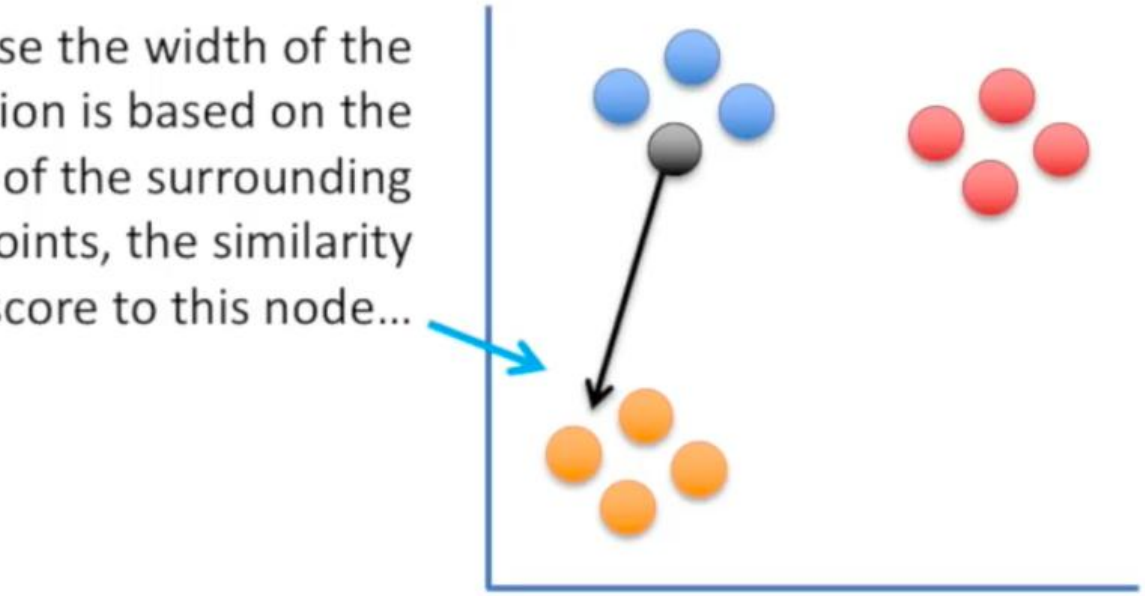




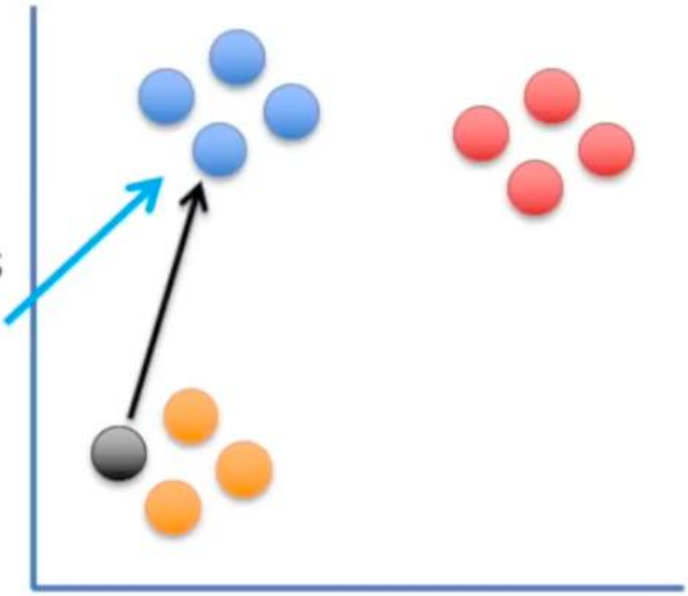
One last thing and the scatter plot will be all set with similarity scores!!!

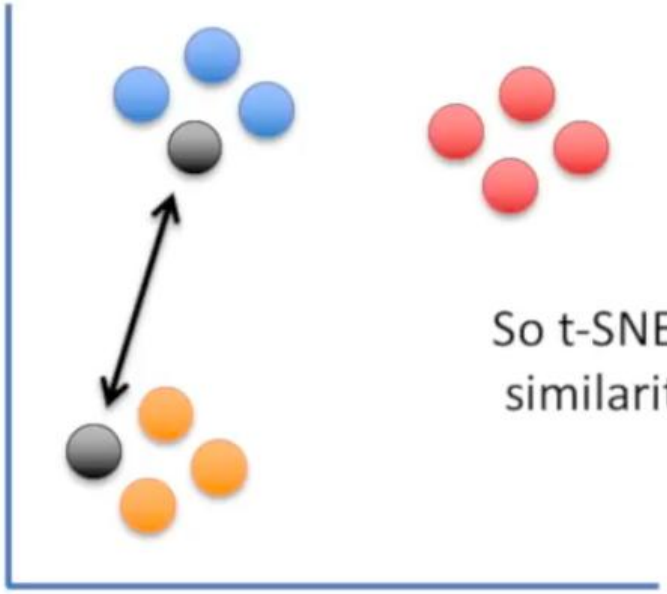


Because the width of the distribution is based on the density of the surrounding data points, the similarity score to this node...

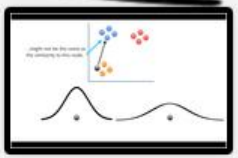


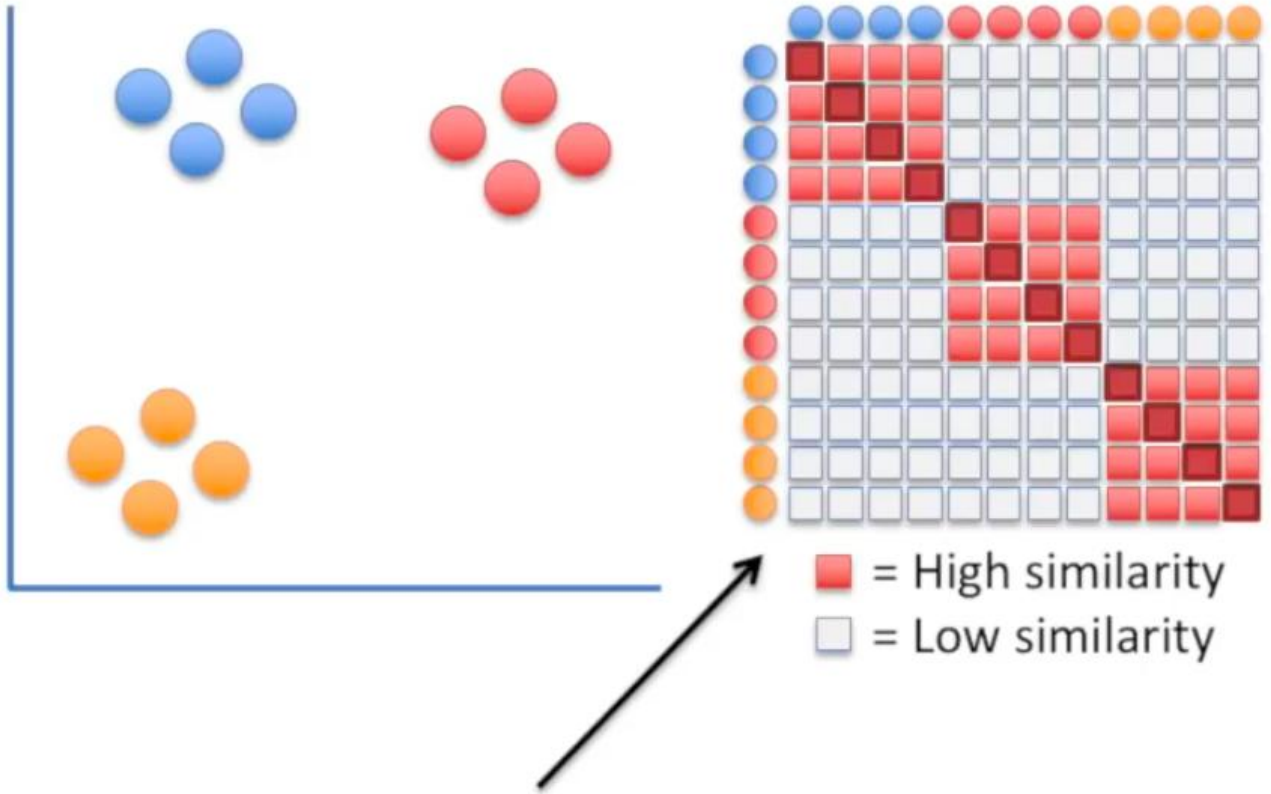
...might not be the same as
the similarity to this node.



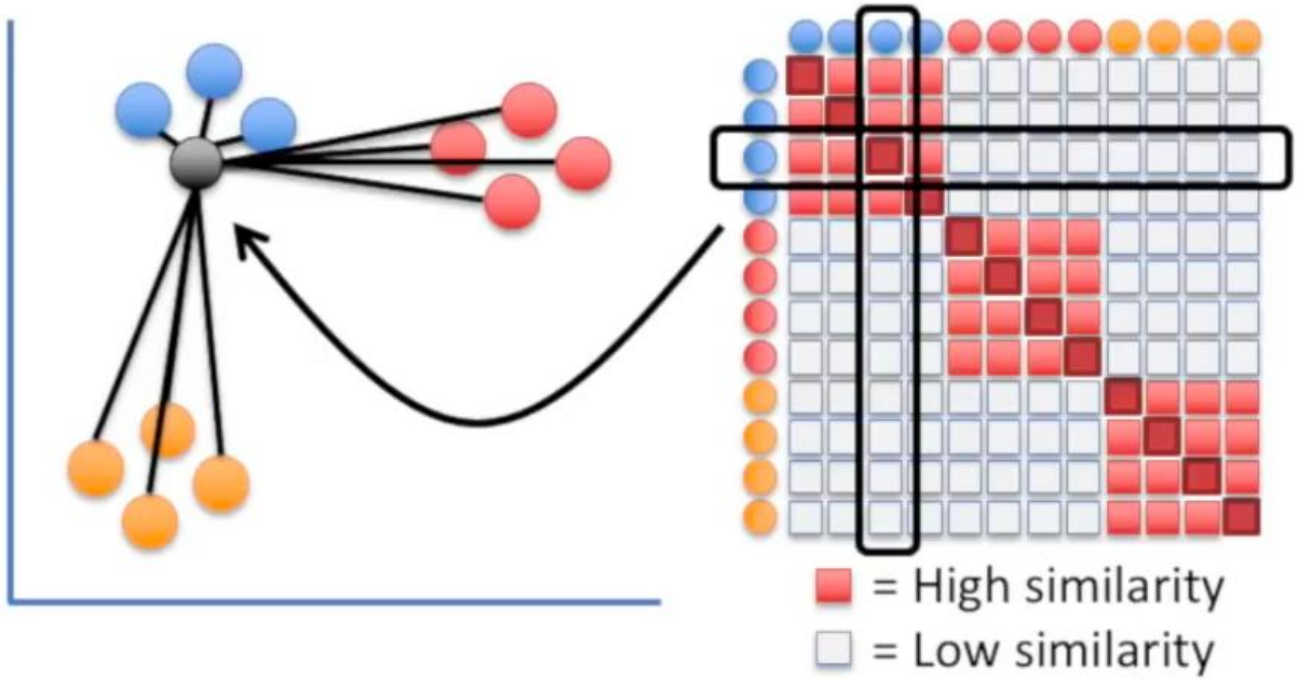


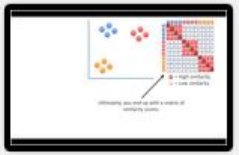
So t-SNE just averages the two similarity scores from the two directions…



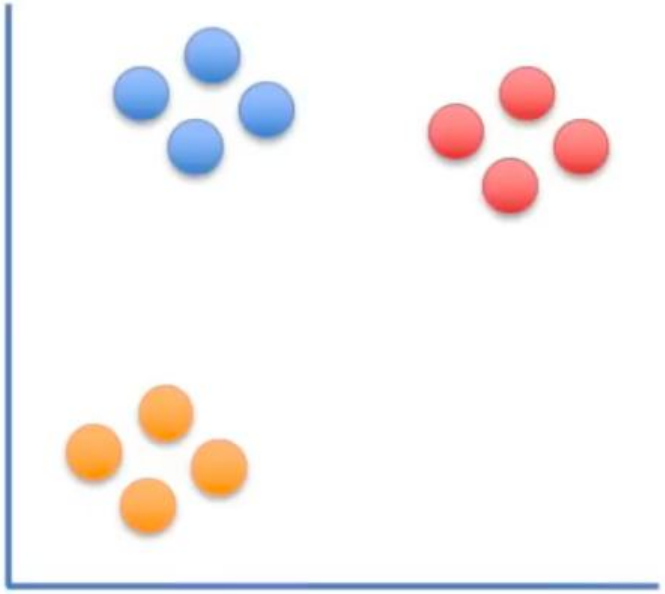


Ultimately, you end up with a matrix of similarity scores.

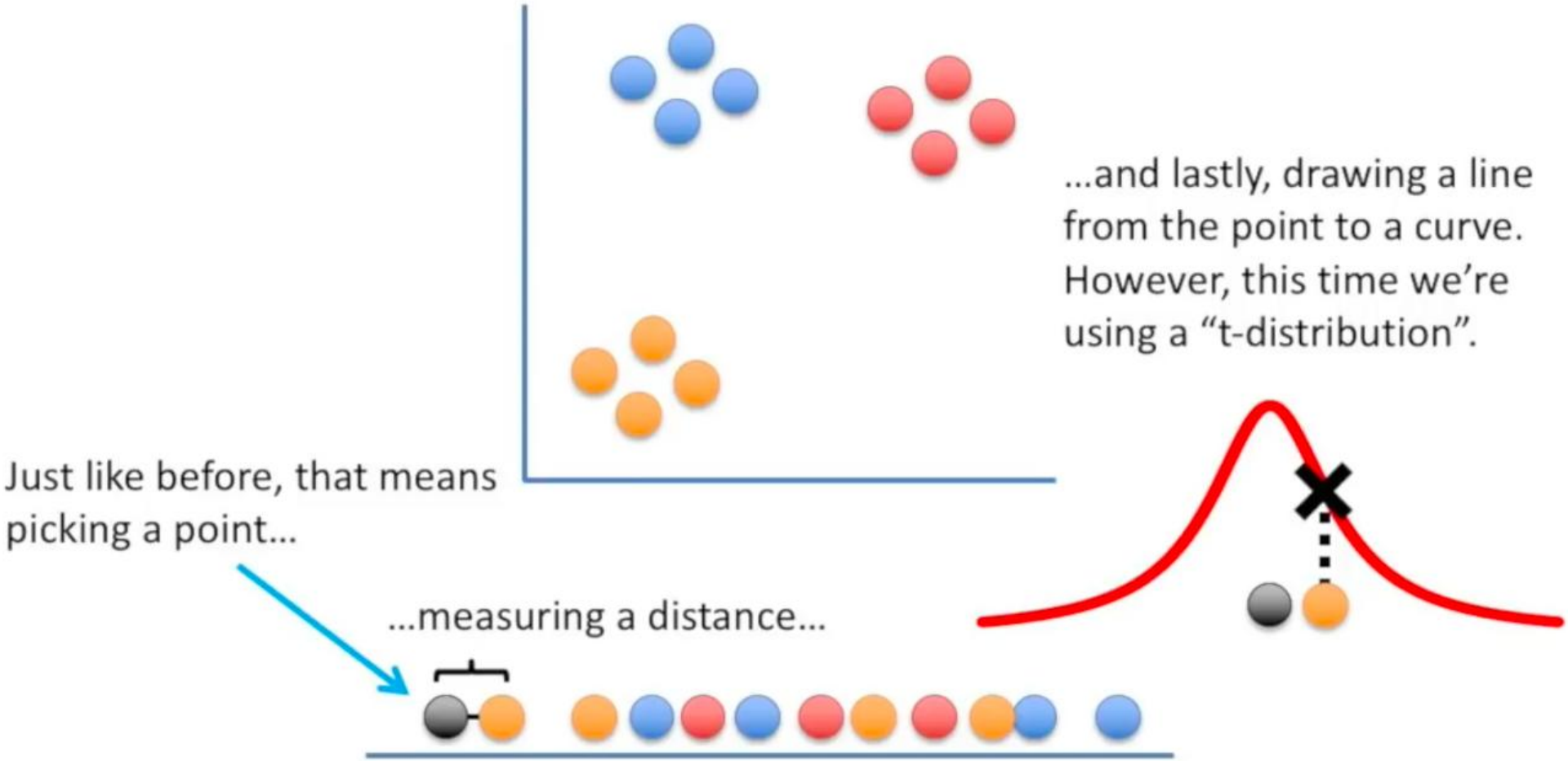

= High similarity
= Low similarity



Now we randomly project the data onto the number line...

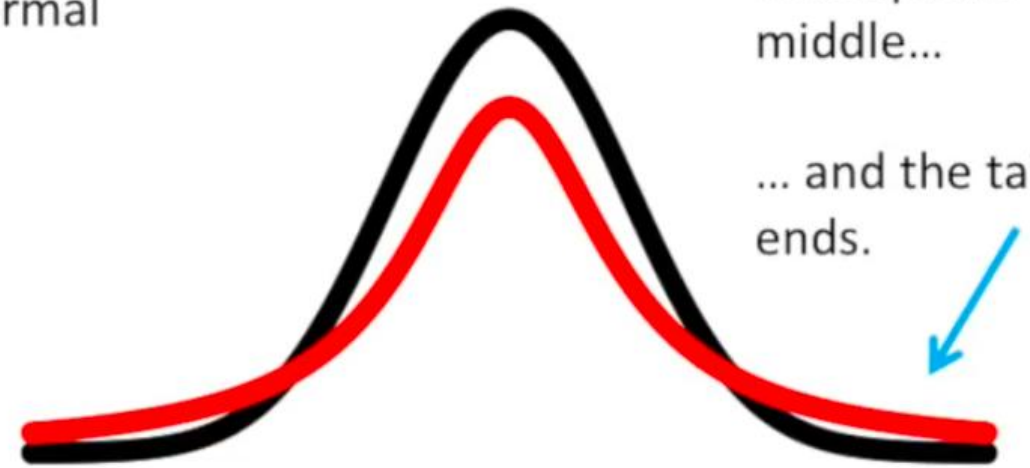


... and calculate similarity scores for the points on the number line.


Just like before, that means picking a point…
…measuring a distance…
…and lastly, drawing a line from the point to a curve. However, this time we're using a "t-distribution".

A “t-distribution”…

…is a lot like a normal distribution…

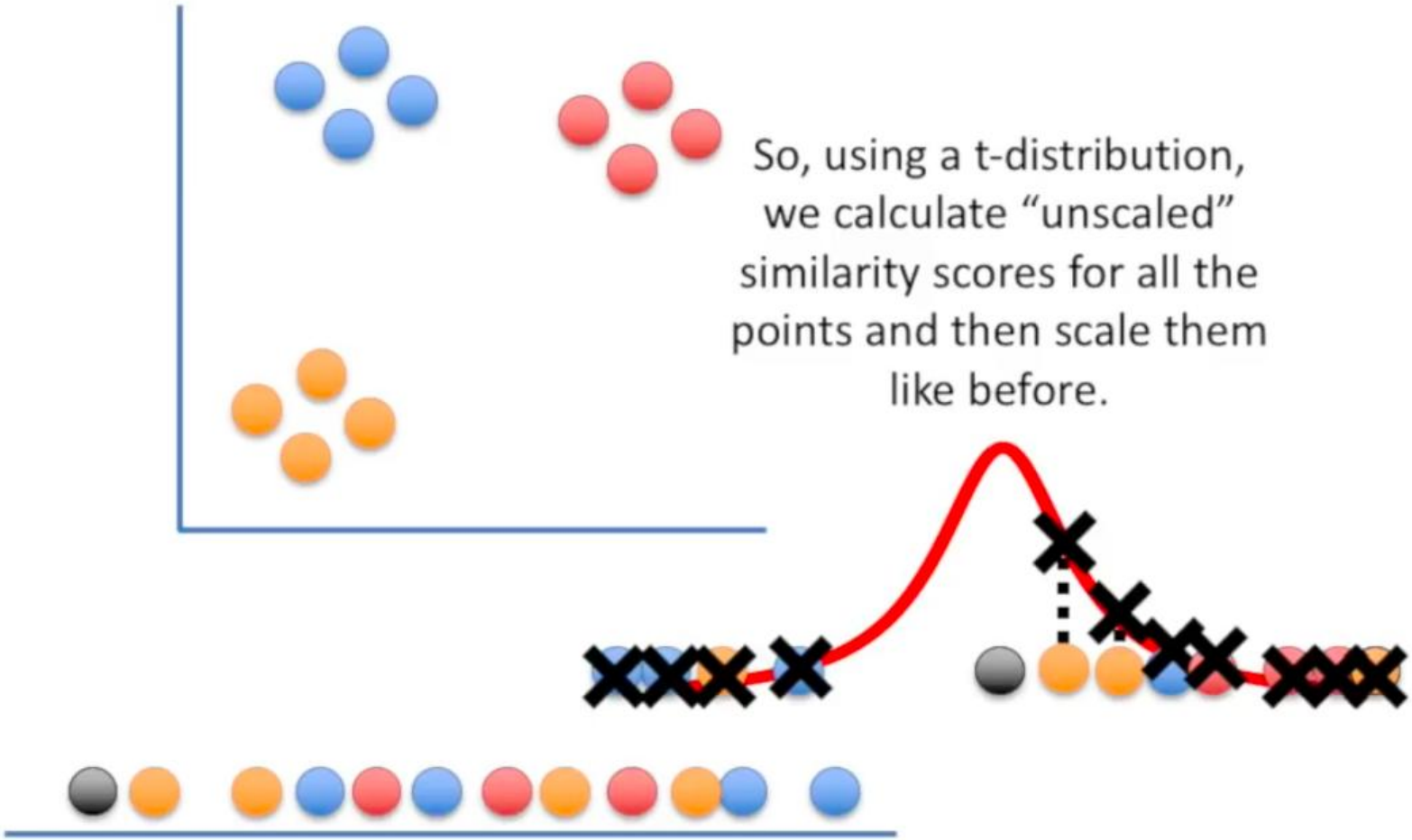


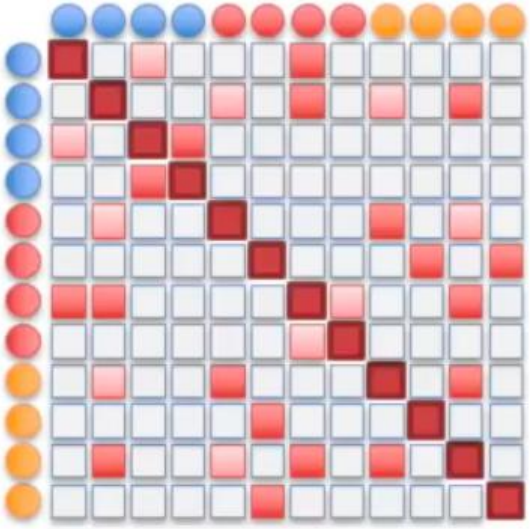
…except the “t” isn’t as tall in the middle…

… and the tails are taller on the ends.

The “t-distribution” is the “t” in t-SNE.

We’ll talk about why the t-distribution is used in a bit…

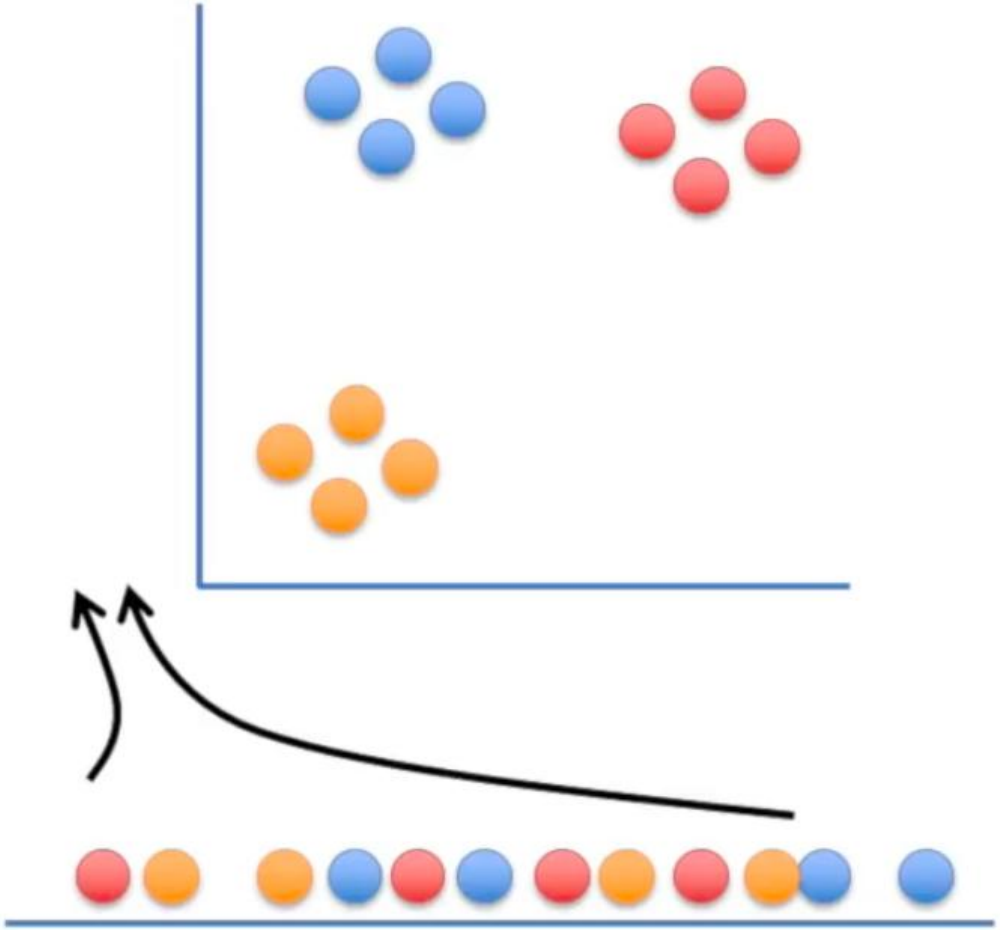

So, using a t-distribution, we calculate “unscaled” similarity scores for all the points and then scale them like before.

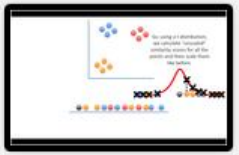


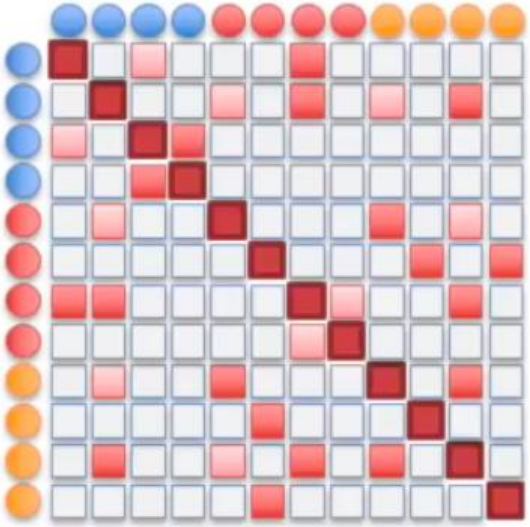
= High similarity

= Low similarity

Like before, we end up with a matrix of similarity scores, but this matrix is a mess…

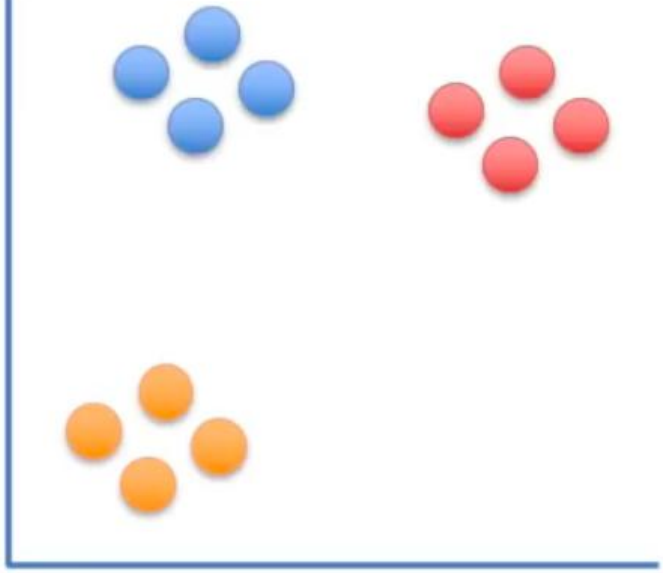


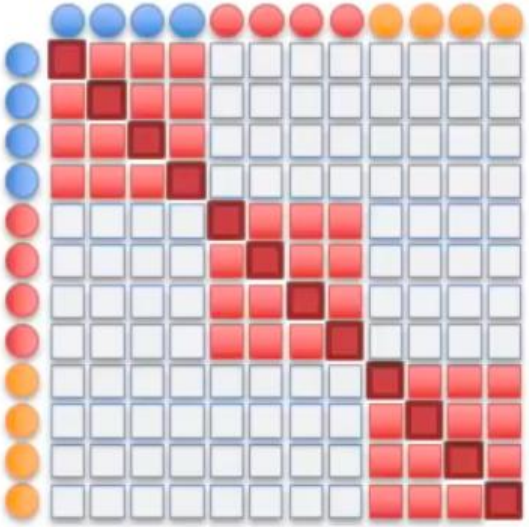




= High similarity

= Low similarity

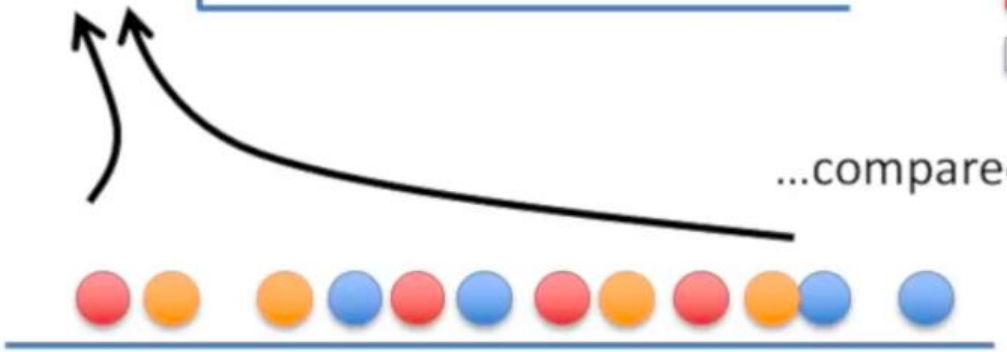




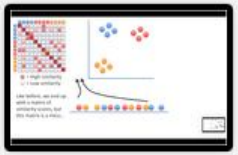
= High similarity

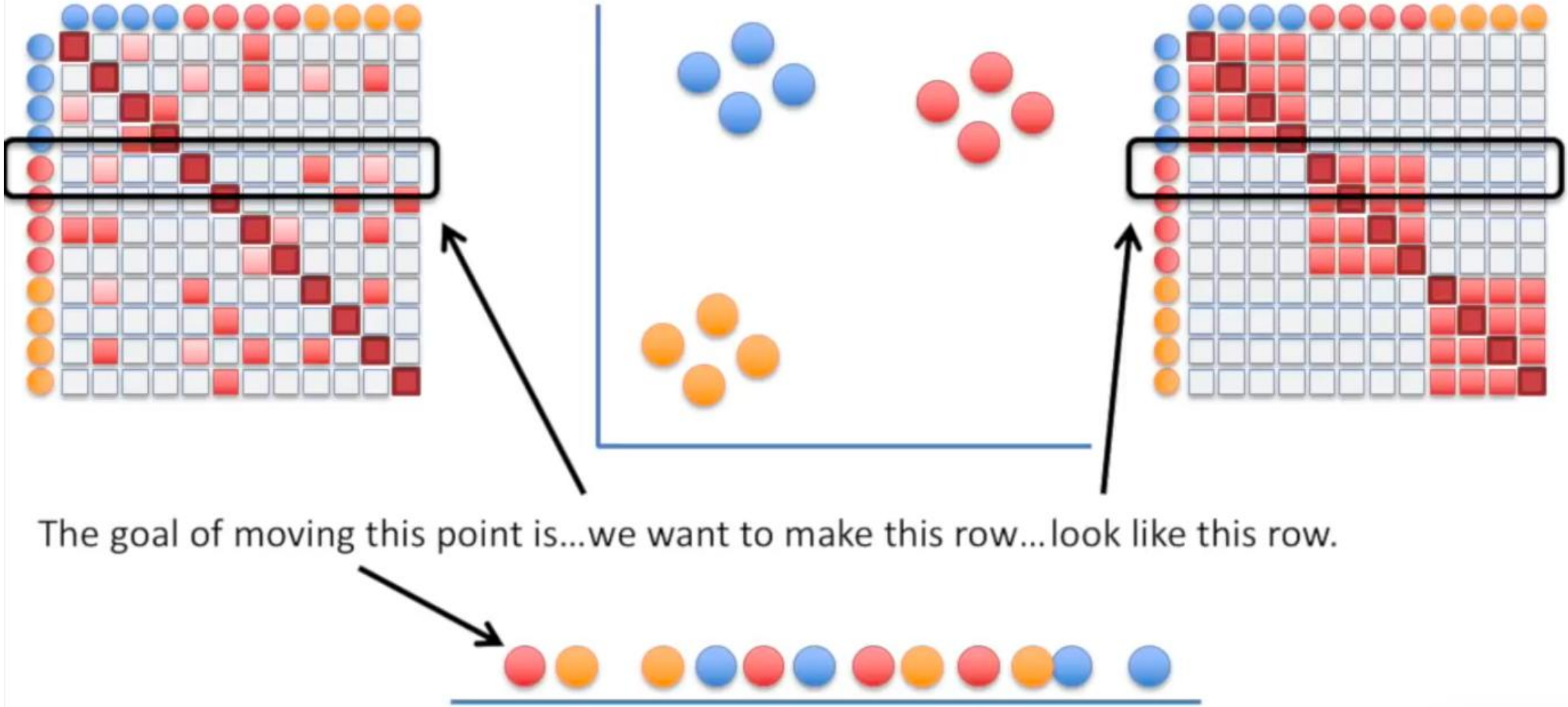
= Low similarity

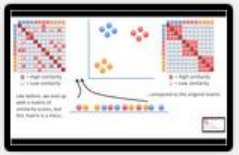
Like before, we end up with a matrix of similarity scores, but this matrix is a mess…



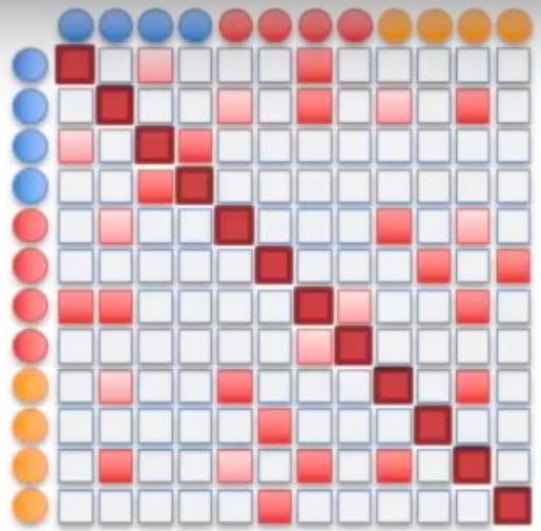
…compared to the original matrix.

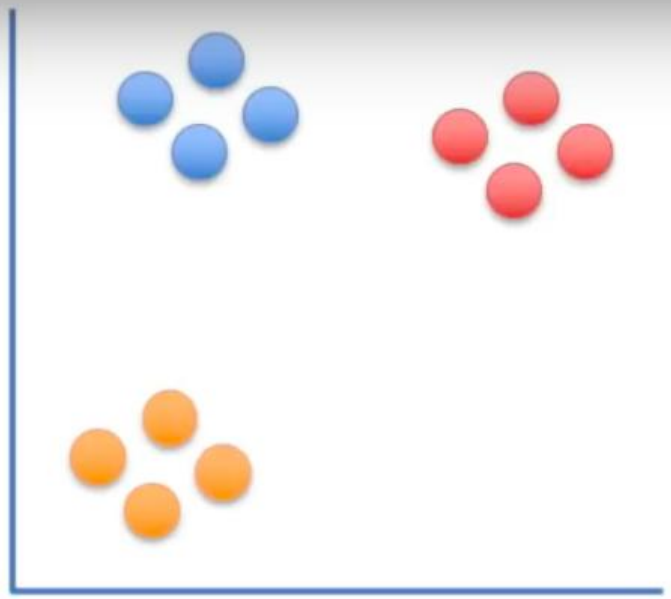


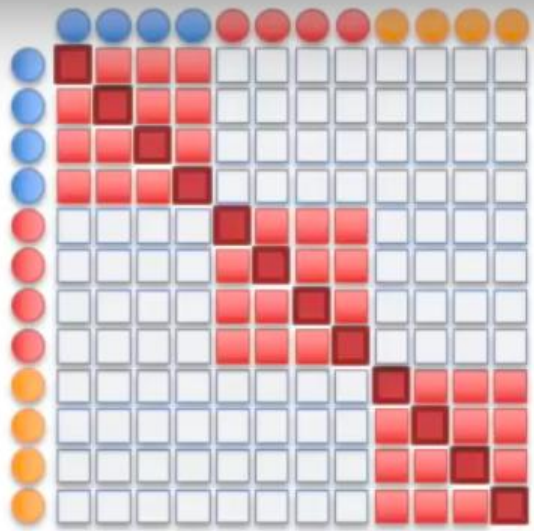

The goal of moving this point is…we want to make this row…look like this row.



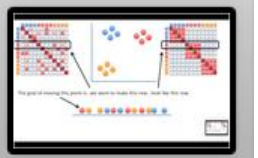
StatQuest: t-SNE, Clearly Explained



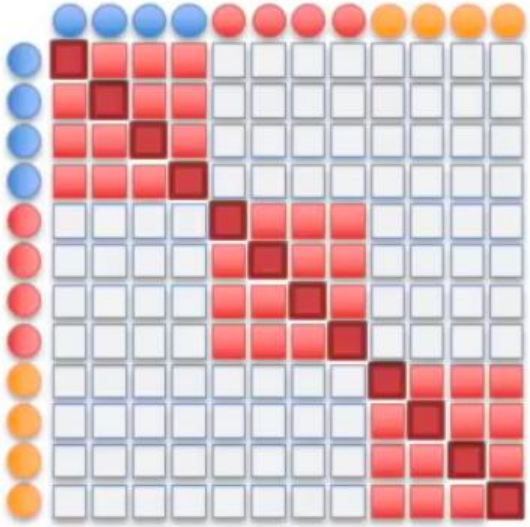


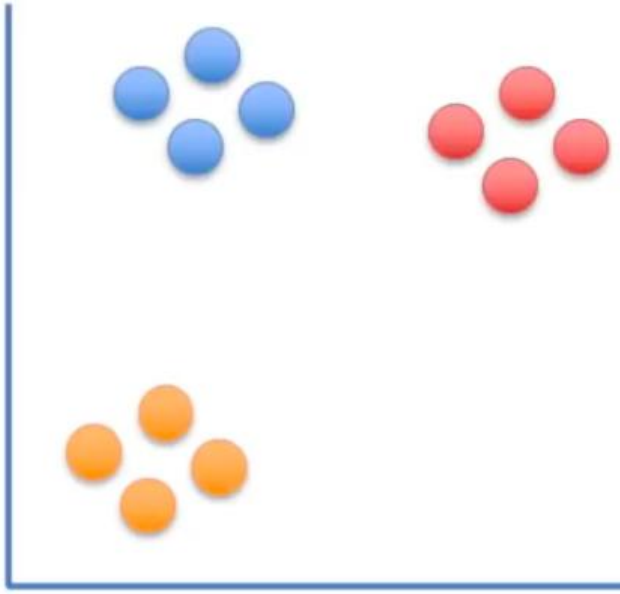


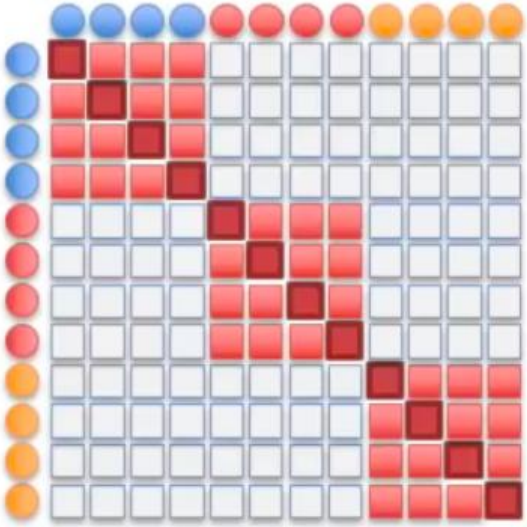
t-SNE moves the points a little bit at a time, and each step it chooses a direction that makes the matrix on the left more like the matrix on the right.



10:45 / 11:47 • Step 3: Move points in low-d >

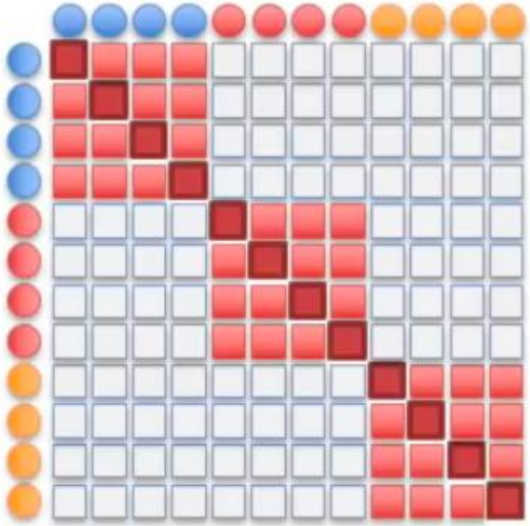


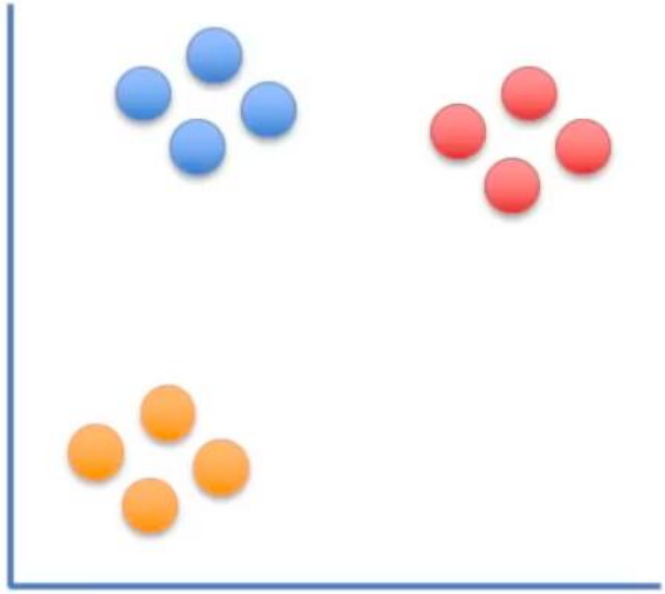


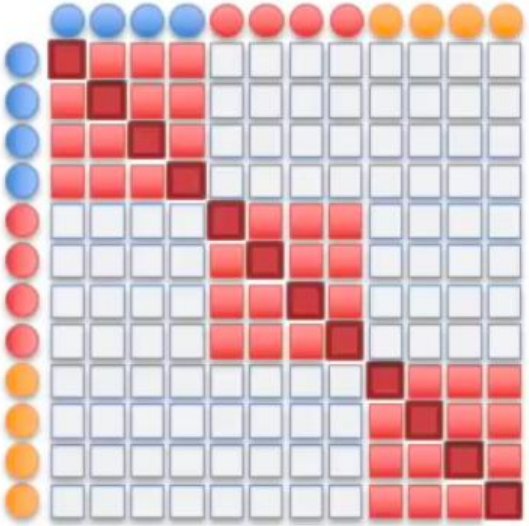


t-SNE moves the points a little bit at a time, and each step it chooses a direction that makes the matrix on the left more like the matrix on the right.

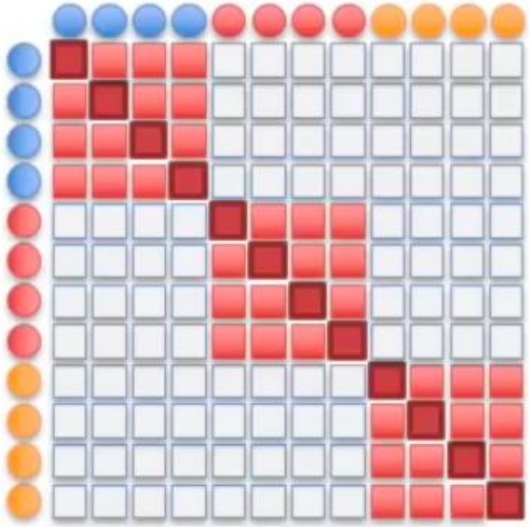
It uses small steps, because it's a little bit like a chess game and can't be solved all at once. Instead, it goes one move at at time.

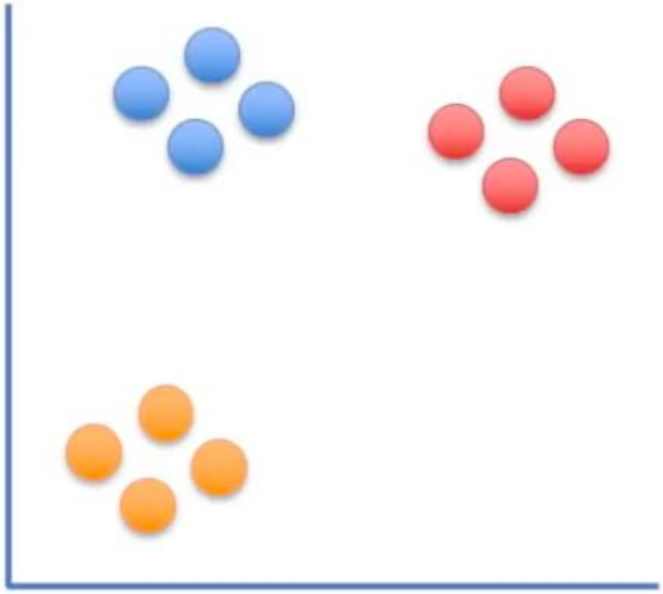


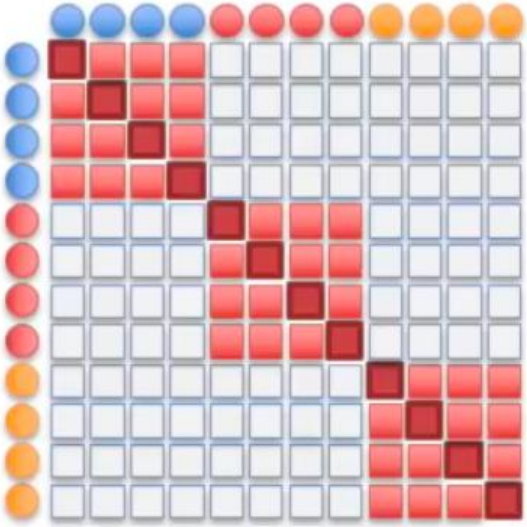




Now to finally tell you why the “t-distribution” is used…







...without it the clusters would all clump up in the middle and be harder to see.