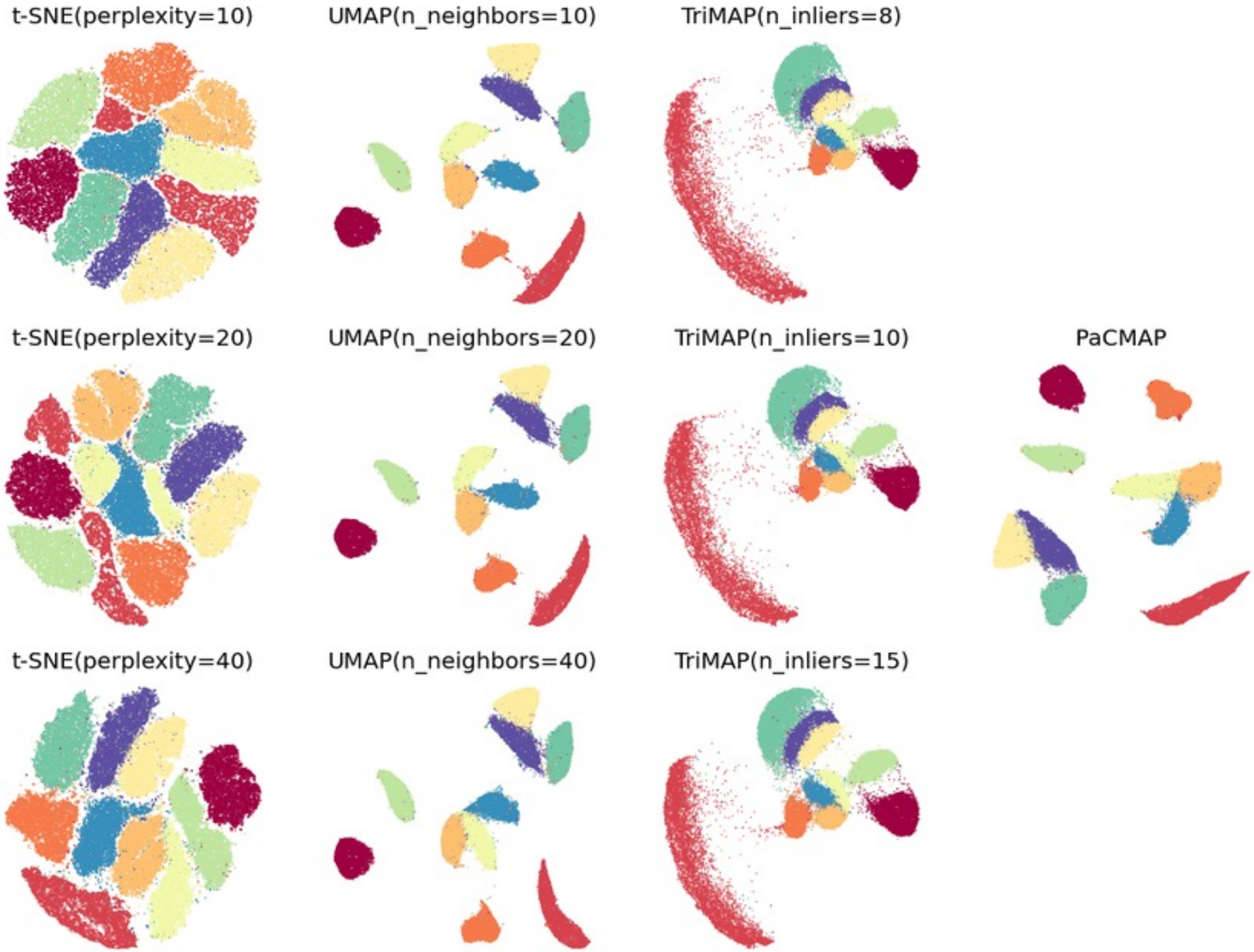
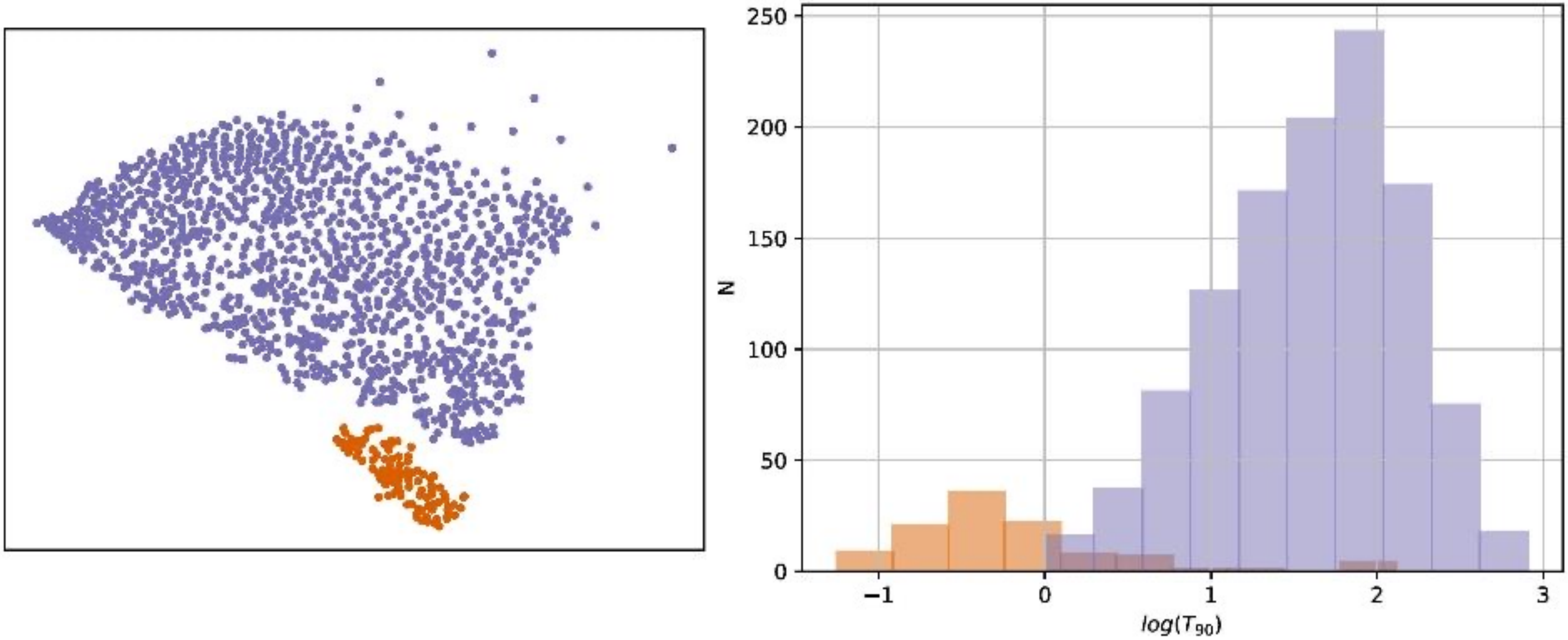
Group "similar" things together

Wang et al. 2020



t-SNE map for *Swift* light curves

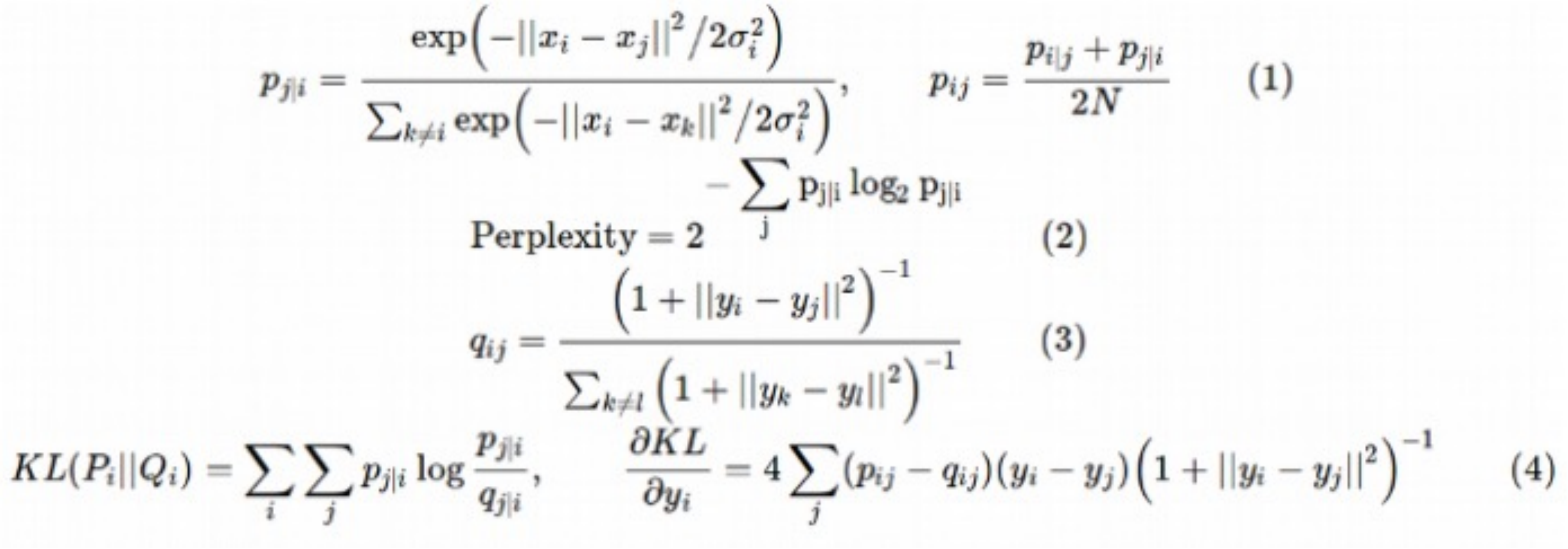
Kragh Jespersen et al. 2020



OK, but can you tell me how t-SNE actually works?

t-Stochastic Neighbor Embedding (t-SNE)

N. Oskolkov, towardsdatascience.com



$$p_{j|i} = \frac{\exp\left(-||x_i - x_j||^2 / 2\sigma_i^2\right)}{\sum_{k\neq i}\exp\left(-||x_i - x_k||^2 / 2\sigma_i^2\right)}, \qquad p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N} \qquad (1)$$

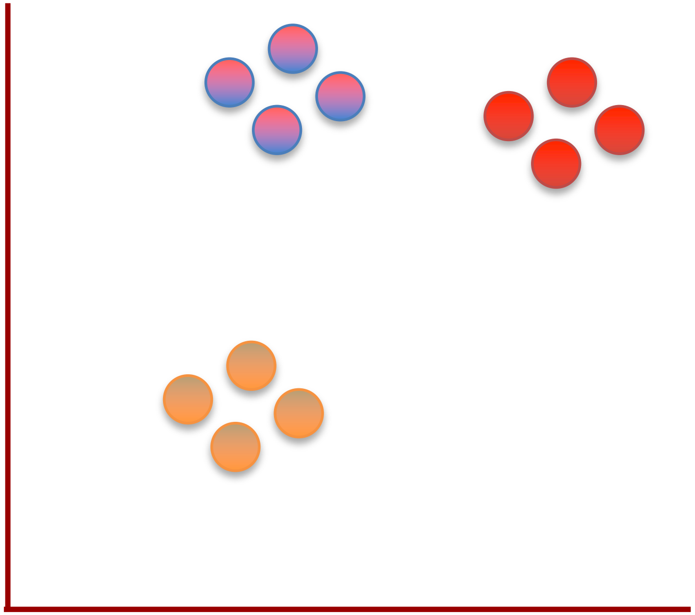
$$\text{Perplexity} = 2^{-\sum_{j} p_{j|i}\log_2 p_{j|i}} \qquad (2)$$

$$q_{ij} = \frac{\left(1 + ||y_i - y_j||^2\right)^{-1}}{\sum_{k\neq l}\left(1 + ||y_k - y_l||^2\right)^{-1}} \qquad (3)$$

$$KL(P_i||Q_i) = \sum_i \sum_j p_{j|i}\log\frac{p_{j|i}}{q_{j|i}}, \qquad \frac{\partial KL}{\partial y_i} = 4\sum_j (p_{ij} - q_{ij})(y_i - y_j)\left(1 + ||y_i - y_j||^2\right)^{-1} \qquad (4)$$

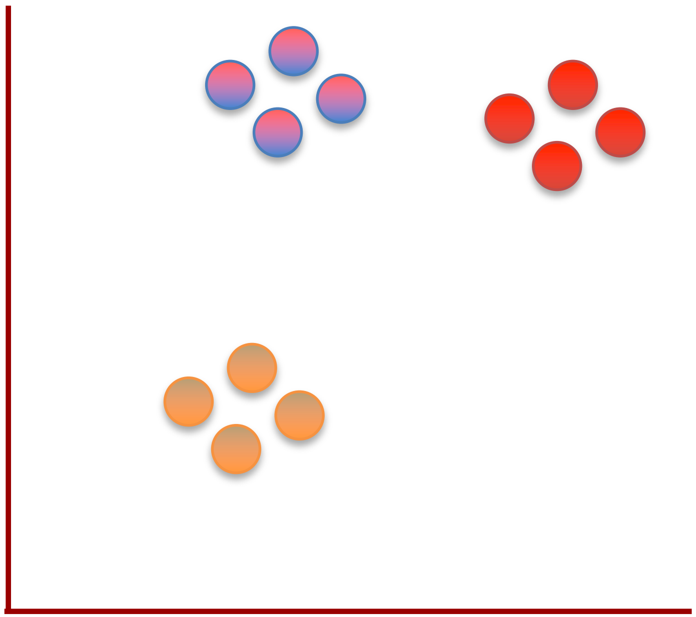
Wait…I'm a visual learner. Maybe in a cartoon?

Here’s a basic 2-D scatter plot.



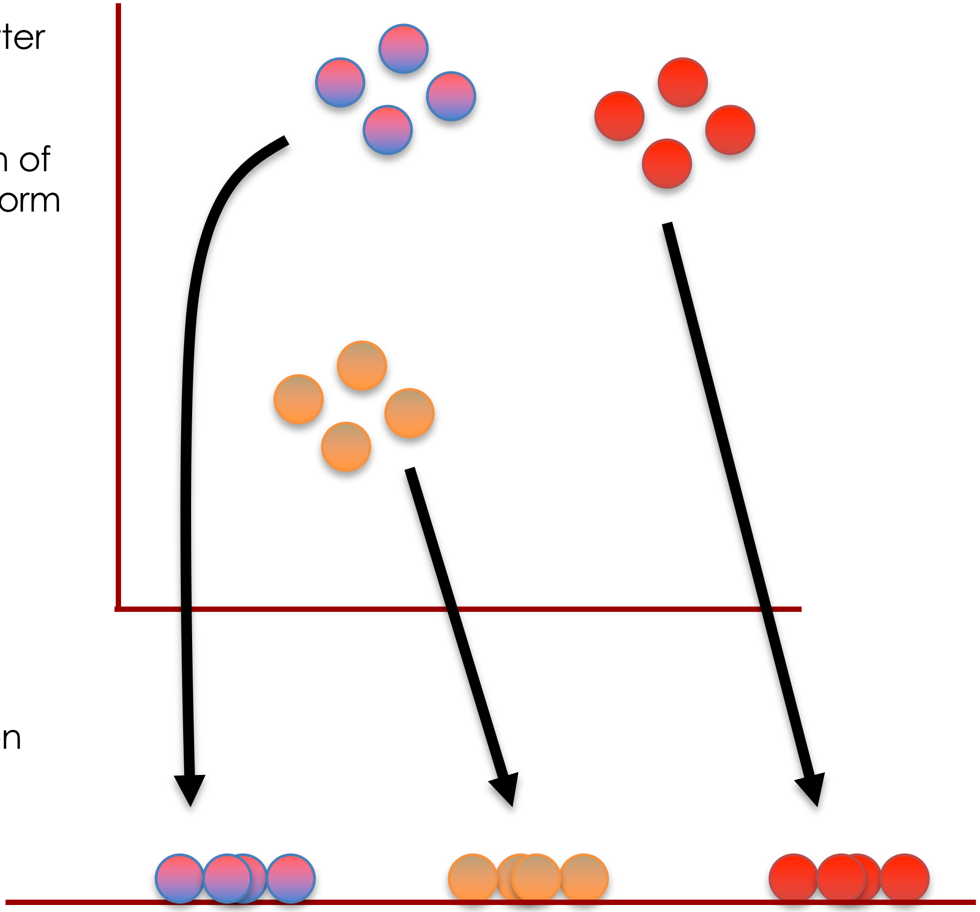
Here’s a basic 2-D scatter plot.

Let’s do a walk through of how t-SNE would transform this graph…

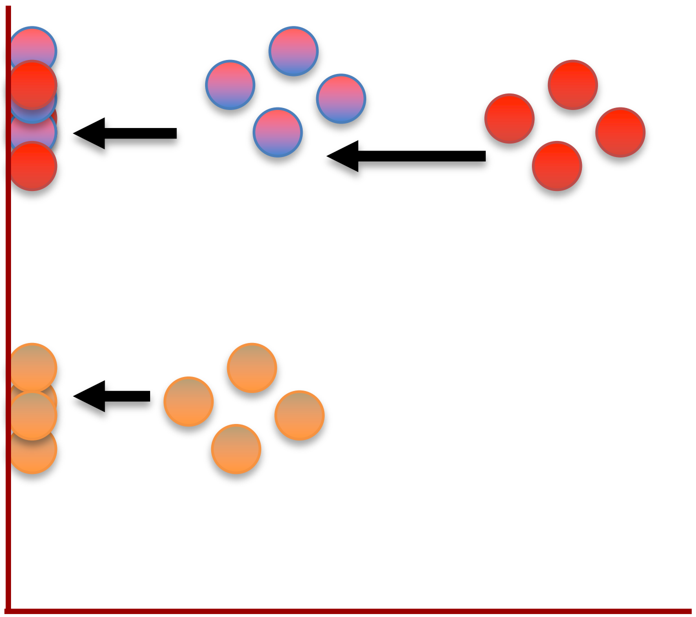


Here's a basic 2-D scatter plot.

Let's do a walk through of how t-SNE would transform this graph…

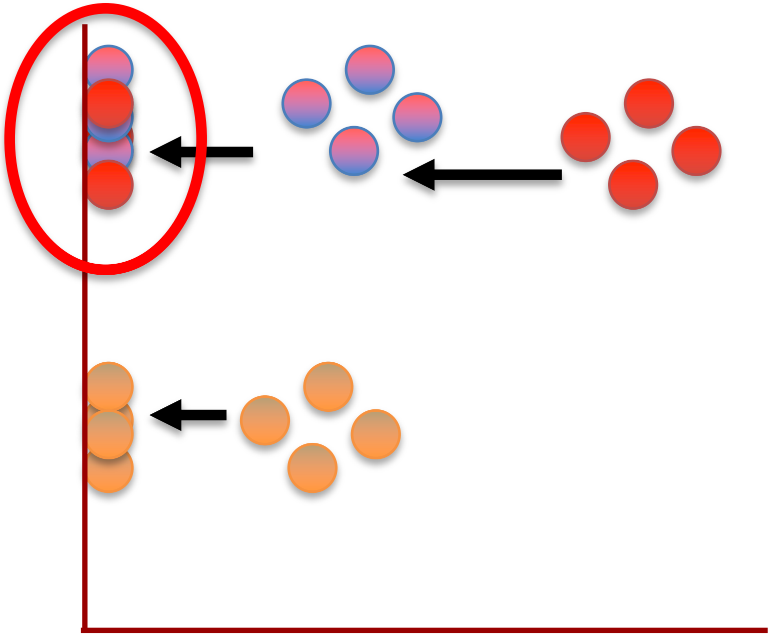


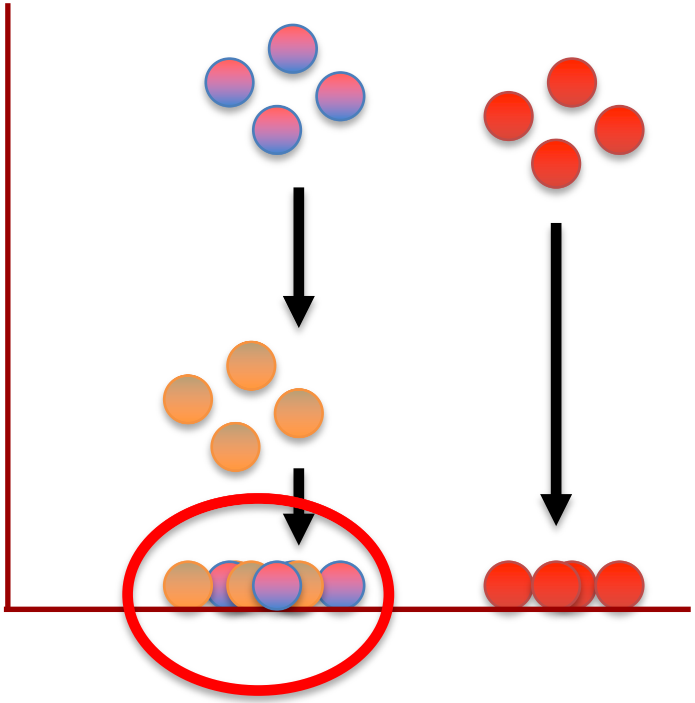
…into a flat, 1-D plot on a number line.



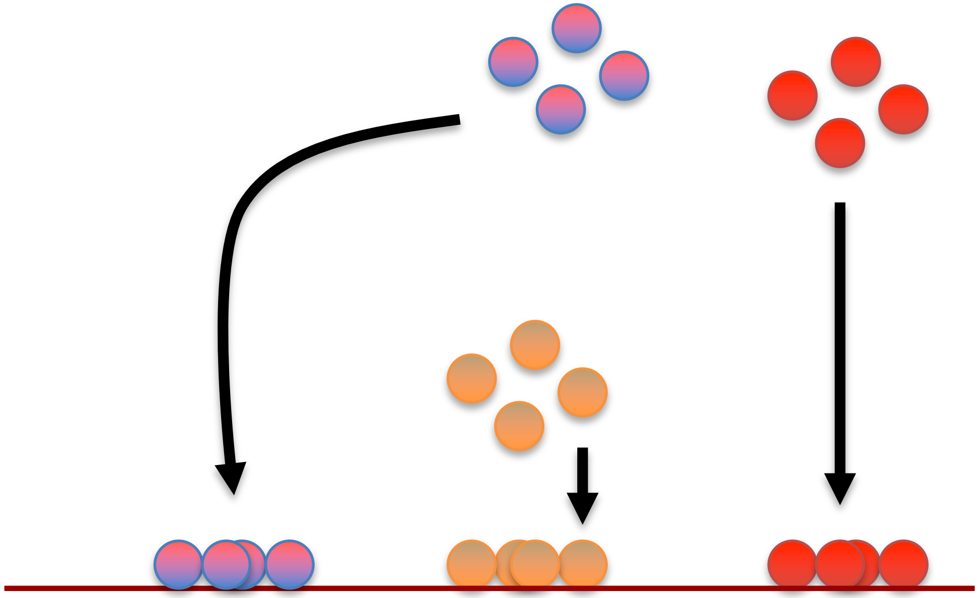
NOTE: If we just projected the data onto one of the axes, we'd just get a big mess that doesn't preserve the original clustering.

Instead of two distinct clusters, we just see a mishmash.

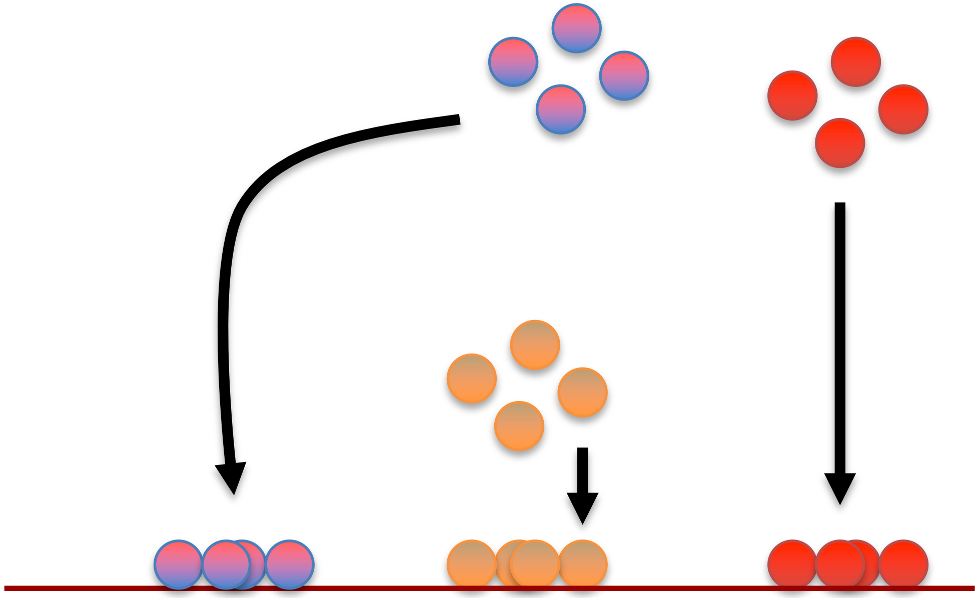




Same here...

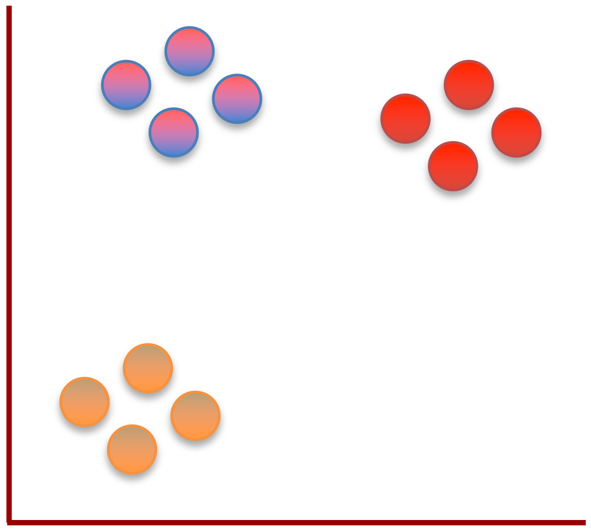


What t-SNE does is find a way to project data into a low dimensional space (in this case, the 1-D number line) so that the clustering in the high dimensional space (in this case, the 2-D scatter plot) is preserved.



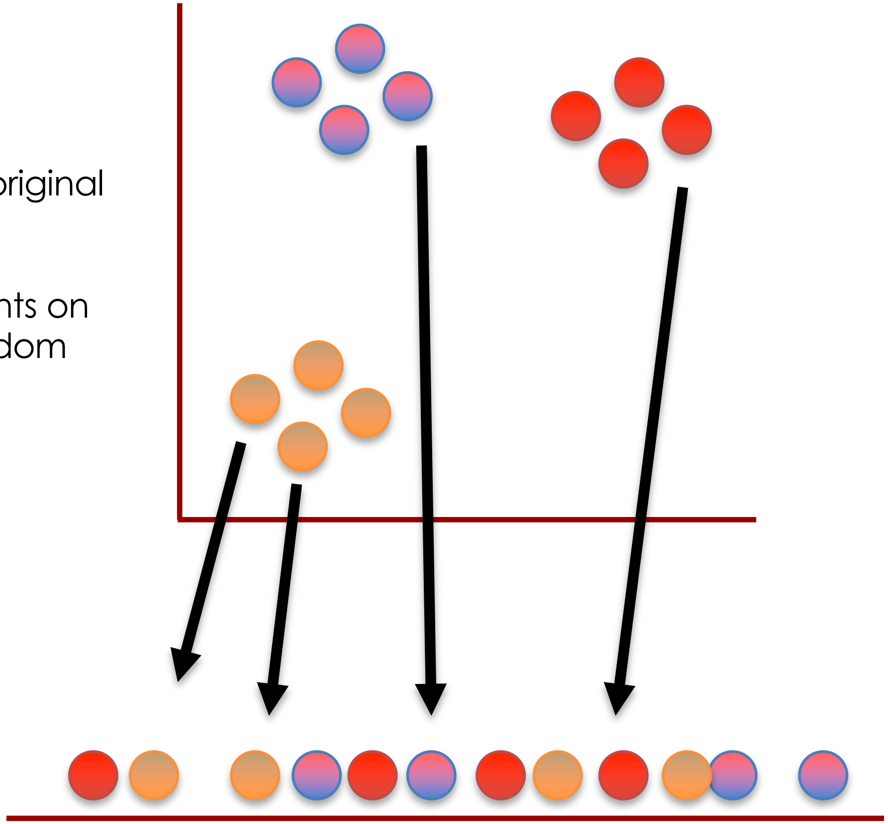
So let's step through the basic ideas of how t-SNE does this.

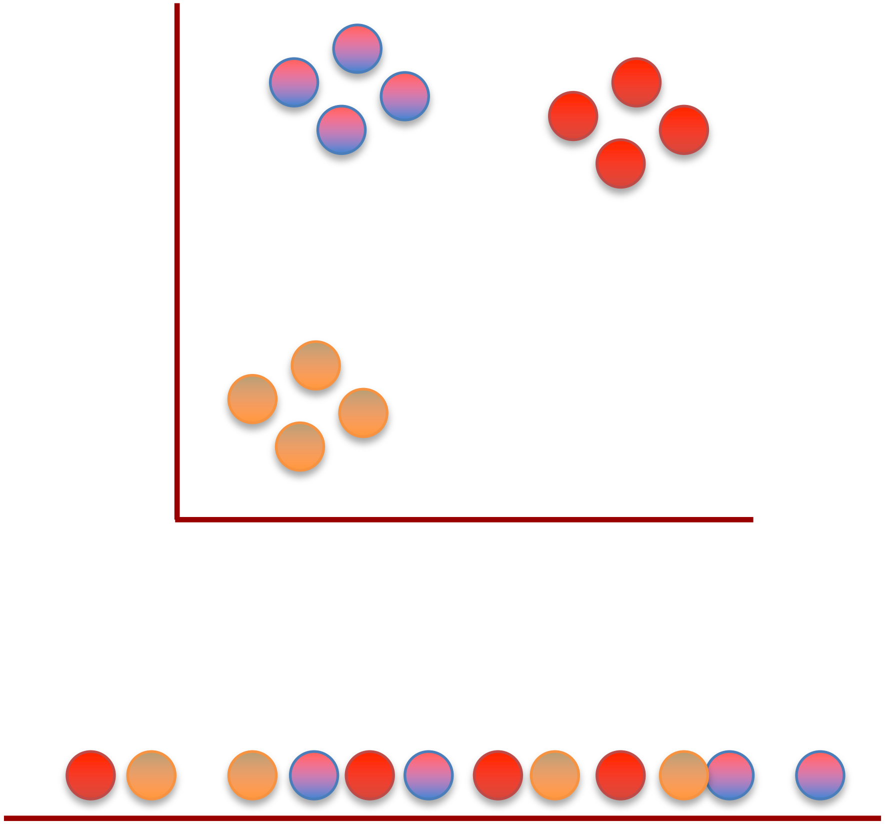
We'll start start with the original scatter plot…



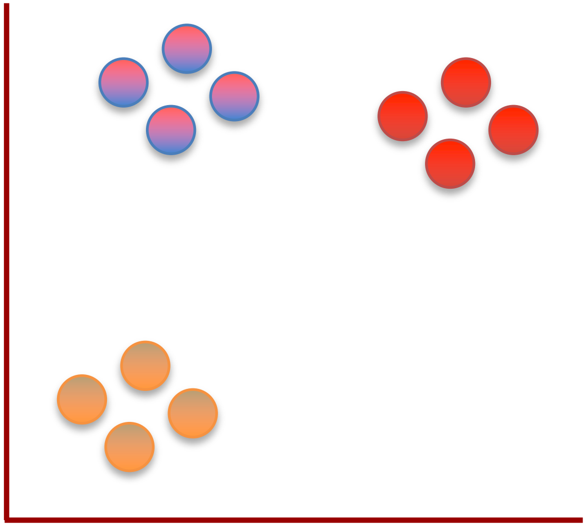
We'll start start with the original scatter plot…

… then we'll put the points on the number line in a random order.

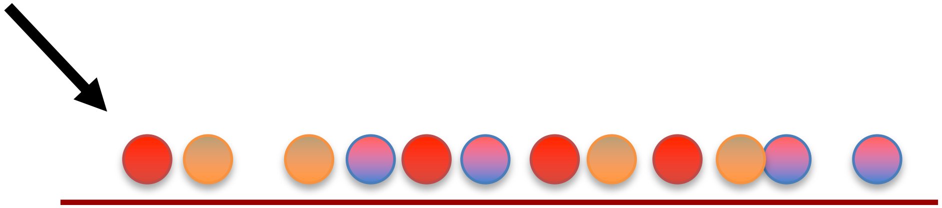


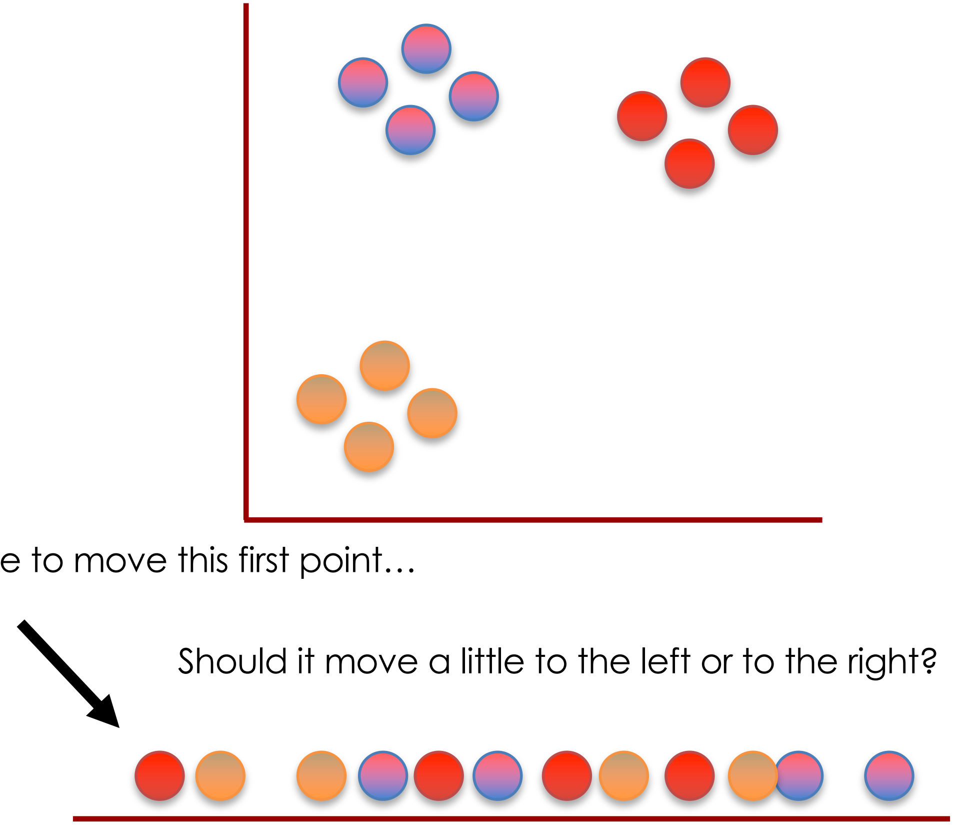


From here on out, t-SNE moves these points, a little bit at a time, until it has clustered them.



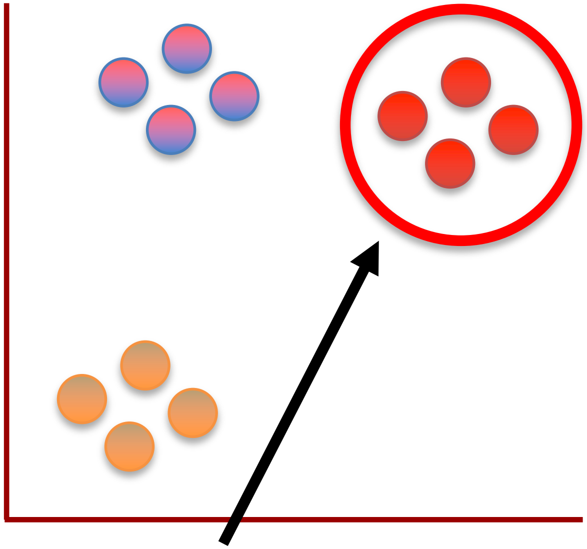
Let’s figure out where to move this first point…



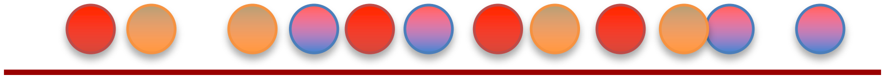


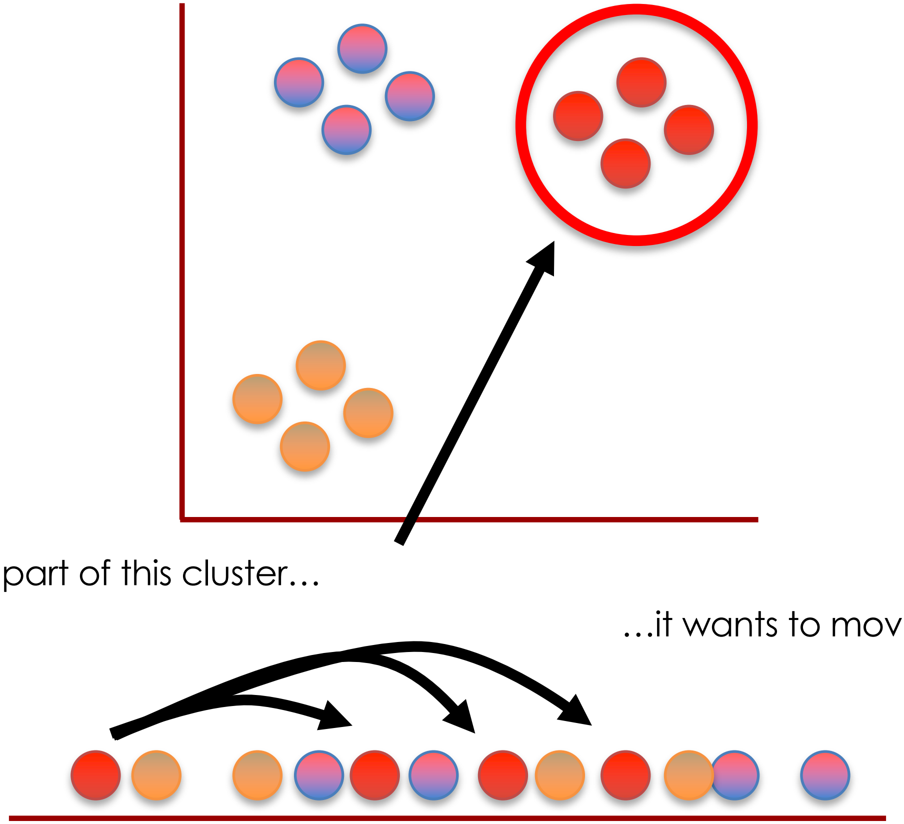
Let’s figure out where to move this first point…

Should it move a little to the left or to the right?



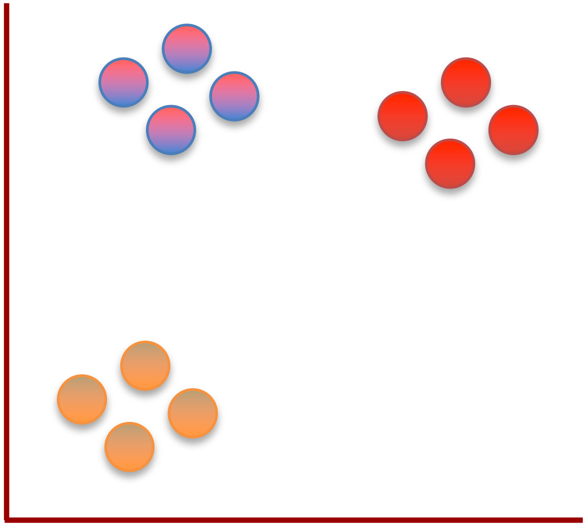
Because it is part of this cluster…



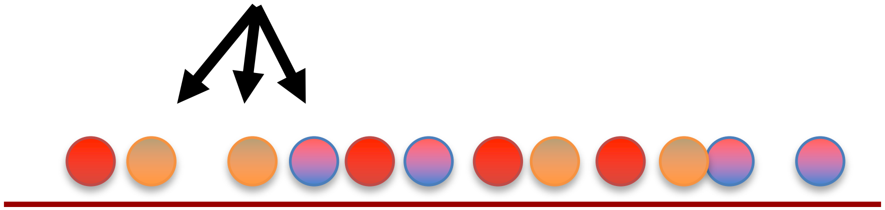


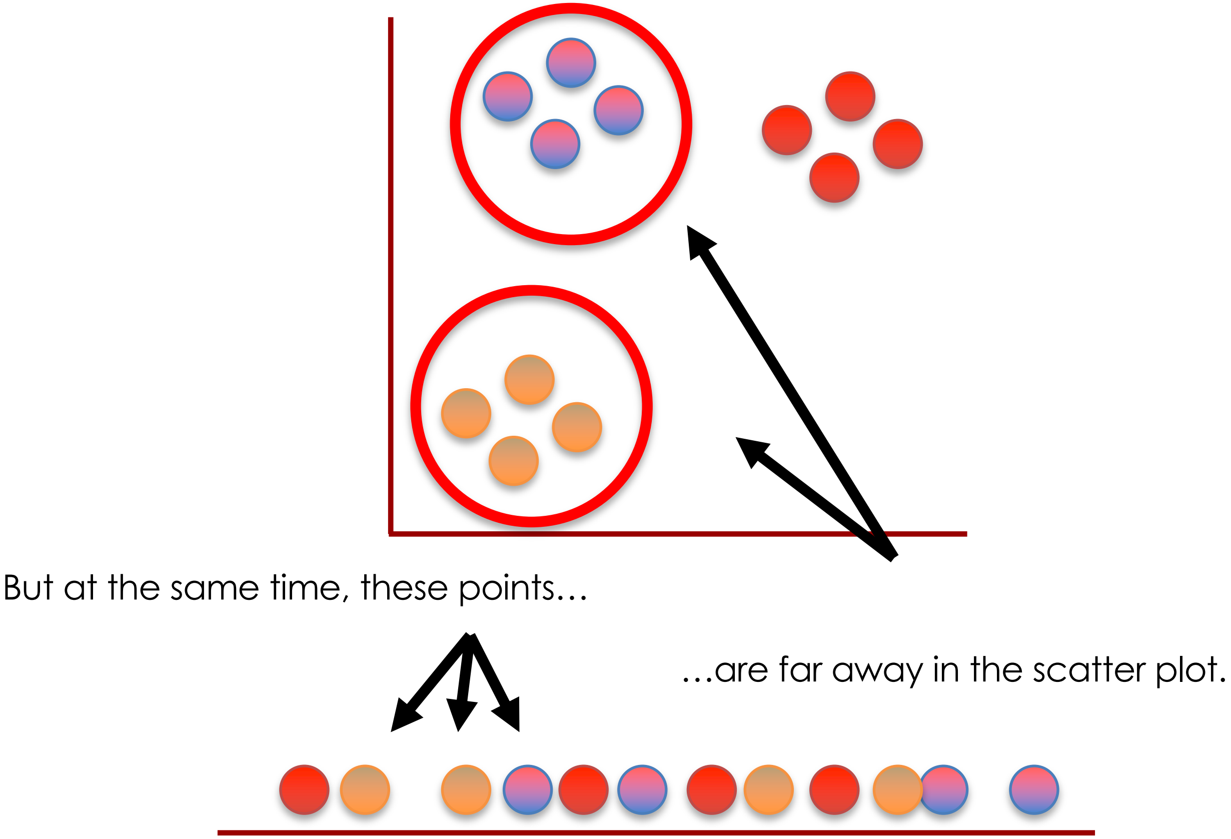
Because it is part of this cluster…

…it wants to move closer to these points.



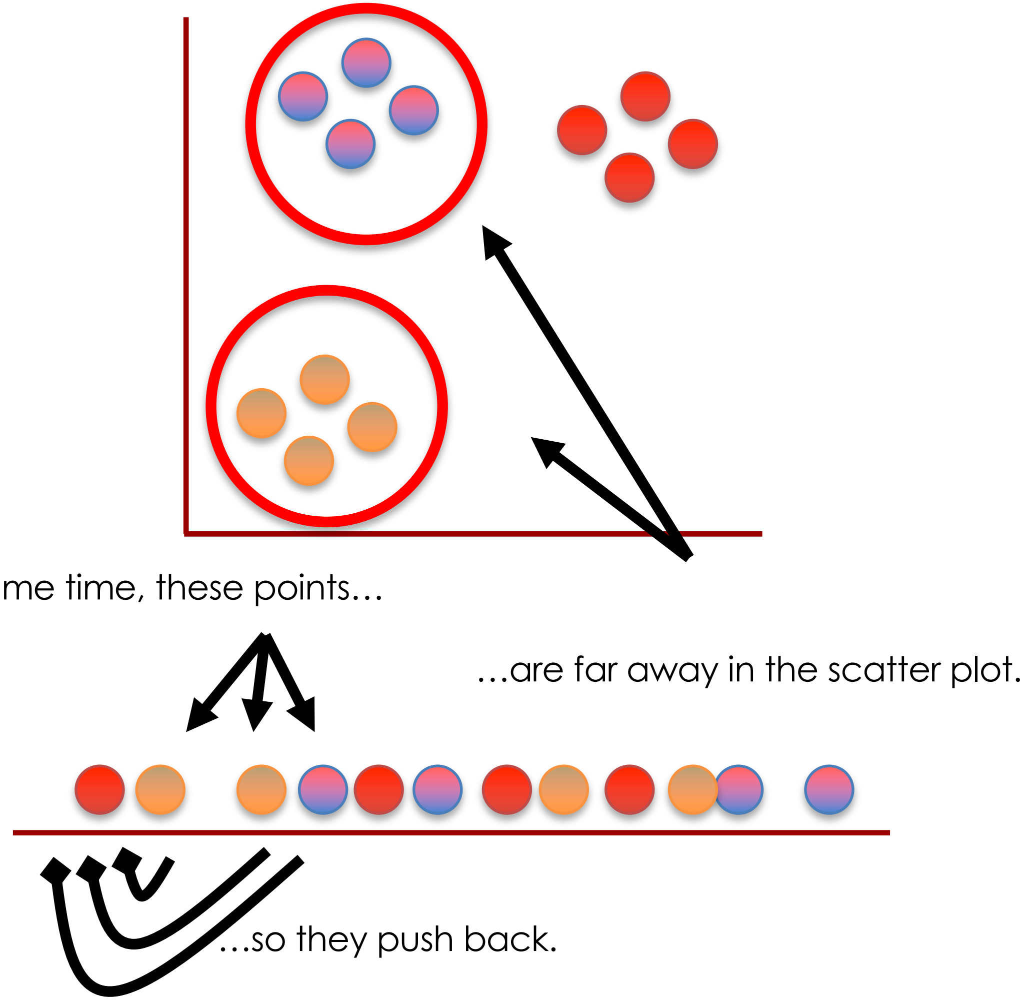
But at the same time, these points…





But at the same time, these points…

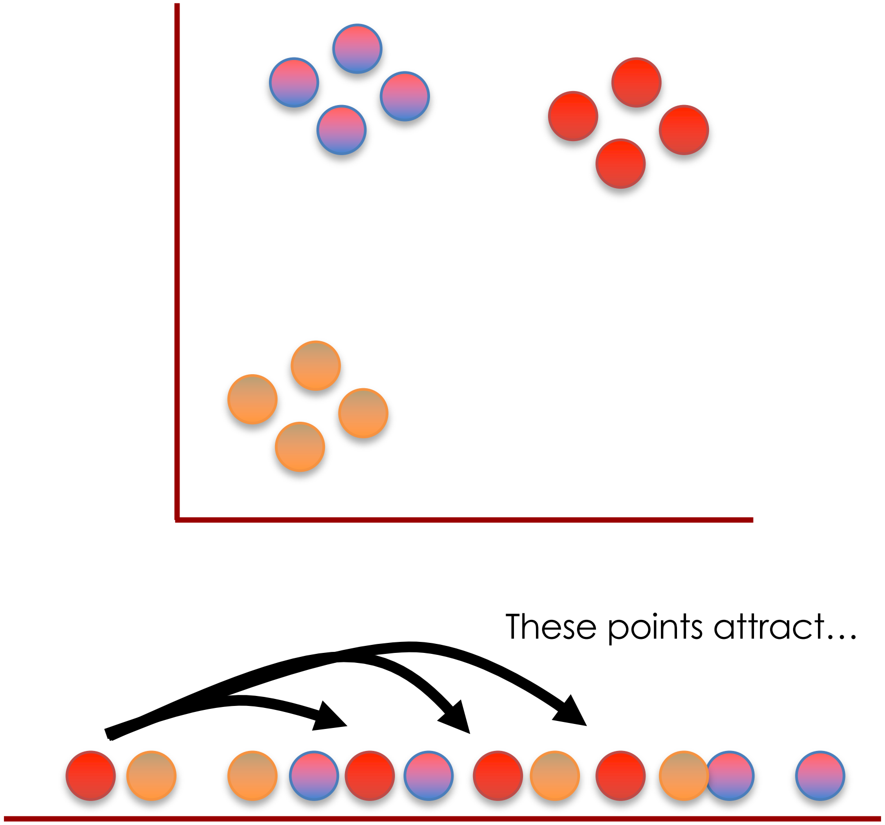
…are far away in the scatter plot.

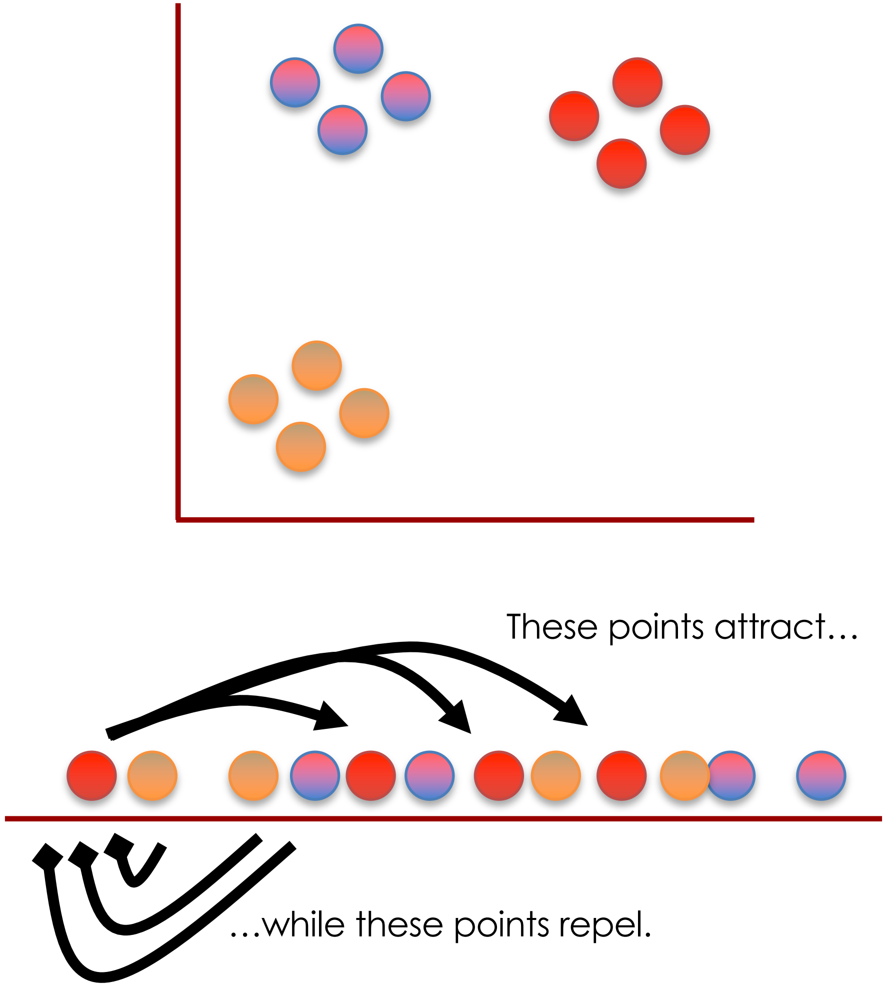


But at the same time, these points…

…are far away in the scatter plot.

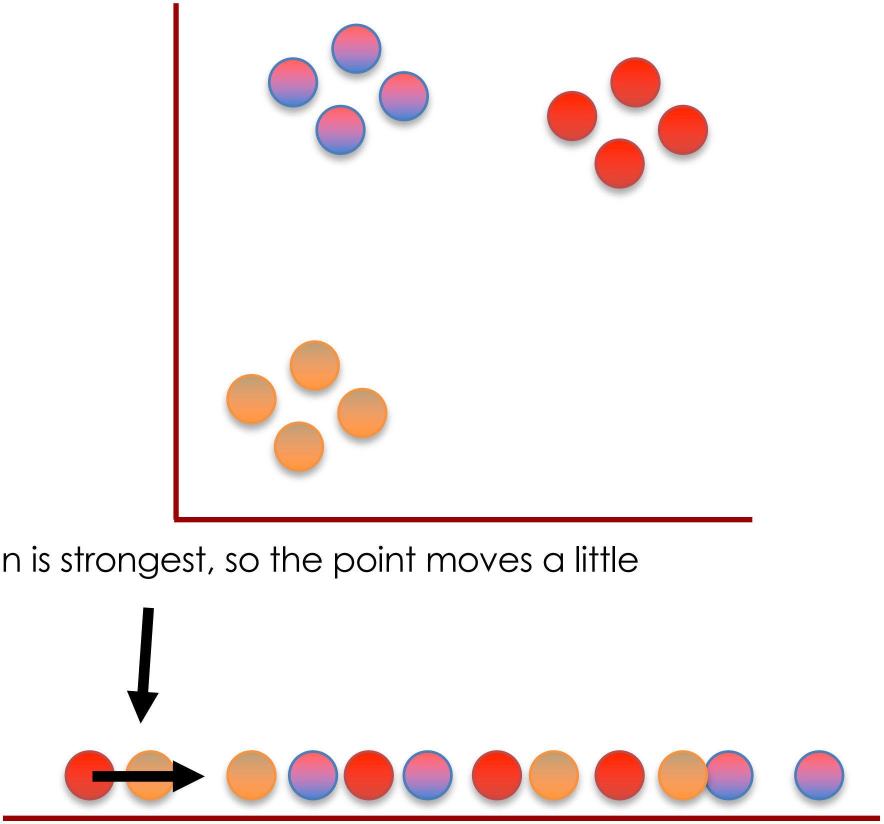
…so they push back.


These points attract…

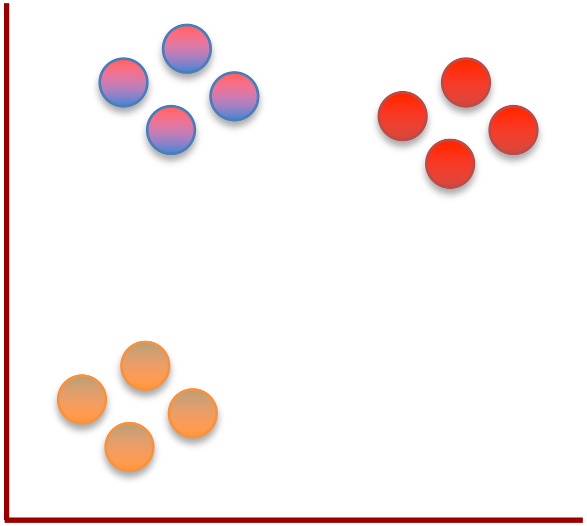


These points attract…

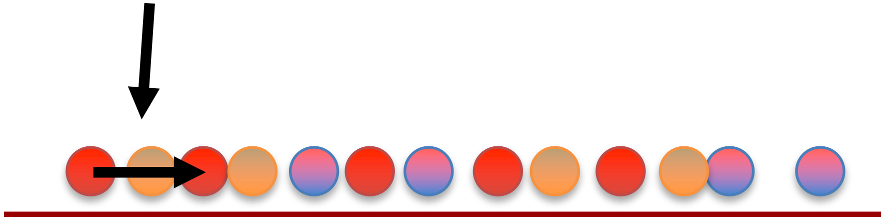
…while these points repel.

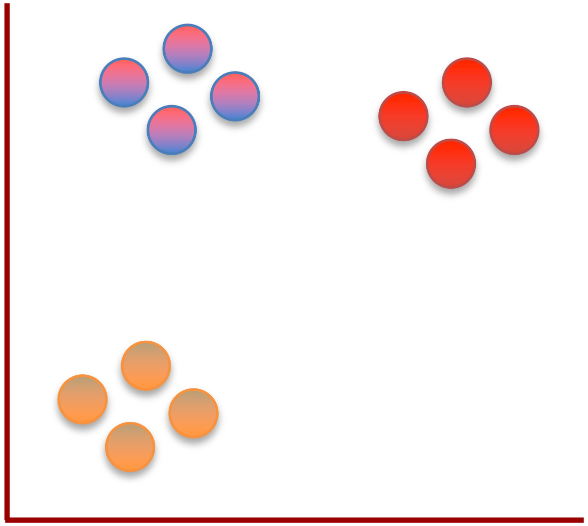


In this case, the attraction is strongest, so the point moves a little to the right.

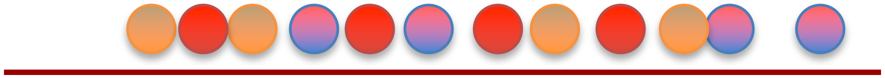


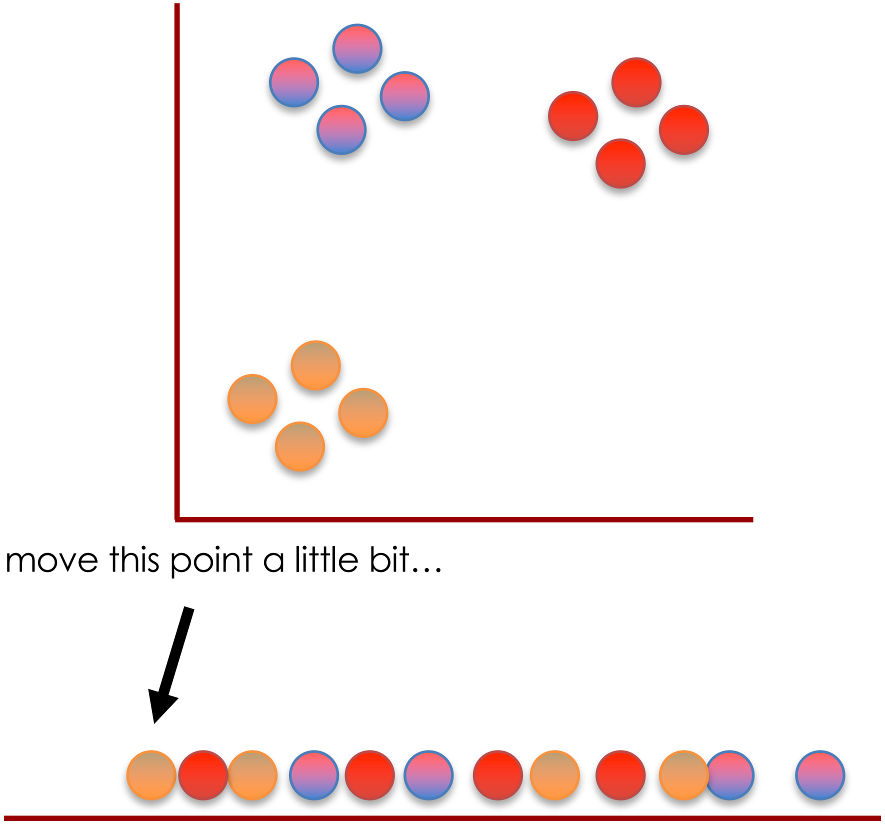
In this case, the attraction is strongest, so the point moves a little to the right.



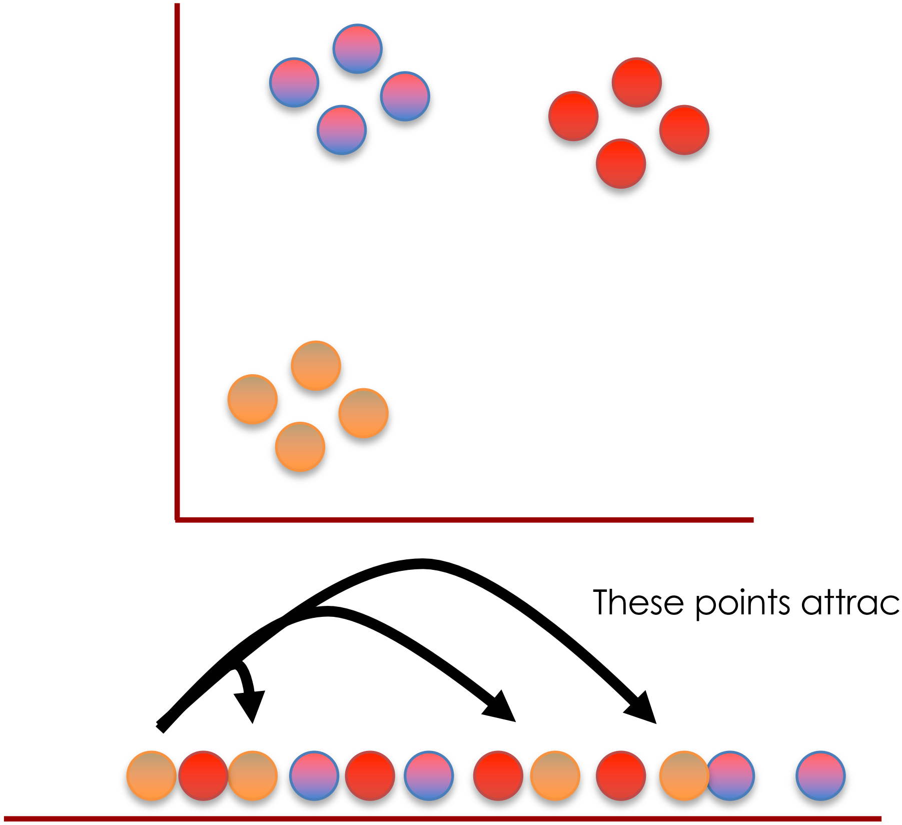


BAM!

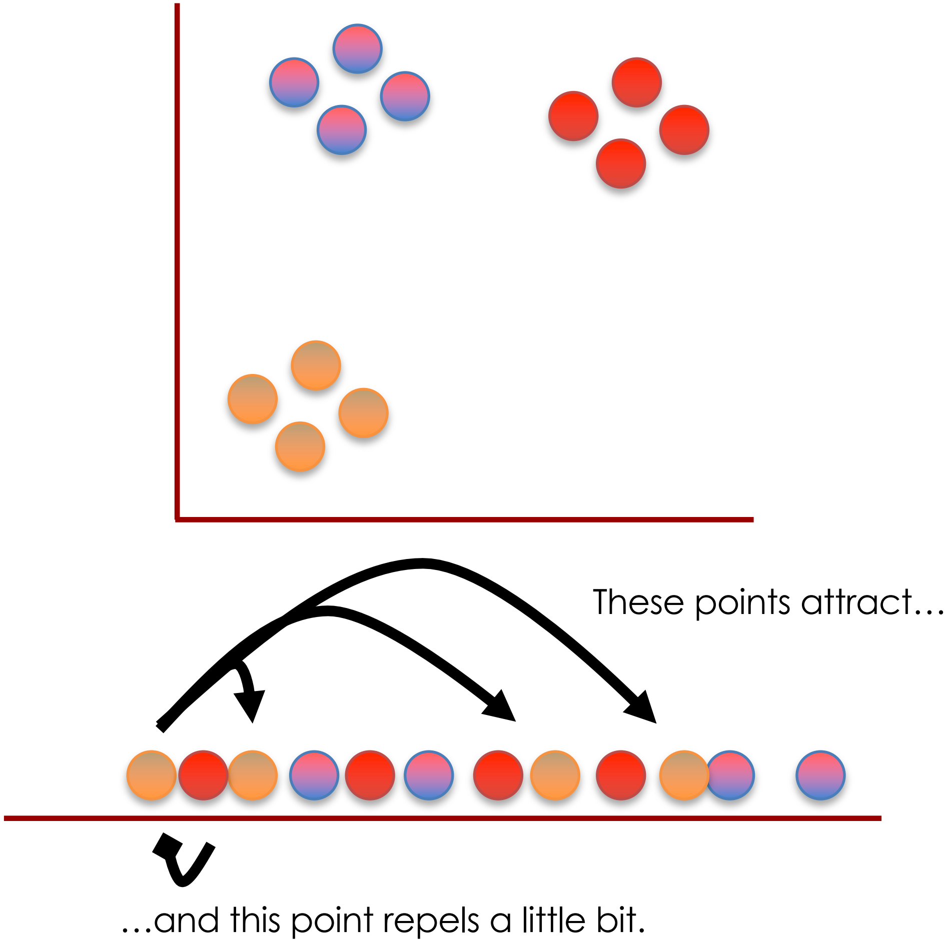




Now let's move this point a little bit…

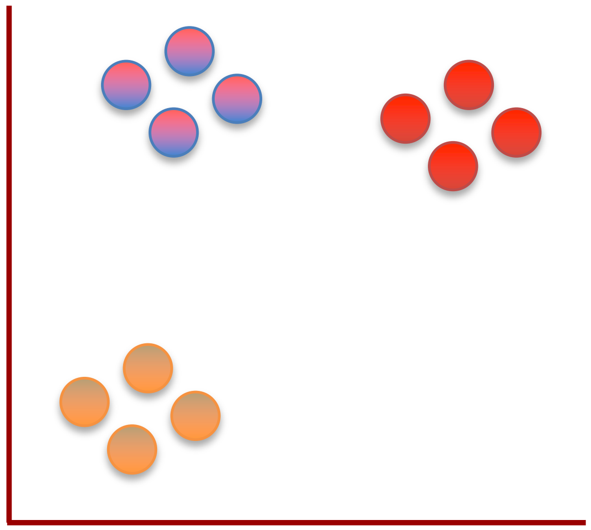


These points attract…

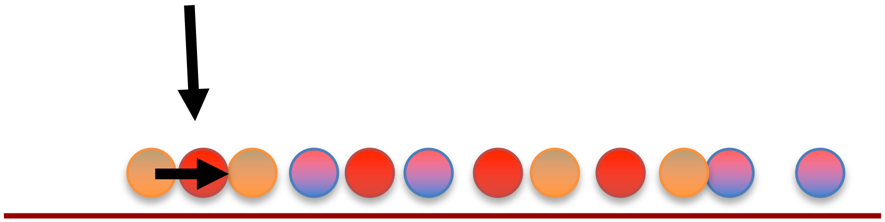


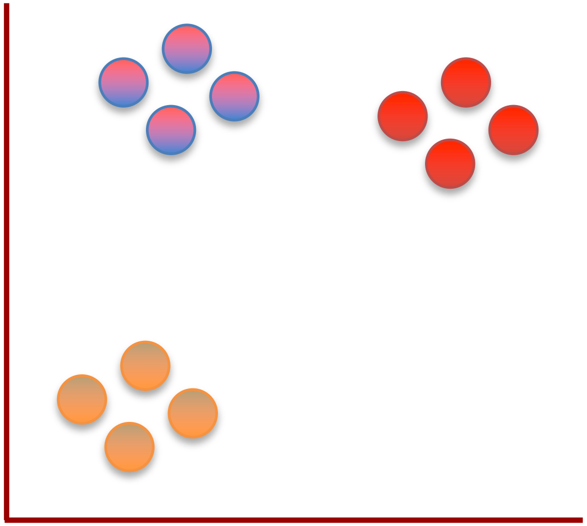
These points attract…

…and this point repels a little bit.

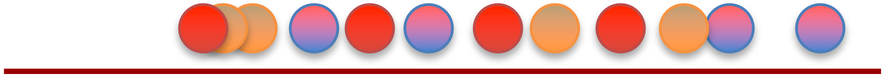


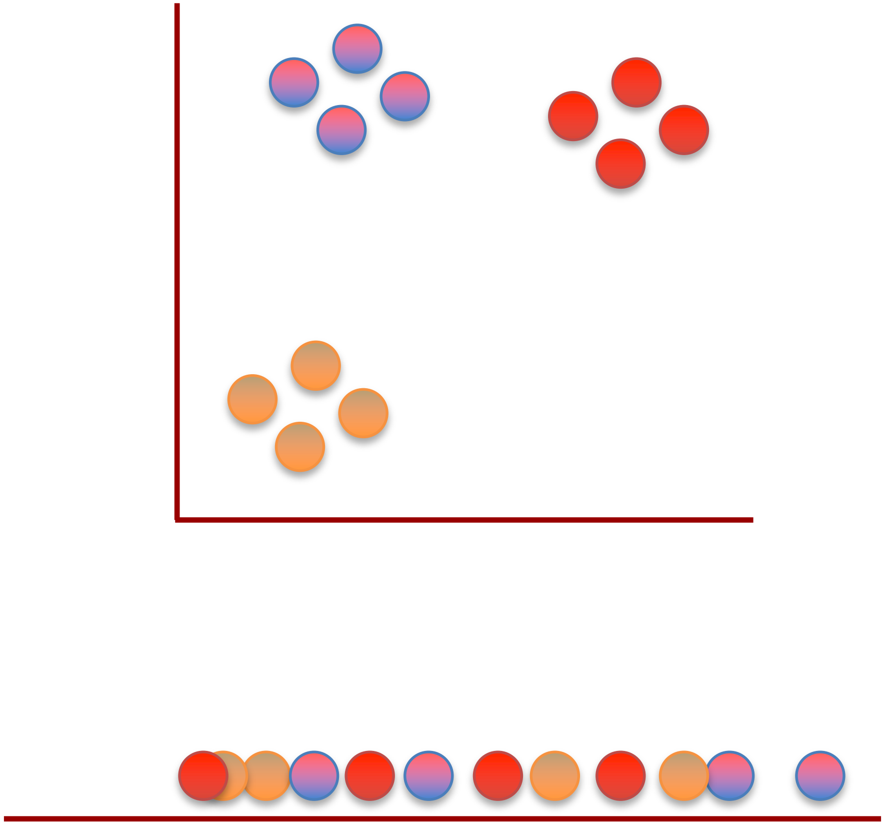
So it moves a little to closer to the other orange points.





Double BAM!

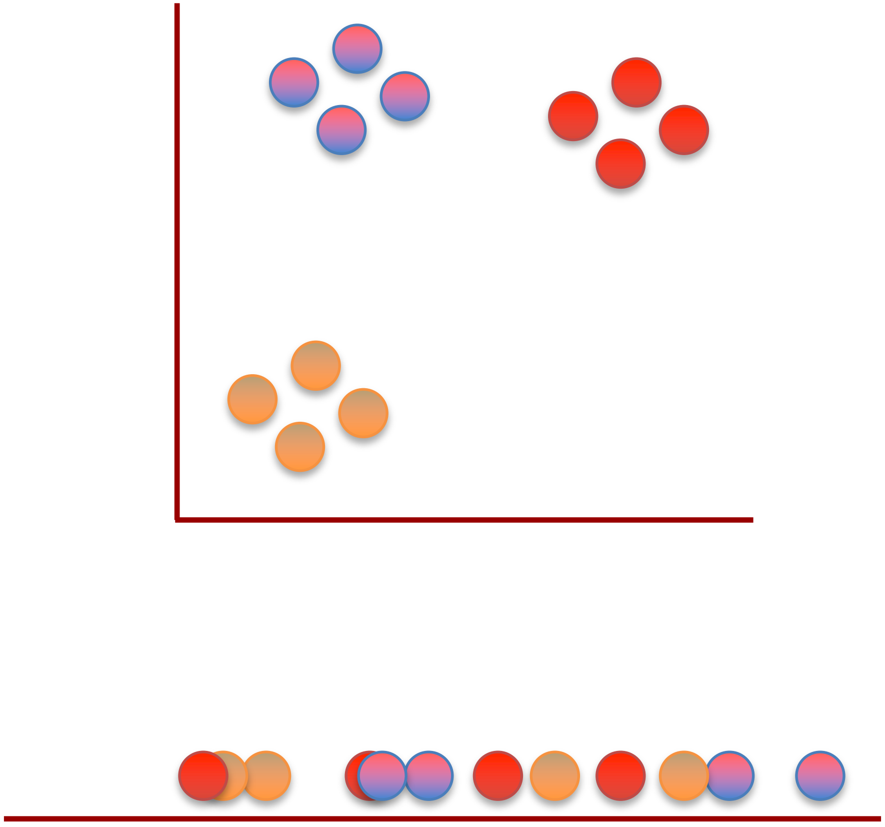


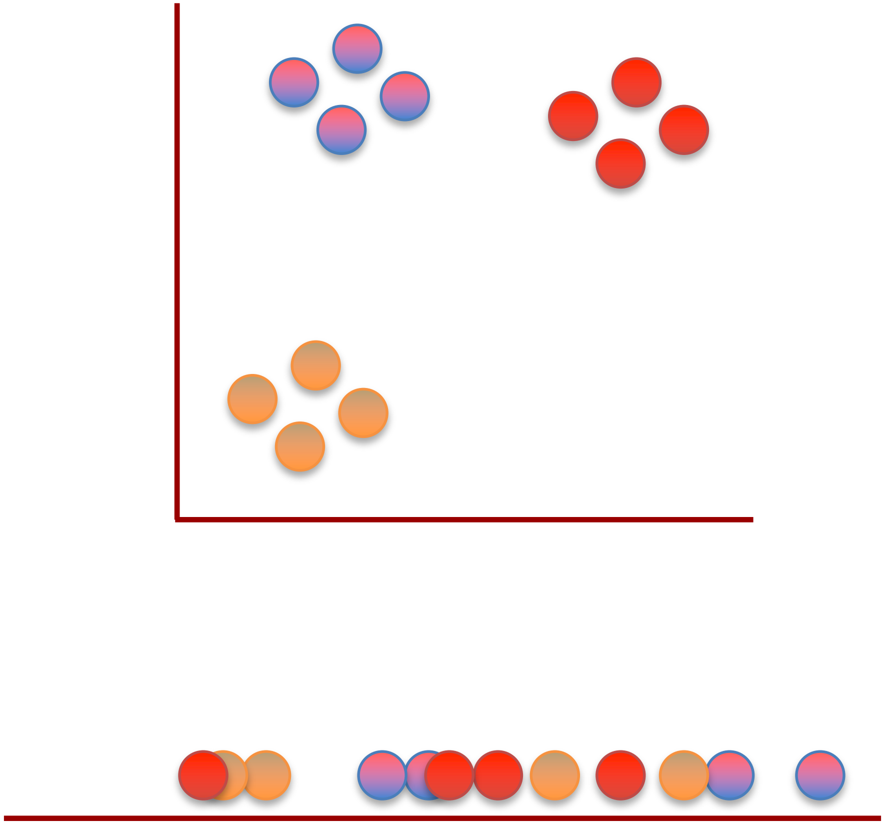


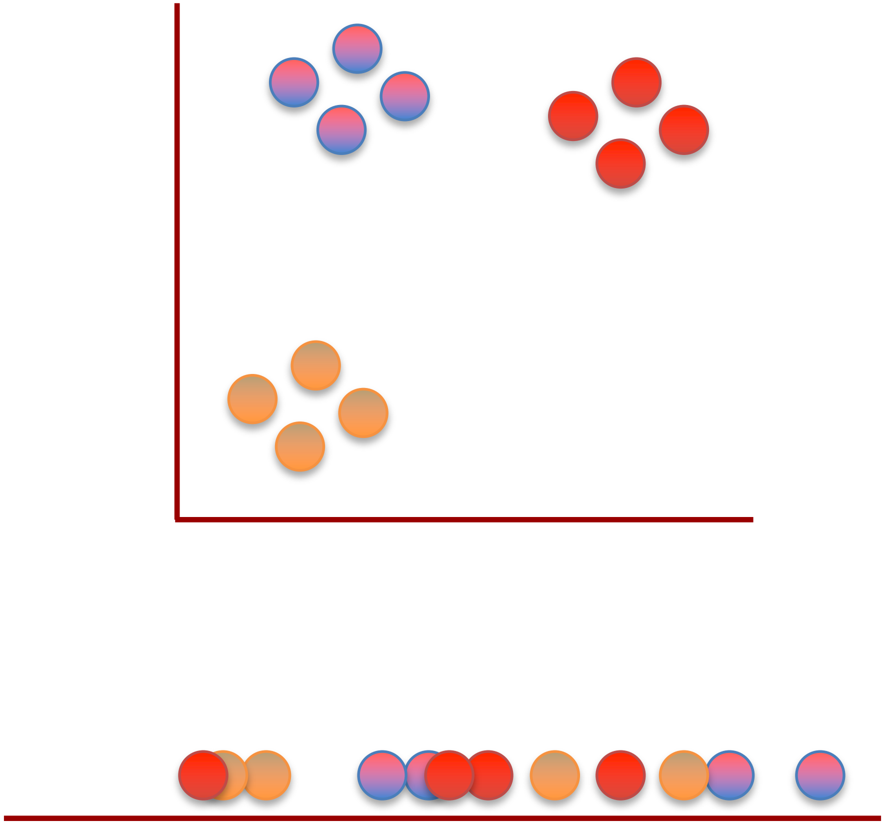
At each step, a point on the line is attracted to points it is near in the scatter plot, and repelled by points it is far from…

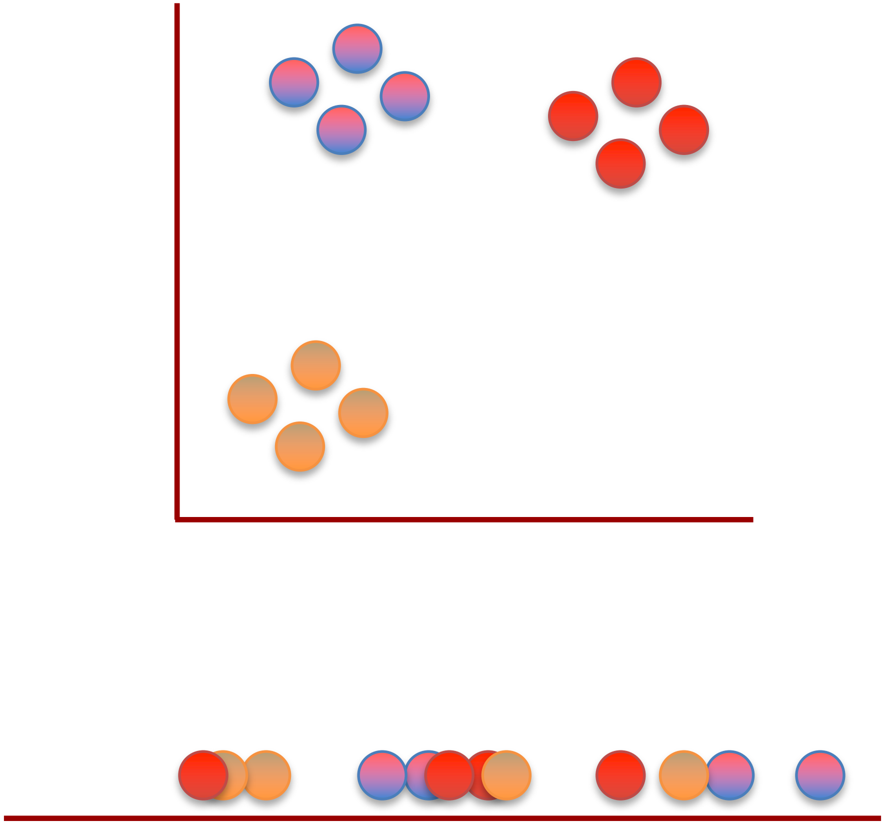
COSMIC DAWN CENTER

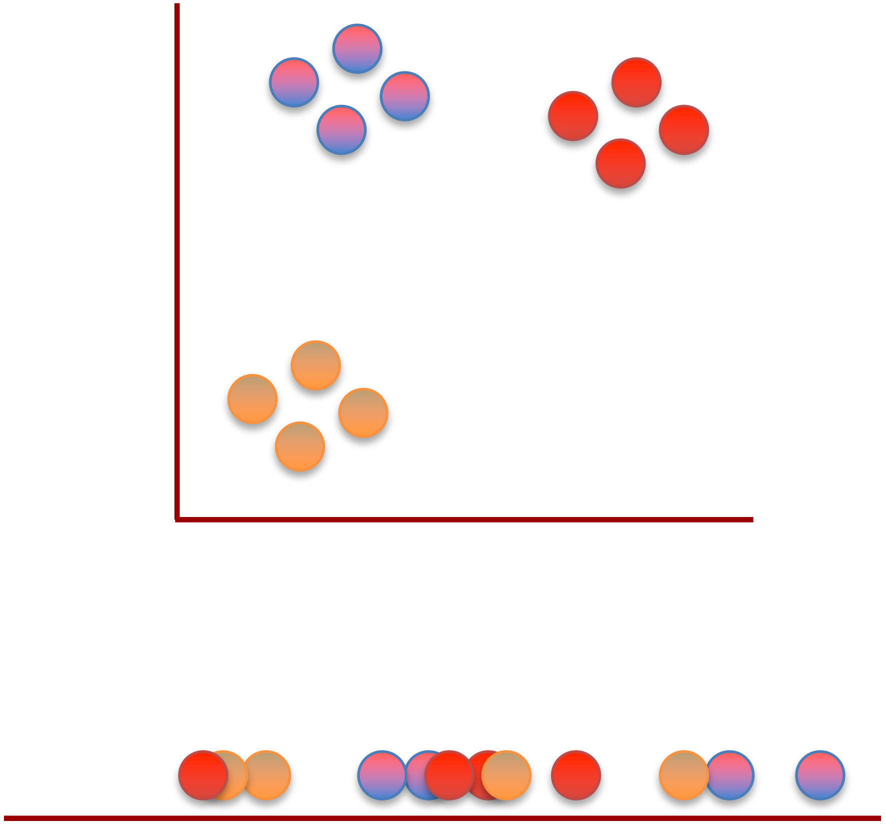
DAWN

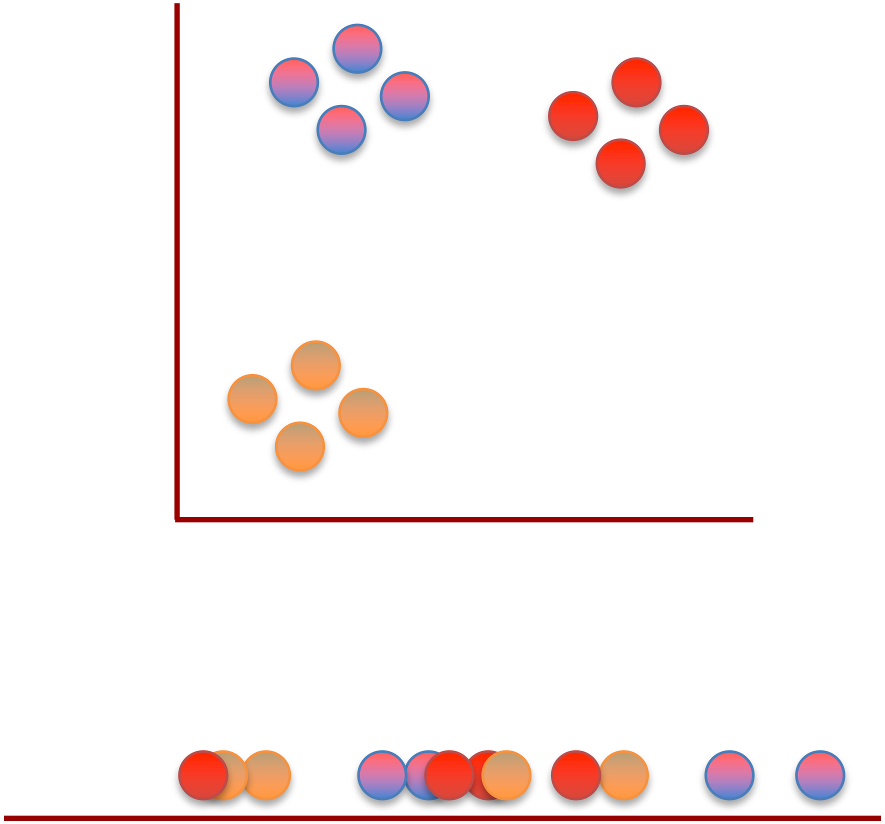








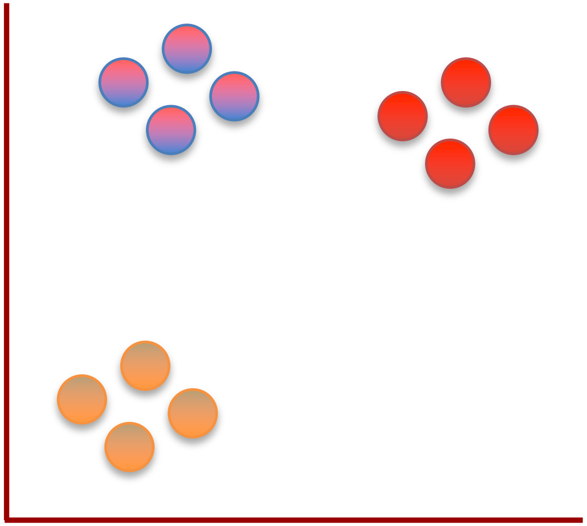


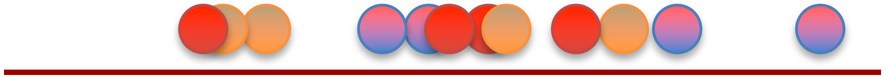


COSMIC DAWN CENTER

DAWN

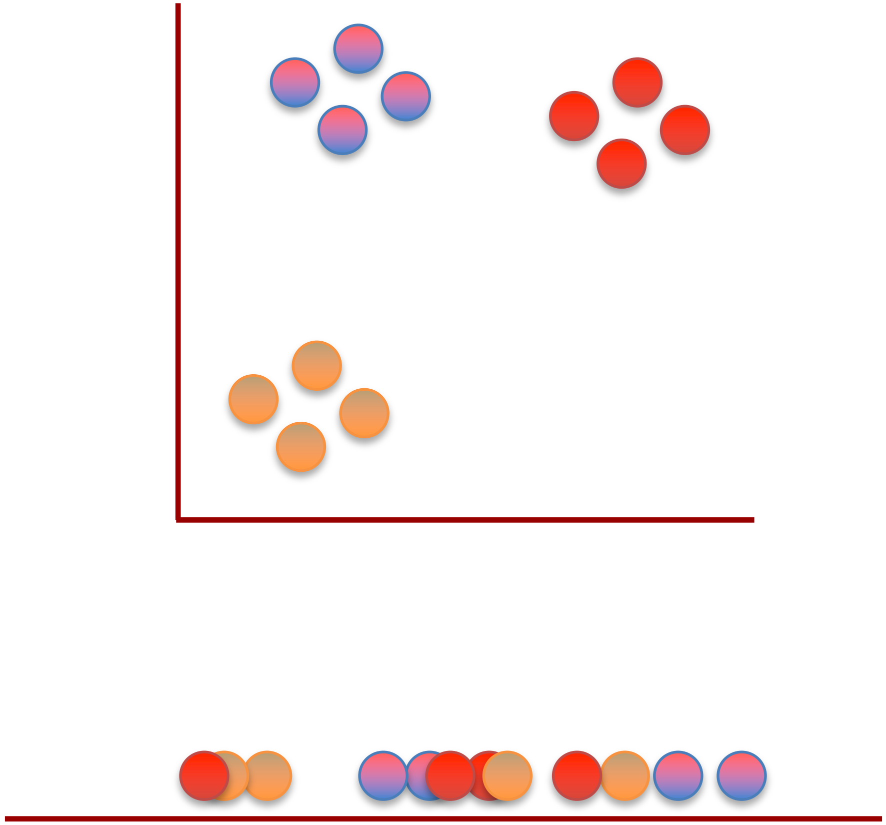
Wednesday May 3, 2023

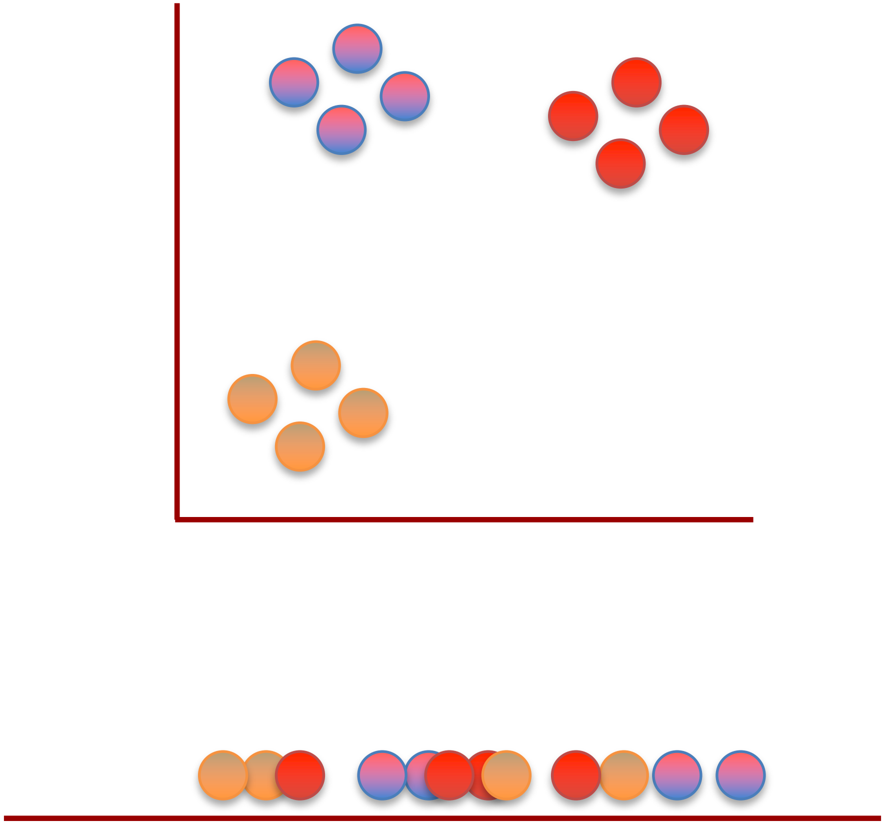


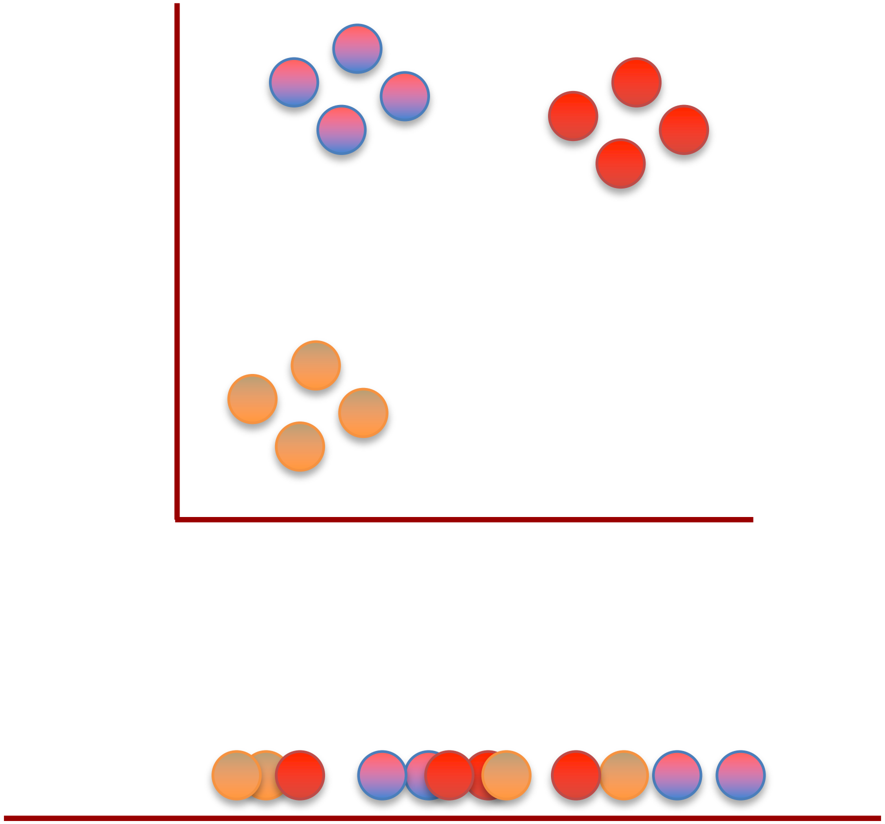


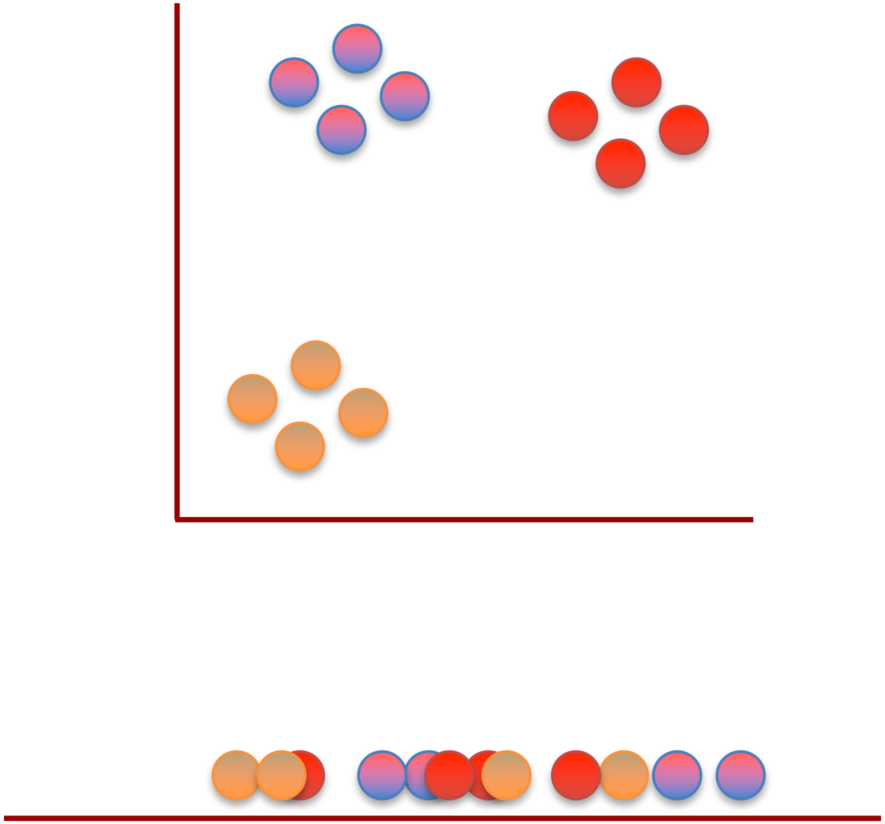
COSMIC DAWN CENTER

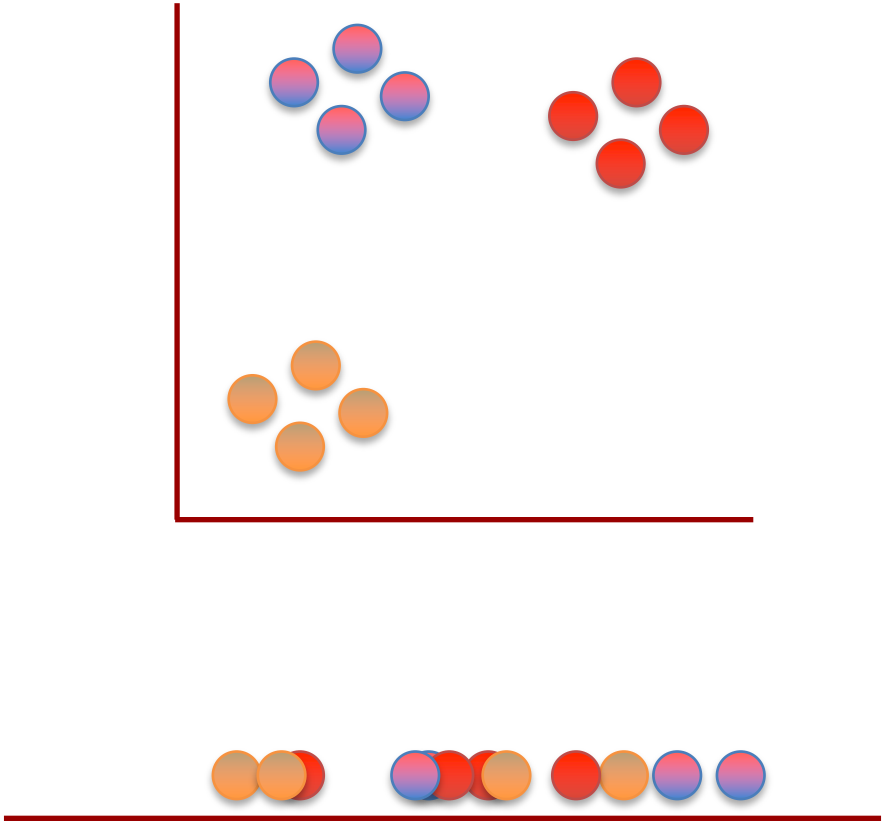
DAWN

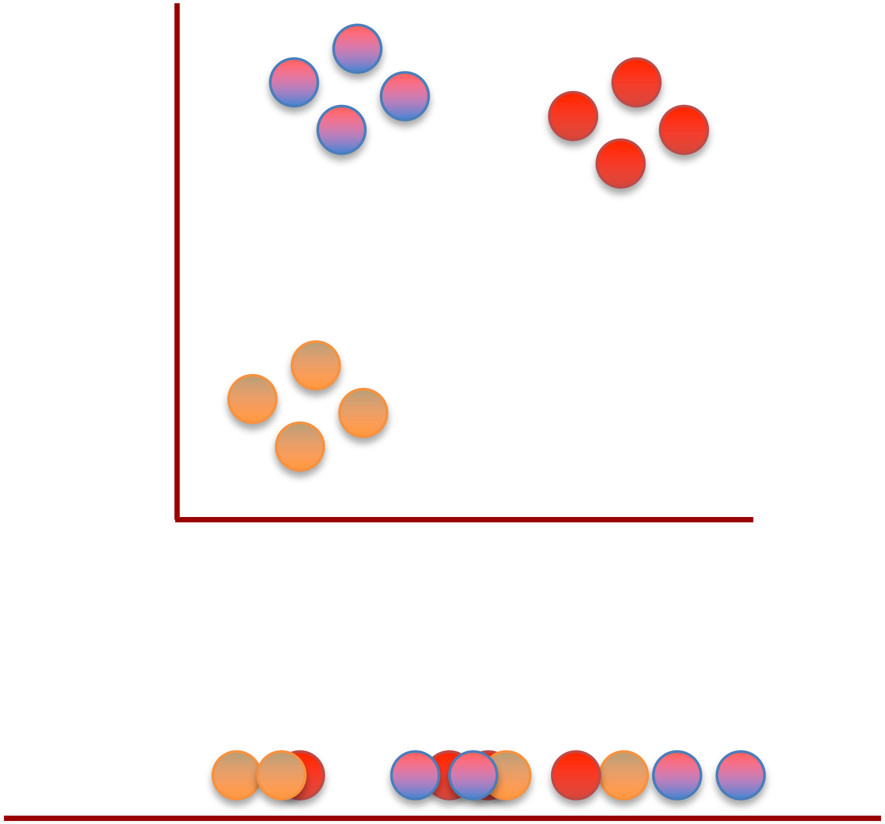


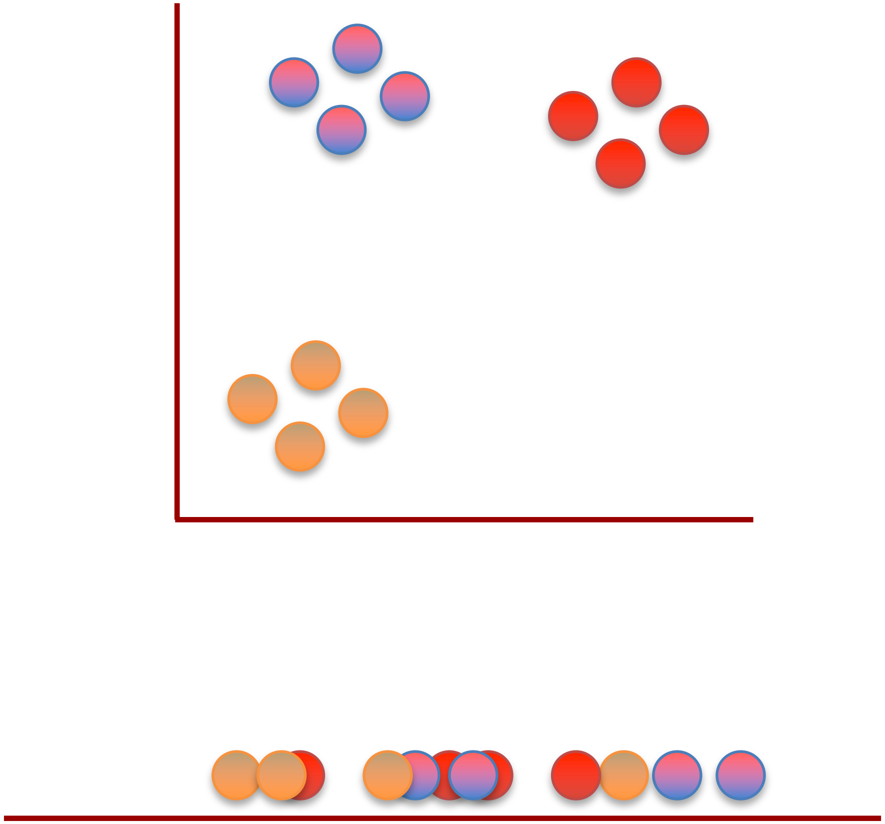


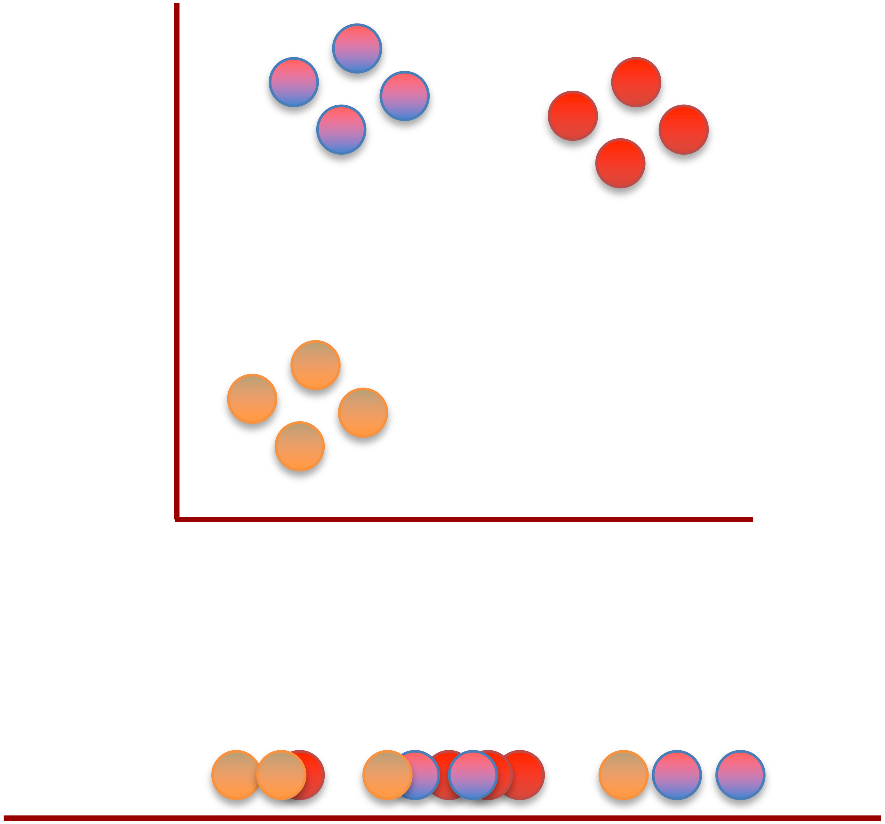


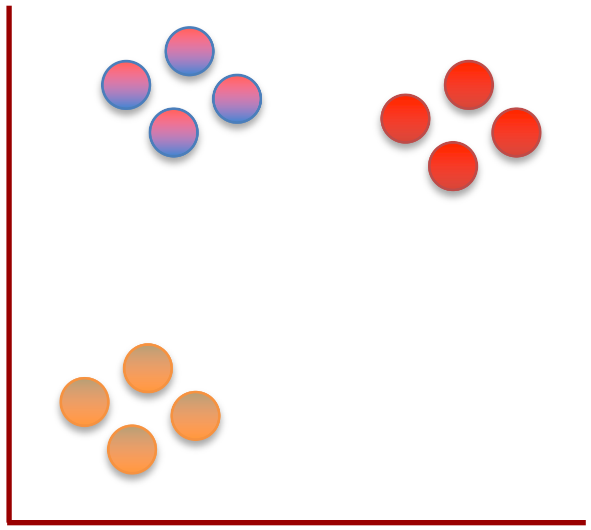


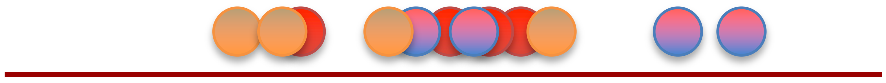






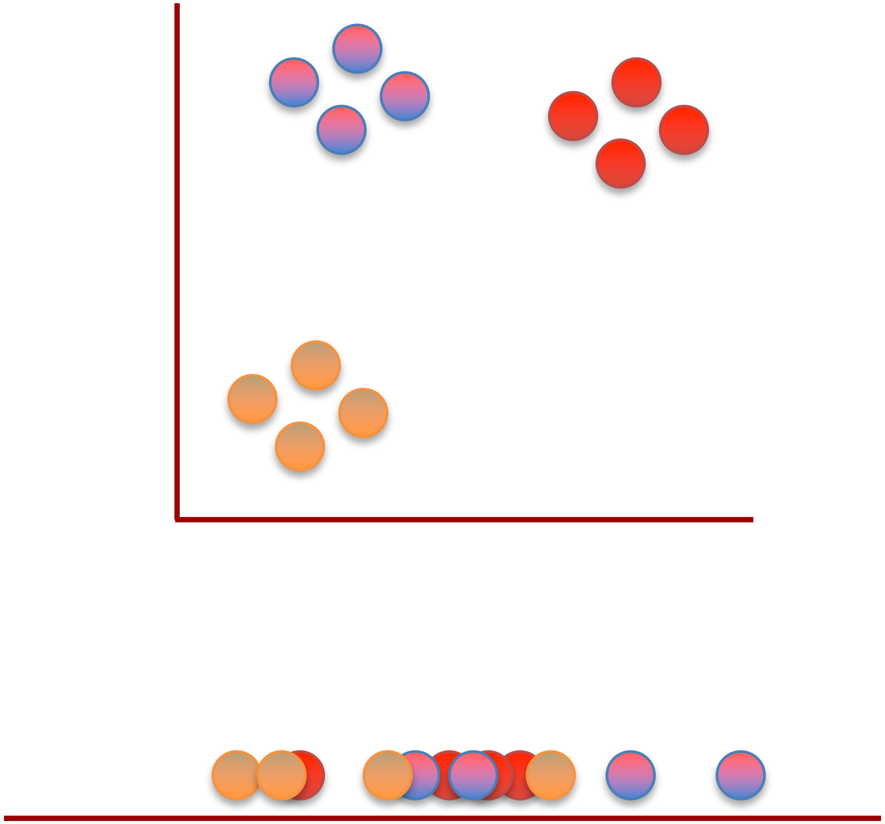


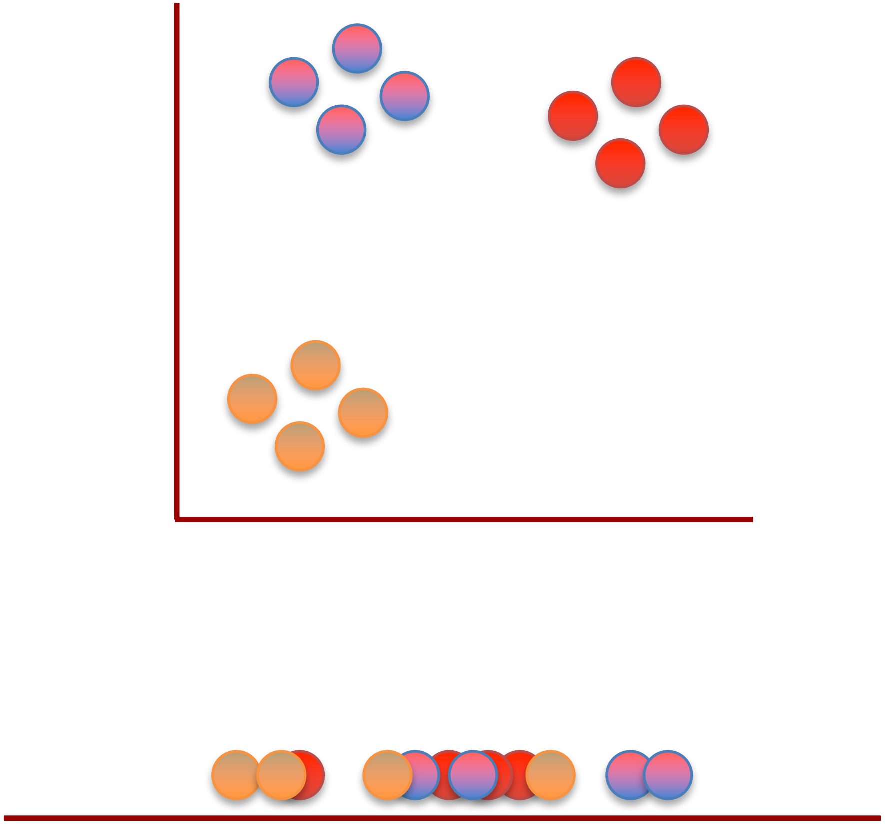


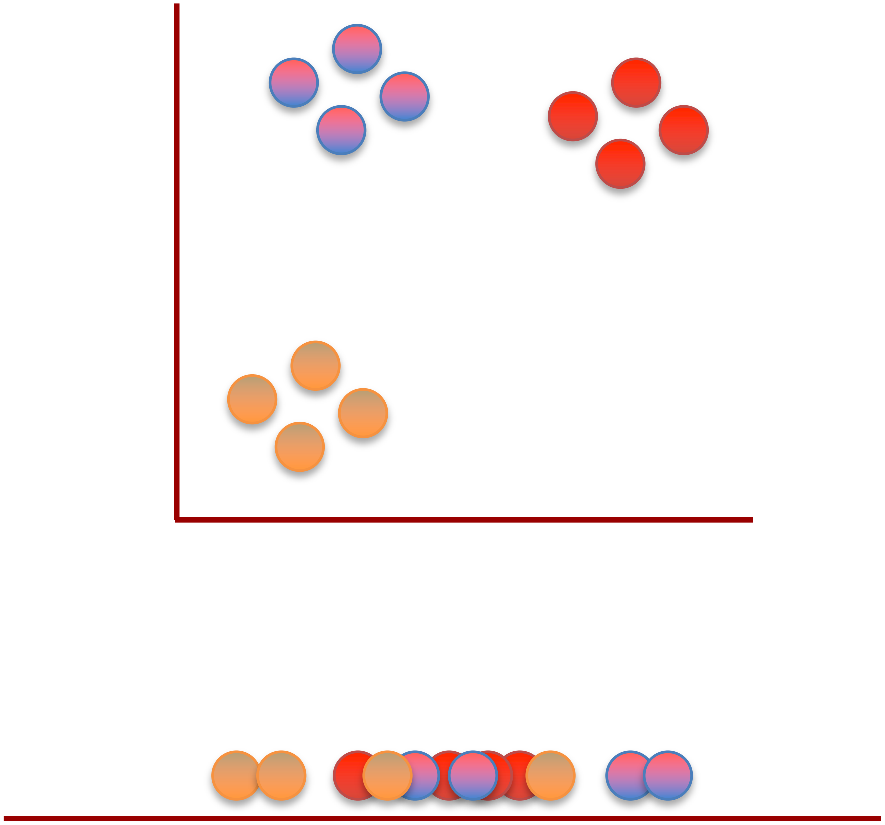


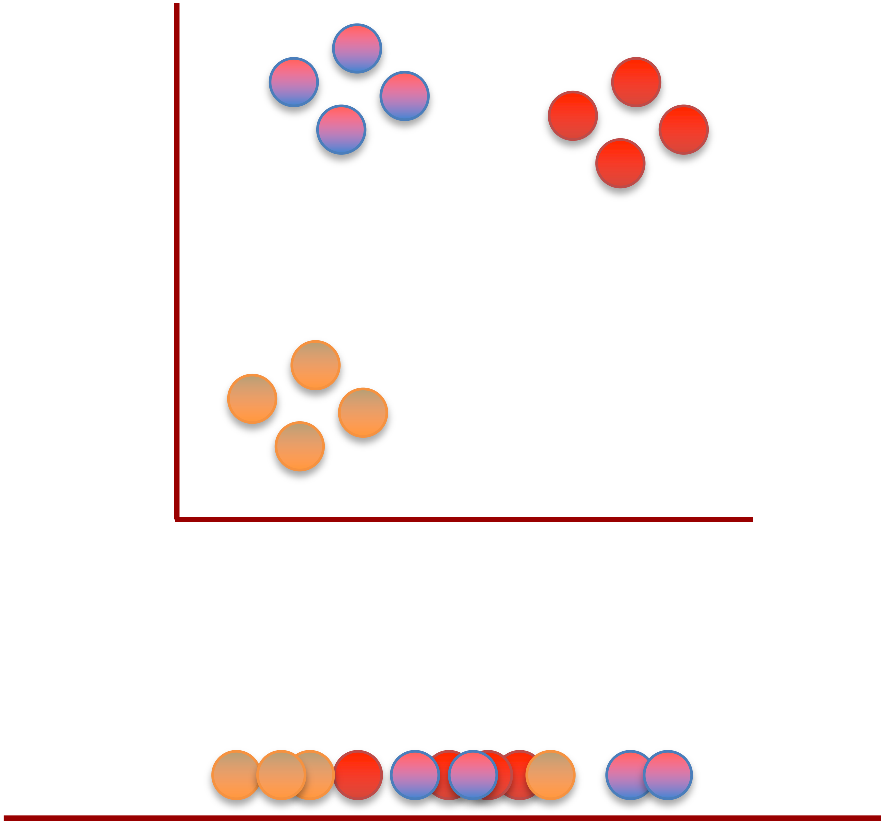
COSMIC DAWN CENTER

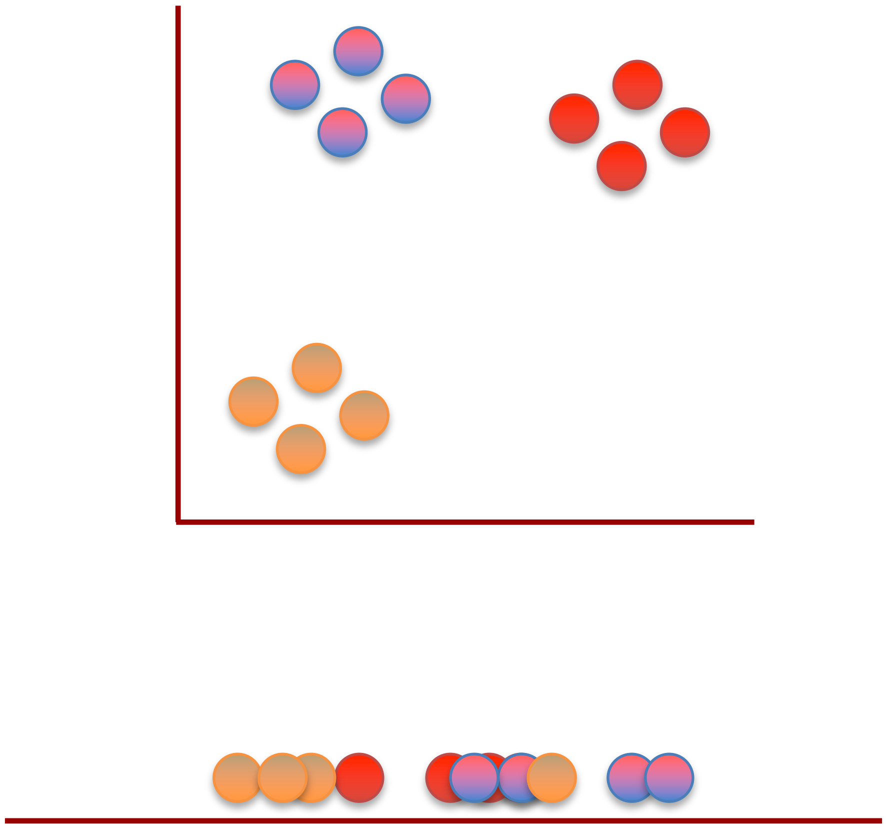
DAWN

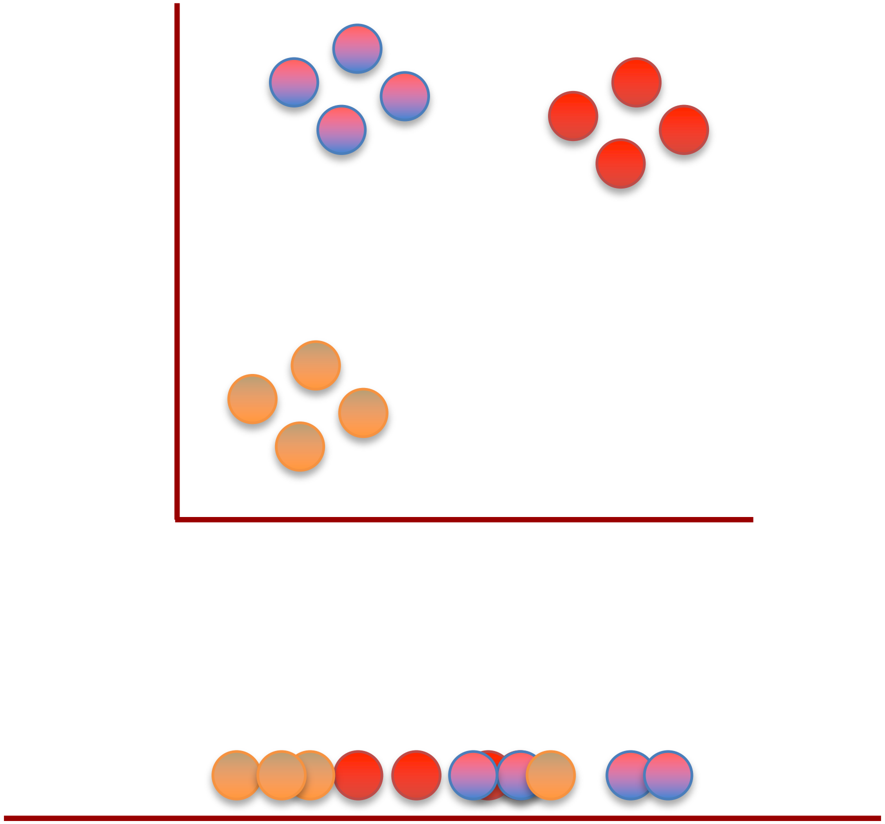


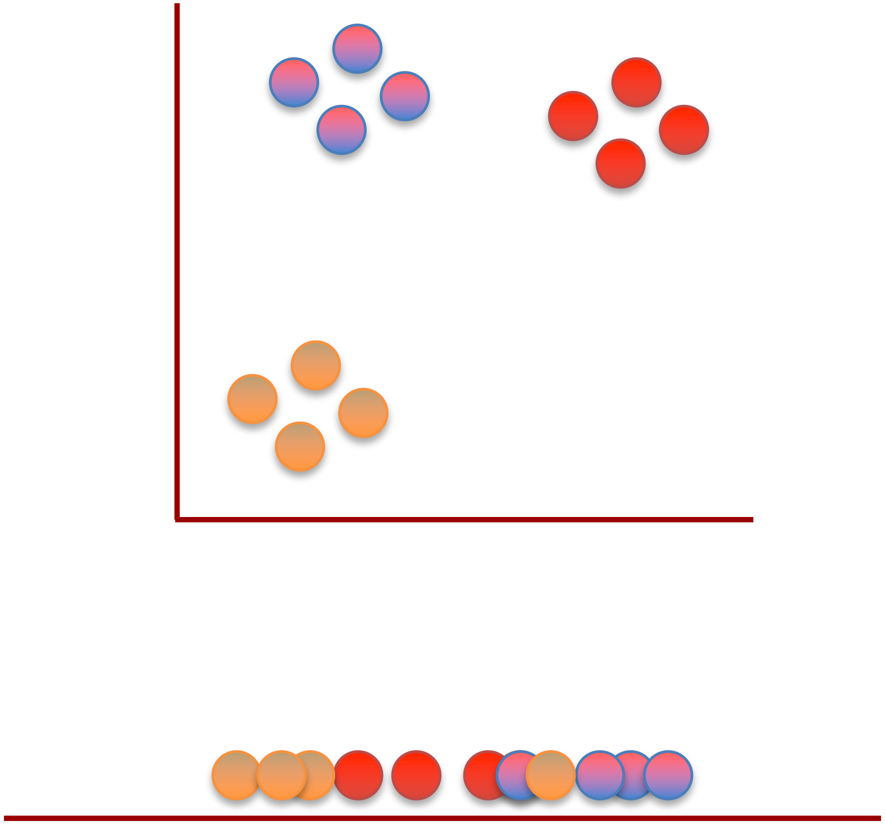


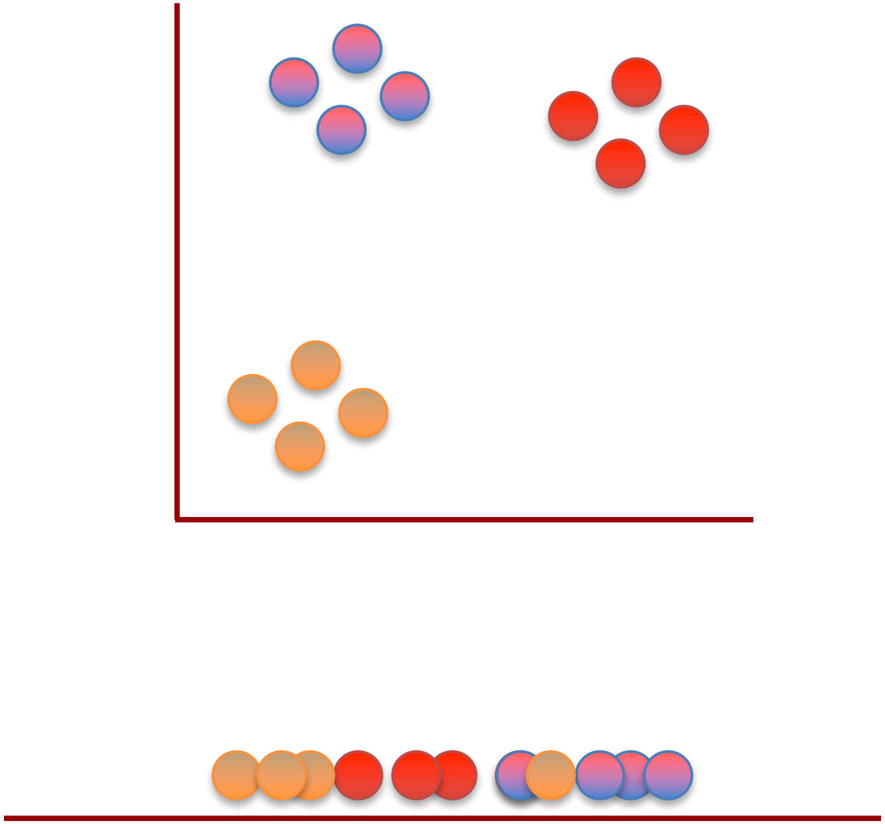






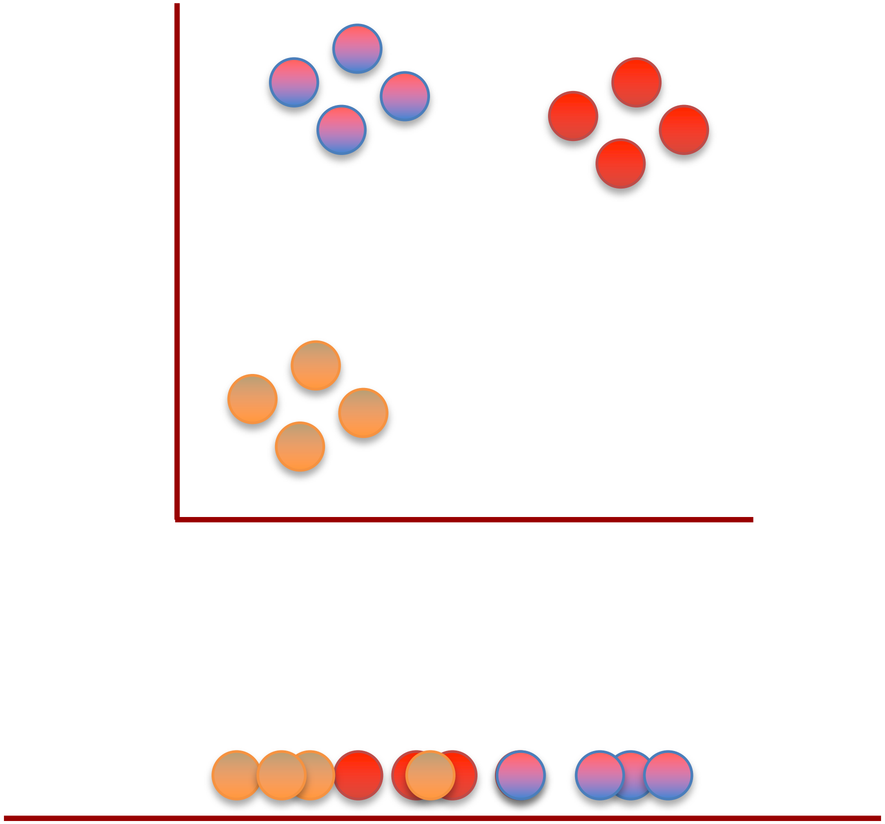






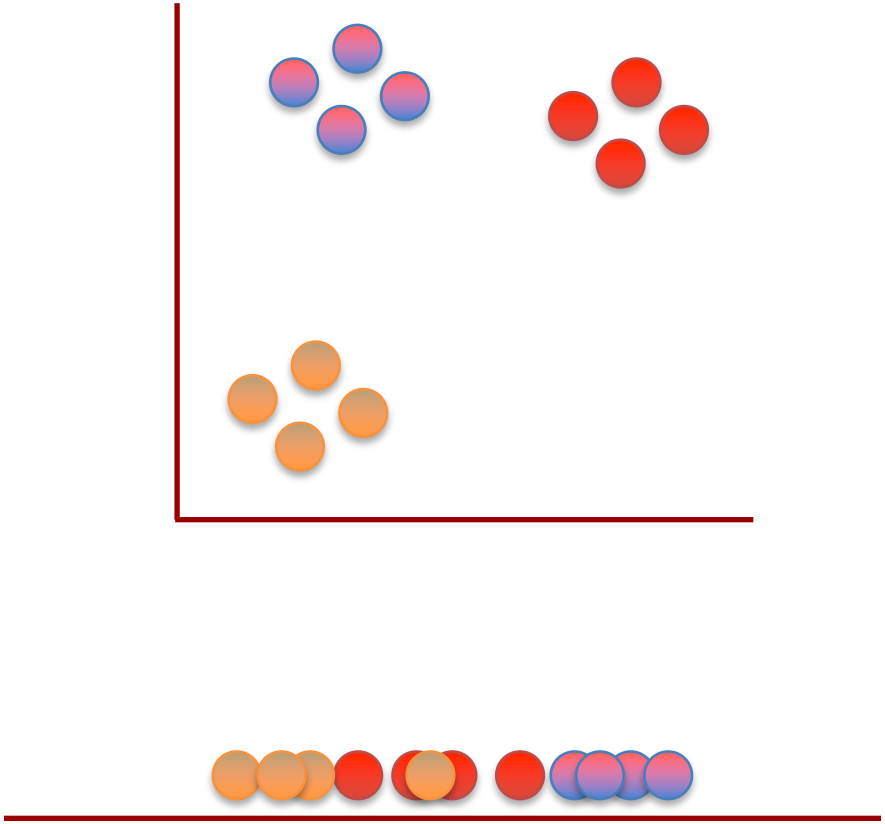
COSMIC DAWN CENTER

DAWN



COSMIC DAWN CENTER
DAWN

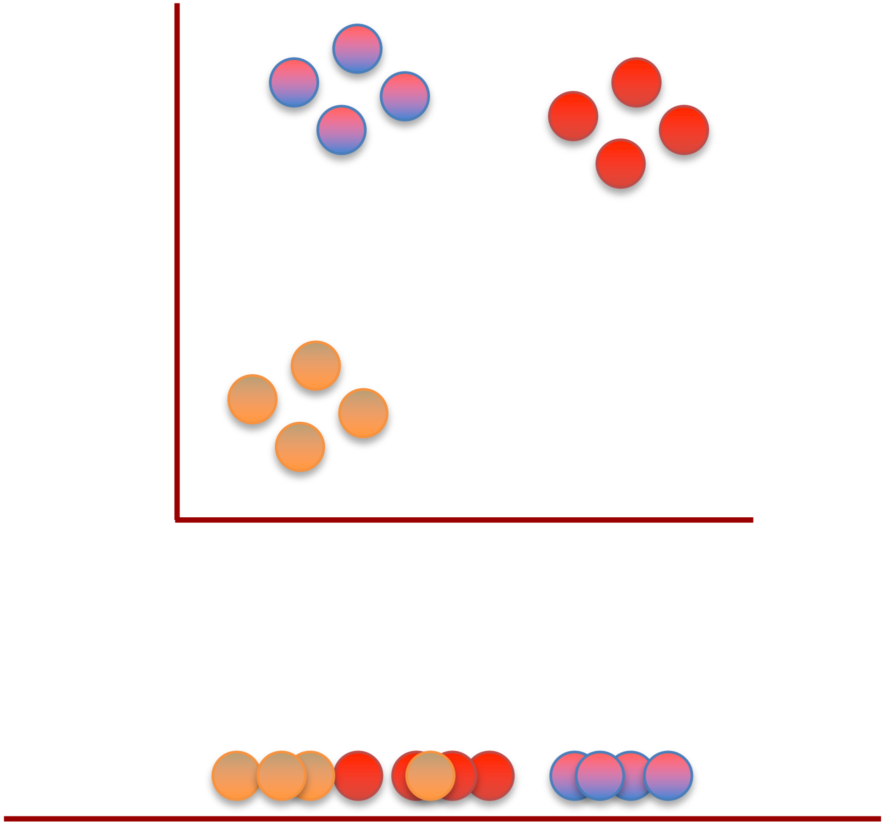
Wednesday May 3, 2023



COSMIC DAWN CENTER

DAWN

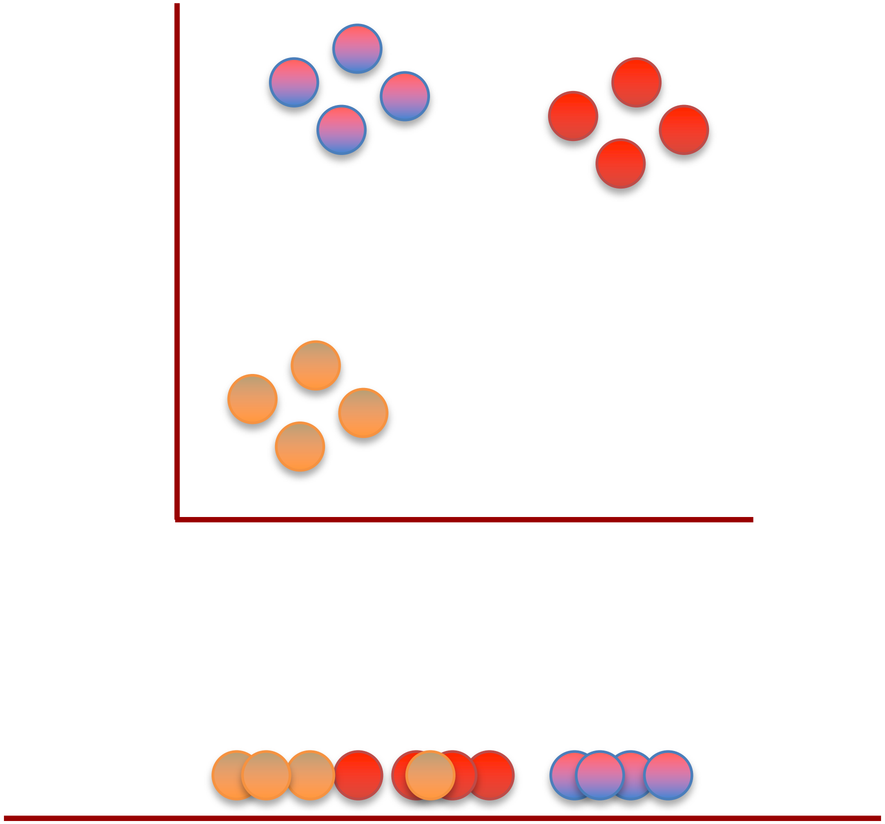
Wednesday May 3, 2023

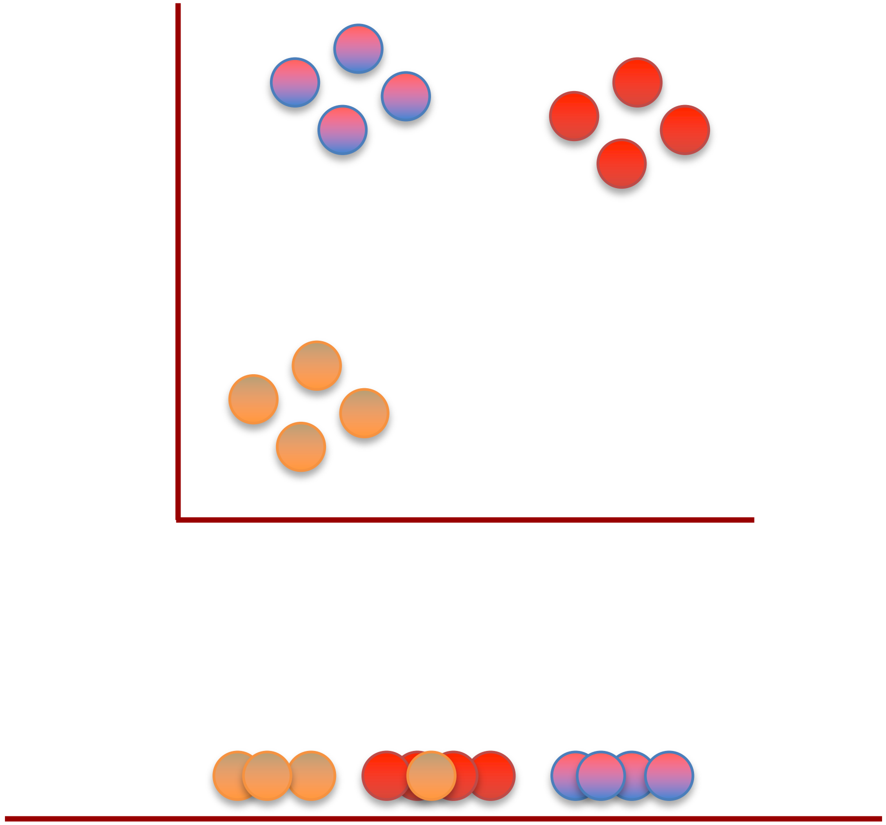


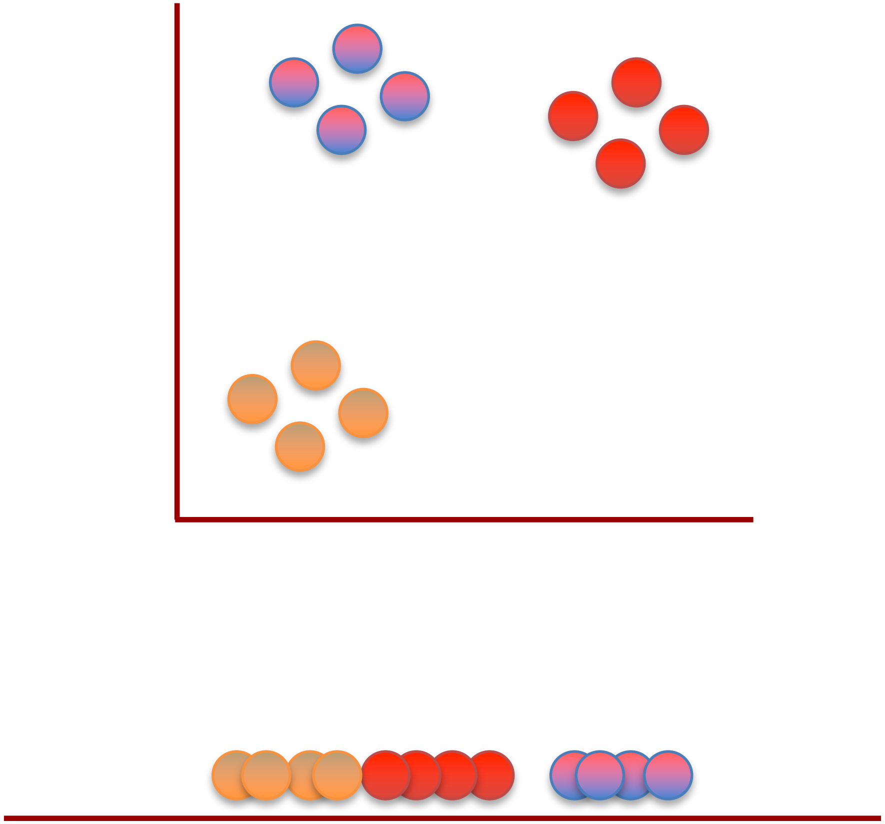
COSMIC DAWN CENTER

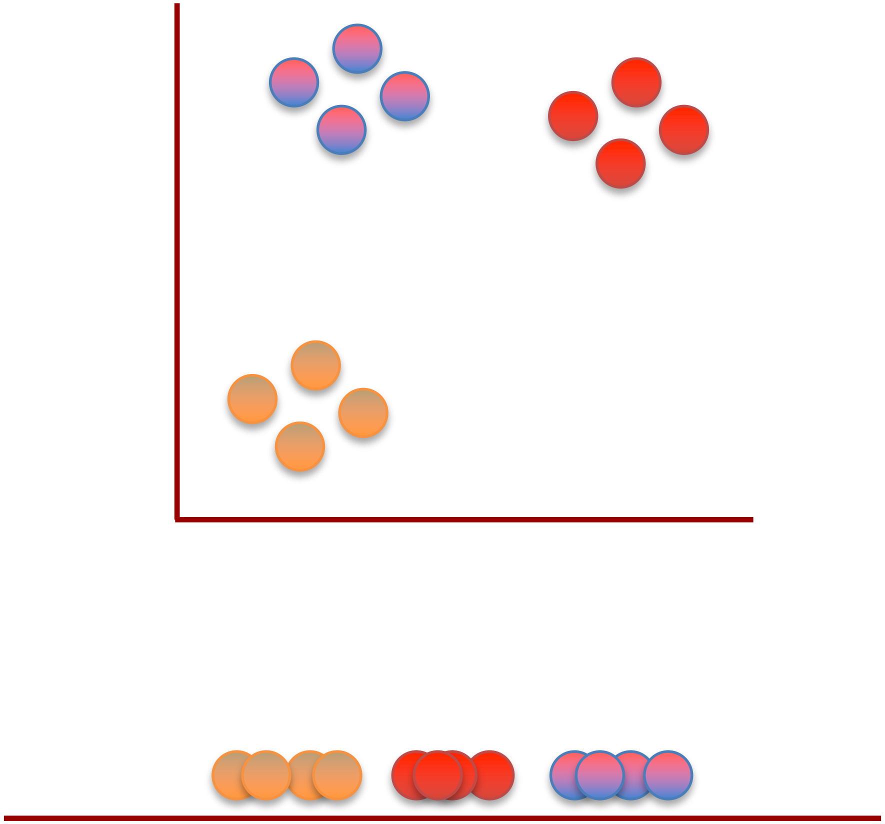
DAWN

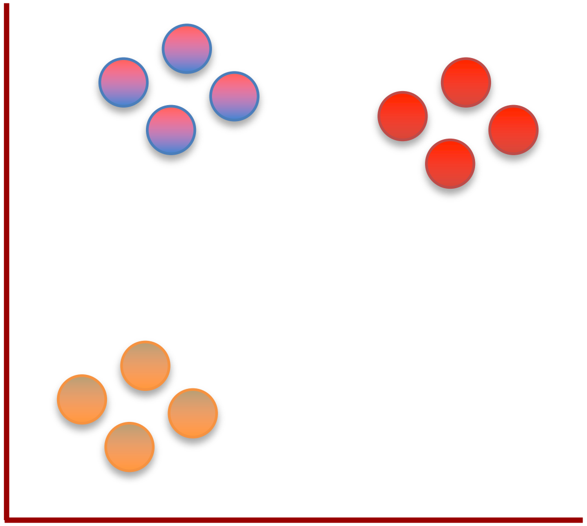
Wednesday May 3, 2023



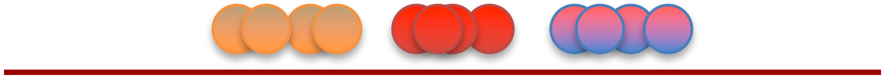




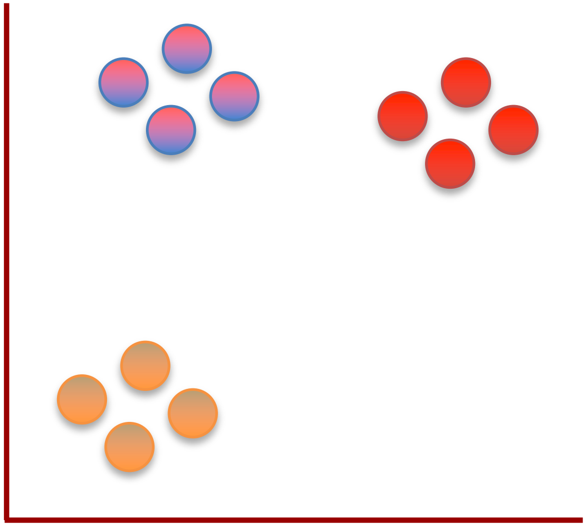


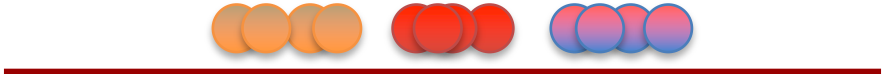


Triple BAM!!!!!

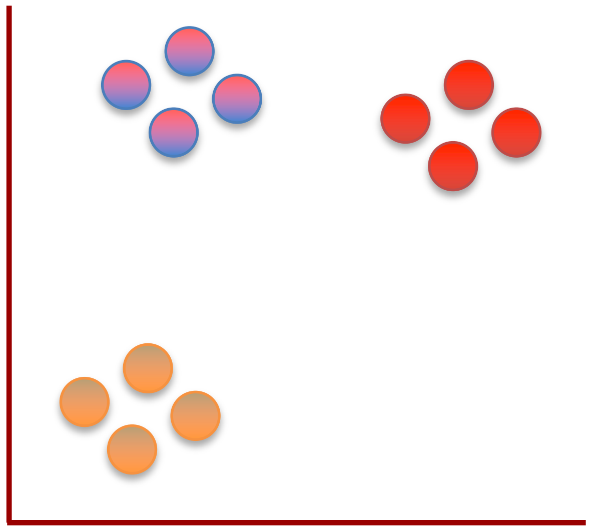


Now that we’ve seen the what t-SNE tries to do, let’s dive into the nitty-gritty details of how it does what it does.

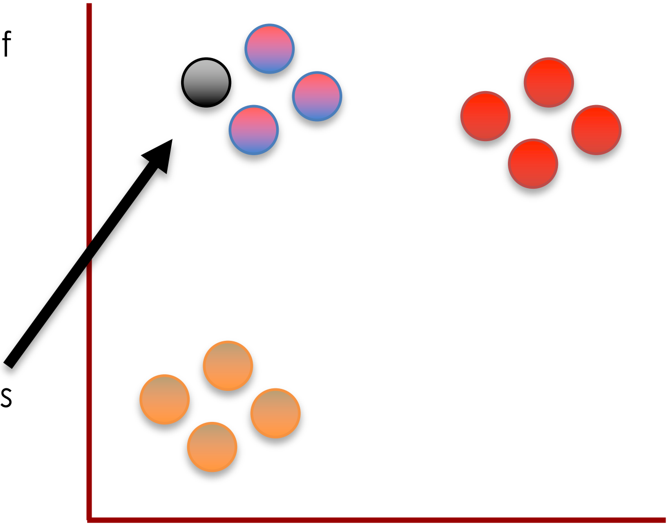




Step 1: Determine the "similarity" of all the points in the scatter plot.

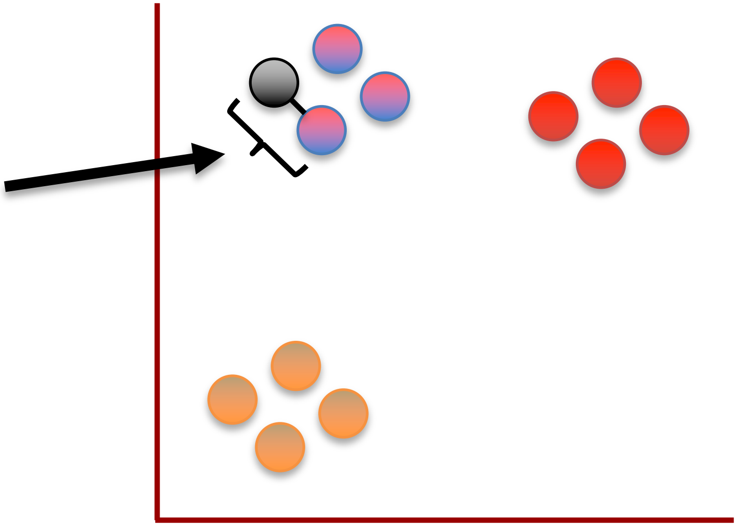


Step 1: Determine the "similarity" of all the points in the scatter plot.



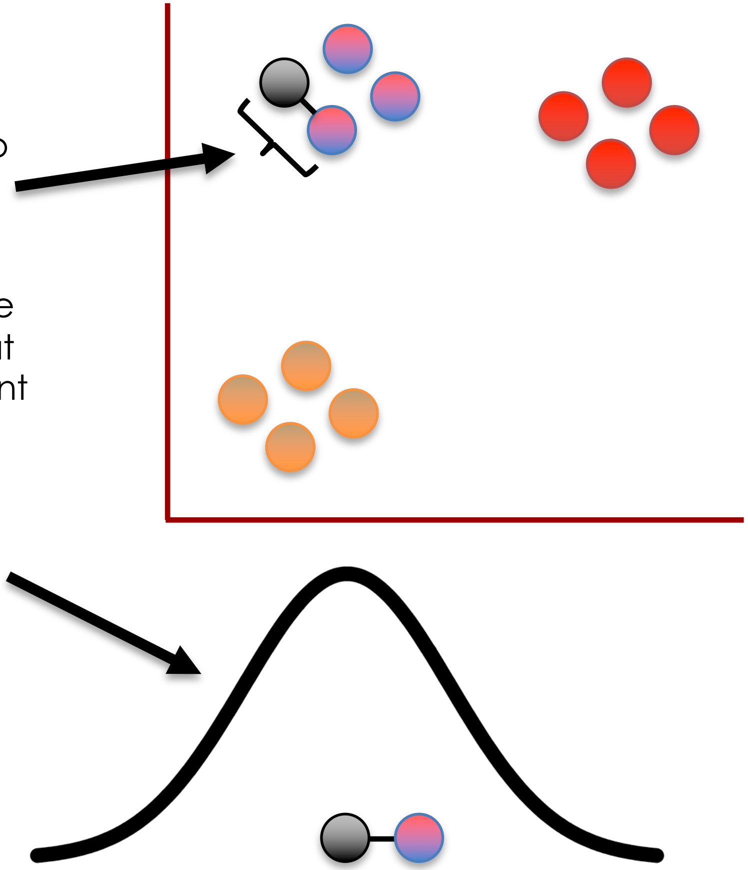
For this example, let's focus on determining the similarities between this point and all of the other points.

First, measure the distance between two points…



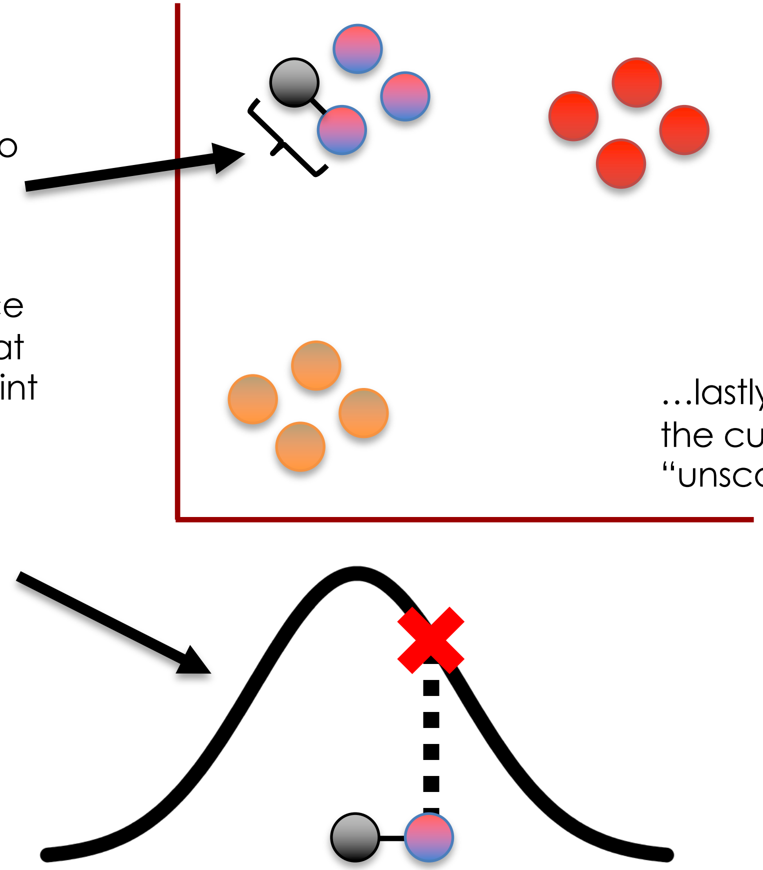
First, measure the distance between two points…

Then plot that distance on a normal curve that is centered on the point of interest…



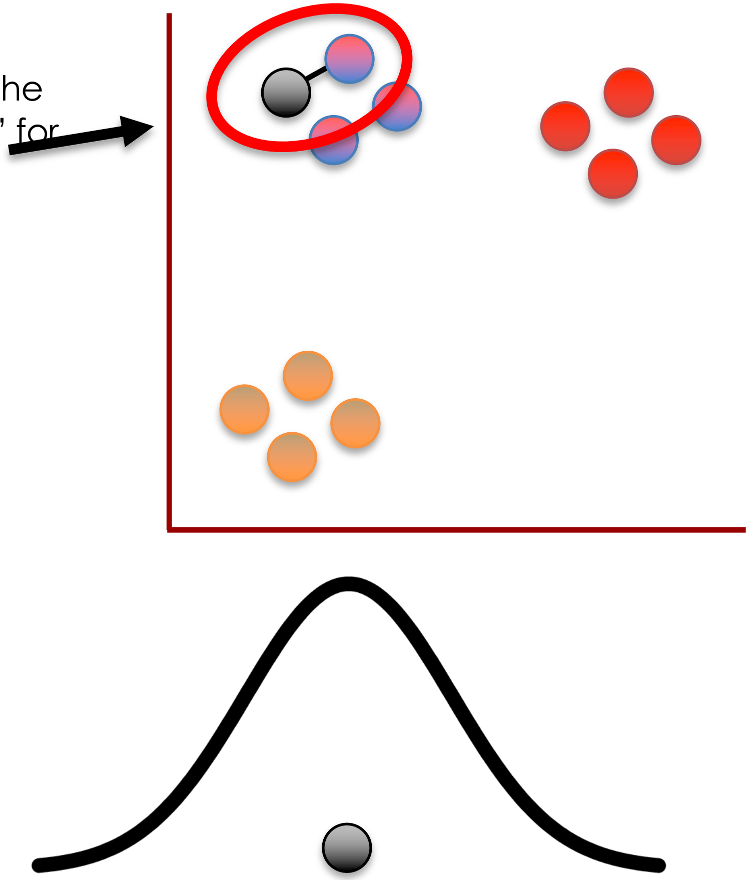
First, measure the distance between two points…

Then plot that distance on a normal curve that is centered on the point of interest…

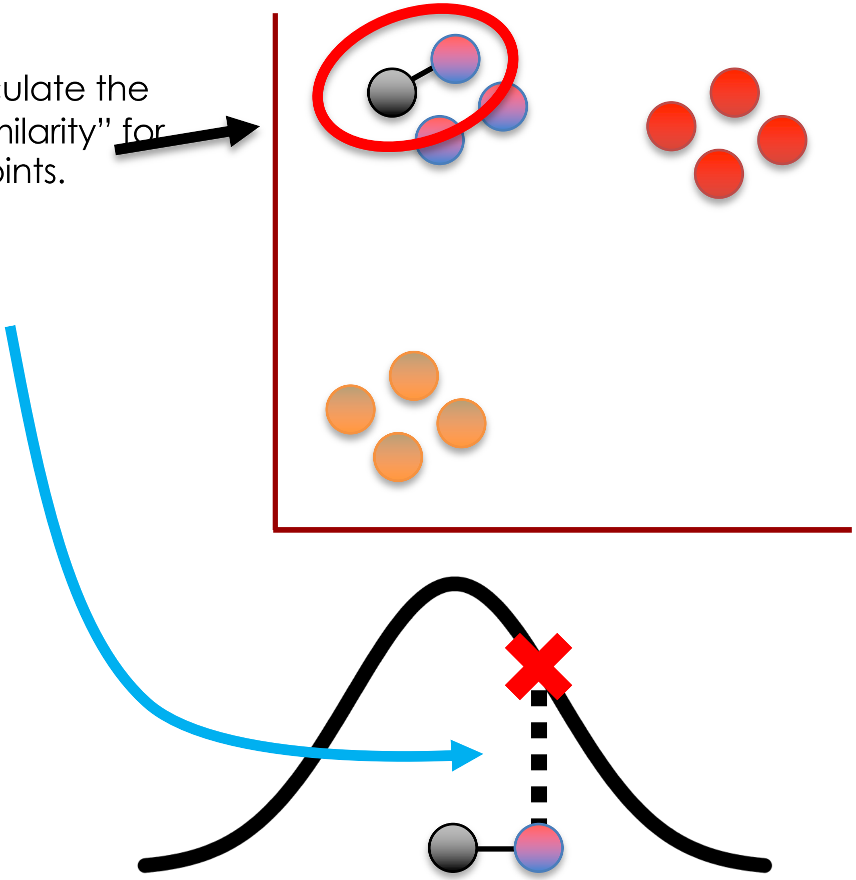


…lastly, draw a line from the point to the curve. The length of that line is the "unscaled similarity".

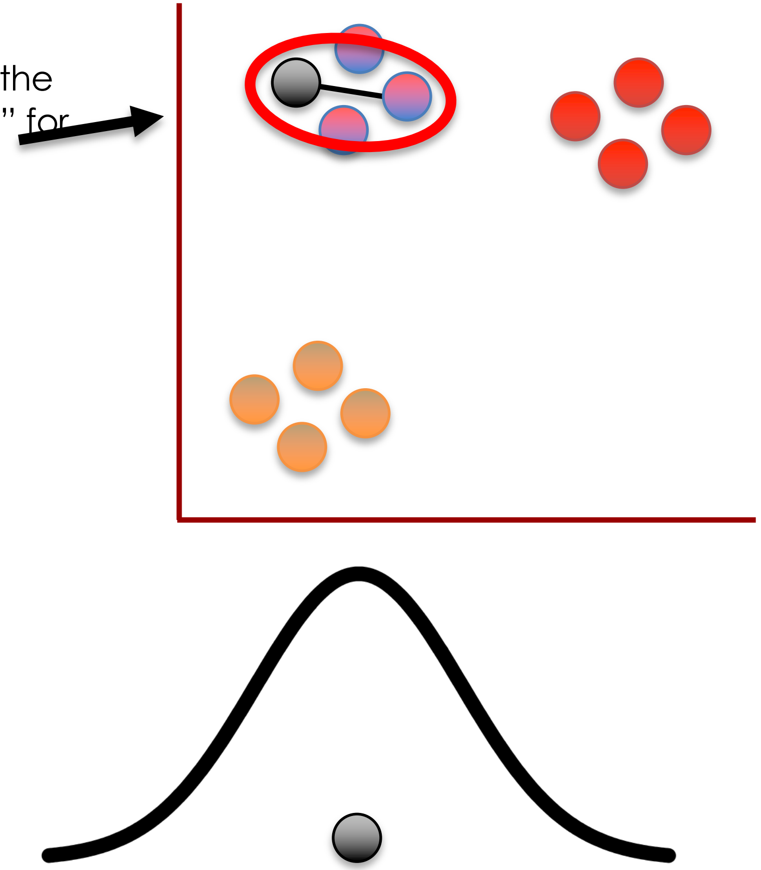
Now we calculate the “unscaled similarity” for this pair of points.



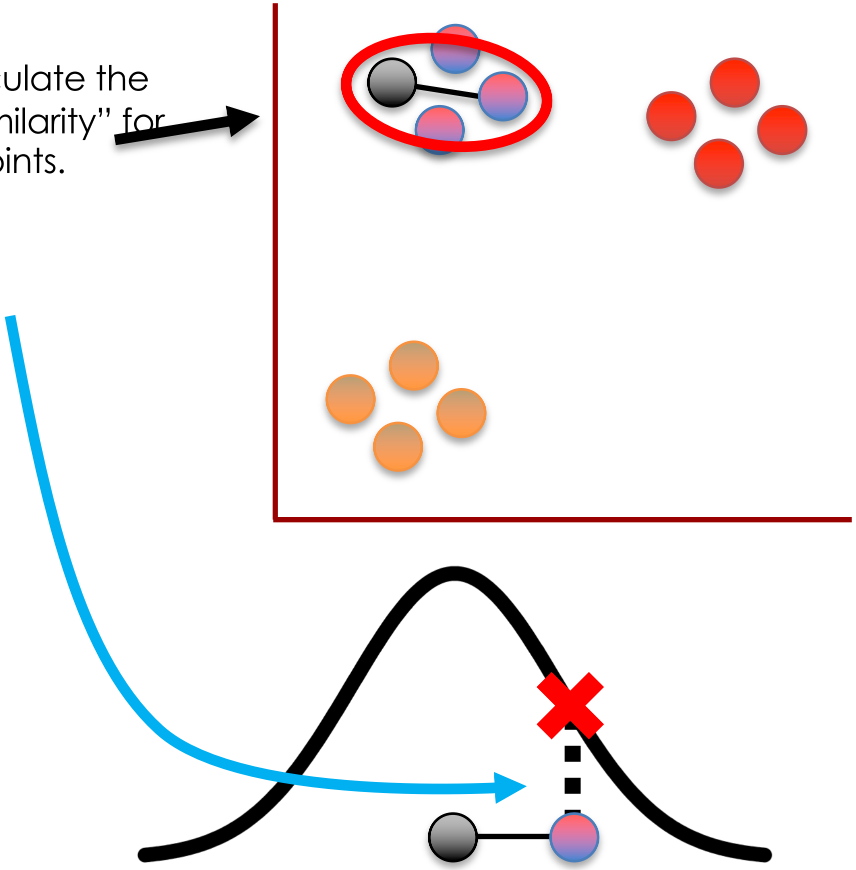
Now we calculate the "unscaled similarity" for this pair of points.



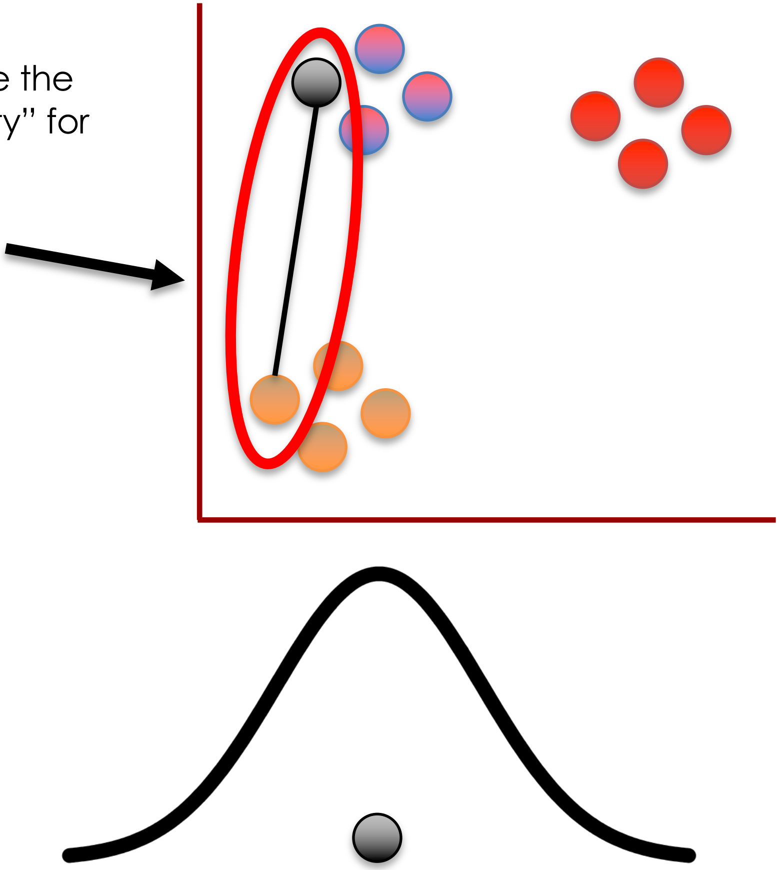
Now we calculate the "unscaled similarity" for this pair of points.



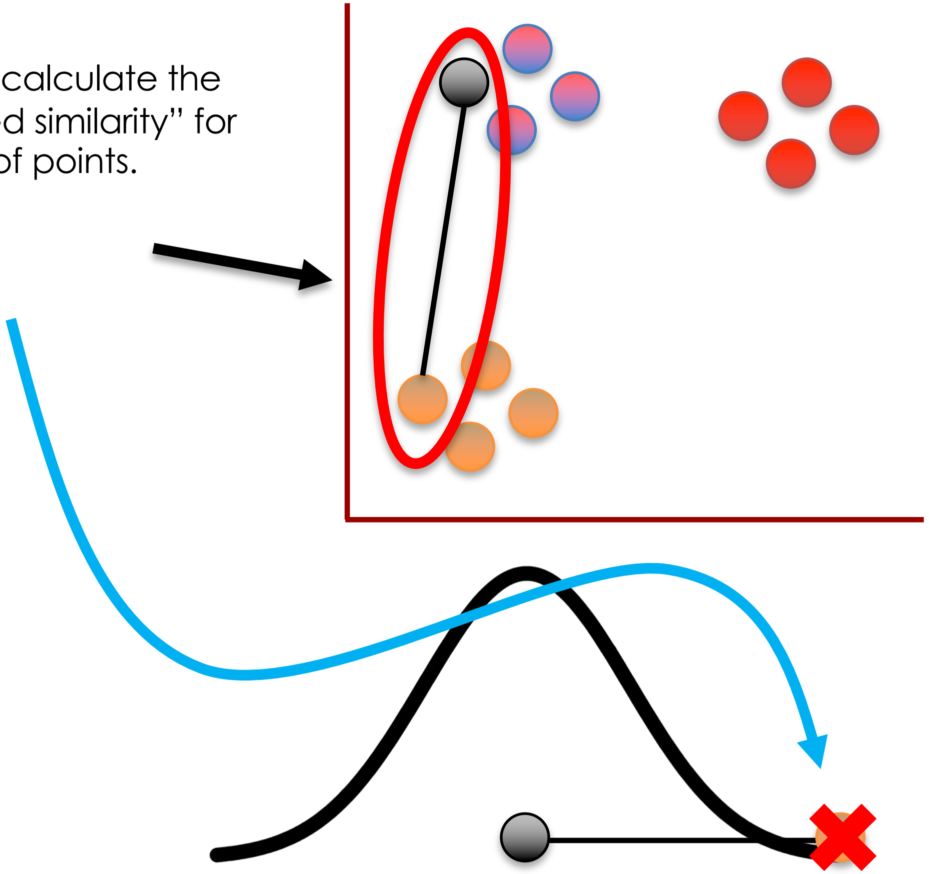
Now we calculate the “unscaled similarity” for this pair of points.



Now we calculate the “unscaled similarity” for this pair of points.



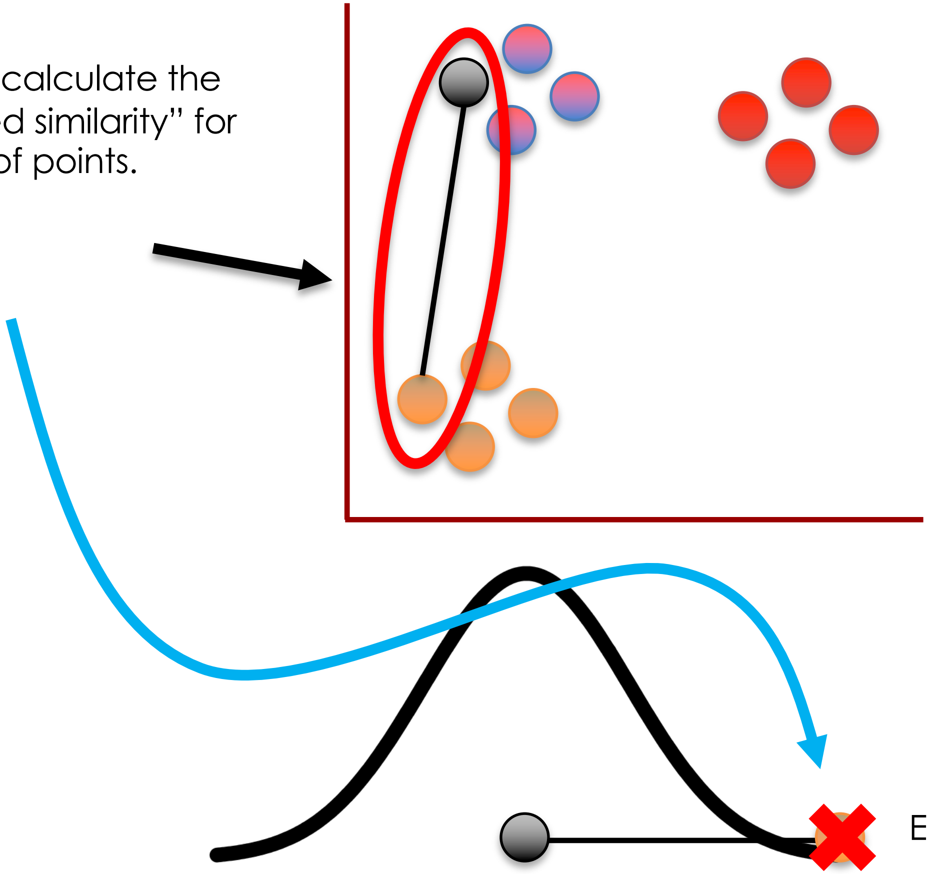
Now we calculate the "unscaled similarity" for this pair of points.



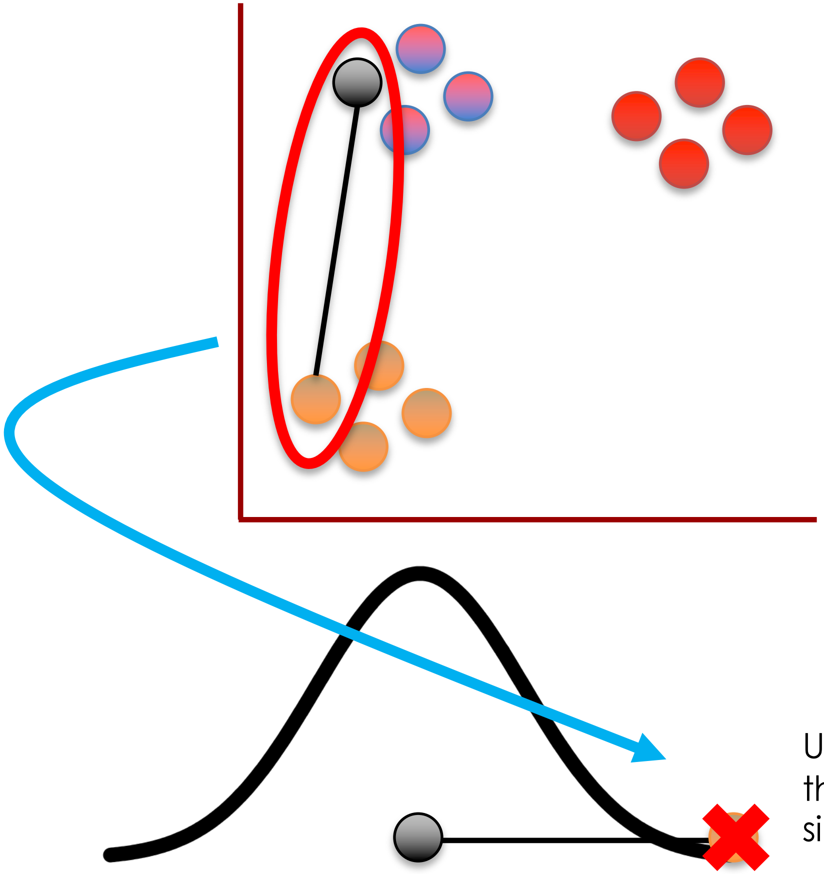
COSMIC DAWN CENTER

DAWN

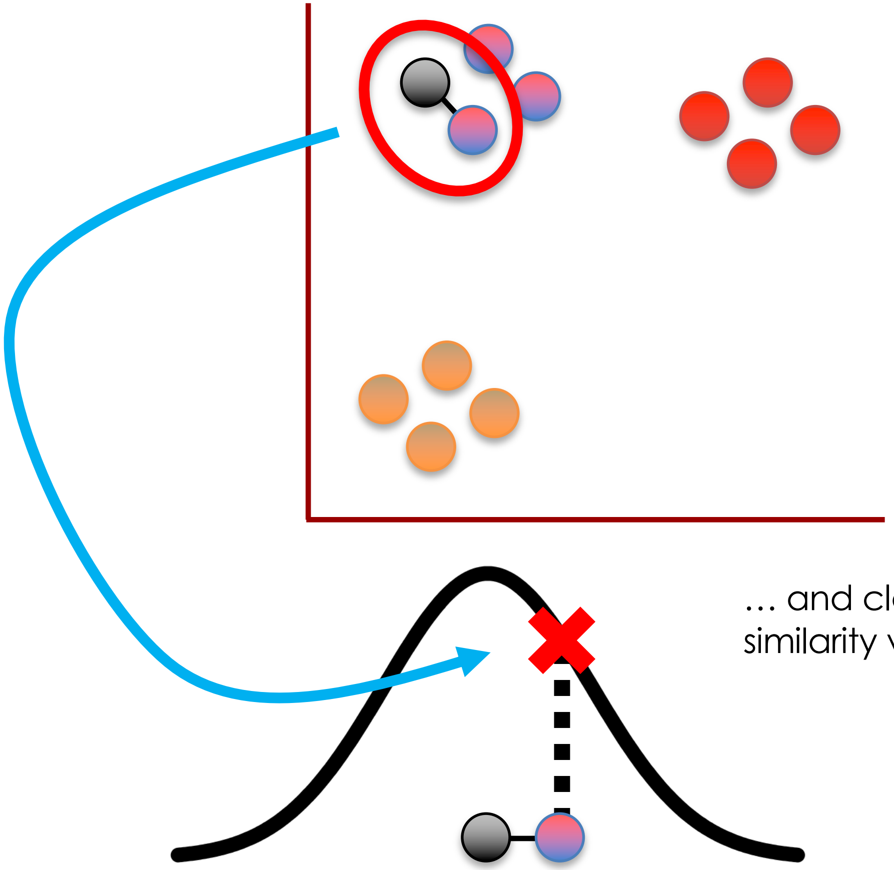
Now we calculate the "unscaled similarity" for this pair of points.



Etc. etc…

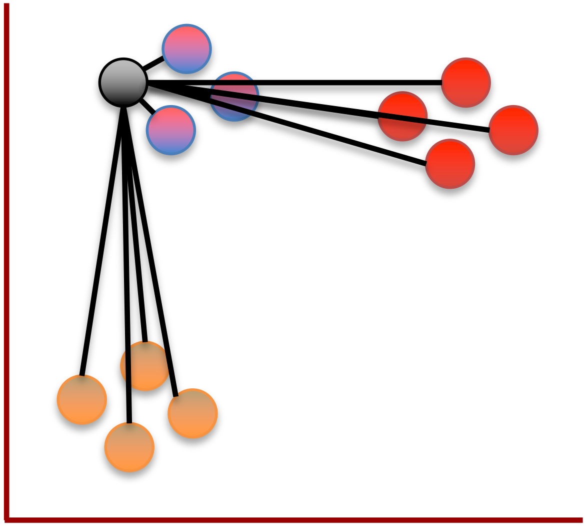


Using a normal distribution means that distant points have very low similarity values….

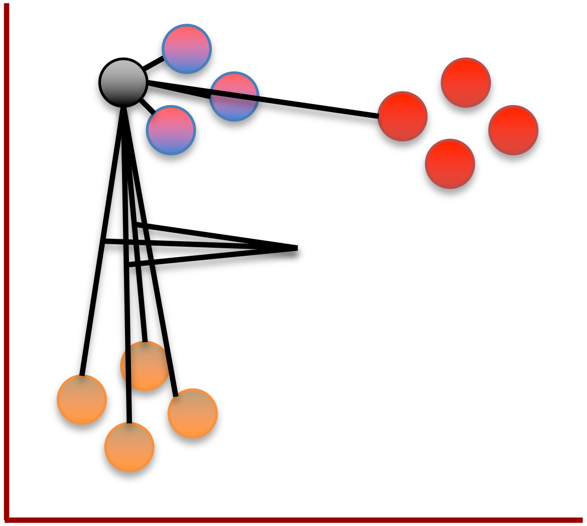


… and close points have high similarity values.

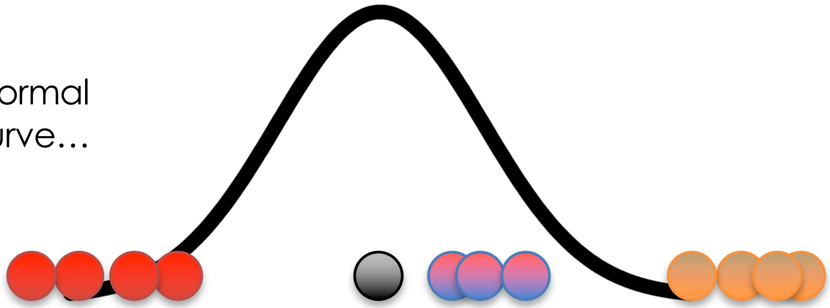
Ultimately, we measure the distances between all of the points and the point of interest…



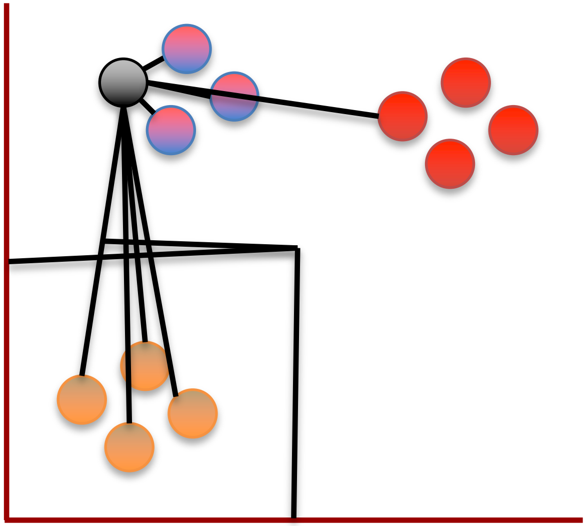
Ultimately, we measure the distances between all of the points and the point of interest…



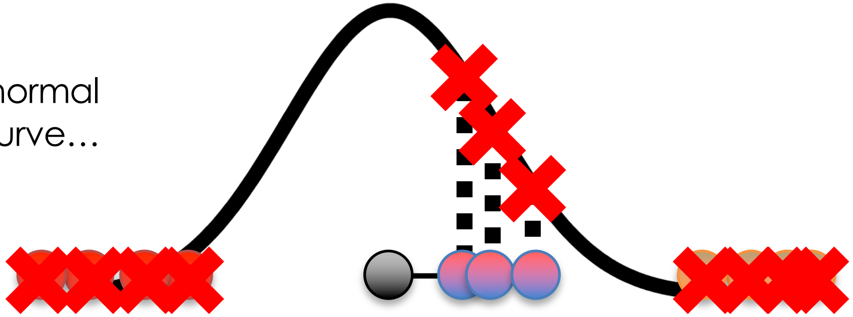
Plot them on the normal curve…



Ultimately, we measure the distances between all of the points and the point of interest…



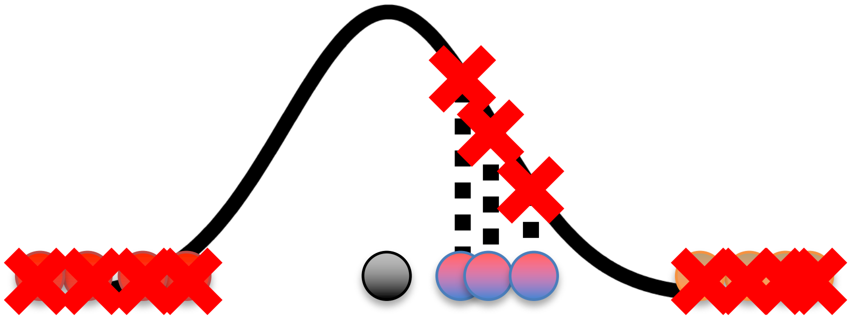
Plot them on the normal curve…



…and then measure the distances from the points to the curve to get the unscaled similarity scores with respect to the point of interest.

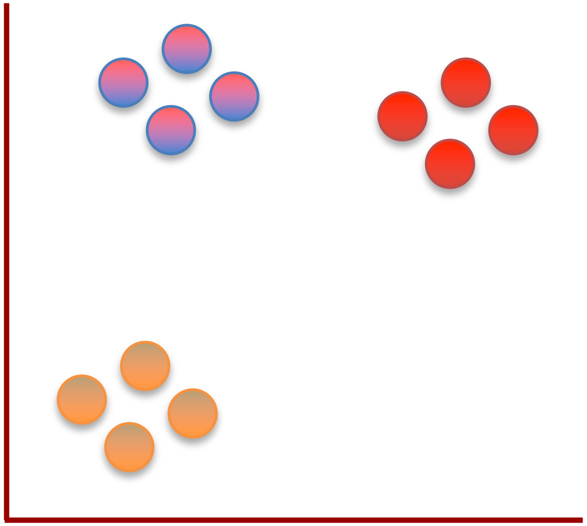
The next step is to scale the unscaled similarities so that they add up to 1.

Umm… Why do the similarity scores need to add up to 1?

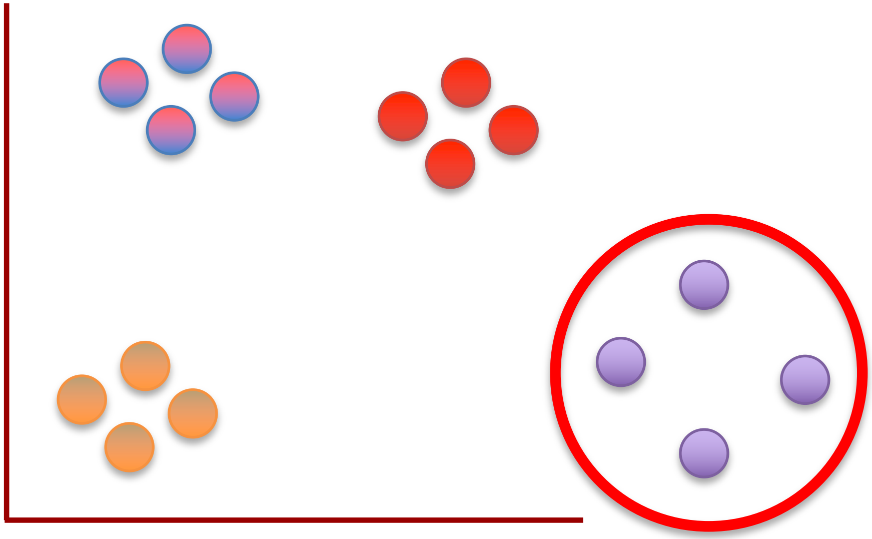


COSMIC DAWN CENTER DAWN

It has to do with something I didn't tell you earlier…

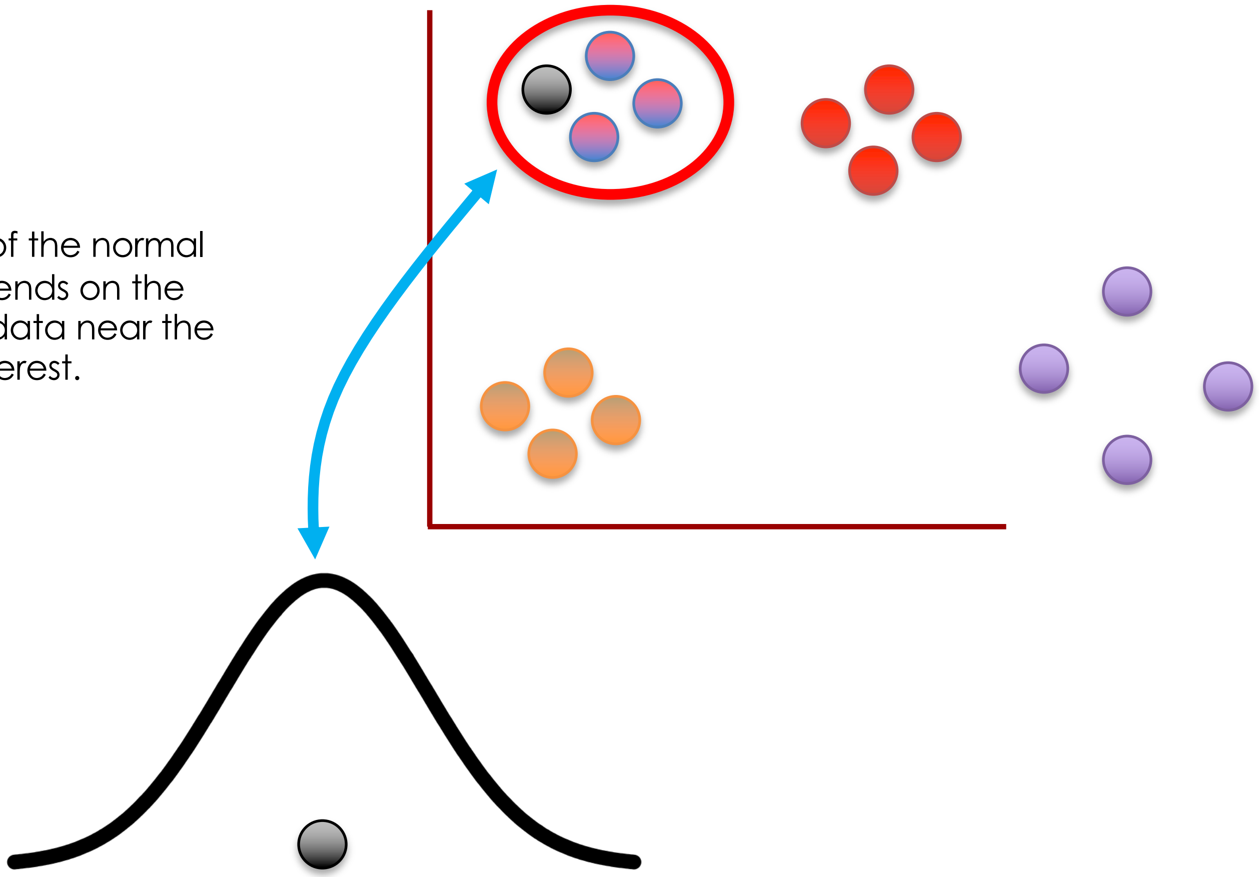


It has to do with something I didn't tell you earlier…

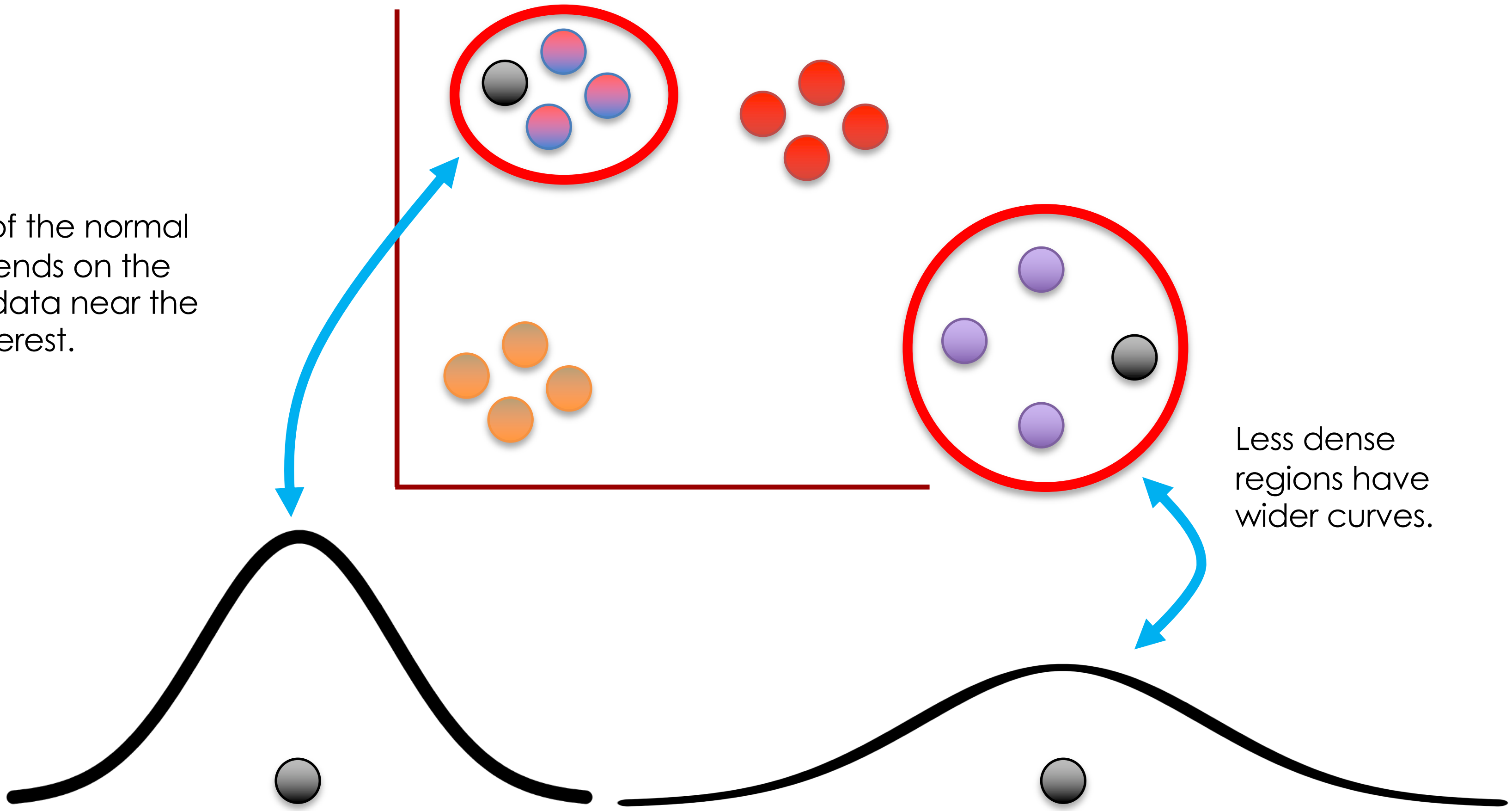


…and to illustrate the concept, I need to add a cluster that is half as dense as the others.

The width of the normal curve depends on the density of data near the point of interest.

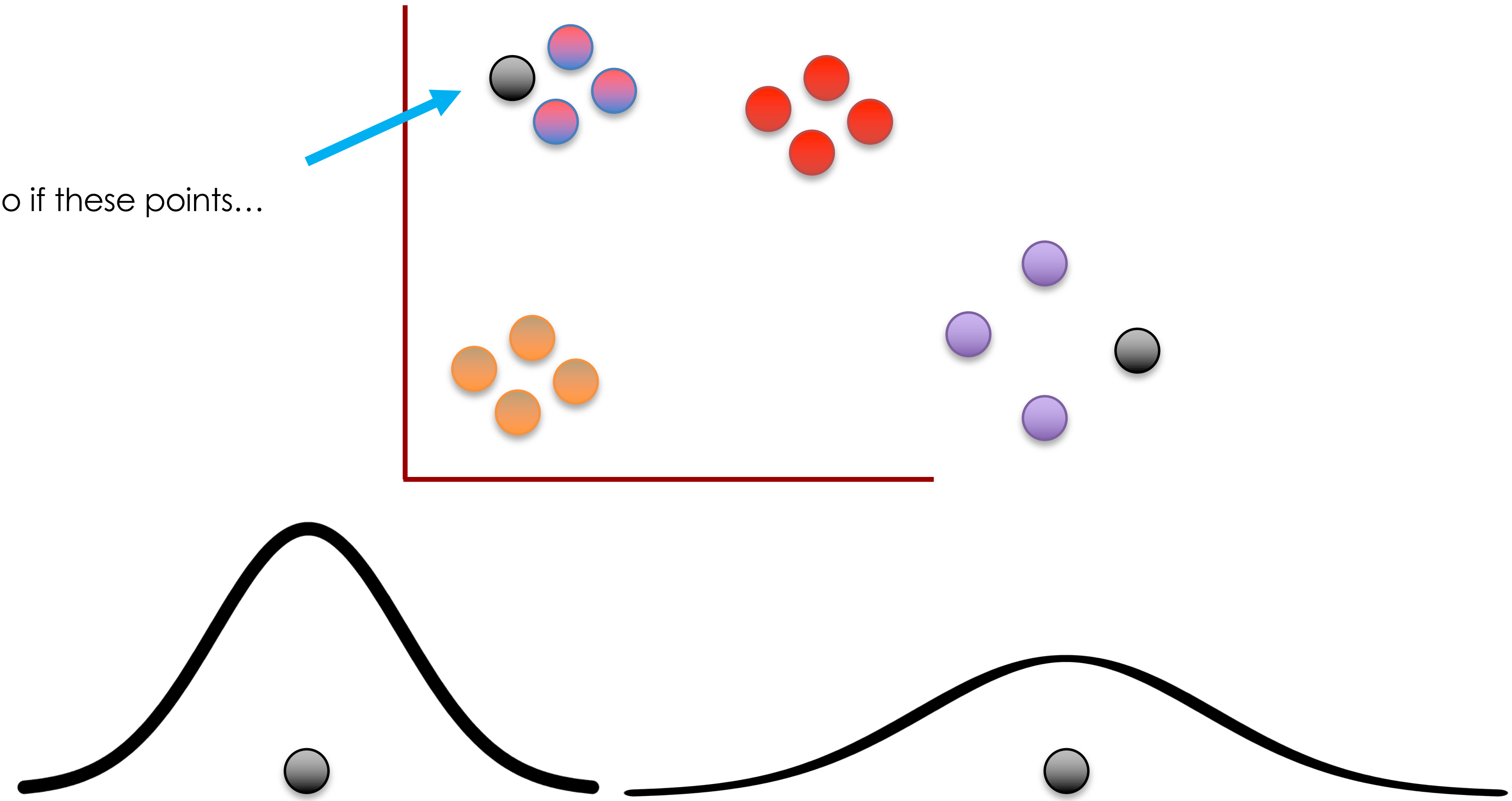


The width of the normal curve depends on the density of data near the point of interest.



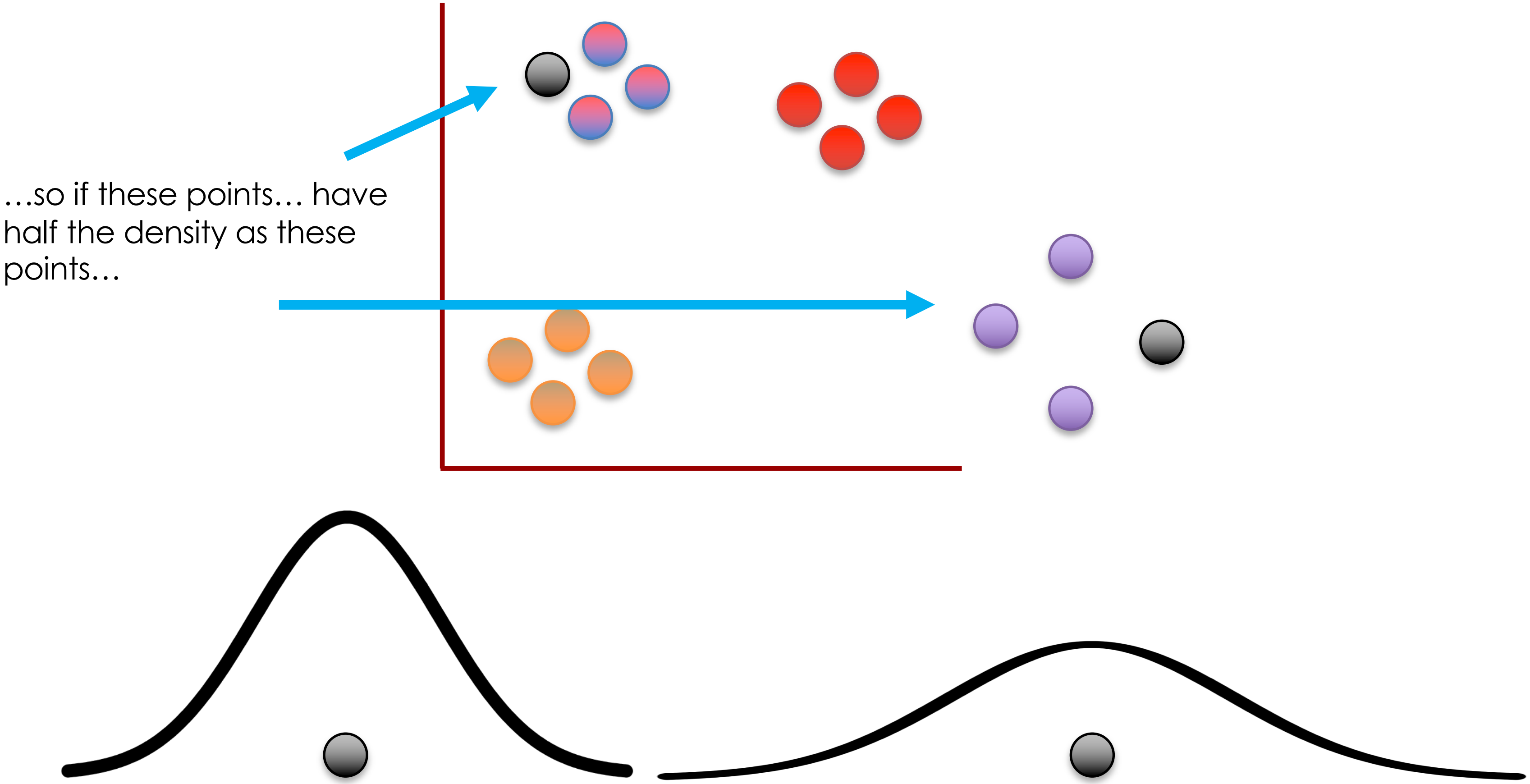
Less dense regions have wider curves.

…so if these points…

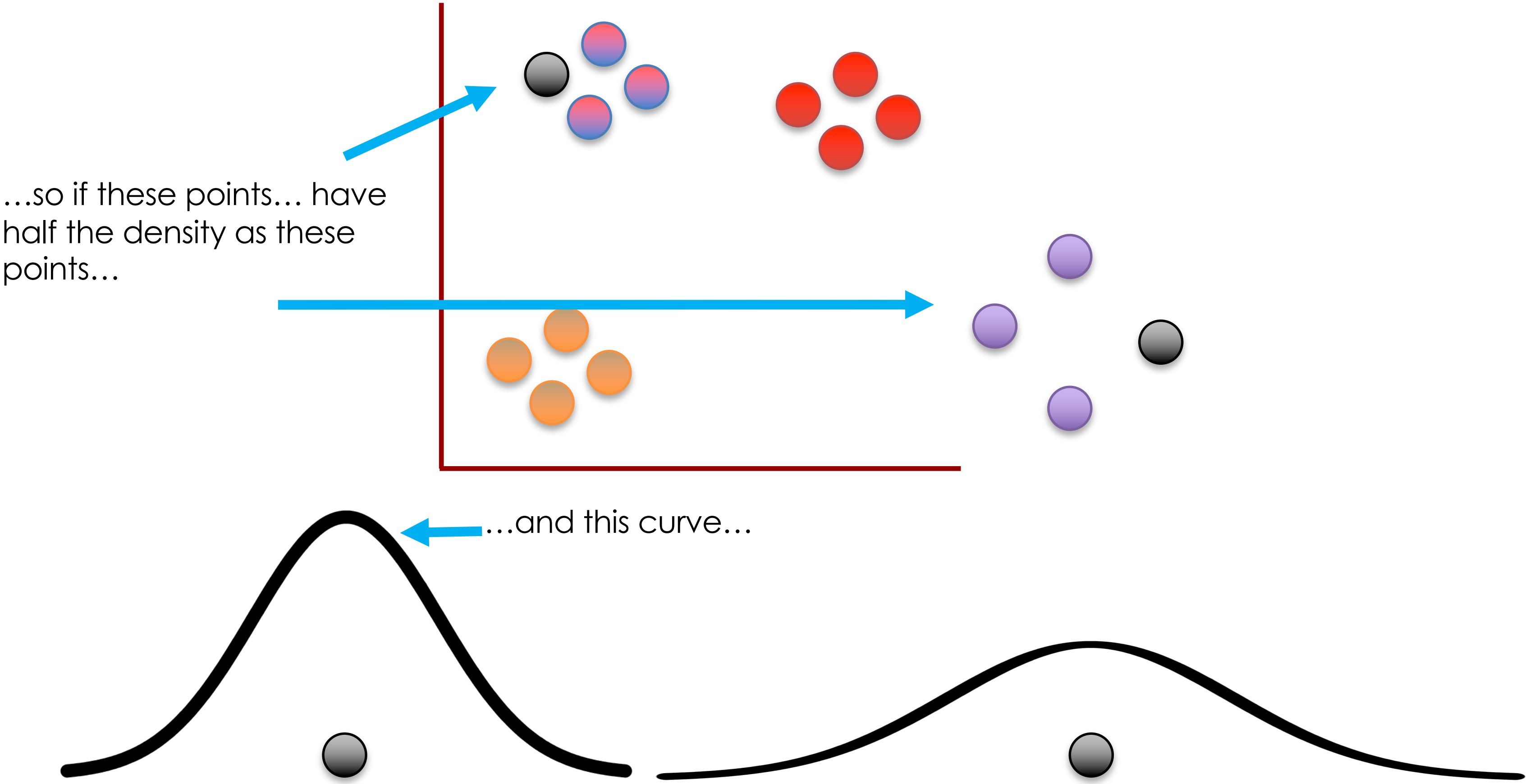


COSMIC DAWN CENTER
DAWN

Wednesday May 3, 2023

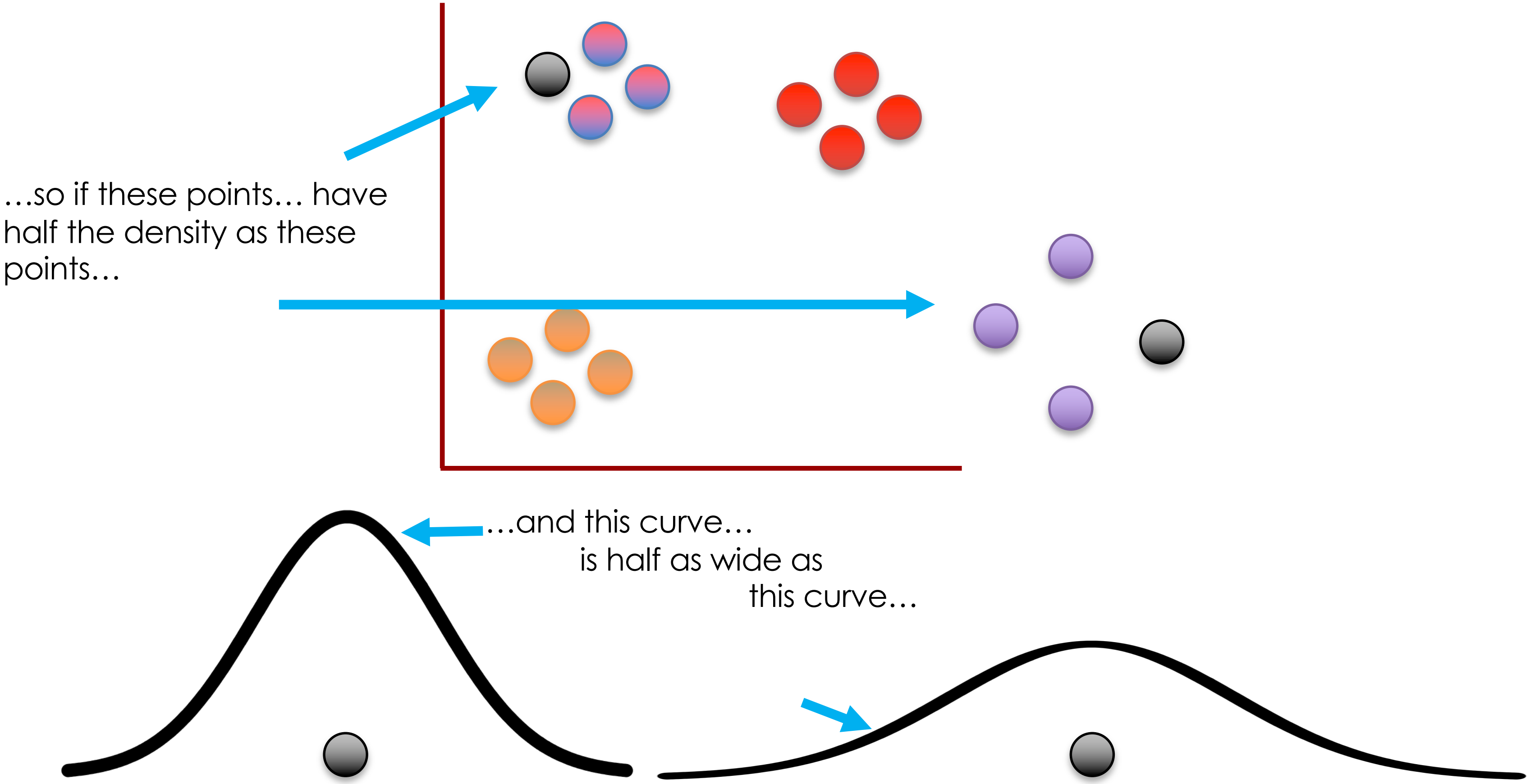


…so if these points… have half the density as these points…



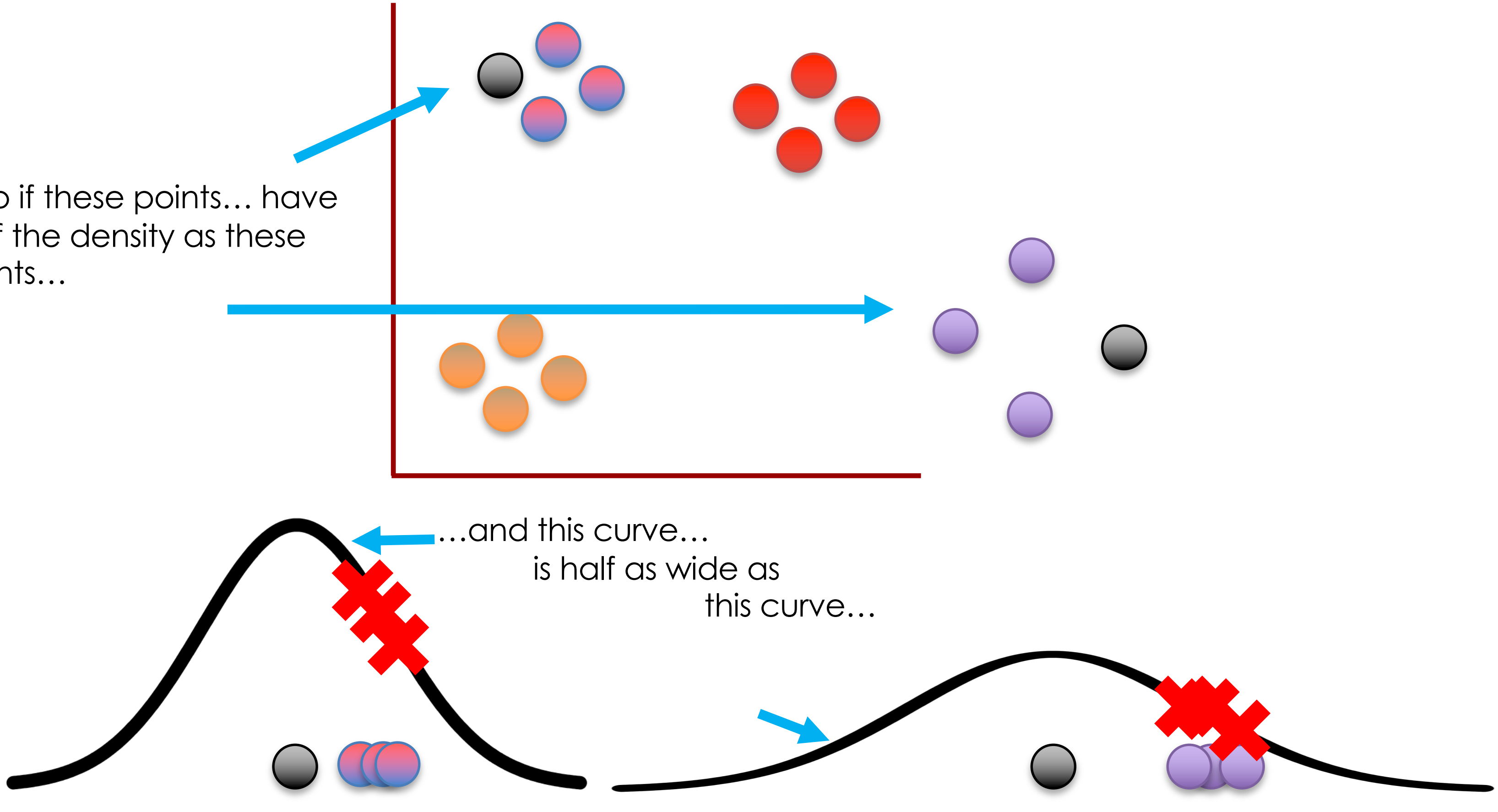
…so if these points… have half the density as these points…

…and this curve…



…so if these points… have half the density as these points…

…and this curve… is half as wide as this curve…



…so if these points… have half the density as these points…

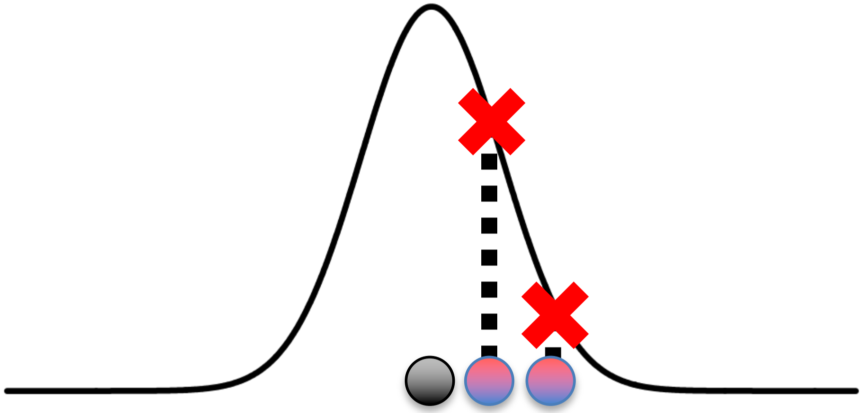
…and this curve…
is half as wide as
this curve…

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

Here’s an example…

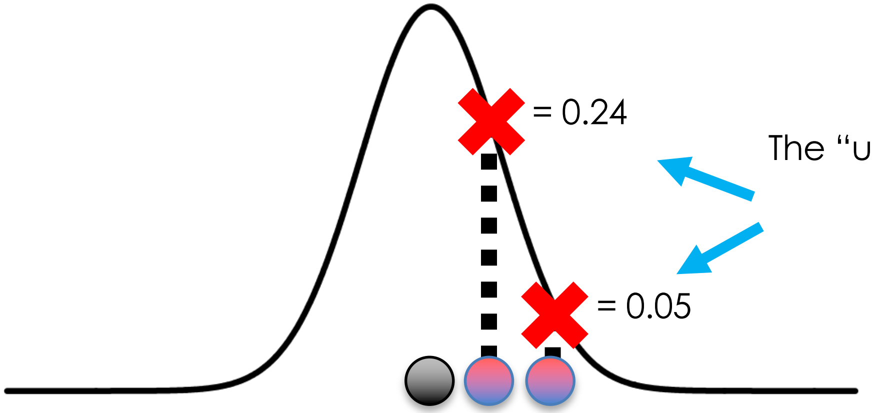
Here’s an example…

This curve has a std = 1.



Here's an example…

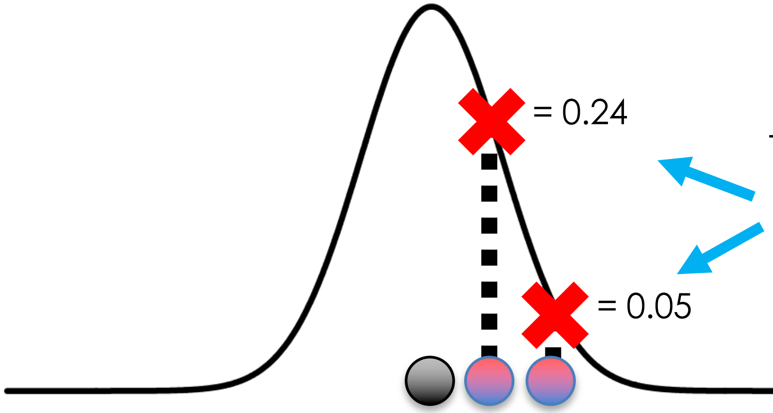
This curve has a std = 1.



The "unscaled" similarity values

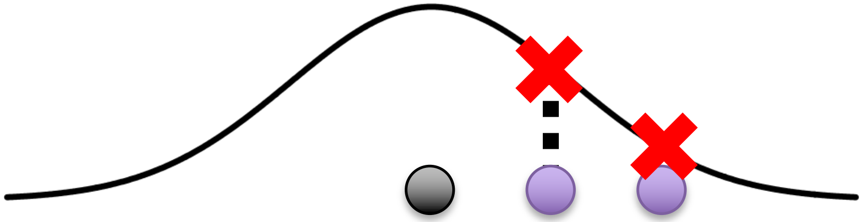
Here’s an example…

This curve has a std = 1.



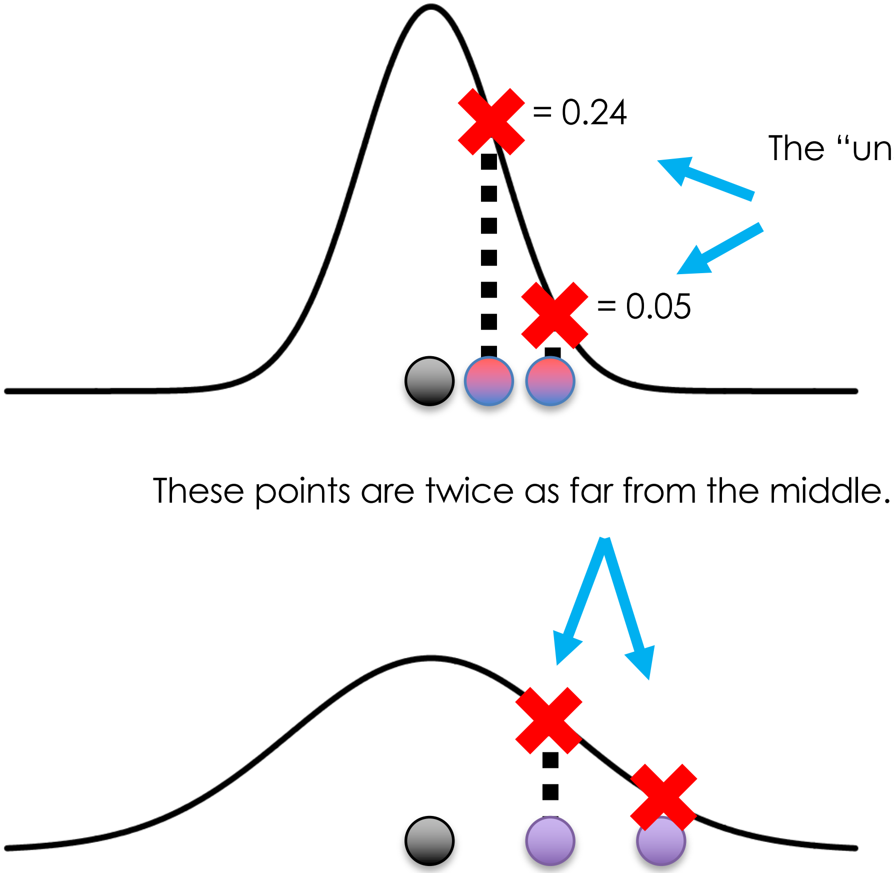
The “unscaled” similarity values

This curve has a std = 2.



Here’s an example…

This curve has a std = 1.



The “unscaled” similarity values

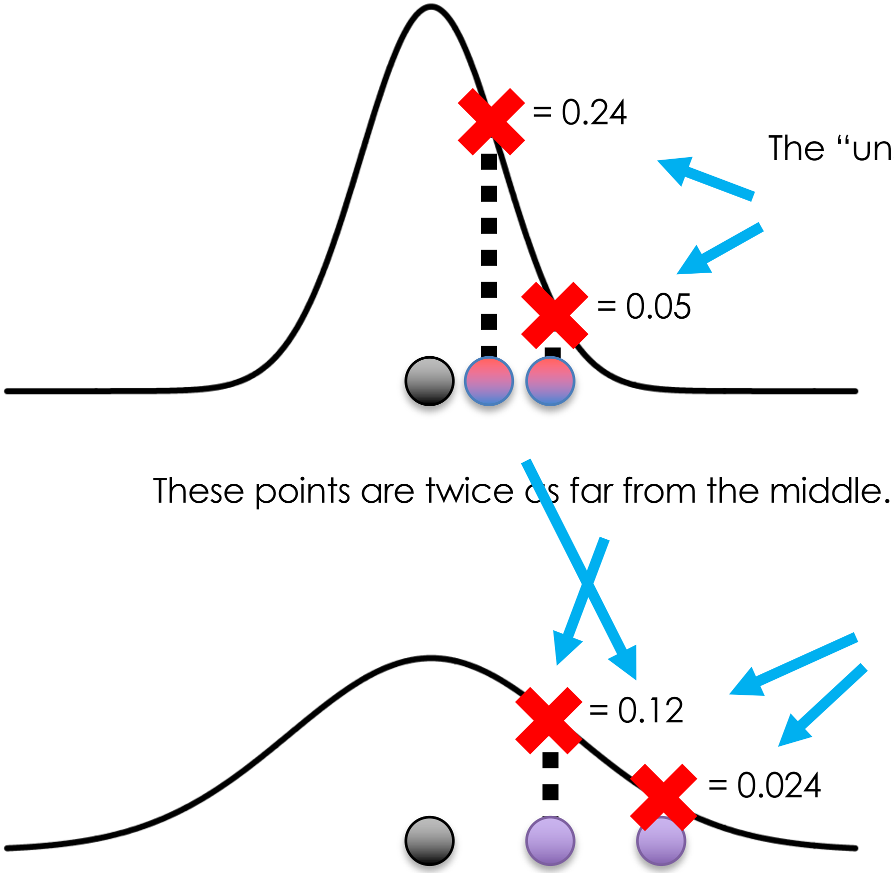
These points are twice as far from the middle.

This curve has a std = 2.

Here’s an example…

This curve has a std = 1.

This curve has a std = 2.



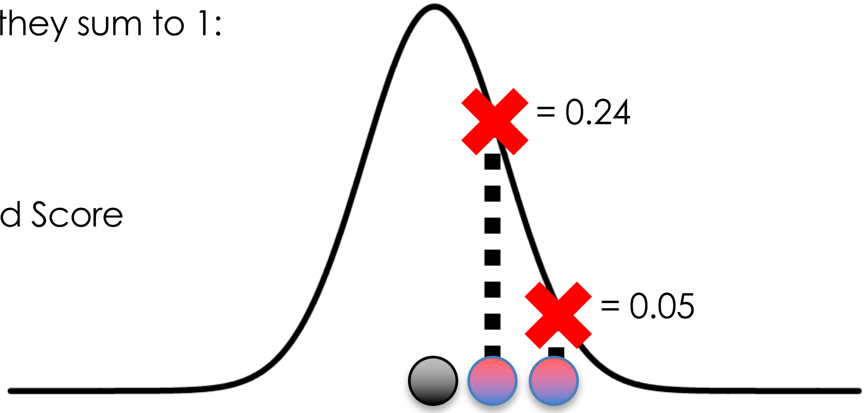
The “unscaled” similarity values

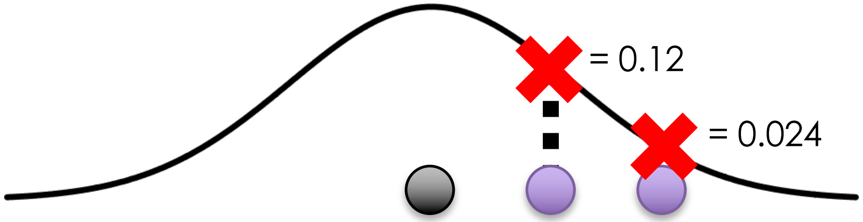
These points are twice as far from the middle.

The “unscaled” similarity values are half of the other ones.

To scale the similarity scores so they sum to 1:

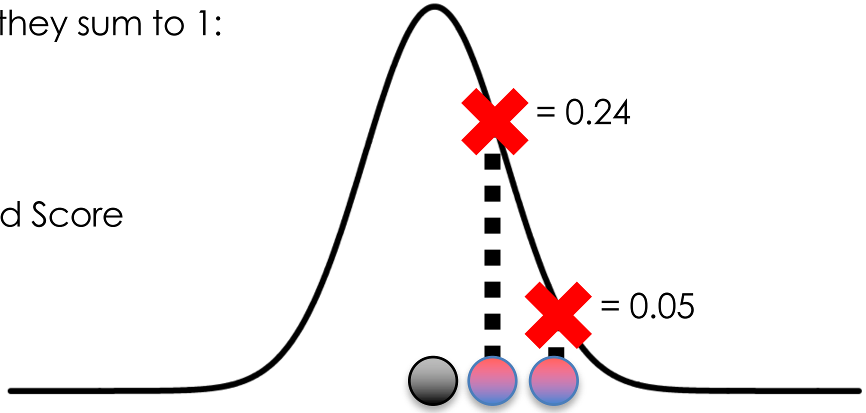
$$\frac{\text{Score}}{\text{Sum of all scores}} = \text{Scaled Score}$$





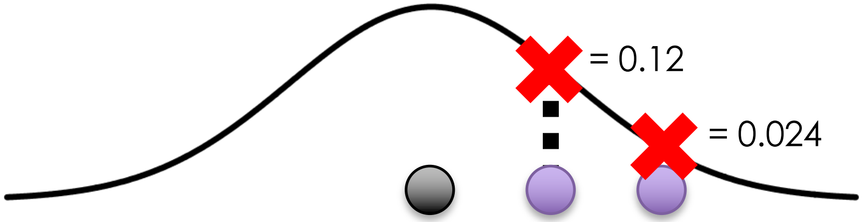
To scale the similarity scores so they sum to 1:

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



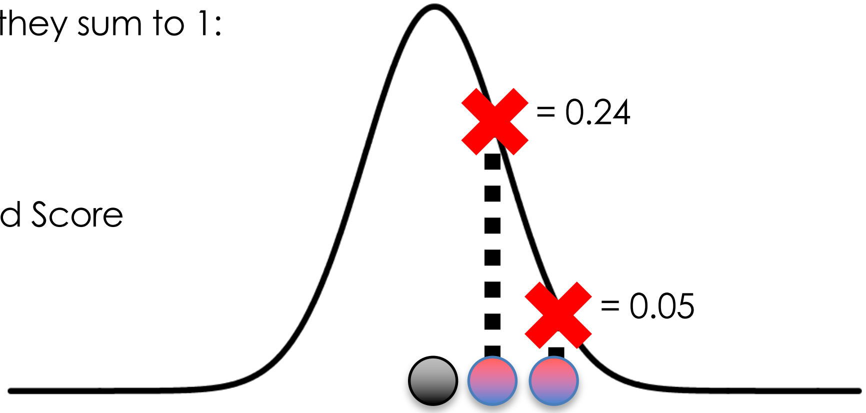
$$\frac{0.24}{0.24 + 0.05} = 0.82$$

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



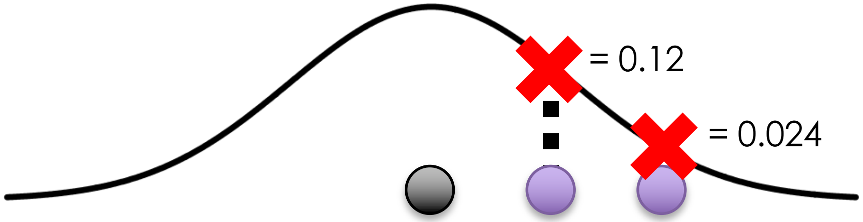
To scale the similarity scores so they sum to 1:

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



$$\frac{0.24}{0.24 + 0.05} = 0.82$$

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

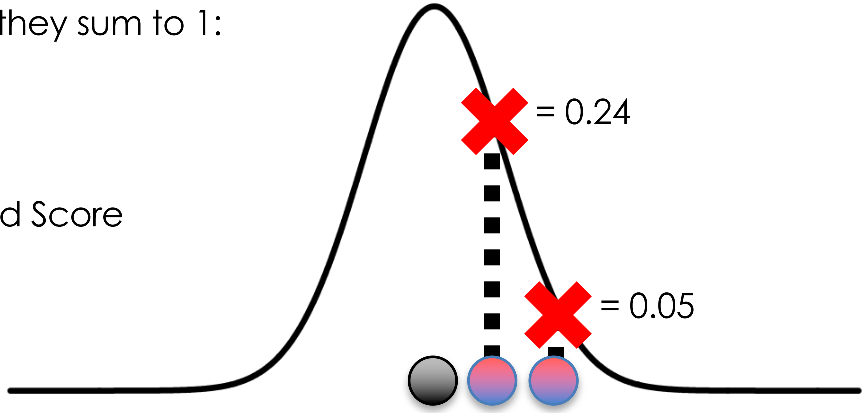


$$\frac{0.12}{0.12 + 0.024} = 0.82$$

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

To scale the similarity scores so they sum to 1:

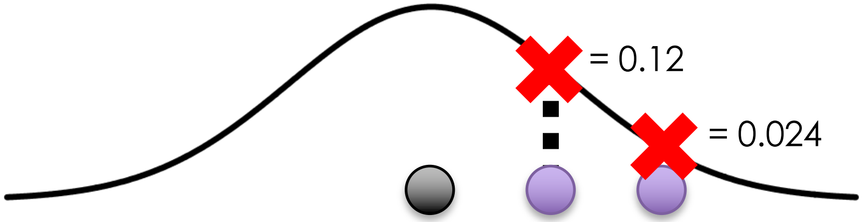
$$\frac{\text{Score}}{\text{Sum of all scores}} = \text{Scaled Score}$$



$$\frac{0.24}{0.24 + 0.05} = 0.82$$

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

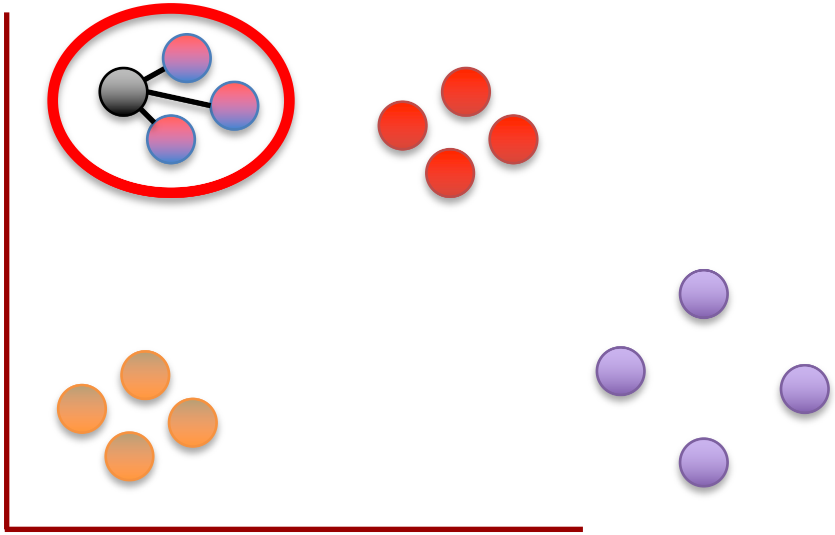
Same values!



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

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

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



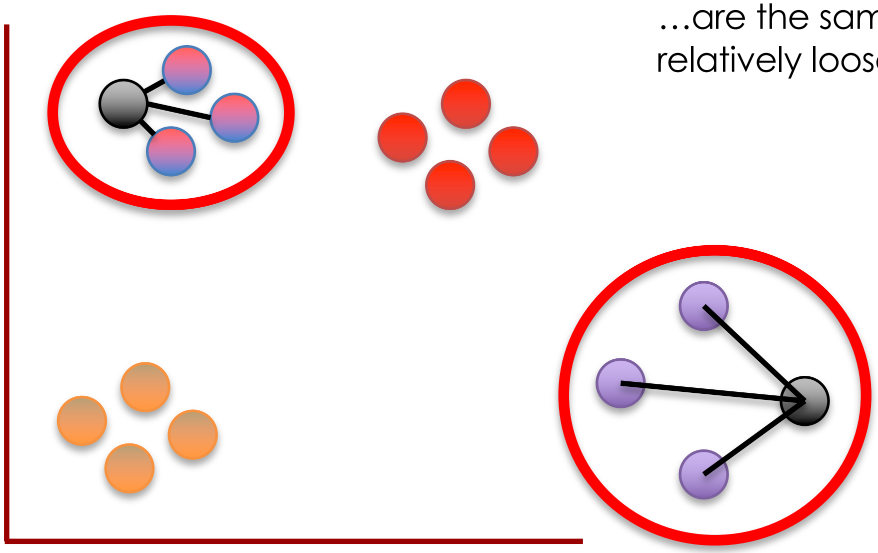
$$\frac{0.24}{0.24 + 0.05} = 0.82$$

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

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

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

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



$$\frac{0.24}{0.24 + 0.05} = 0.82$$

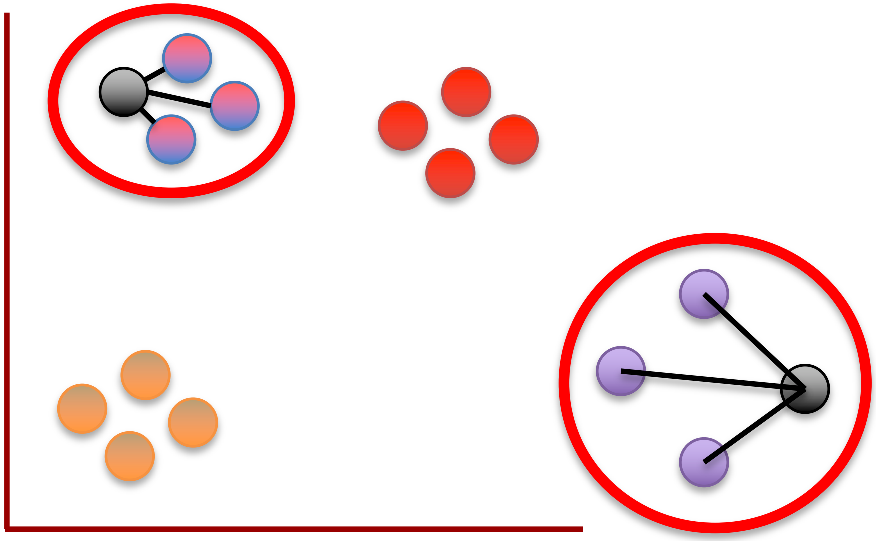
$$\frac{0.05}{0.24 + 0.05} = 0.18$$

…are the same for this
relatively loose cluster!

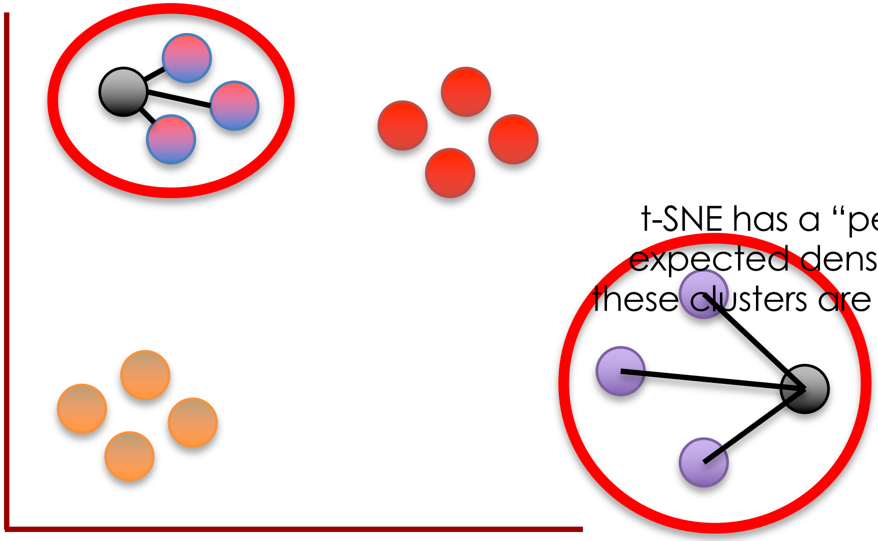
$$\frac{0.12}{0.12 + 0.024} = 0.82$$

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

The reality is a little more complicated, but only slightly.

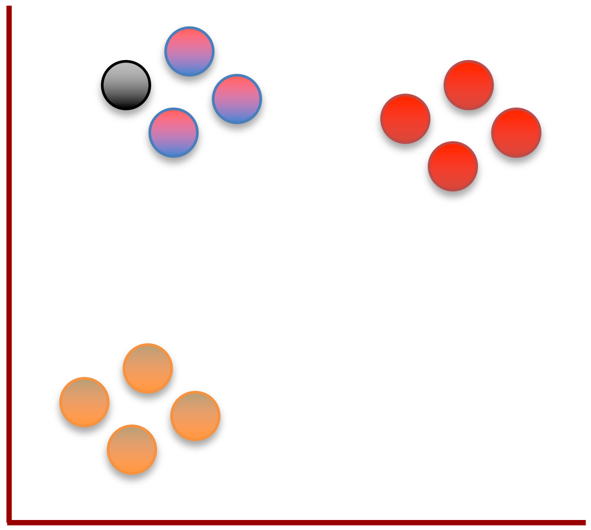


The reality is a little more complicated, but only slightly.

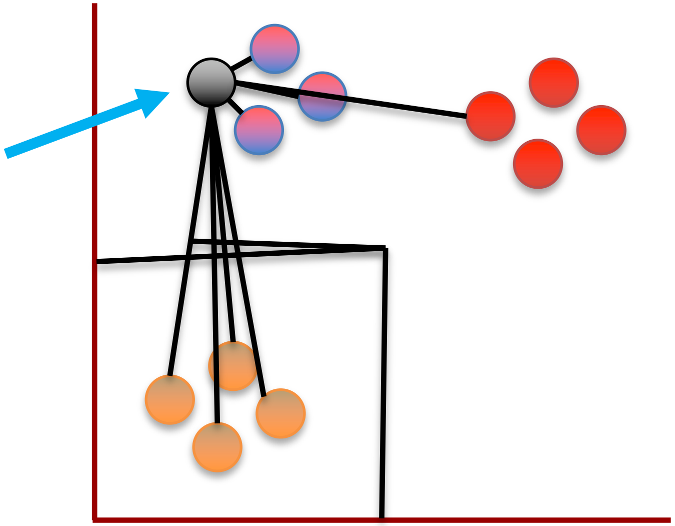


t-SNE has a "perplexity" parameter equal to the expected density, and that comes into play, but these clusters are still more "similar" than you might expect.

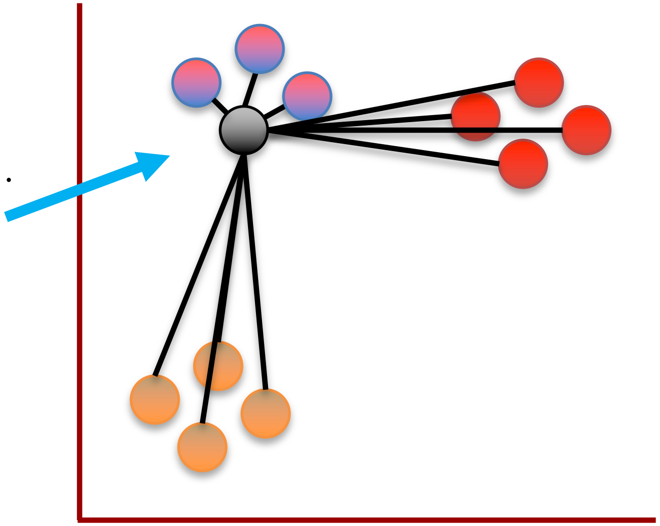
Now back to the original scatter plot…



We’ve calculated similarity scores for this point.

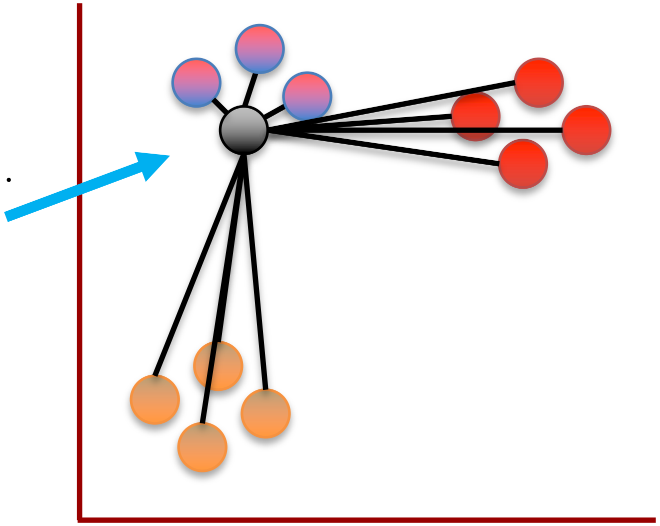


Now we do it for this point…

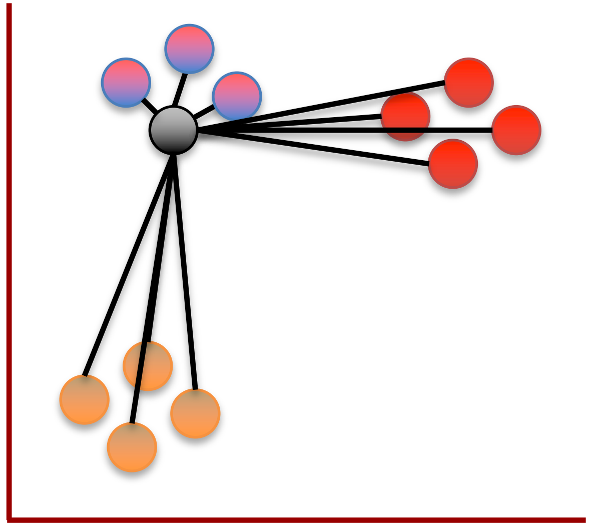


Now we do it for this point…

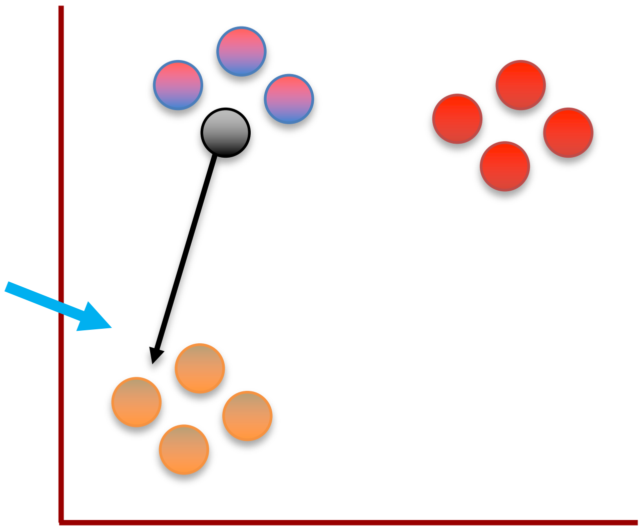
…and we do it for all the points.

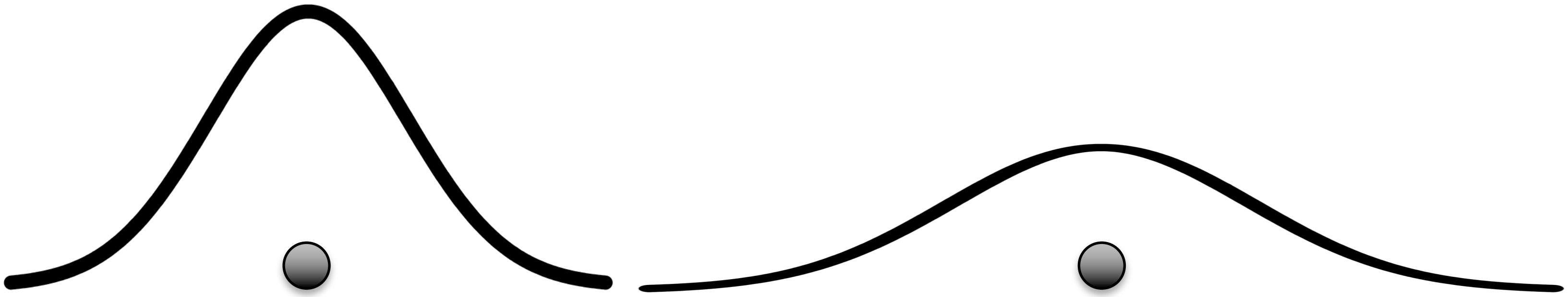


One last thing and the scatter plot will be all set with similarity scores!!!

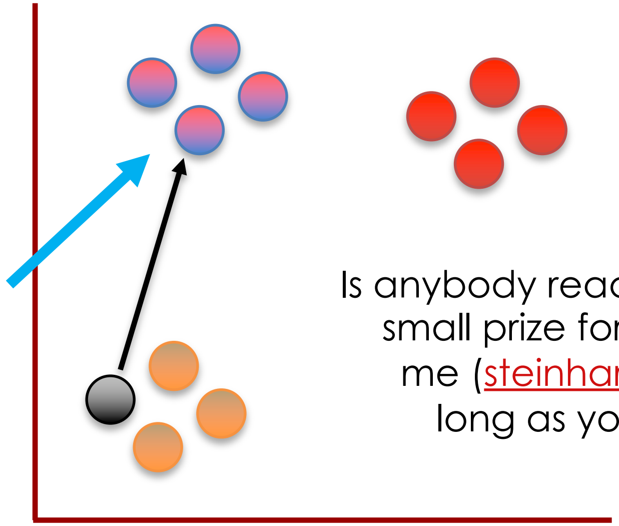


Because the width of the distribution is based on the density of the surrounding data points, the similarity score to this node…

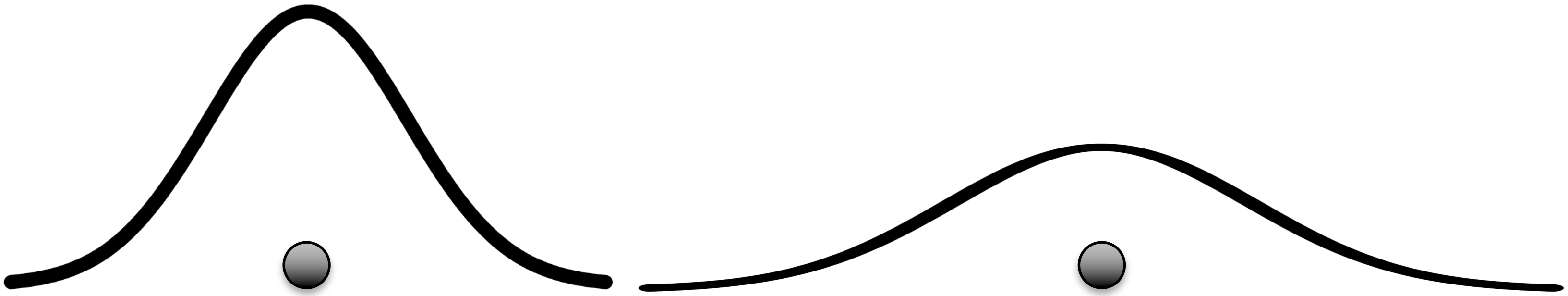


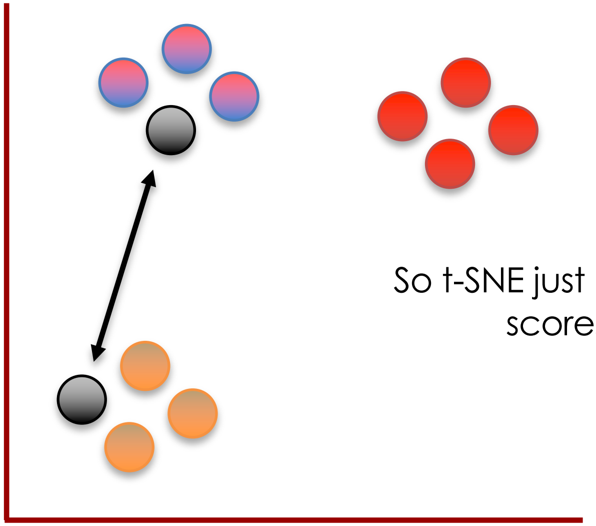


…might not be the same as the similarity to this node.

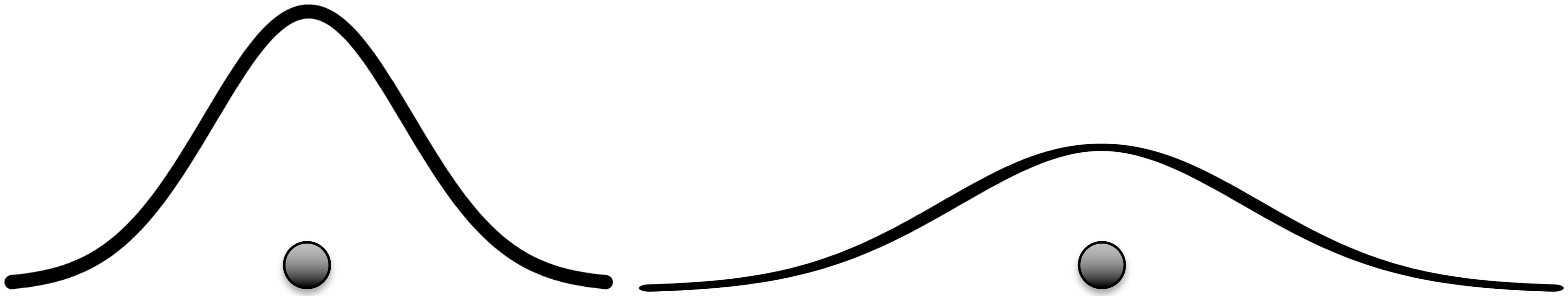


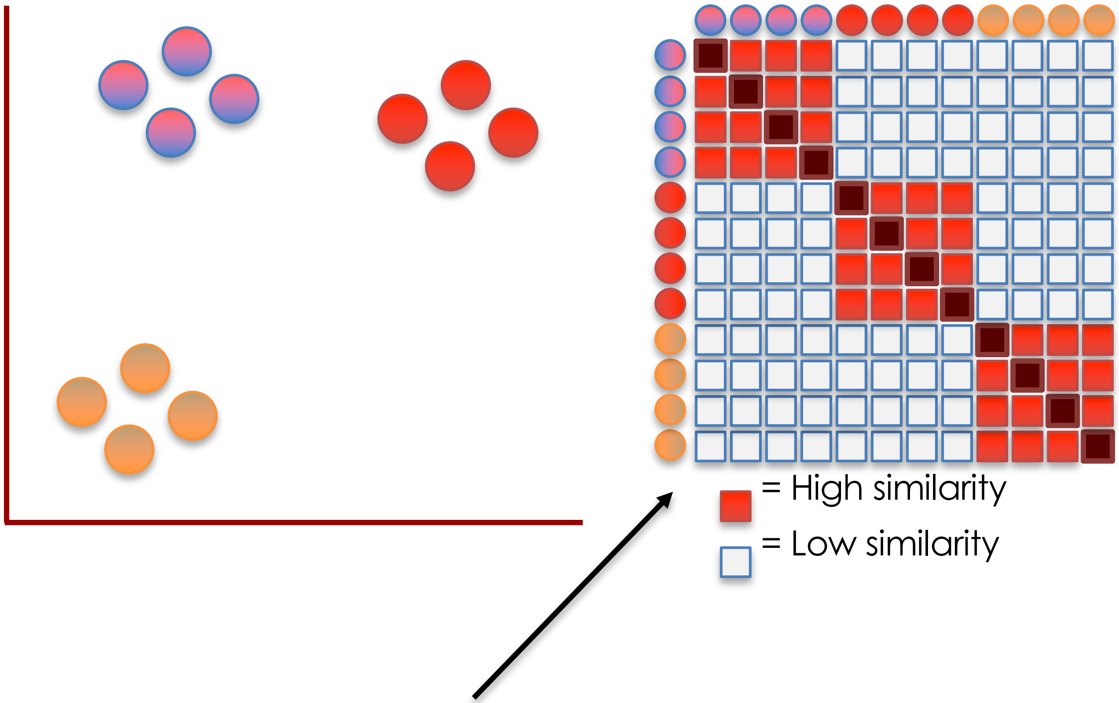
Is anybody reading these slides? I'll have a small prize for the first student who emails me (steinhardt@nbi.ku.dk) about this, as long as you email by midnight, May 1.



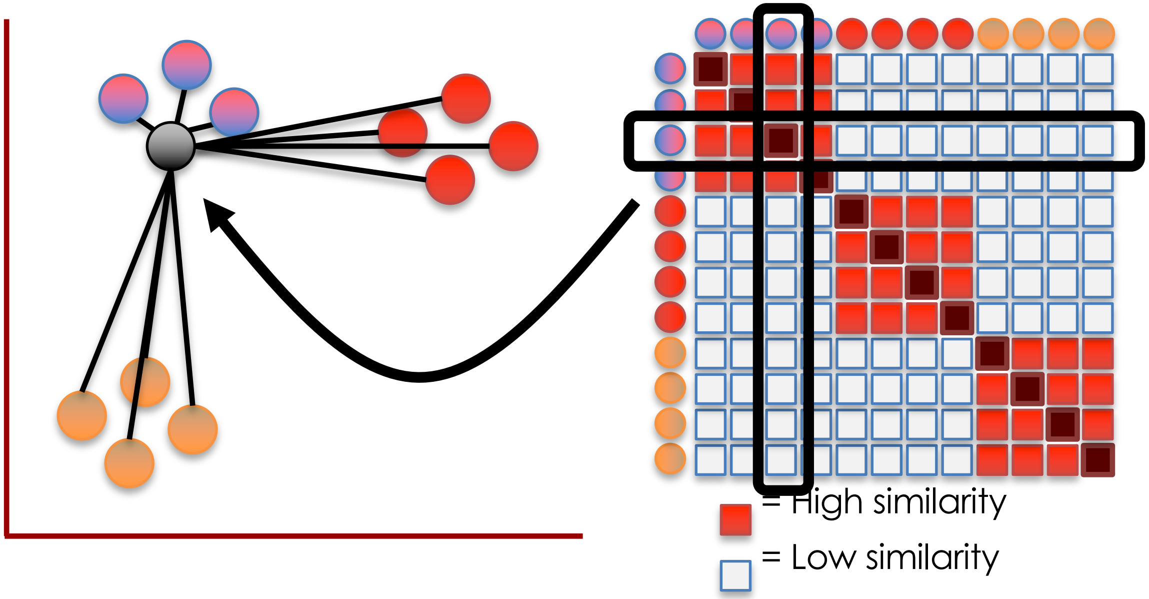


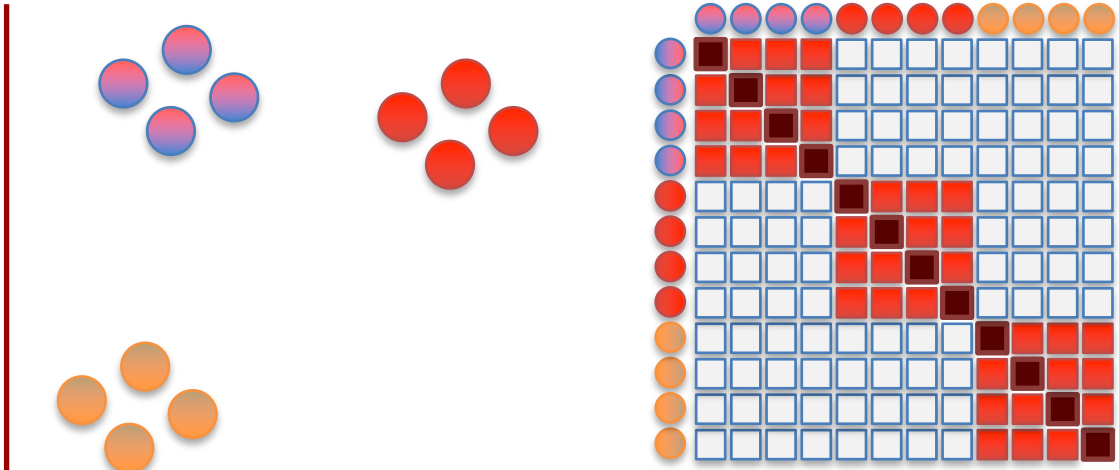
So t-SNE just averages the two similarity scores from the two directions…





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


🟥 = High similarity
□ = Low similarity

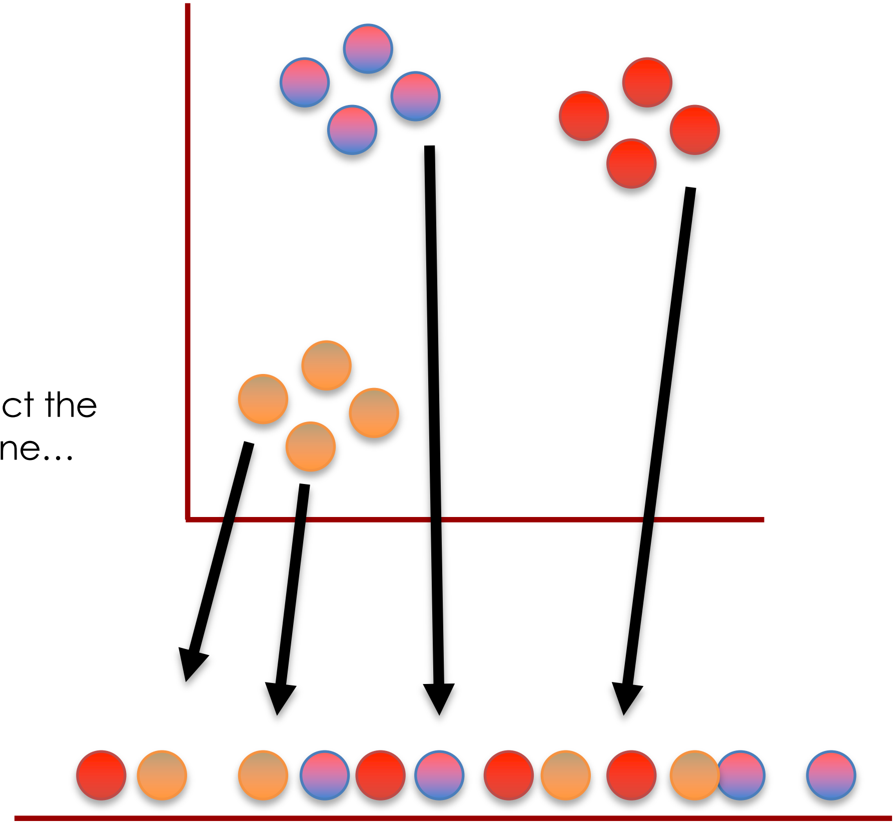


🟥 = High similarity

□ = Low similarity

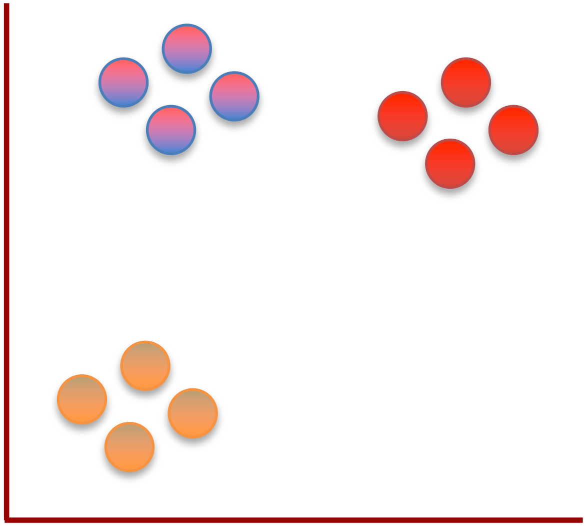
Hooray!!! We’re done doing calculating similarity scores for the scatter plot!

Now we randomly project the data onto the number line…

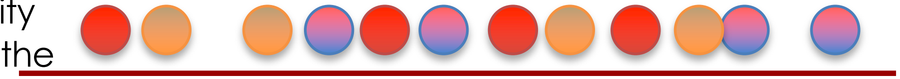


COSMIC DAWN CENTER
DAWN

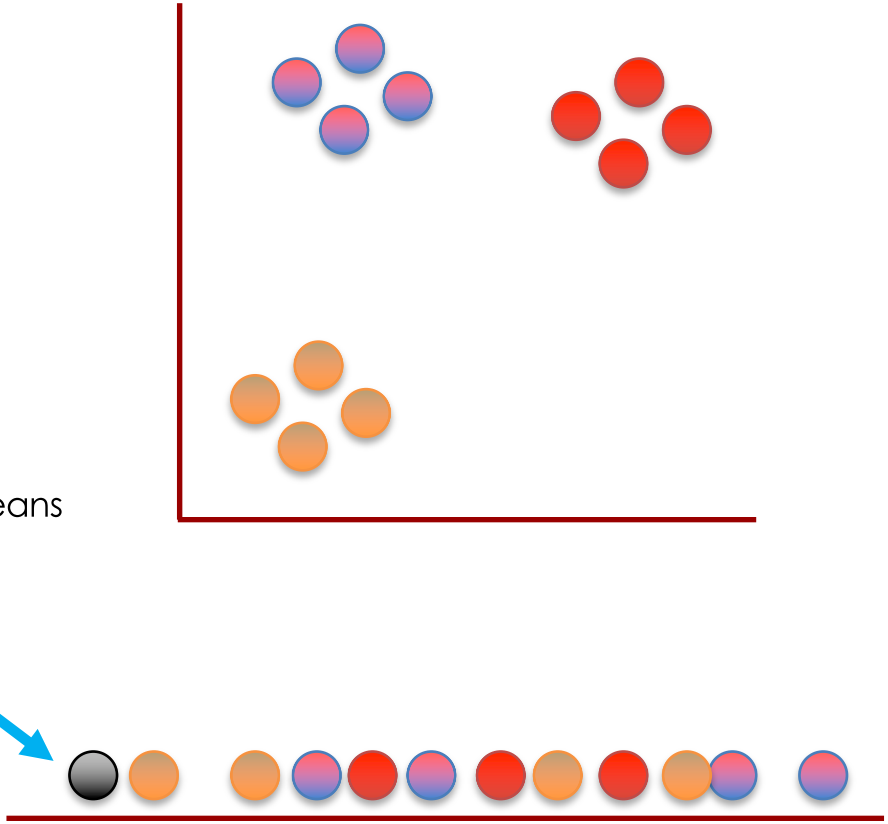
Now we randomly project the data onto the number line…

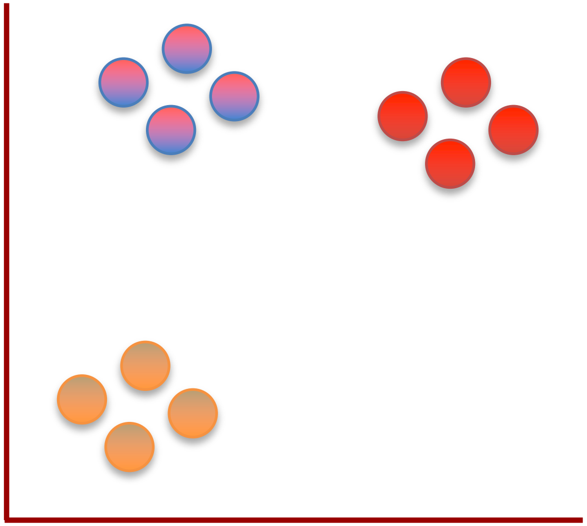


… and calculate similarity scores for the points on the number line.

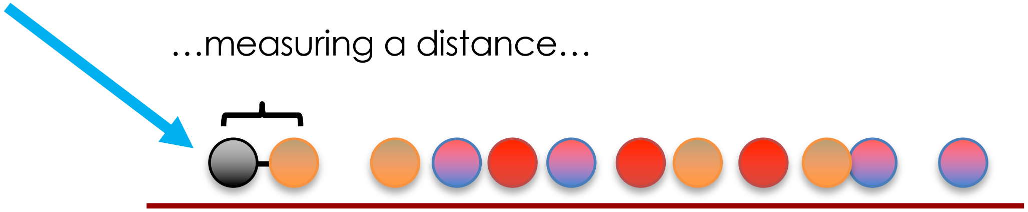


Just like before, that means picking a point…





Just like before, that means picking a point…

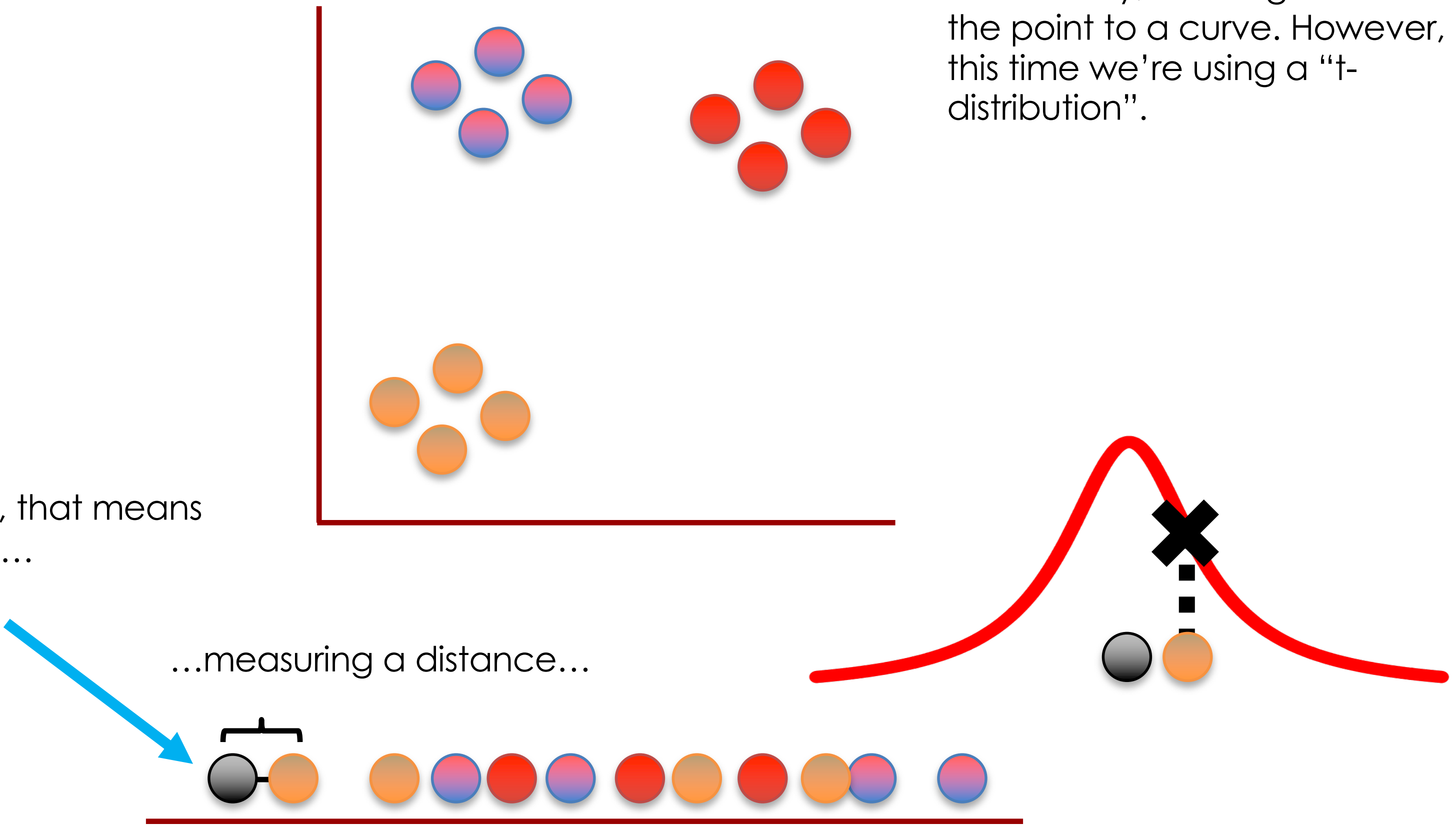


…measuring a distance…

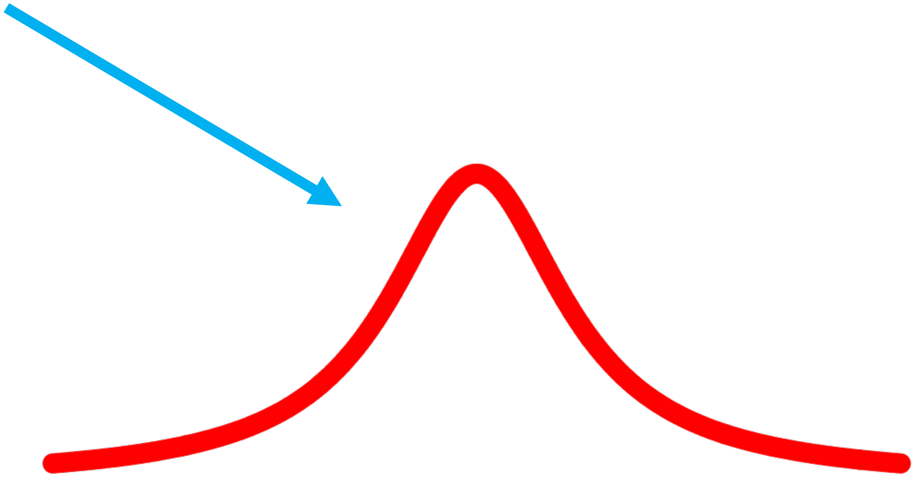
…and lastly, drawing a line from the point to a curve. However, this time we're using a "t-distribution".

Just like before, that means picking a point…

…measuring a distance…

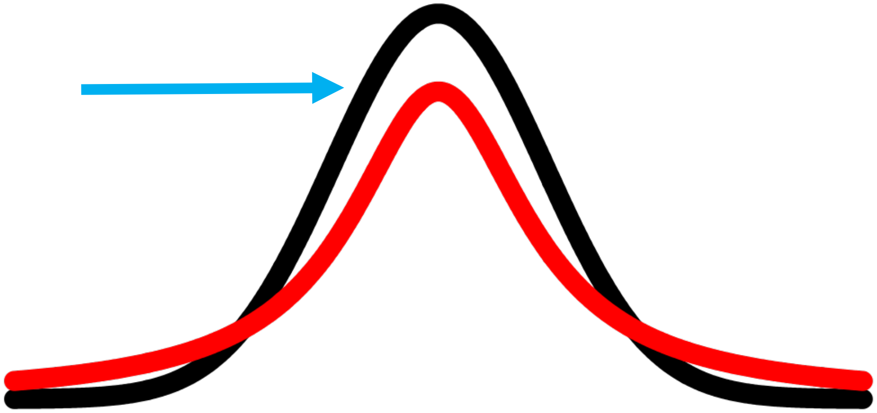


A “t-distribution”…



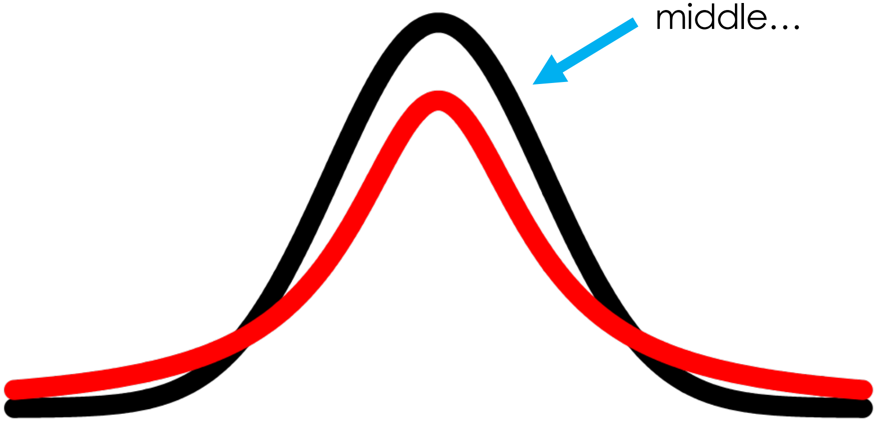
A “t-distribution”…

…is a lot like a normal distribution



A “t-distribution”…

…is a lot like a normal distribution…

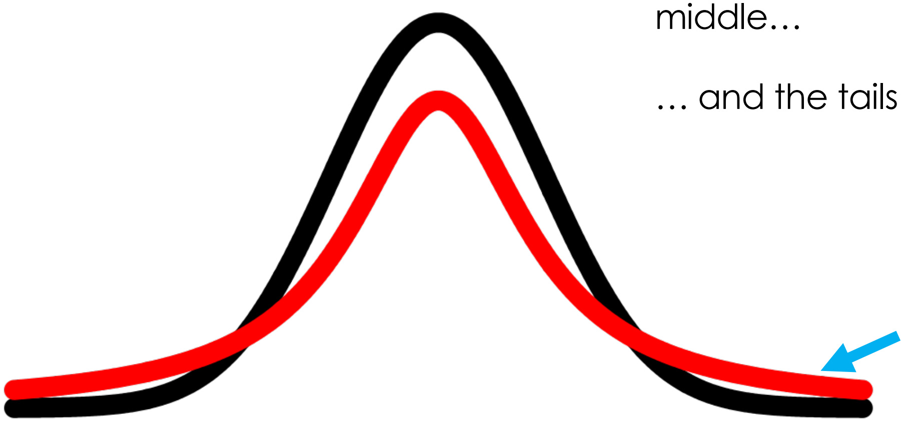


…except the “t” isn’t as tall in the middle…

COSMIC DAWN CENTER
DAWN

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.

COSMIC DAWN CENTER

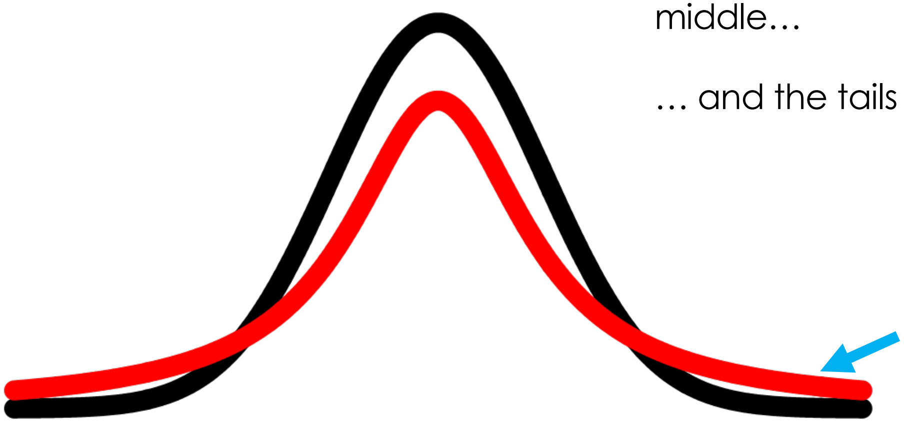
DAWN

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.



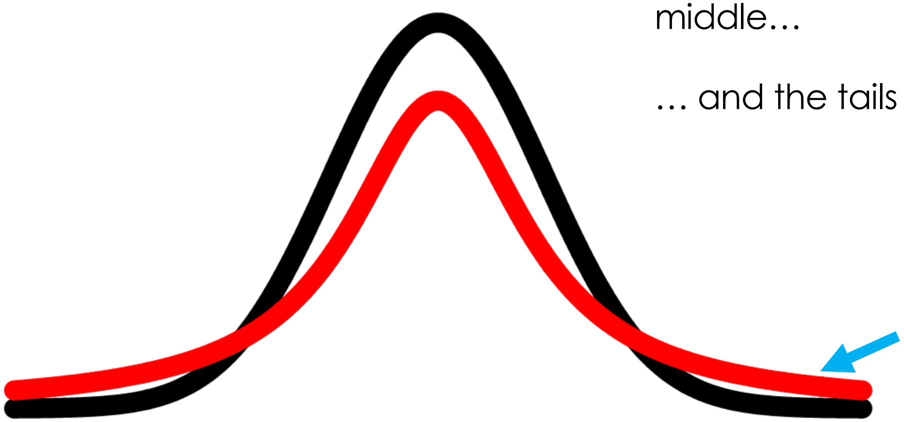
The “t-distribution” is the “t” in t-SNE.

A “t-distribution”…

…is a lot like a normal distribution…

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

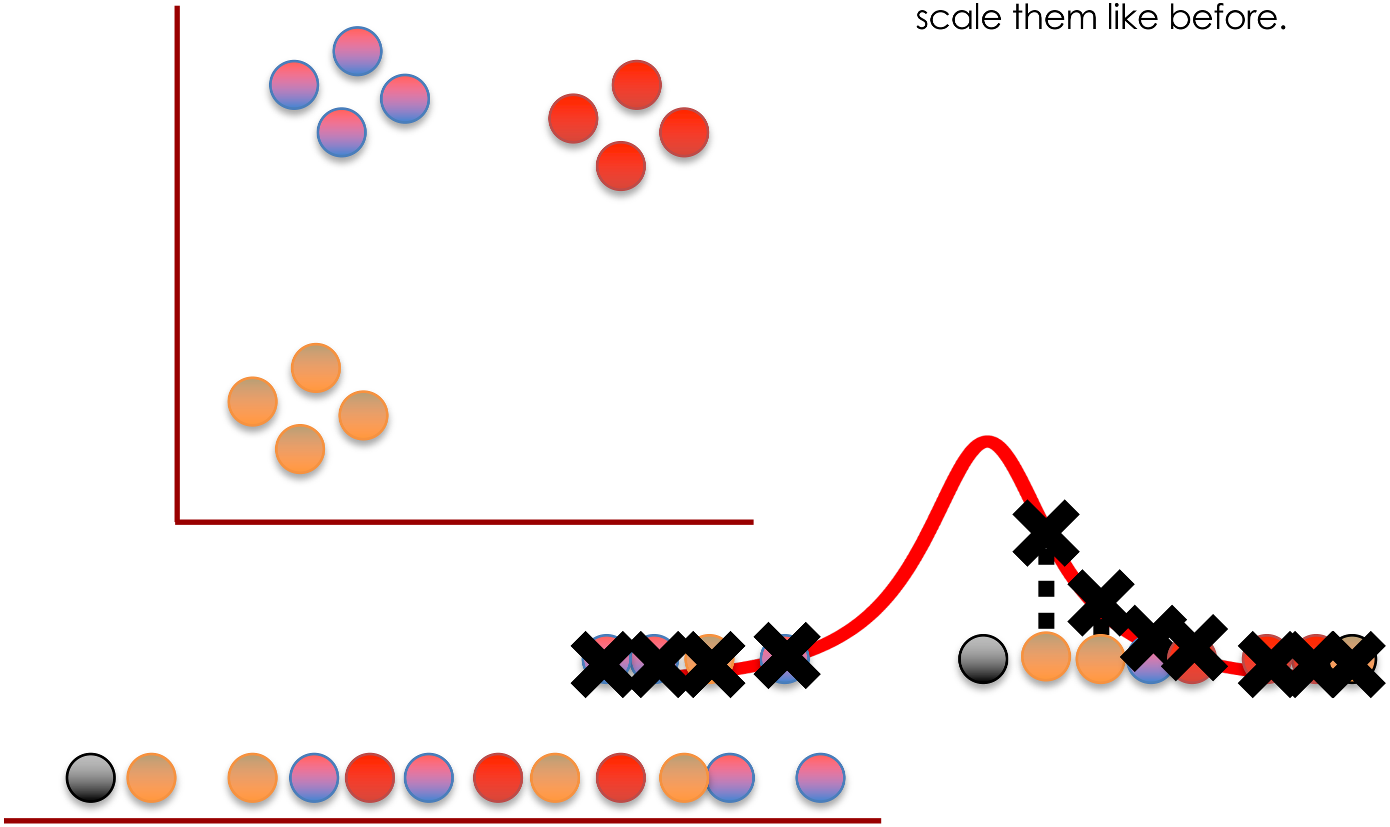
… and the tails are taller on the ends.

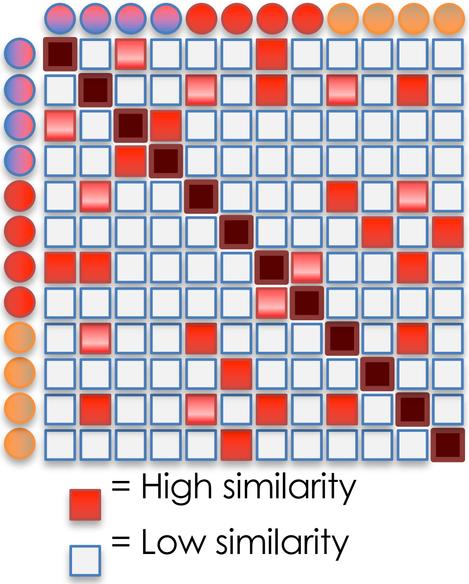


The “t-distribution” is the “t” in t-SNE.

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.

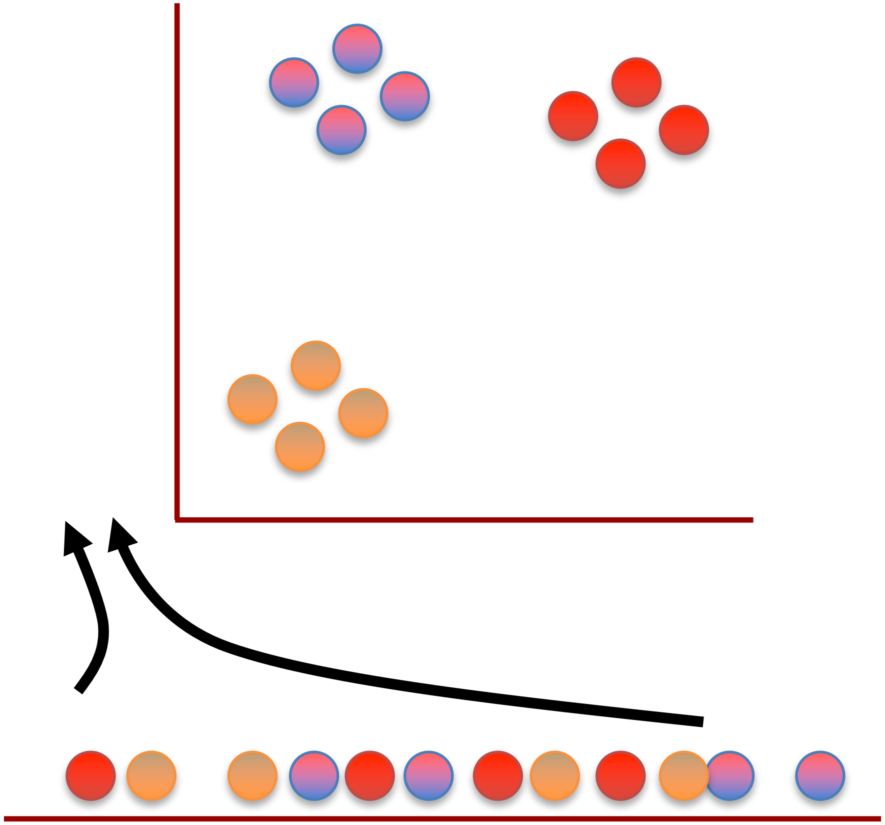


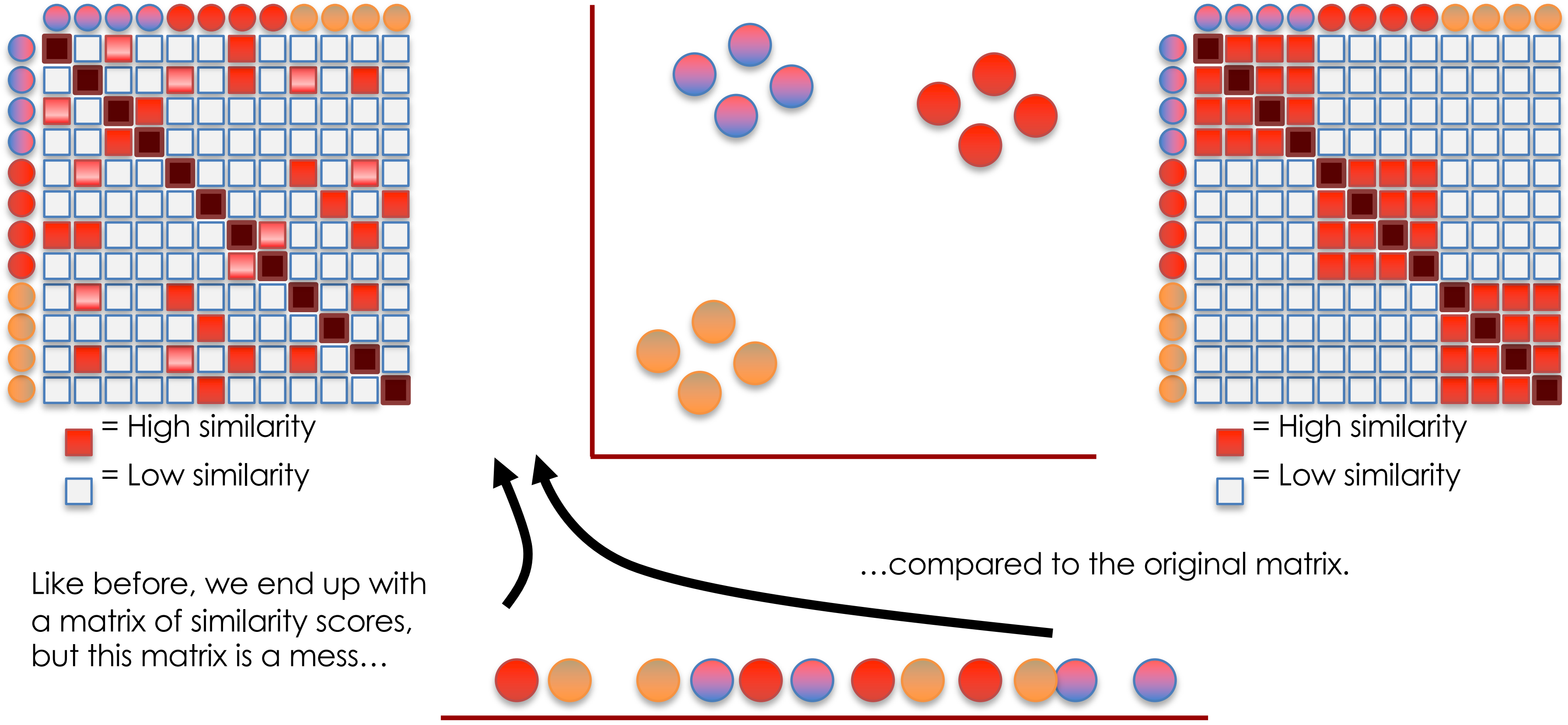


🟥 = High similarity

⬜ = Low similarity

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





🟥 = High similarity

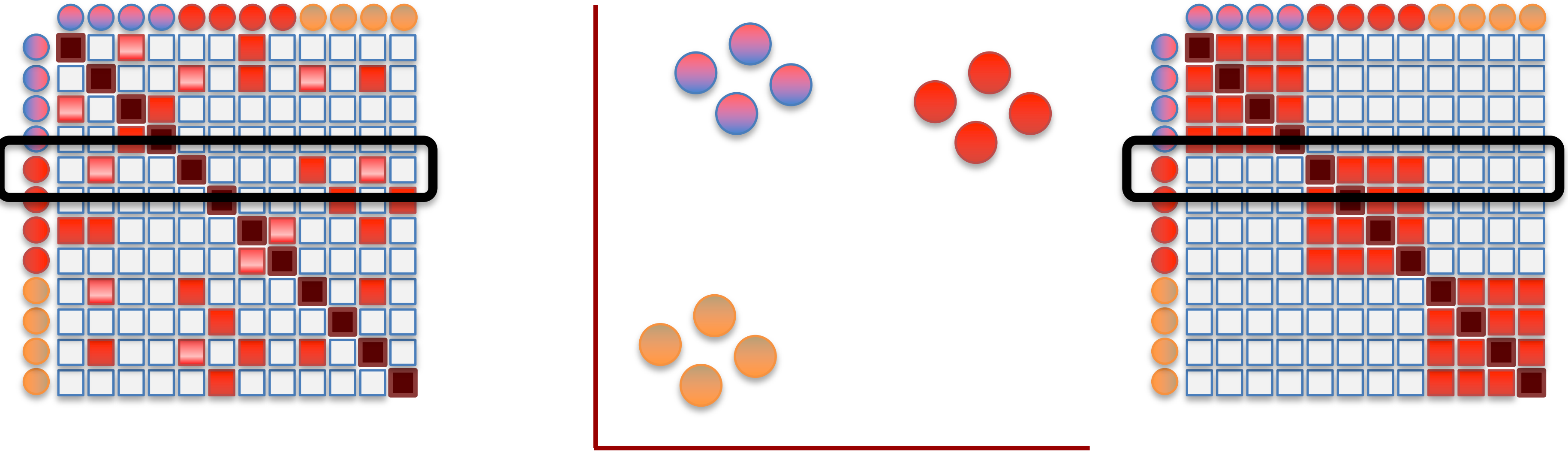
□ = Low similarity

🟥 = High similarity

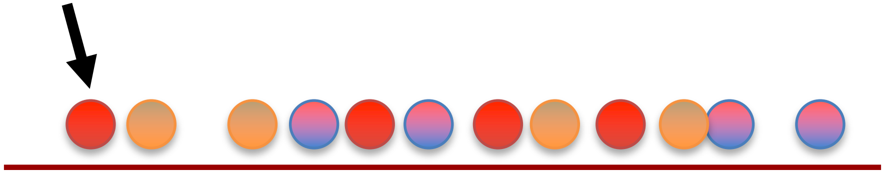
□ = Low similarity

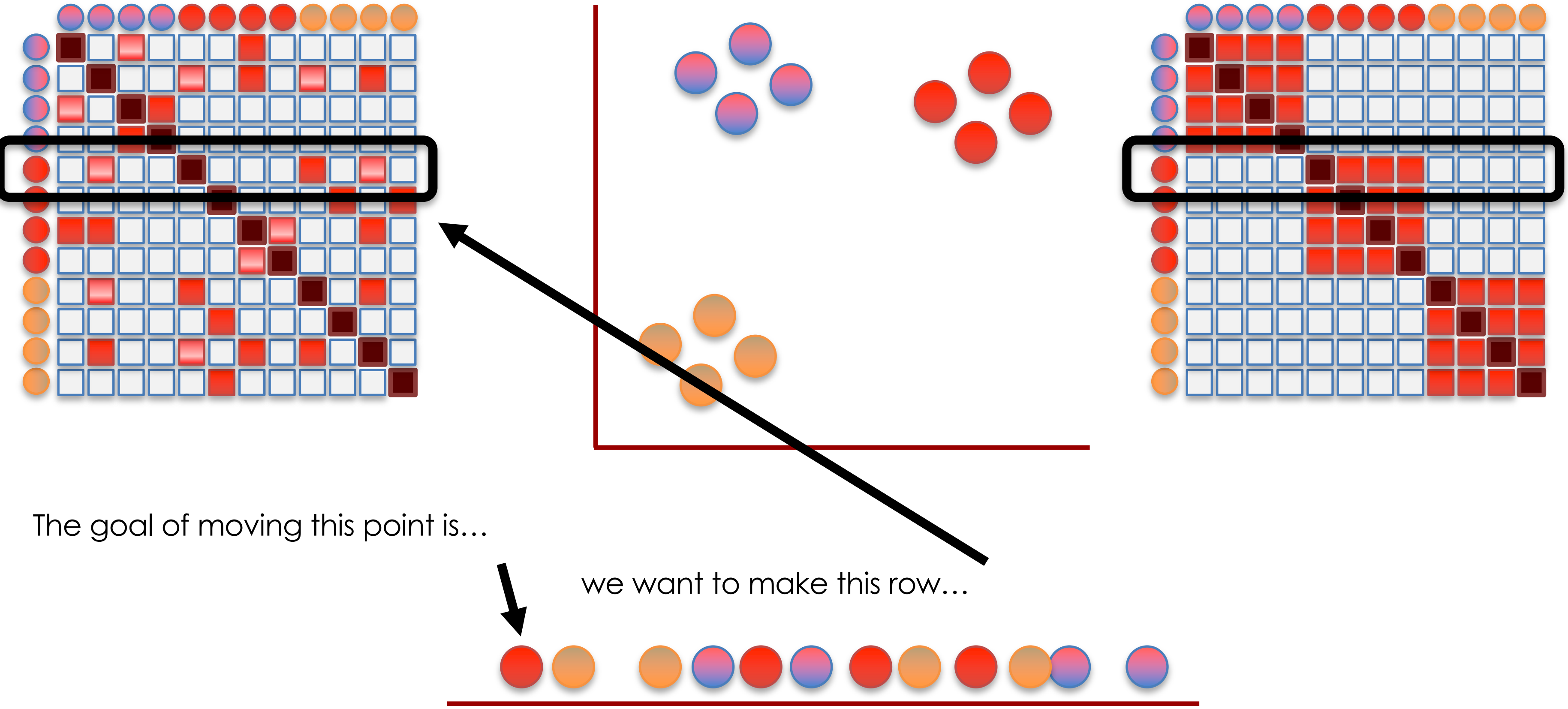
Like before, we end up with a matrix of similarity scores, but this matrix is a mess…

…compared to the original matrix.



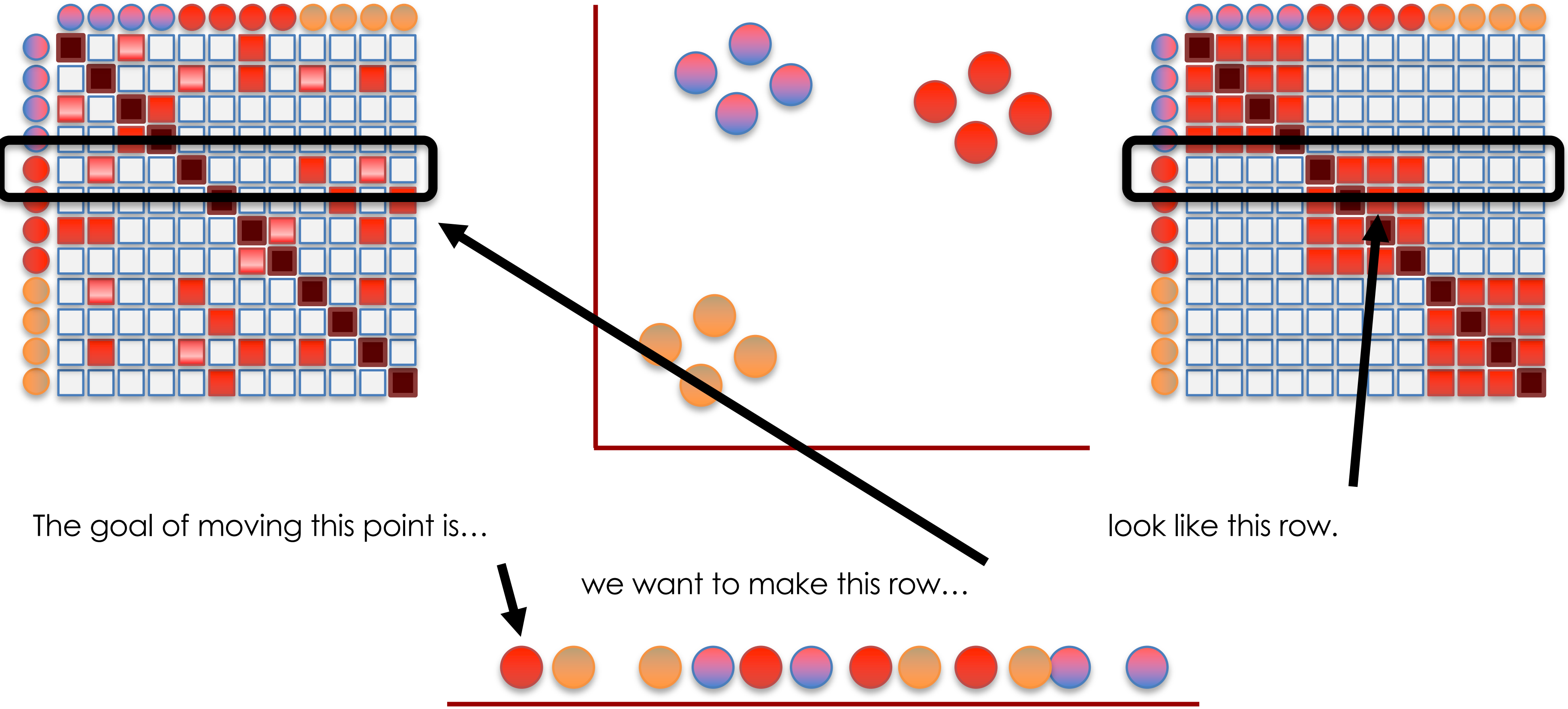
The goal of moving this point is…





The goal of moving this point is…

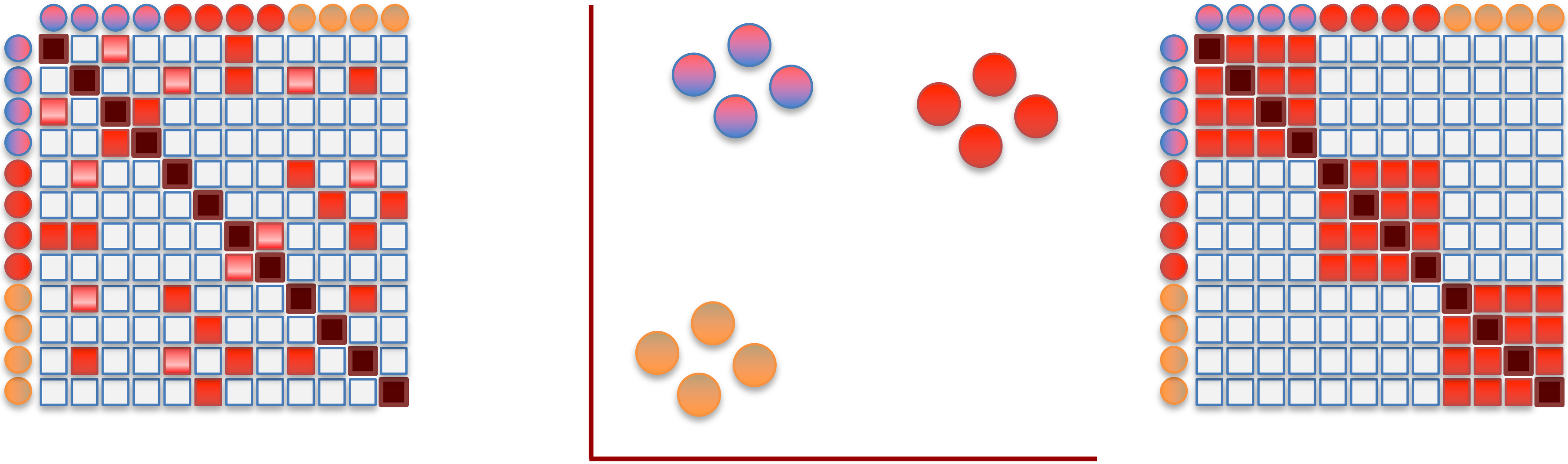
we want to make this row…



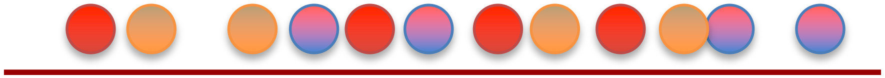
The goal of moving this point is…

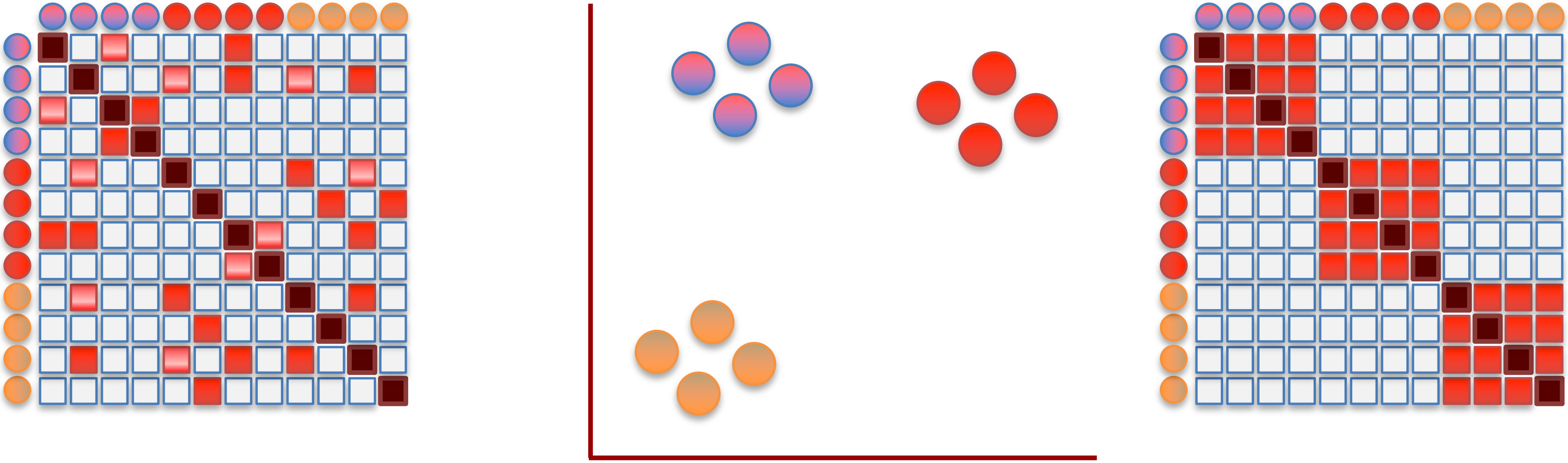
we want to make this row…

look like this row.

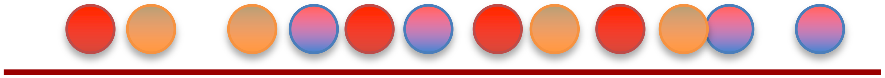


t-SNE moves the points a little bit at a time, and each step it chooses a direction that makes the matrix on the left more like the matrix on the right.

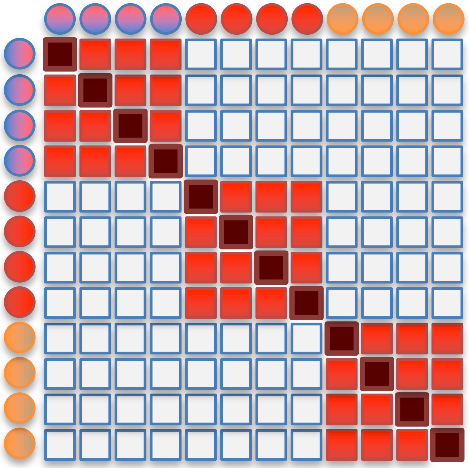


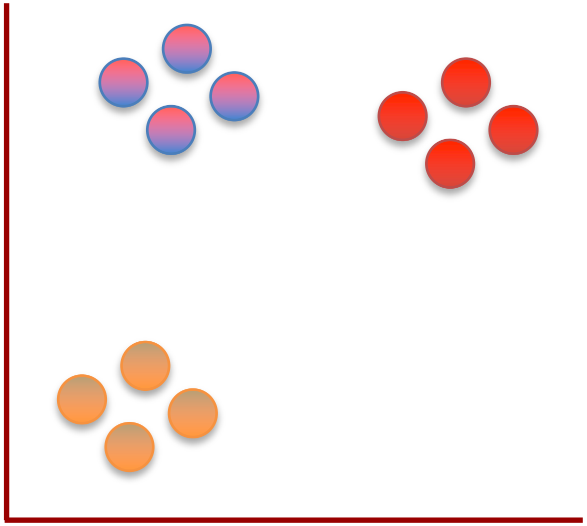


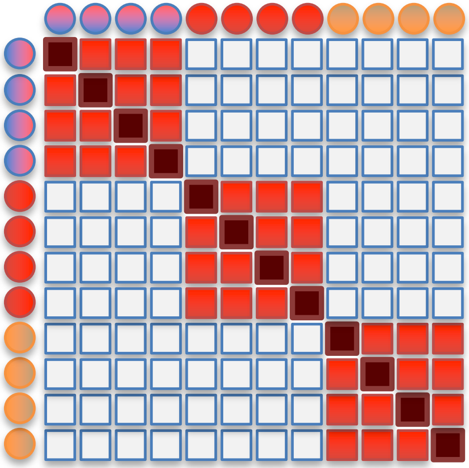
t-SNE moves the points a little bit at a time, and each step it chooses a direction that makes the matrix on the left more like the matrix on the right.



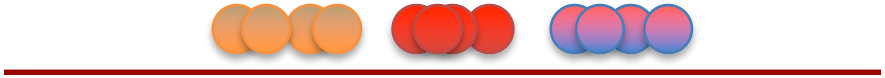
It uses small steps, because it's a little bit like a chess game and can't be solved all at once. Instead, it goes one move at at time.

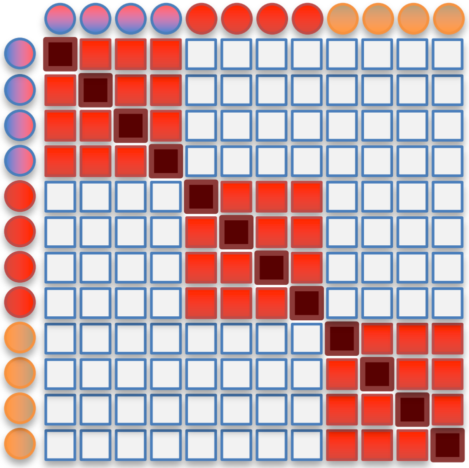


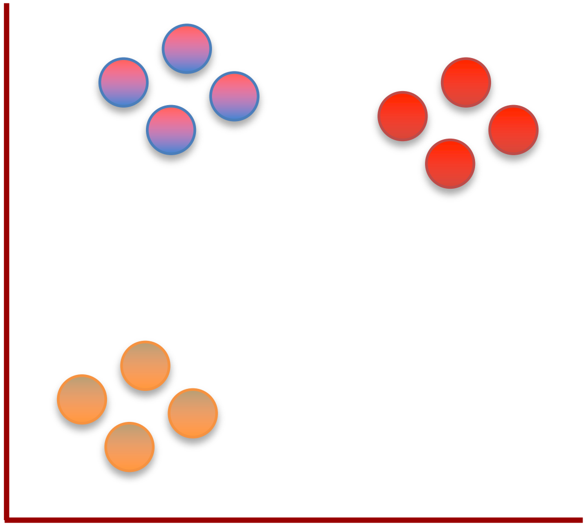


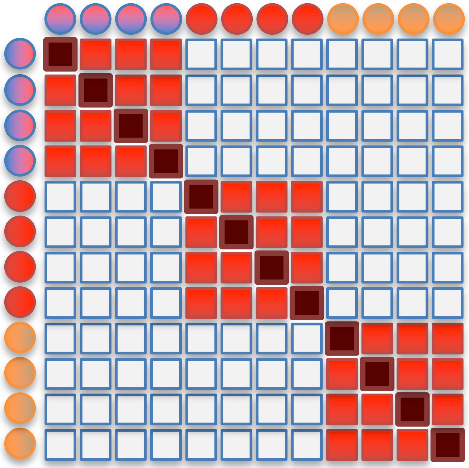


BAM!!!

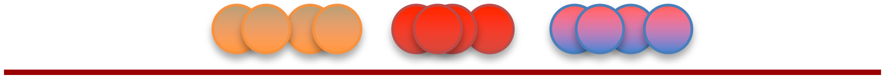


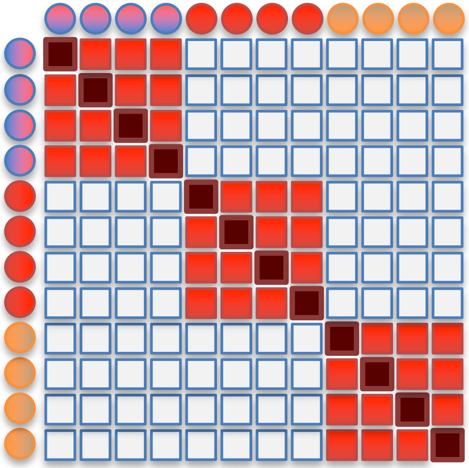


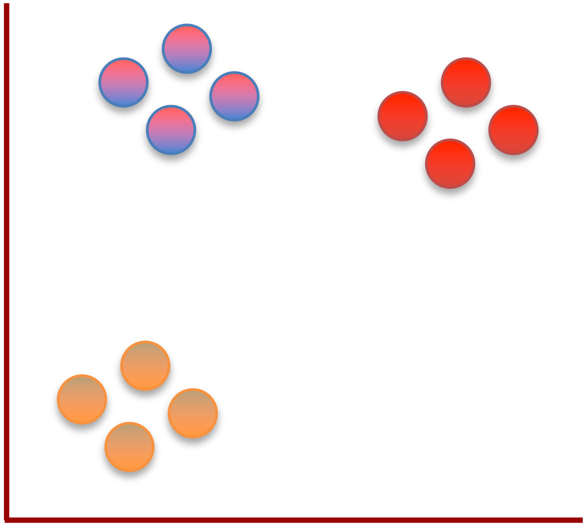


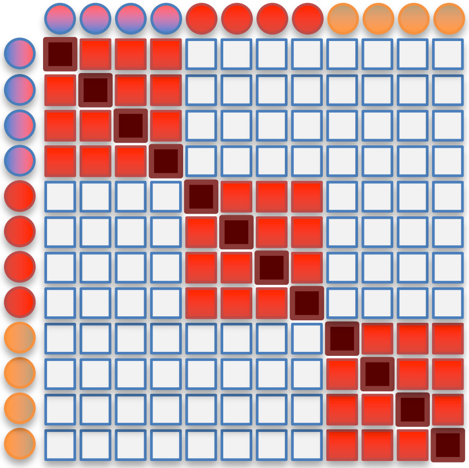


Now to finally tell you why the “t-distribution” is used…

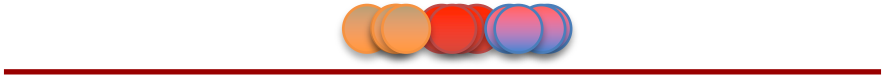


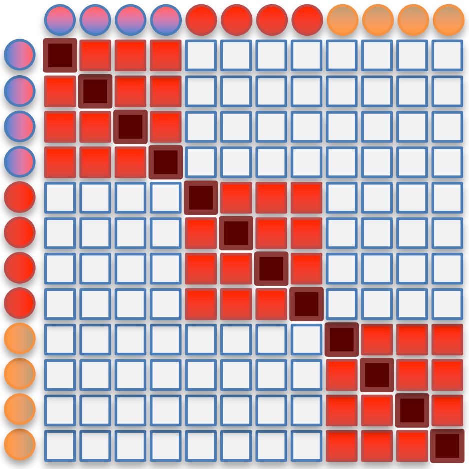


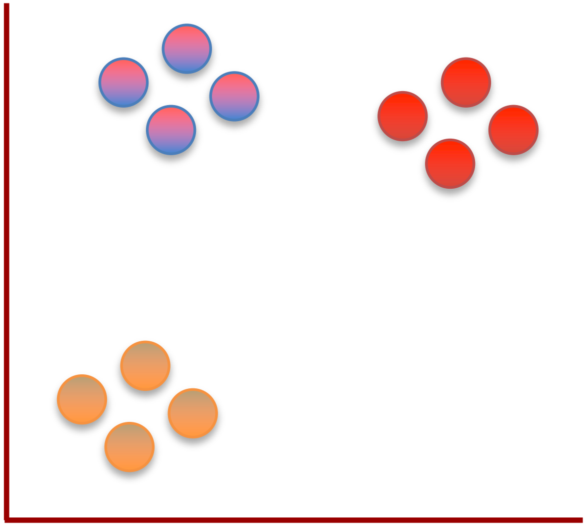


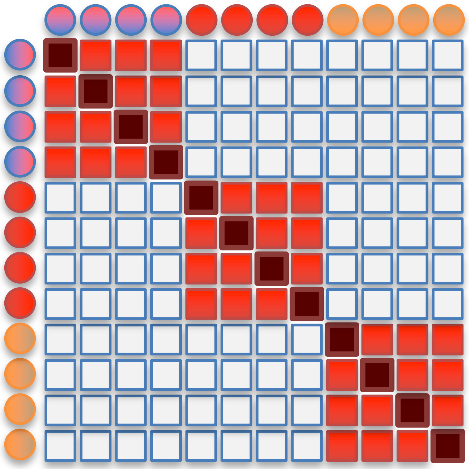


Originally, the “SNE” algorithm used a normal distribution throughout and the clusters clumped up in the middle and were harder to see.

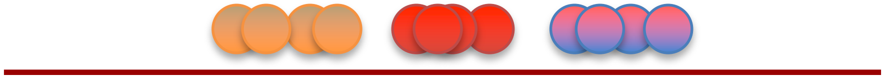




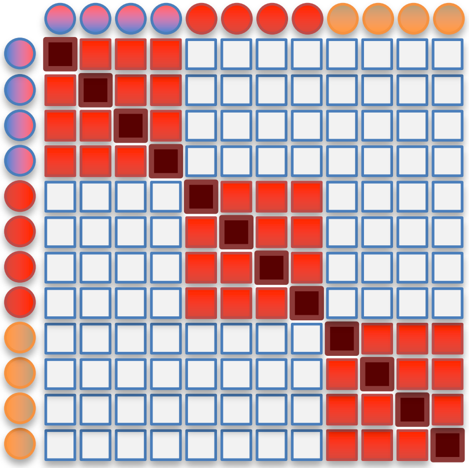


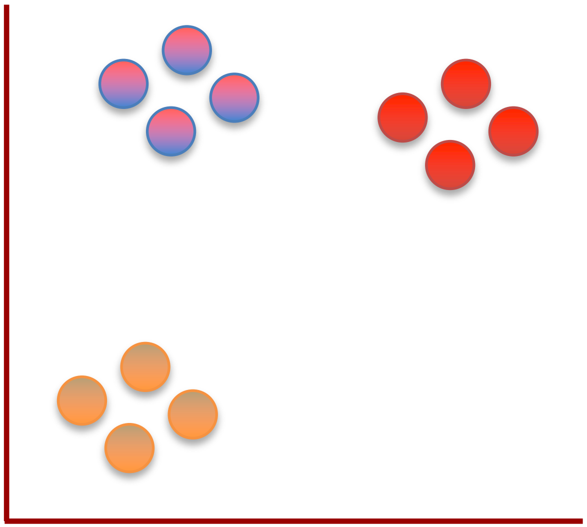


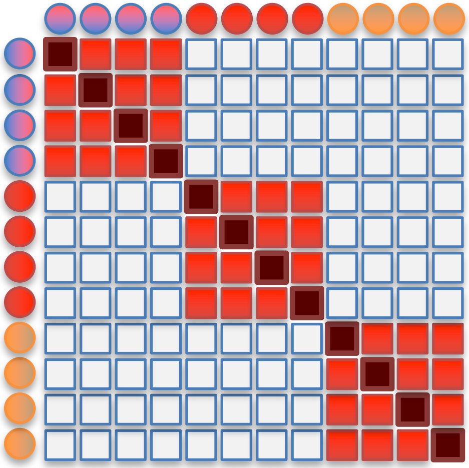
The t-distribution forces some space between the points.



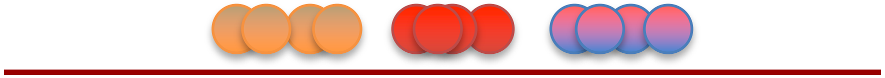
COSMIC DAWN CENTER
DAWN





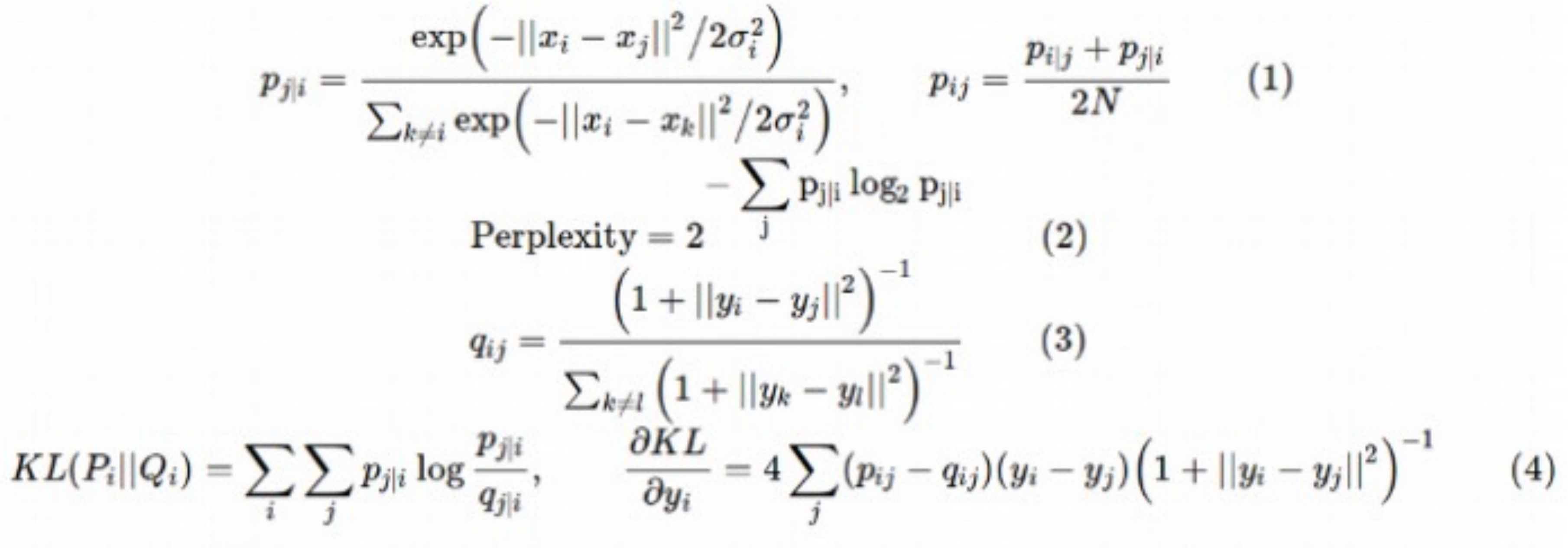


Triple Bam!!!



t-Stochastic Neighbor Embedding (t-SNE)

N. Oskolkov, towardsdatascience.com



$$p_{j|i} = \frac{\exp\left(-||x_i - x_j||^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-||x_i - x_k||^2 / 2\sigma_i^2\right)}, \qquad p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N} \qquad (1)$$

$$\text{Perplexity} = 2^{-\sum_j p_{j|i} \log_2 p_{j|i}} \qquad (2)$$

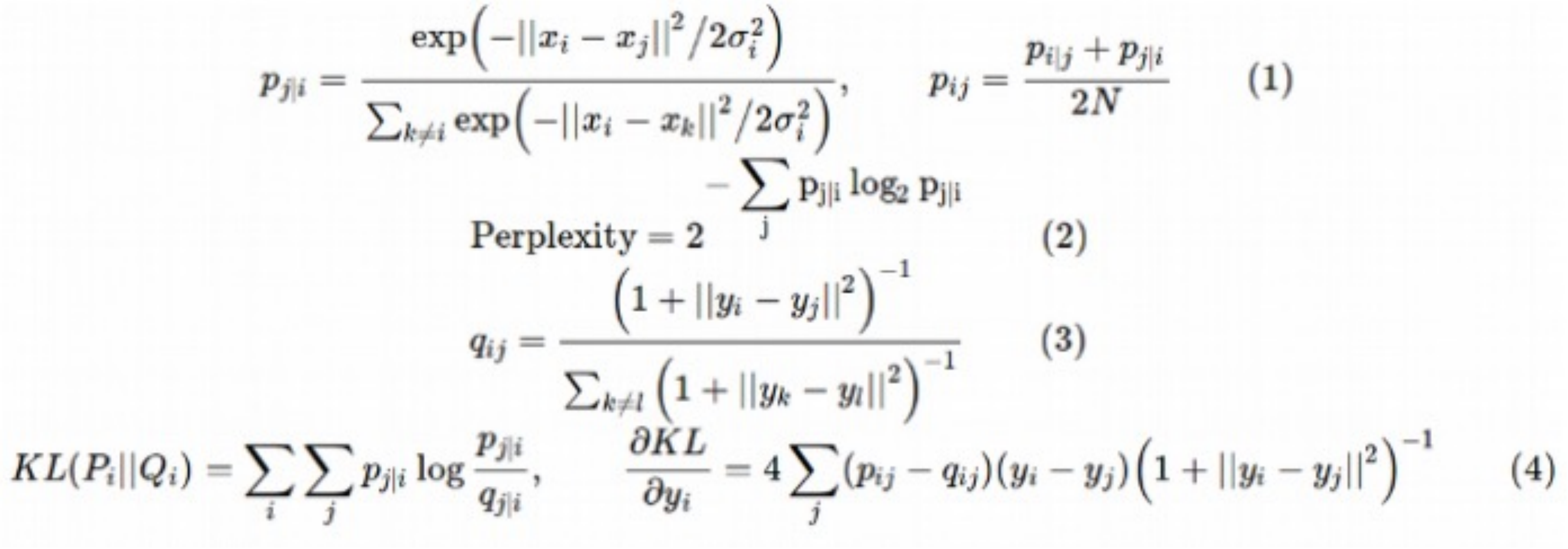
$$q_{ij} = \frac{\left(1 + ||y_i - y_j||^2\right)^{-1}}{\sum_{k \neq l} \left(1 + ||y_k - y_l||^2\right)^{-1}} \qquad (3)$$

$$KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}, \qquad \frac{\partial KL}{\partial y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j)\left(1 + ||y_i - y_j||^2\right)^{-1} \qquad (4)$$

t-Stochastic Neighbor Embedding (t-SNE)

N. Oskolkov, towardsdatascience.com

Defines distance probability as Gaussian



$$p_{j|i} = \frac{\exp\left(-||x_i - x_j||^2/2\sigma_i^2\right)}{\sum_{k\neq i}\exp\left(-||x_i - x_k||^2/2\sigma_i^2\right)}, \qquad p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N} \qquad (1)$$

$$\text{Perplexity} = 2^{-\sum_{j} p_{j|i}\log_2 p_{j|i}} \qquad (2)$$

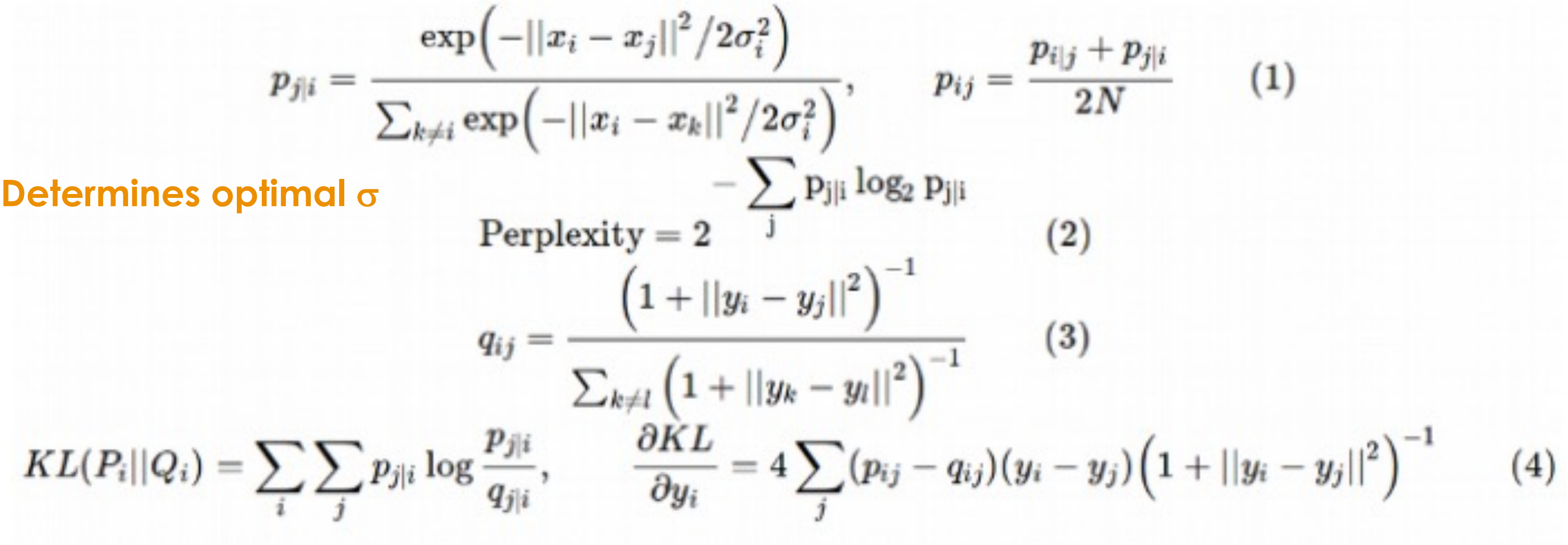
$$q_{ij} = \frac{\left(1 + ||y_i - y_j||^2\right)^{-1}}{\sum_{k\neq l}\left(1 + ||y_k - y_l||^2\right)^{-1}} \qquad (3)$$

$$KL(P_i||Q_i) = \sum_i \sum_j p_{j|i}\log\frac{p_{j|i}}{q_{j|i}}, \qquad \frac{\partial KL}{\partial y_i} = 4\sum_j (p_{ij} - q_{ij})(y_i - y_j)\left(1 + ||y_i - y_j||^2\right)^{-1} \qquad (4)$$

t-Stochastic Neighbor Embedding (t-SNE)

N. Oskolkov, towardsdatascience.com

Defines distance probability as Gaussian



$$p_{j|i} = \frac{\exp\left(-||x_i - x_j||^2/2\sigma_i^2\right)}{\sum_{k\neq i}\exp\left(-||x_i - x_k||^2/2\sigma_i^2\right)}, \qquad p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N} \qquad (1)$$

Determines optimal σ

$$\text{Perplexity} = 2^{-\sum_{j} p_{j|i} \log_2 p_{j|i}} \qquad (2)$$

$$q_{ij} = \frac{\left(1 + ||y_i - y_j||^2\right)^{-1}}{\sum_{k\neq l}\left(1 + ||y_k - y_l||^2\right)^{-1}} \qquad (3)$$

$$KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}, \qquad \frac{\partial KL}{\partial y_i} = 4\sum_j (p_{ij} - q_{ij})(y_i - y_j)\left(1 + ||y_i - y_j||^2\right)^{-1} \qquad (4)$$

COSMIC DAWN CENTER

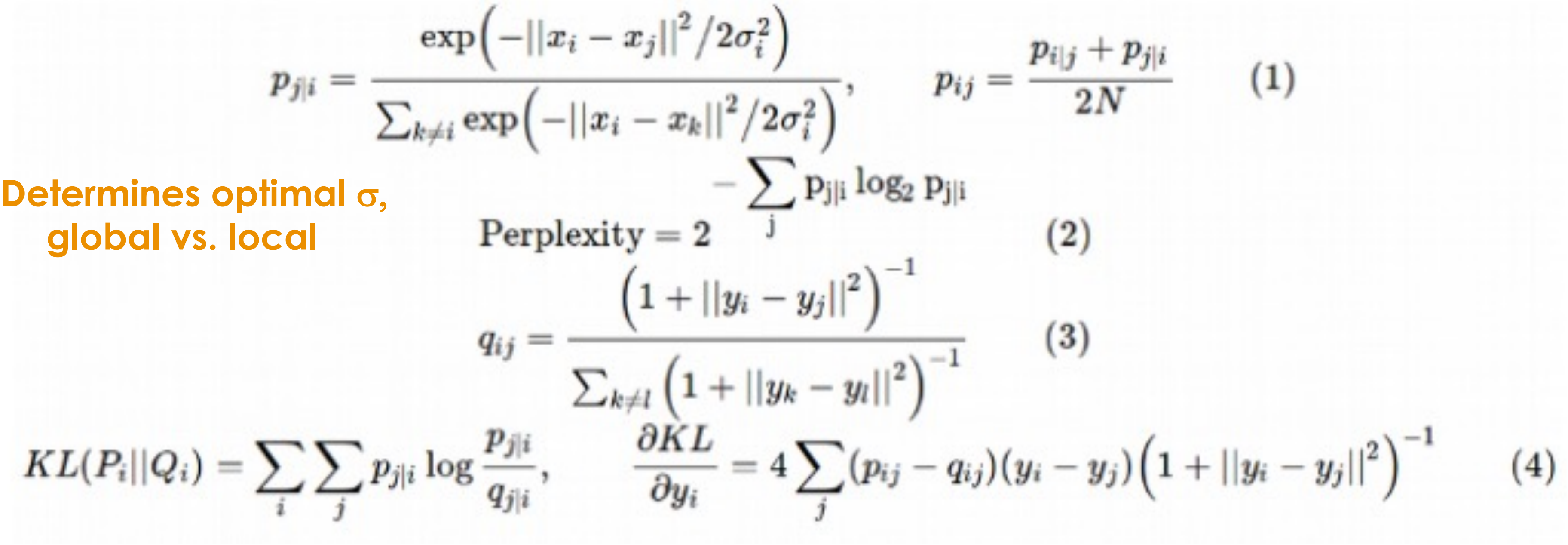
DAWN

t-Stochastic Neighbor Embedding (t-SNE)

N. Oskolkov, towardsdatascience.com

Defines distance probability as Gaussian

Determines optimal σ, global vs. local



$$p_{j|i} = \frac{\exp\left(-||x_i - x_j||^2/2\sigma_i^2\right)}{\sum_{k\neq i}\exp\left(-||x_i - x_k||^2/2\sigma_i^2\right)}, \qquad p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N} \qquad (1)$$

$$\text{Perplexity} = 2^{-\sum_{\text{j}} \text{p}_{\text{j|i}} \log_2 \text{p}_{\text{j|i}}} \qquad (2)$$

$$q_{ij} = \frac{\left(1 + ||y_i - y_j||^2\right)^{-1}}{\sum_{k\neq l}\left(1 + ||y_k - y_l||^2\right)^{-1}} \qquad (3)$$

$$KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}, \qquad \frac{\partial KL}{\partial y_i} = 4\sum_j (p_{ij} - q_{ij})(y_i - y_j)\left(1 + ||y_i - y_j||^2\right)^{-1} \qquad (4)$$

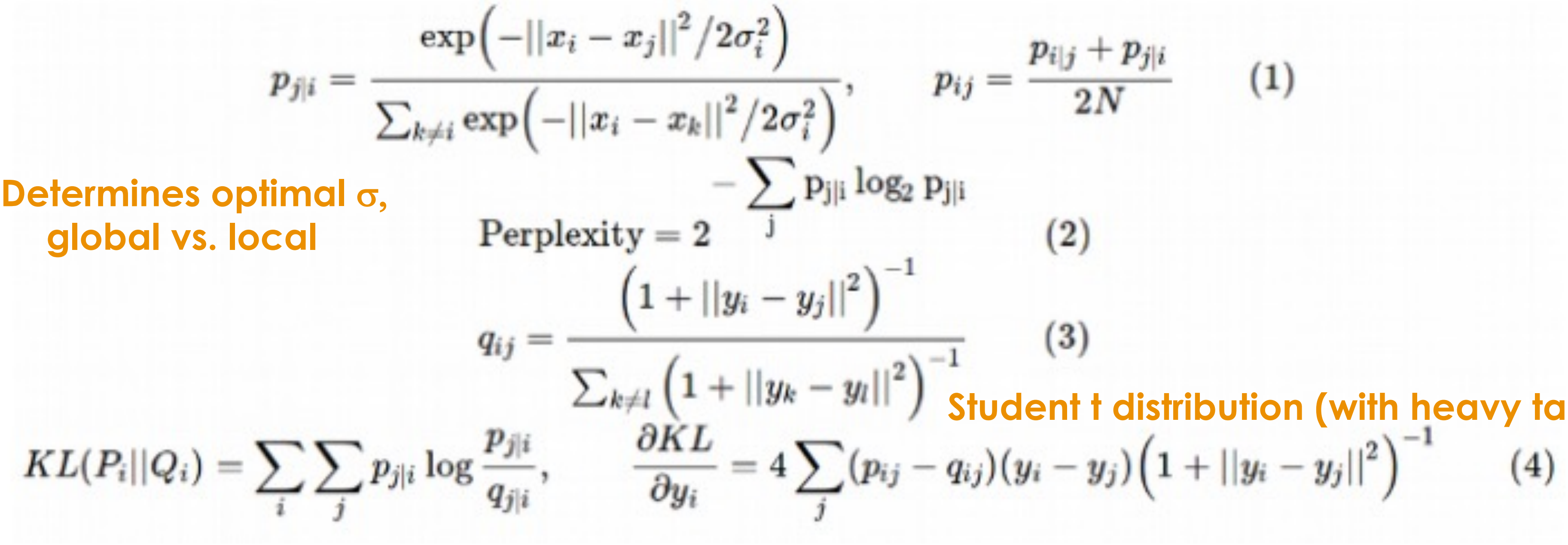
COSMIC DAWN CENTER

DAWN

t-Stochastic Neighbor Embedding (t-SNE)

N. Oskolkov, towardsdatascience.com

Defines distance probability as Gaussian



$$p_{j|i} = \frac{\exp\left(-||x_i - x_j||^2/2\sigma_i^2\right)}{\sum_{k\neq i}\exp\left(-||x_i - x_k||^2/2\sigma_i^2\right)}, \qquad p_{ij} = \frac{p_{i|j}+p_{j|i}}{2N} \qquad (1)$$

Determines optimal σ, global vs. local

$$\text{Perplexity} = 2^{-\sum_{\mathrm{j}} \mathrm{p}_{\mathrm{j|i}} \log_2 \mathrm{p}_{\mathrm{j|i}}} \qquad (2)$$

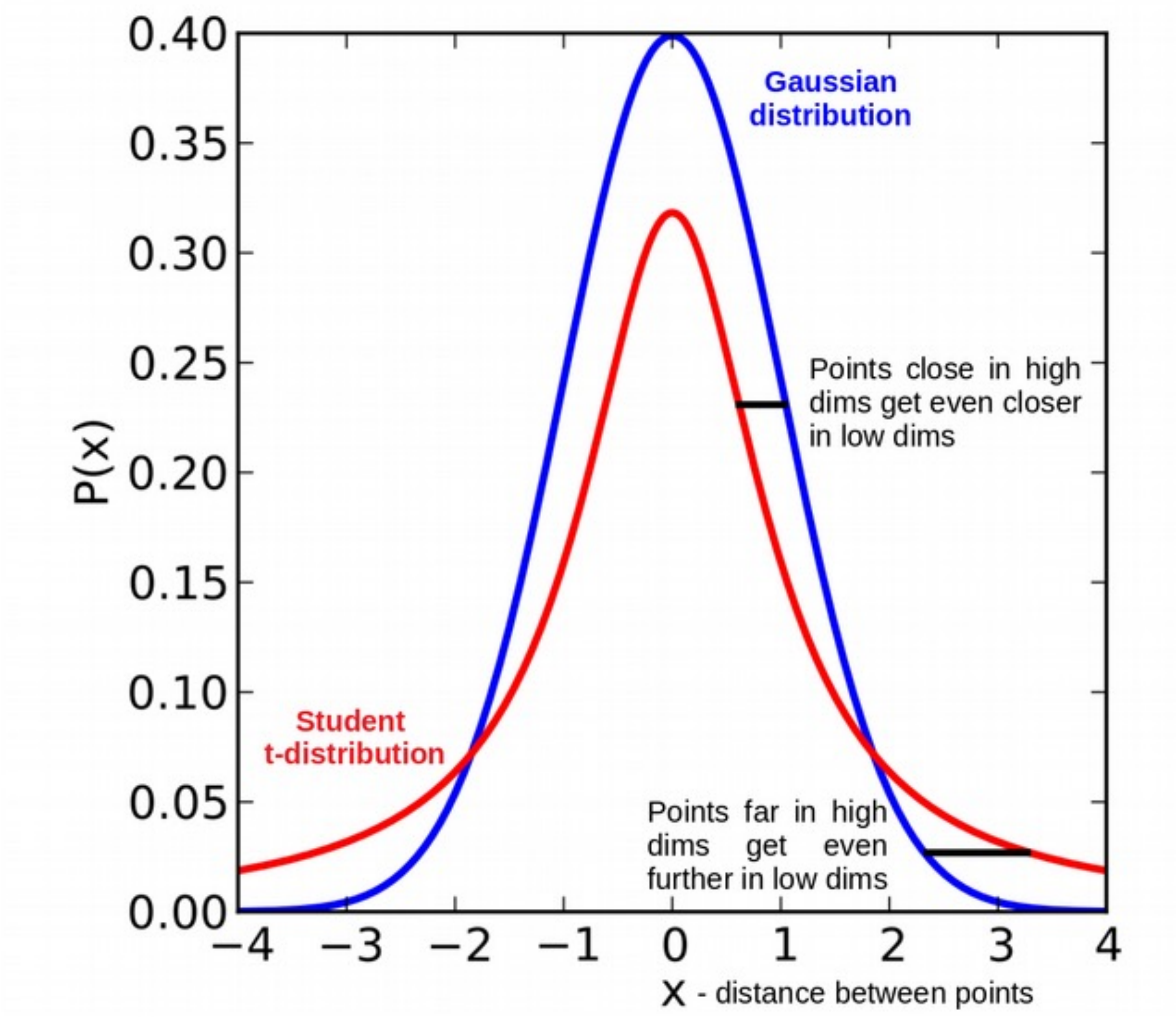
$$q_{ij} = \frac{\left(1 + ||y_i - y_j||^2\right)^{-1}}{\sum_{k\neq l}\left(1 + ||y_k - y_l||^2\right)^{-1}} \qquad (3)$$

Student t distribution (with heavy tails!)

$$KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}, \qquad \frac{\partial KL}{\partial y_i} = 4\sum_j (p_{ij} - q_{ij})(y_i - y_j)\left(1 + ||y_i - y_j||^2\right)^{-1} \qquad (4)$$

t-Stochastic Neighbor Embedding (t-SNE)

N. Oskolkov, towardsdatascience.com

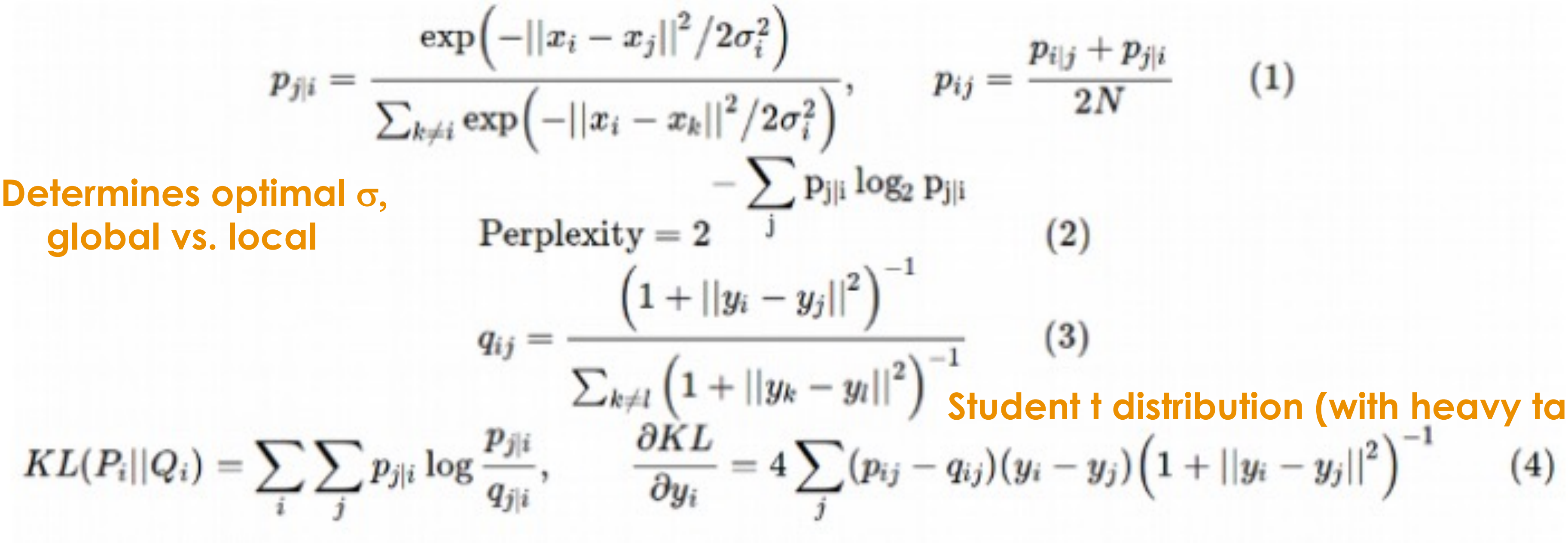


COSMIC DAWN CENTER

DAWN

t-Stochastic Neighbor Embedding (t-SNE)

N. Oskolkov, towardsdatascience.com



Defines distance probability as Gaussian

$$p_{j|i} = \frac{\exp\left(-||x_i - x_j||^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-||x_i - x_k||^2 / 2\sigma_i^2\right)}, \qquad p_{ij} = \frac{p_{i|j} + p_{j|i}}{2N} \qquad (1)$$

Determines optimal σ, global vs. local

$$\text{Perplexity} = 2^{-\sum_{\mathrm{j}} \mathrm{p}_{\mathrm{j|i}} \log_2 \mathrm{p}_{\mathrm{j|i}}} \qquad (2)$$

$$q_{ij} = \frac{\left(1 + ||y_i - y_j||^2\right)^{-1}}{\sum_{k \neq l} \left(1 + ||y_k - y_l||^2\right)^{-1}} \qquad (3)$$

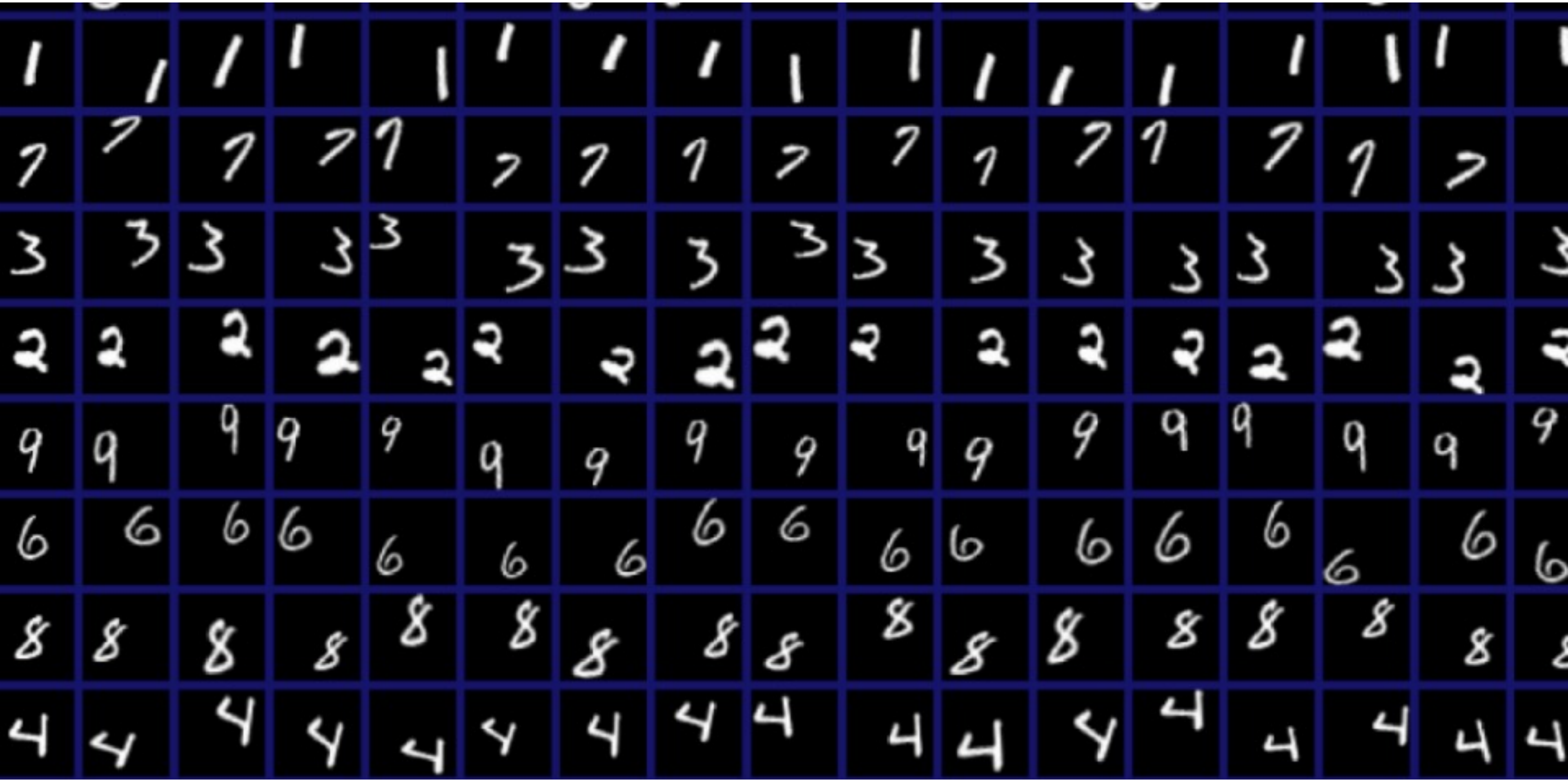
Student t distribution (with heavy tails!)

$$KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}, \qquad \frac{\partial KL}{\partial y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j)\left(1 + ||y_i - y_j||^2\right)^{-1} \qquad (4)$$

Defines loss function for gradient descent

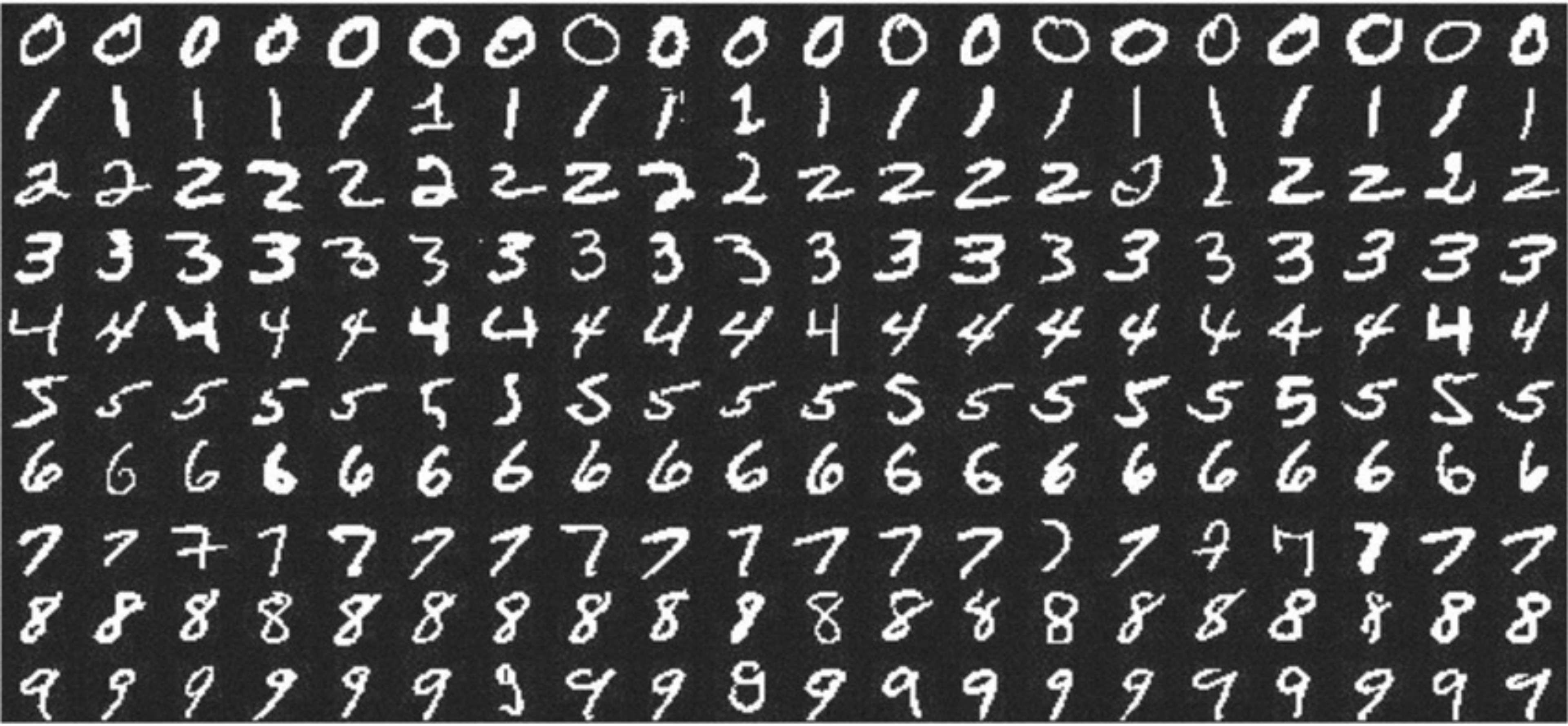
Problem 1: Preprocessing/Standardization

MNIST, raw data



Problem 2 (or 1.5): Distance Metrics

MNIST



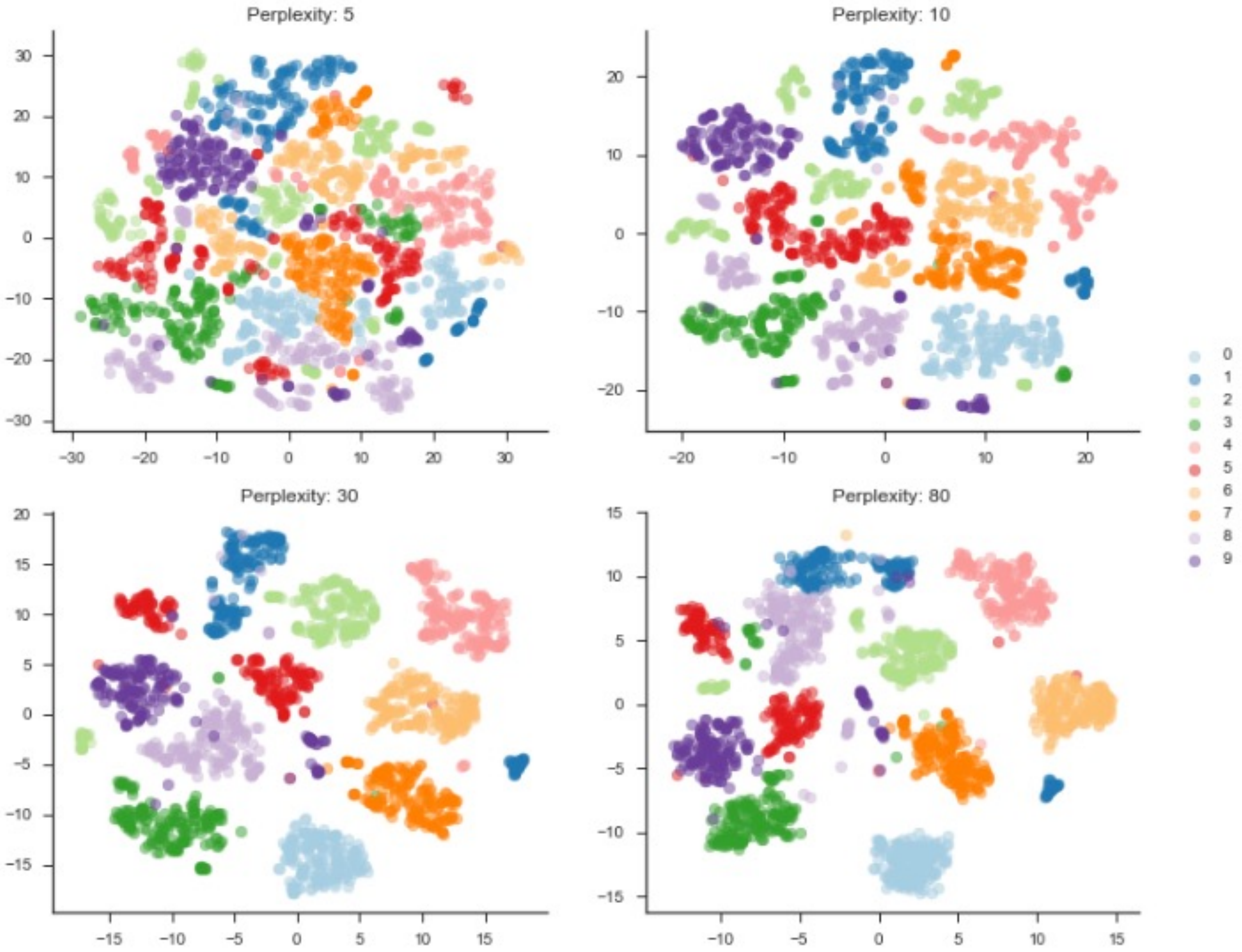
COSMIC DAWN CENTER

DAWN

Wednesday May 3, 2023

Problem 3: Hyperparameters

Cortale 2017



Can we do better?

Can we do better?

- t-SNE is slow and scales poorly

Can we do better?

- t-SNE is slow and scales poorly

- Difficult tradeoff between global and local structure

Can we do better?

- t-SNE is slow and scales poorly
- Difficult tradeoff between global and local structure
- Lack of mathematical formalism

Can we do better?

- t-SNE is slow and scales poorly

Barnes-Hut approximation!

- Difficult tradeoff between global and local structure

- Lack of mathematical formalism

Can we do better?

- t-SNE is slow and scales poorly

~~Barnes-Hut approximation!~~

Barnes-Hut approximation ONLY IN 2D OR 3D!

- Difficult tradeoff between global and local structure

- Lack of mathematical formalism

Can we do better?

- t-SNE is slow and scales poorly

~~Barnes-Hut approximation!~~

Barnes-Hut approximation ONLY IN 2D OR 3D!

- Difficult tradeoff between global and local structure

Perplexity!

- Lack of mathematical formalism

Can we do better?

- t-SNE is slow and scales poorly

~~Barnes-Hut approximation!~~

Barnes-Hut approximation ONLY IN 2D OR 3D!

- Difficult tradeoff between global and local structure

Perplexity! (well, if you have to pick)

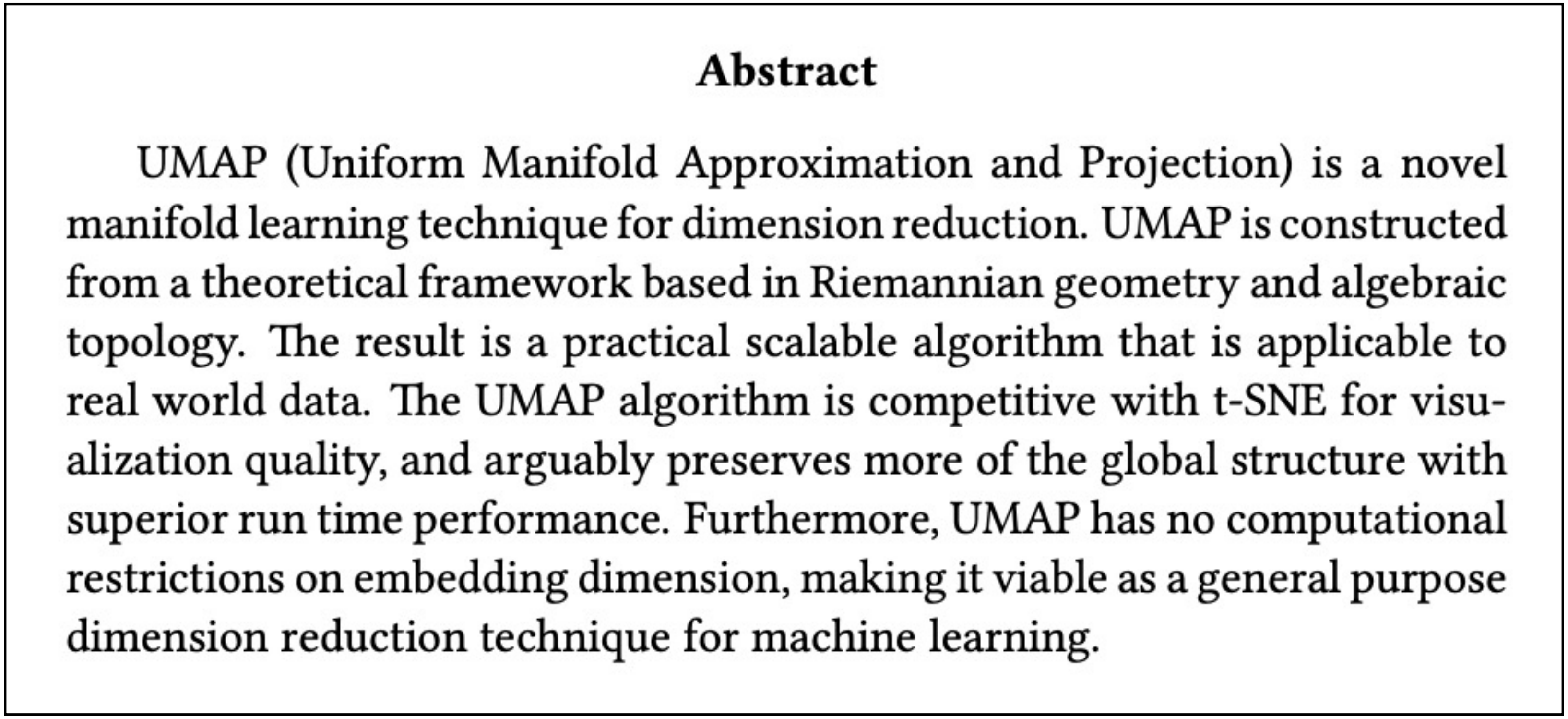
- Lack of mathematical formalism

COSMIC DAWN CENTER

DAWN

UMAP

UMAP builds on using **Riemannian manifolds!** Within differential geometry, this allows the definition of angles, hyper-area, and curvature in high dimensionality.



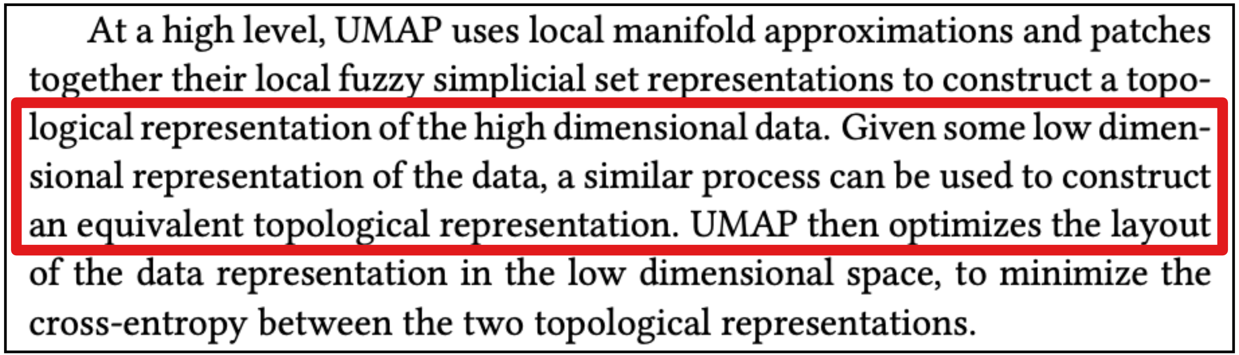
Abstract

UMAP (Uniform Manifold Approximation and Projection) is a novel manifold learning technique for dimension reduction. UMAP is constructed from a theoretical framework based in Riemannian geometry and algebraic topology. The result is a practical scalable algorithm that is applicable to real world data. The UMAP algorithm is competitive with t-SNE for visualization quality, and arguably preserves more of the global structure with superior run time performance. Furthermore, UMAP has no computational restrictions on embedding dimension, making it viable as a general purpose dimension reduction technique for machine learning.

UMAP paper, arXiv 1802.03426, Sep. 2020

UMAP

As in the t-SNE case, UMAP tries to find a metric in both the original (large) space X, and the lower dimension output space Y, which can be (topologically) matched:



At a high level, UMAP uses local manifold approximations and patches together their local fuzzy simplicial set representations to construct a topological representation of the high dimensional data. Given some low dimensional representation of the data, a similar process can be used to construct an equivalent topological representation. UMAP then optimizes the layout of the data representation in the low dimensional space, to minimize the cross-entropy between the two topological representations.

UMAP paper, arXiv 1802.03426, Sep. 2020

However, the metrics in X and Y used by UMAP and t-SNE differ:

For t-SNE these metrics are as follows:

$$v_{j|i} = \exp(-\left\|x_i - x_j\right\|_2^2 / 2\sigma_i^2)$$

$$w_{ij} = \left(1 + \left\|y_i - y_j\right\|_2^2\right)^{-1}$$

For UMAP they are:

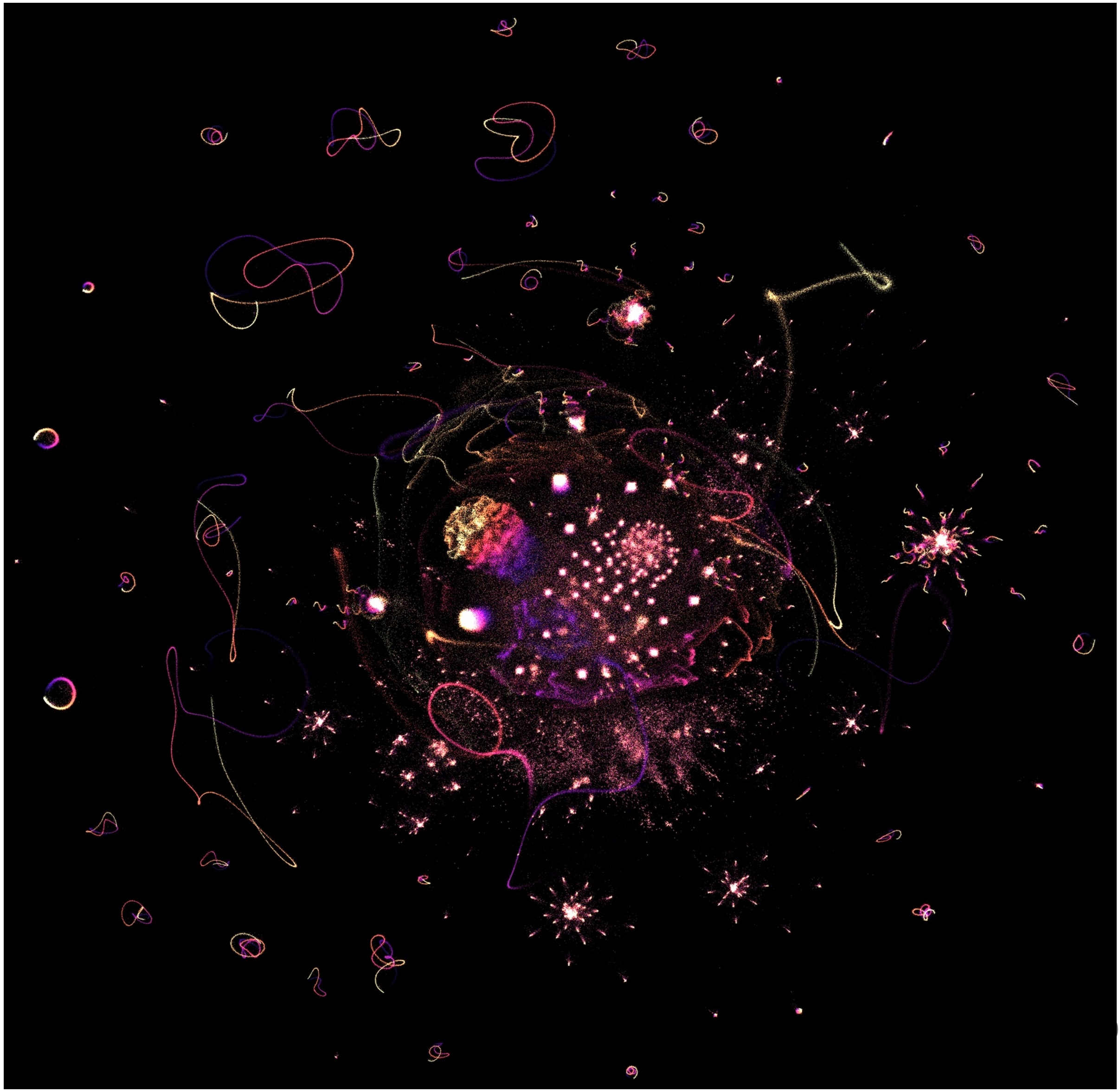
$$v_{j|i} = \exp[(-d\,(x_i, x_j) - \rho_i)/\sigma_i]$$

$$w_{ij} = \left(1 + a\left\|y_i - y_j\right\|_2^{2b}\right)^{-1}$$

First million integers in UMAP

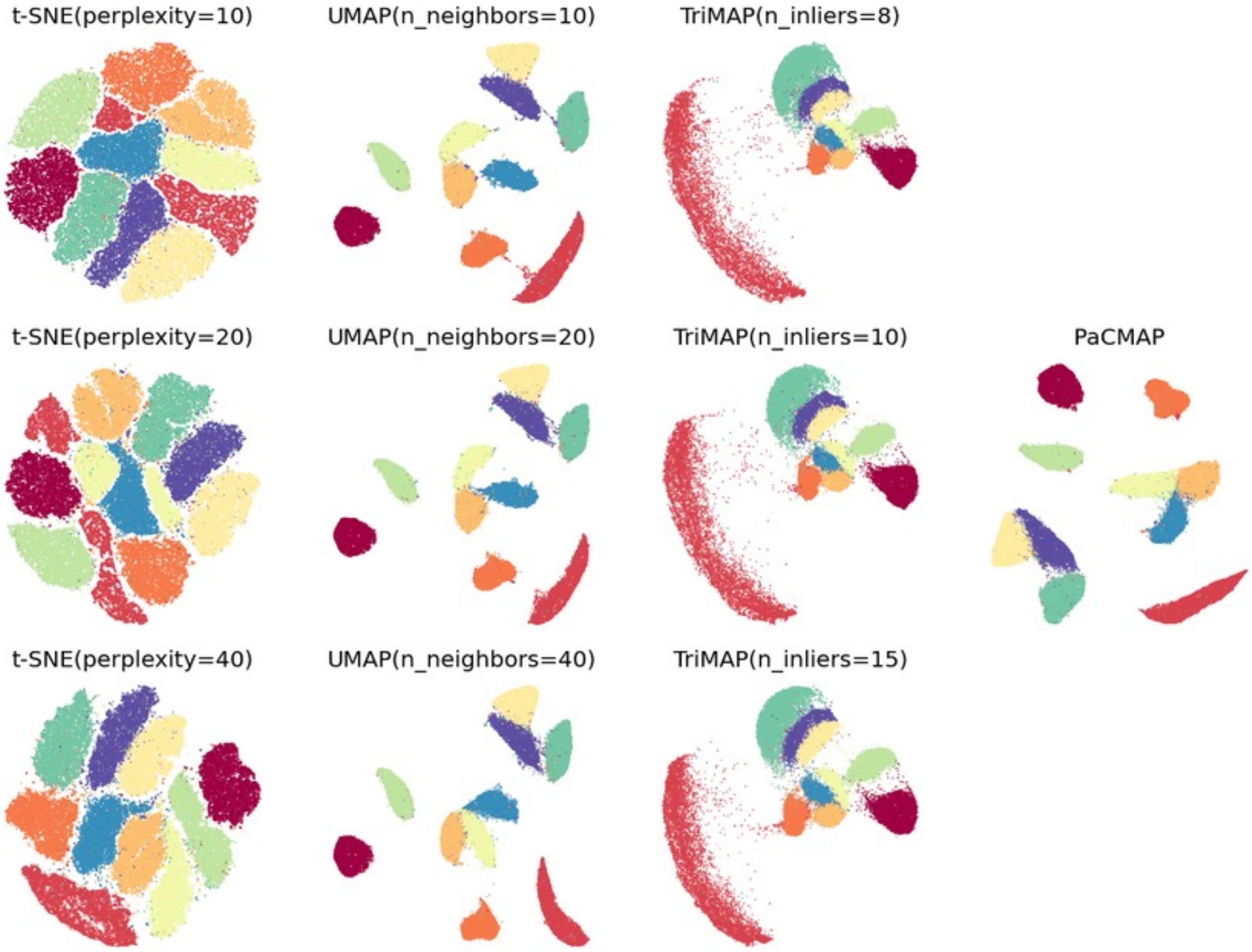
Prime-factorizing the first million integers, and drawing them (artfully) gives the following image.

If nothing else, it's a great piece of ML art.



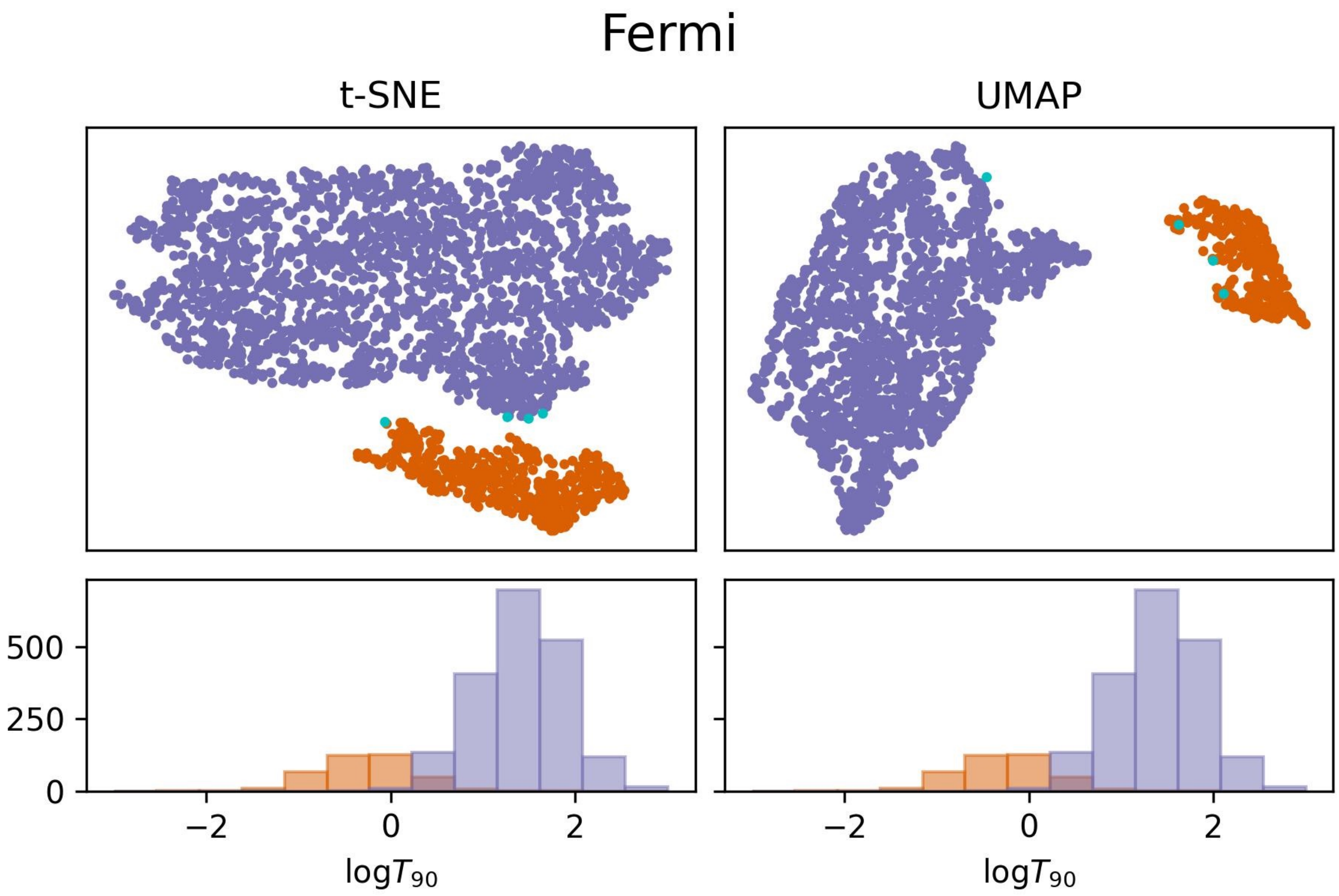
Group "similar" things together

Wang et al. 2020



t-SNE vs. UMAP

Kragh Jespersen et al. 2020

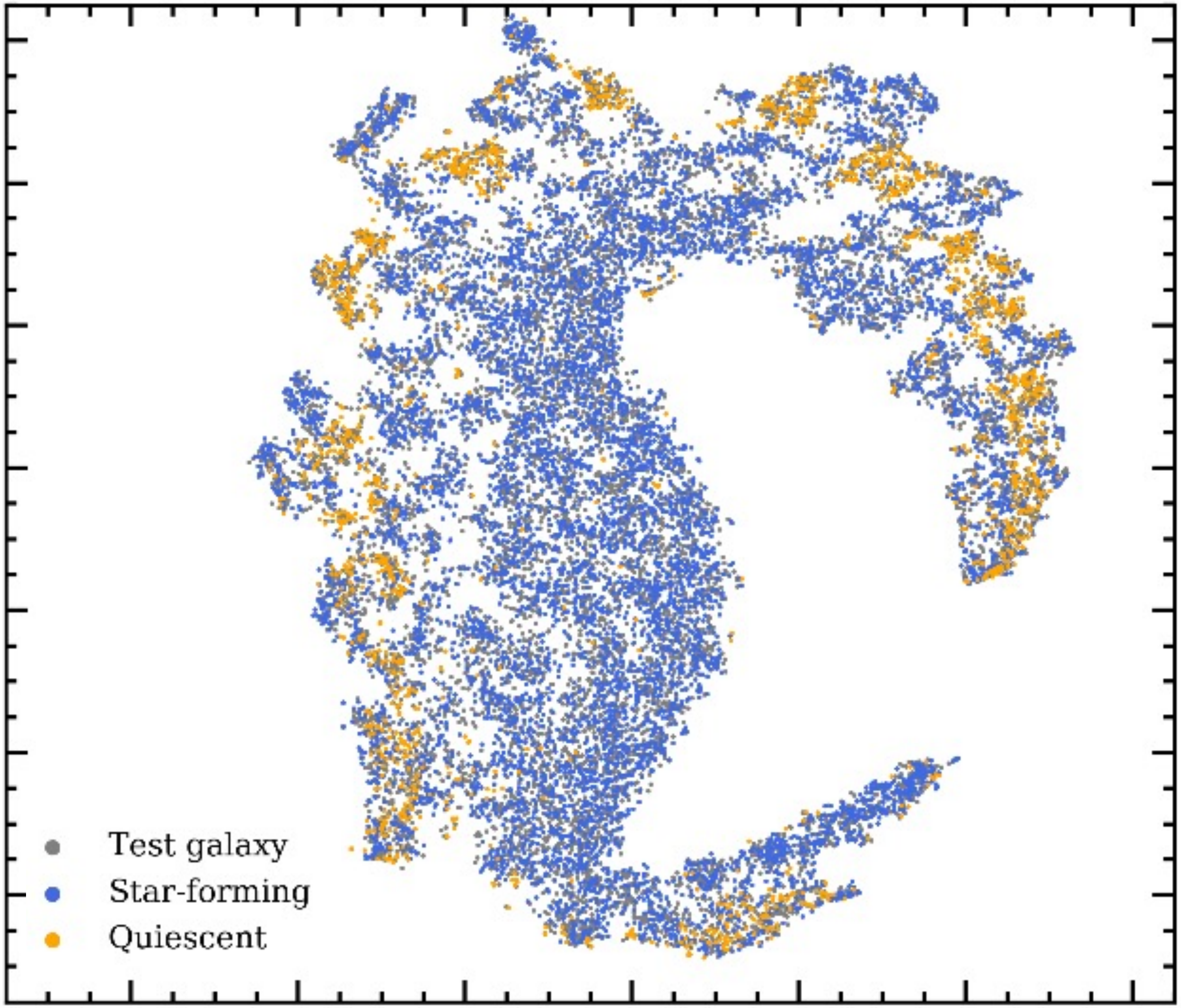


COSMIC DAWN CENTER
DAWN

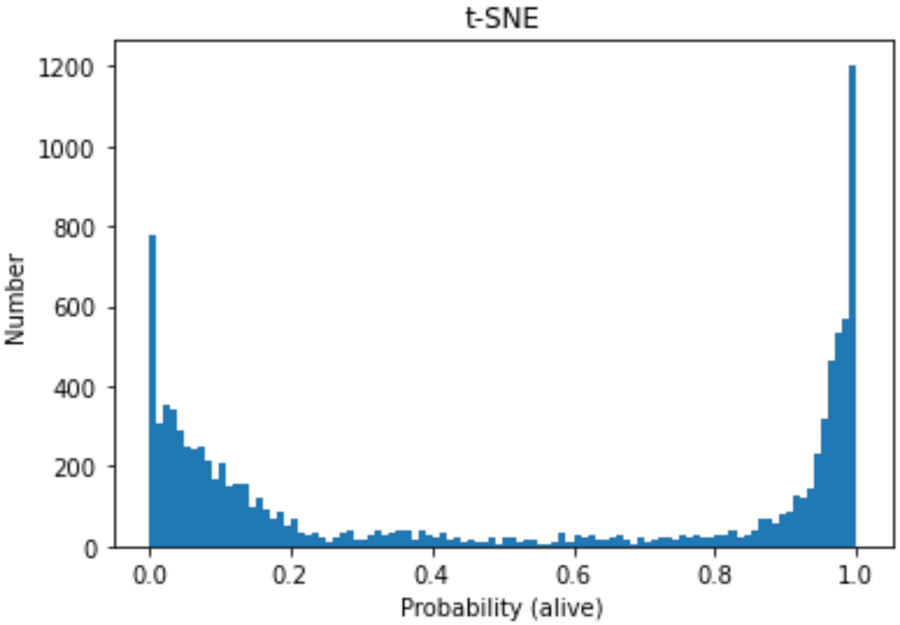
Wednesday May 3, 2023

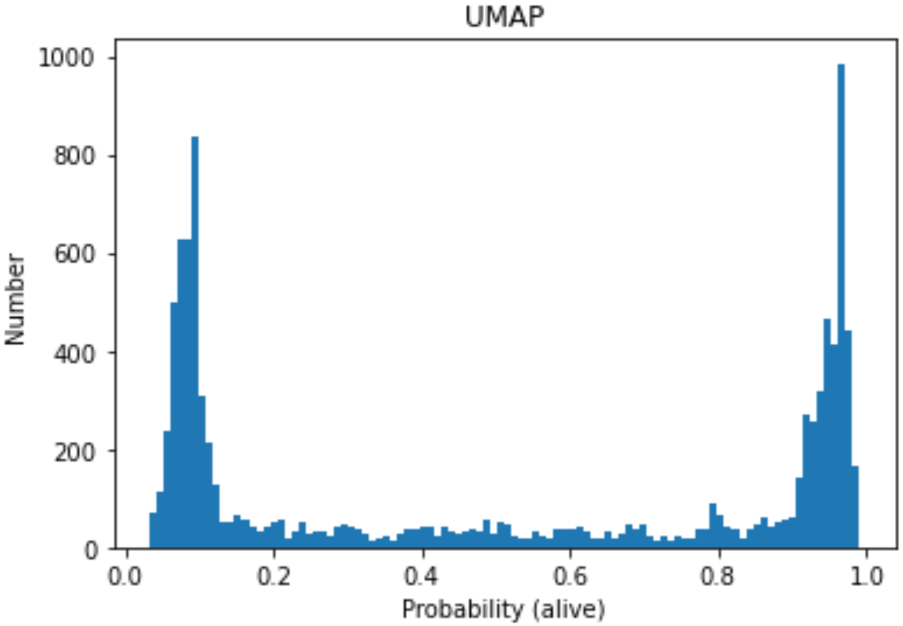
Approach 2: Finding "dead" galaxies

COSMOS2015 catalog, objects at z≈1

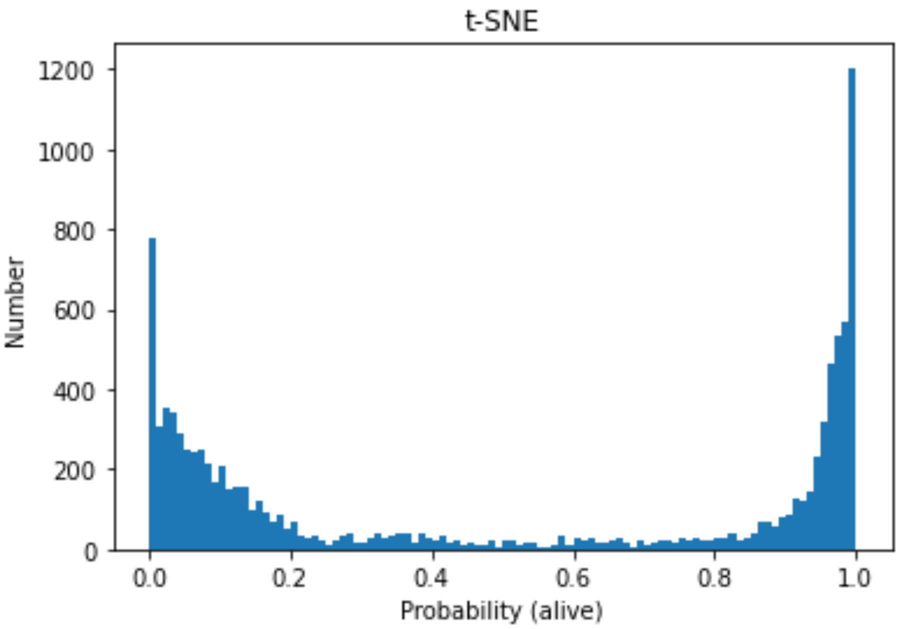


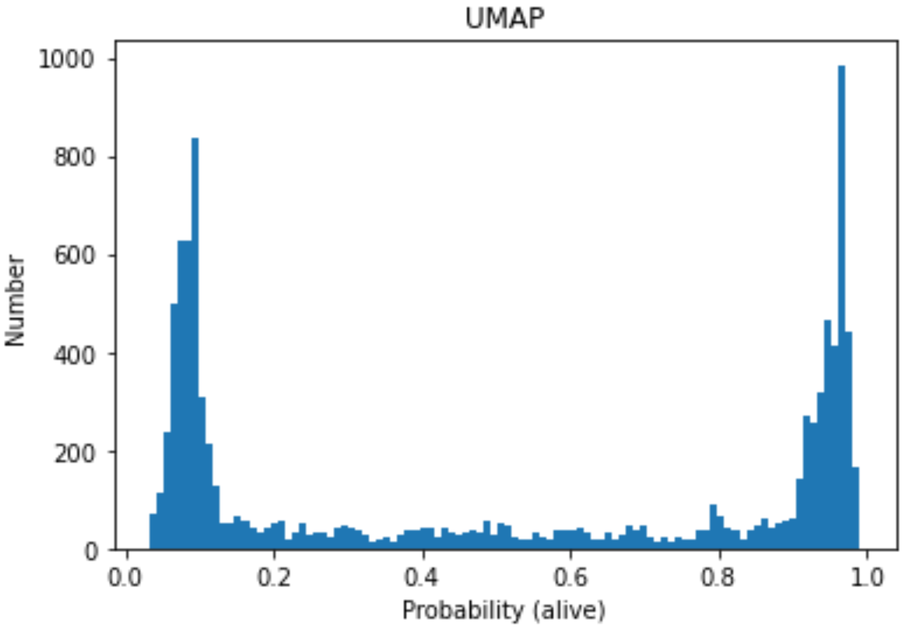
Which is a better separation?

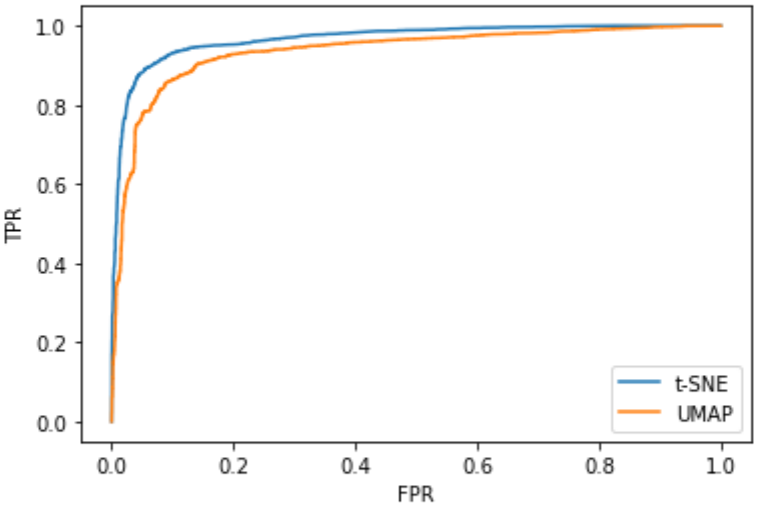




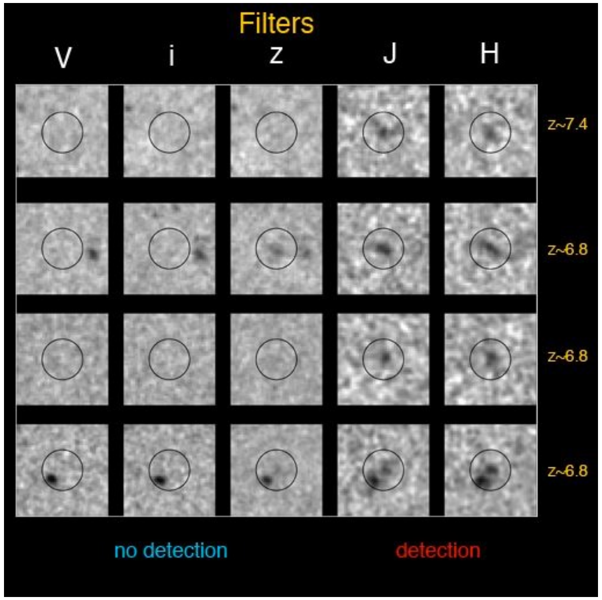
Which is a better separation?







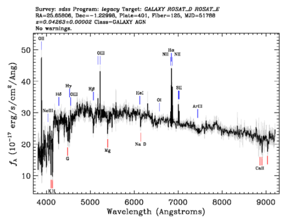
Our goal:



"V band"
"i band"
"z band"
"J band"
"H band"

etc.

Choose a few good examples

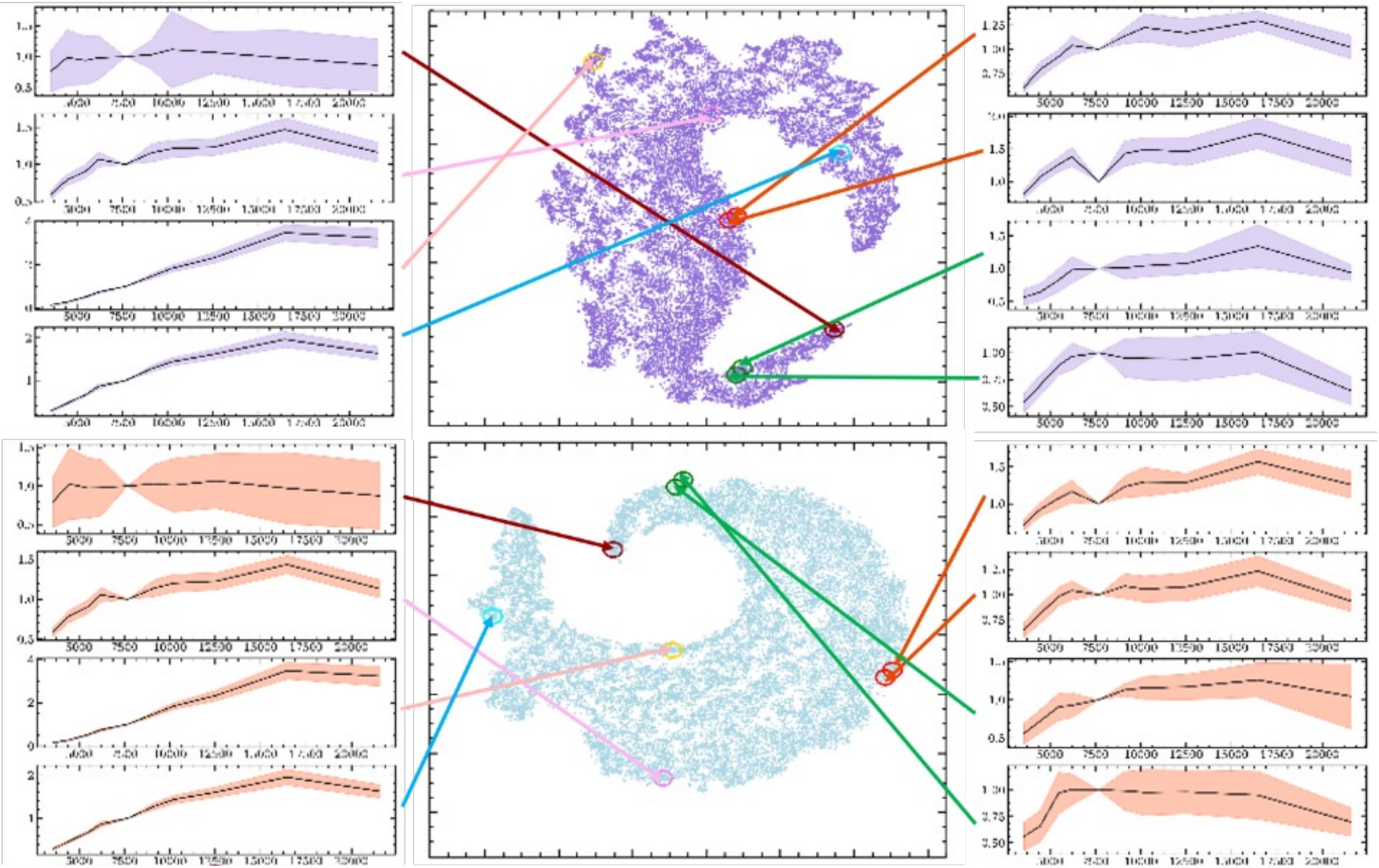


(Stellar) Mass
Star formation rate
Star formation history
Distance/Redshift
Age

etc.

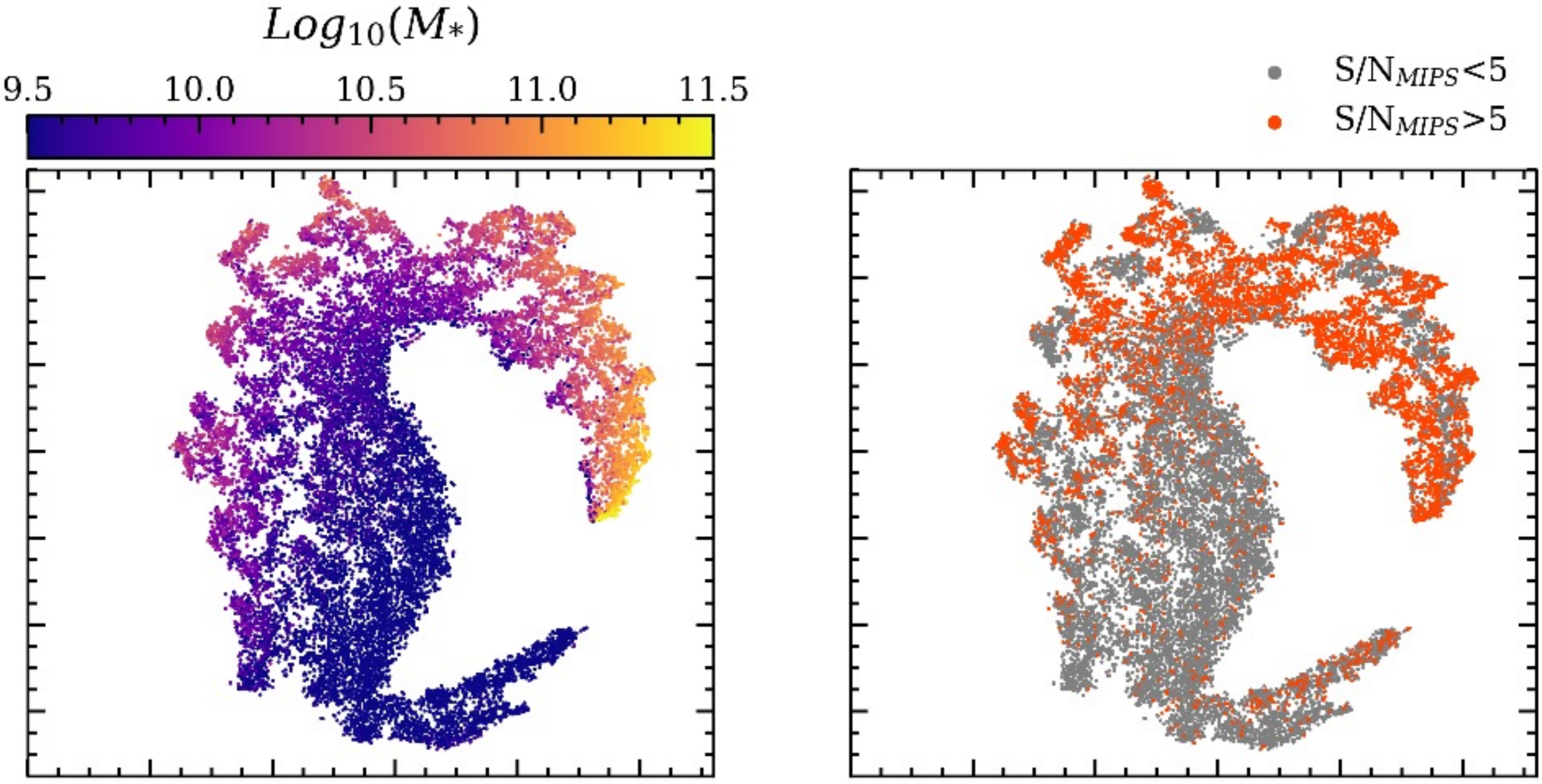
Approach 2: Similar galaxies are nearby

WARNING: positions are neither fixed nor meaningful. Topology is meaningful.



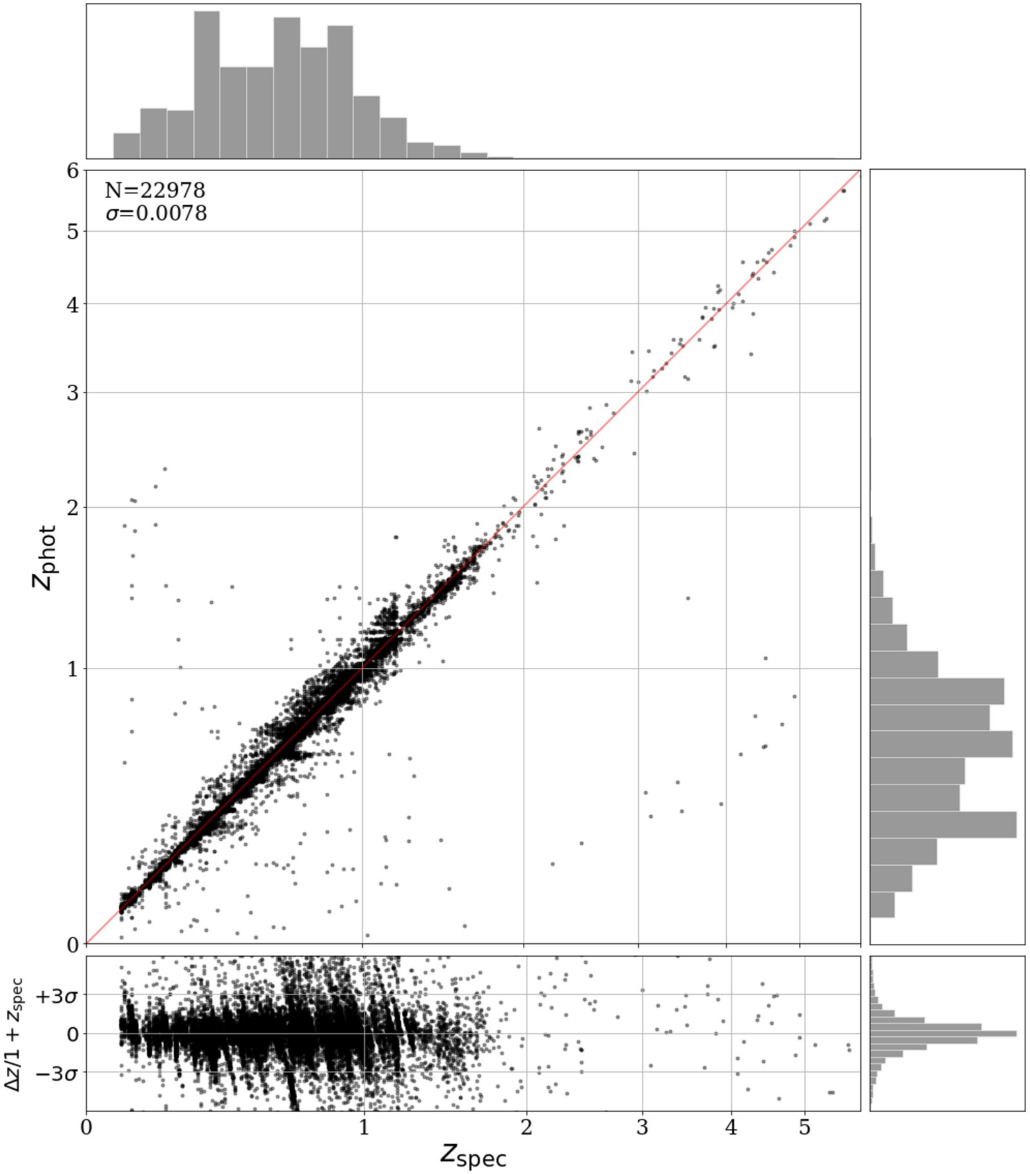
Approach 2: Arranging by photometry also calculates other useful things!

COSMOS2015 catalog, objects at z≈1



Photometric Redshifts Usually Work…

Hovis-Afflerbach, Steinhardt et al. 2020

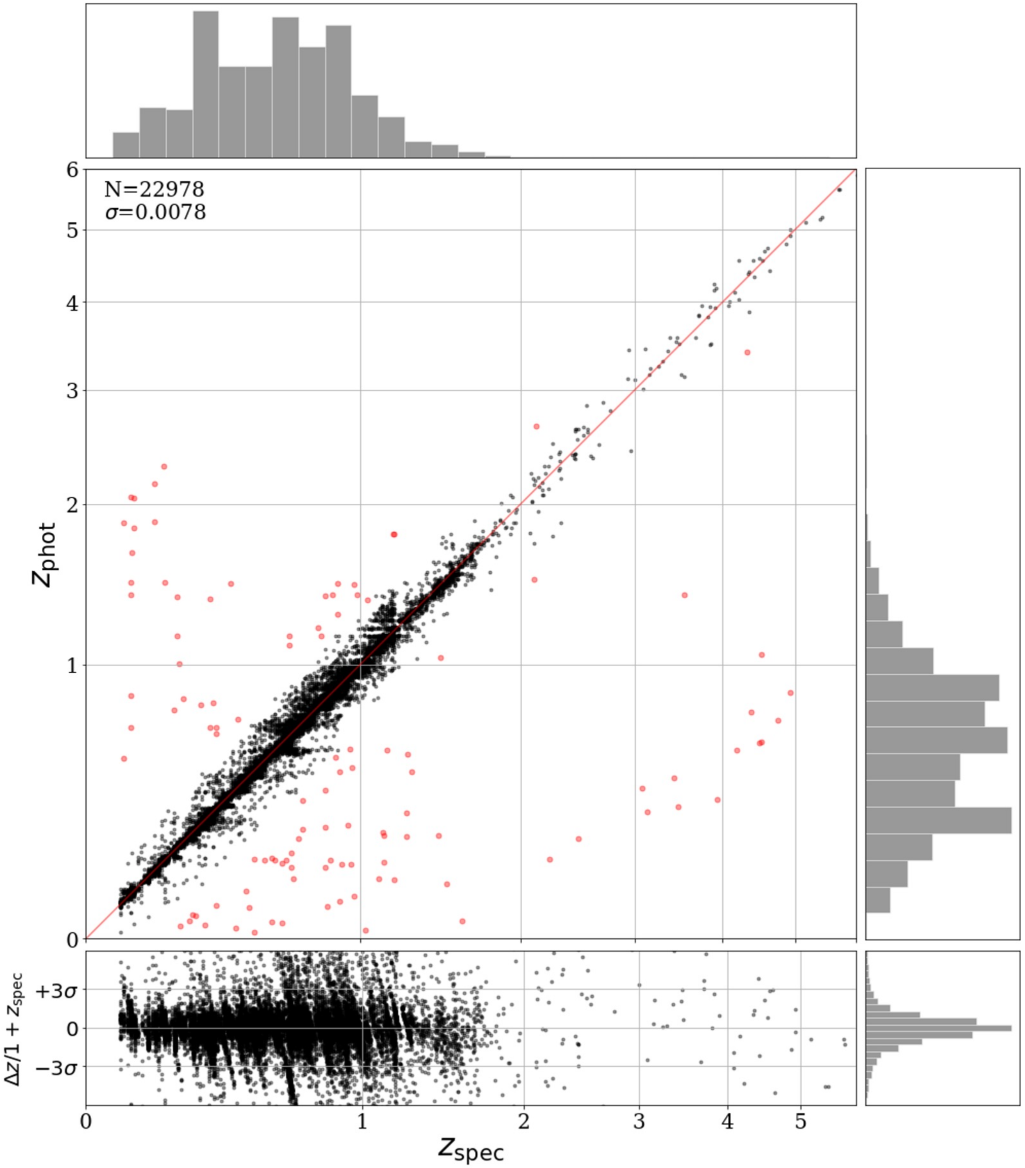


COSMIC DAWN CENTER

DAWN

Photometric Redshifts Usually Work…

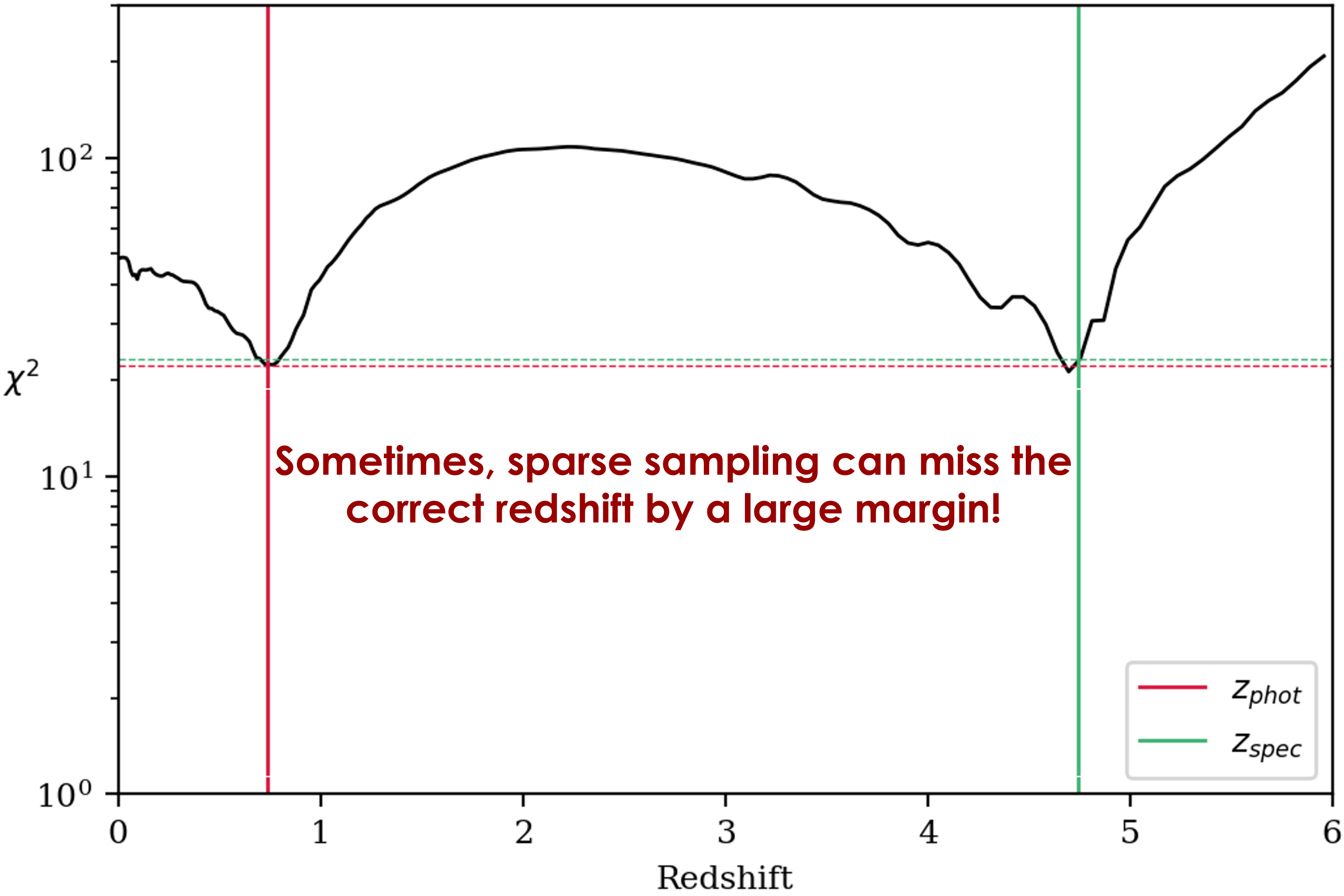
Hovis-Afflerbach, Steinhardt et al. 2020



COSMIC DAWN CENTER

DAWN

Wednesday May 3, 2023



Sometimes, sparse sampling can miss the correct redshift by a large margin!

…But Not Always!

About 5% of objects fail template fitting[1]

Can we identify these objects without comparing with spectroscopy?

Can we fix these objects and determine the correct properties?

[1] Hildebrandt et al. 2010:
12 photo-z methods tested on objects with 18 bands, between 4.9% and 29% of objects had dz/(1+z) > 0.15

~~Two~~ Three Fundamental Assumptions for Photometry

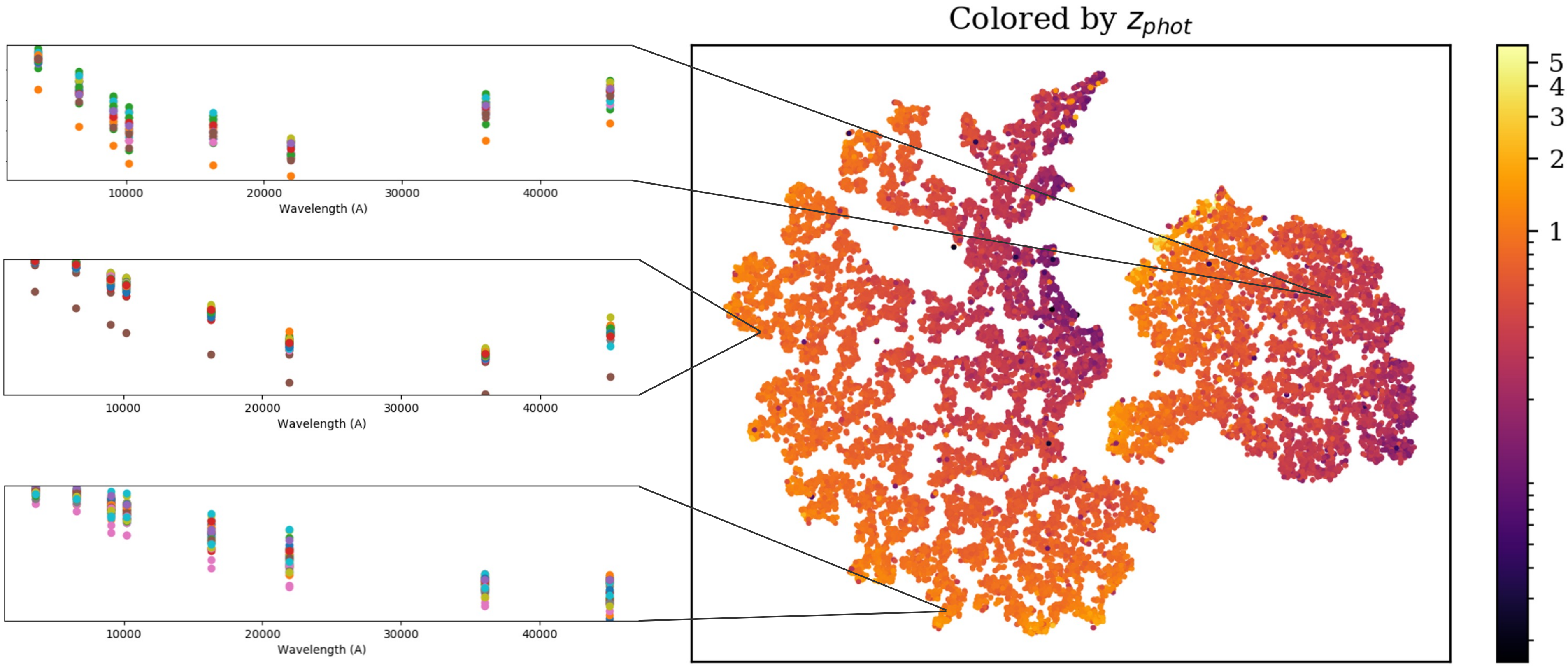
1. If an object is sufficiently well-measured, there is a surjective (one-to-one or many-to-one, but not one-to-many) mapping from photometric fluxes to astrophysical properties.

2. Objects with sufficiently similar photometry should be mapped to similar astrophysical properties.

3. We can map objects from the full, n-dimensional space with all bands to a smaller one with many neighbors, and the other two assumptions will continue to hold.

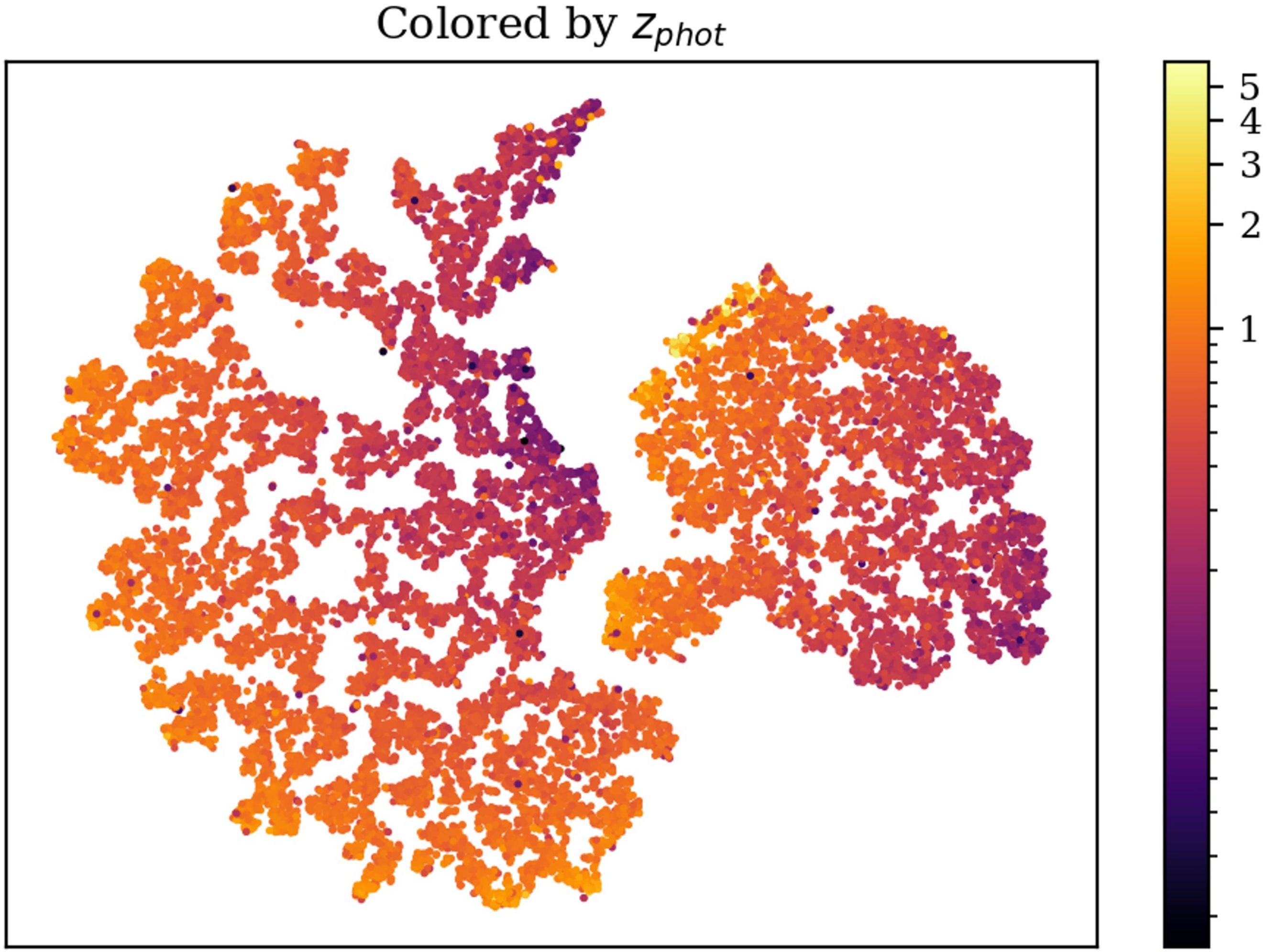
t-SNE Map for COSMOS/Spec-z Dataset

Hovis-Afflerbach, Steinhardt et al. 2020



t-SNE Map for COSMOS/Spec-z Dataset

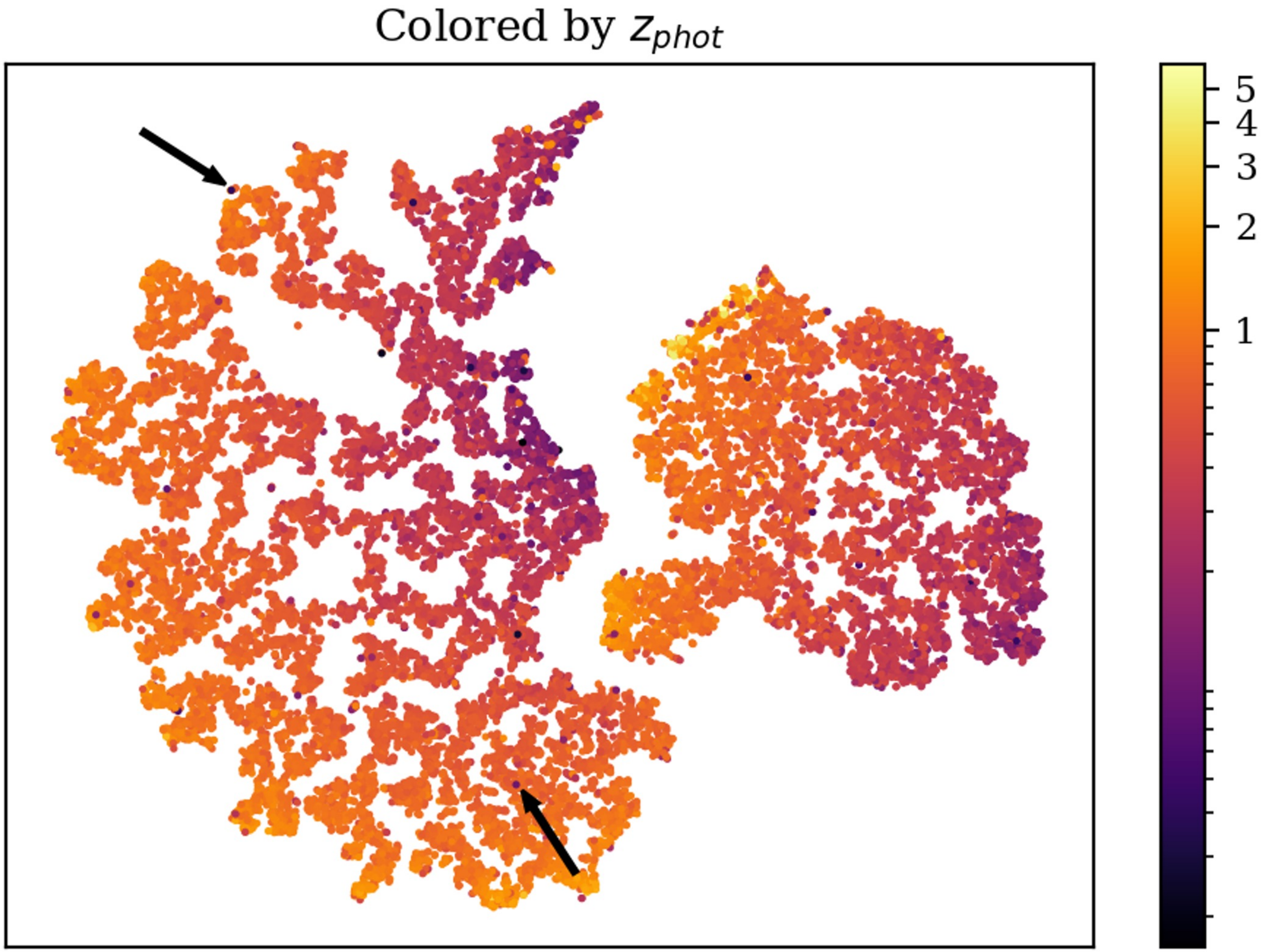
Hovis-Afflerbach, Steinhardt et al. 2020



COSMIC DAWN CENTER
DAWN

Task: Make errors “look” wrong

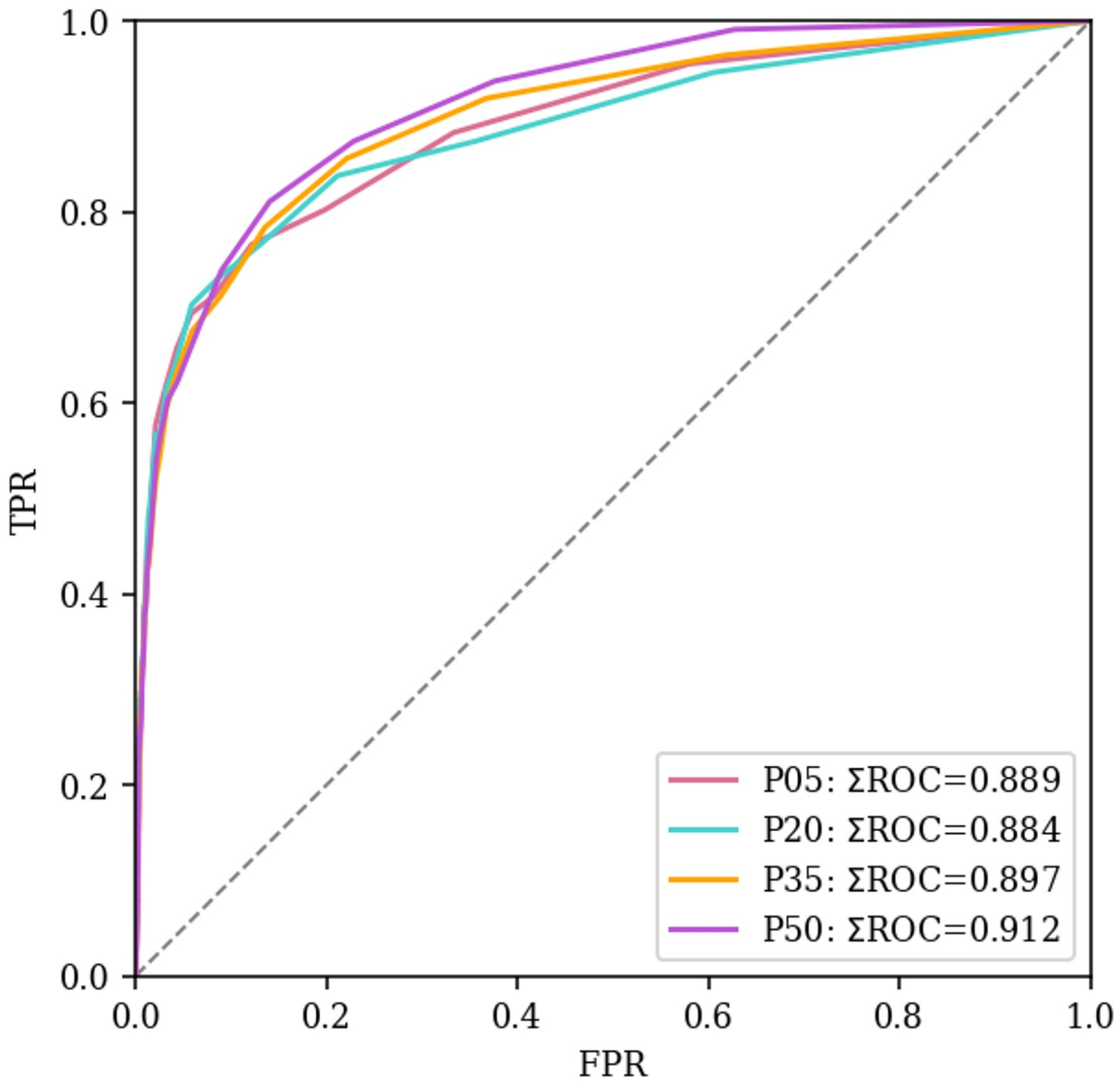
Hovis-Afflerbach, Steinhardt et al. 2020

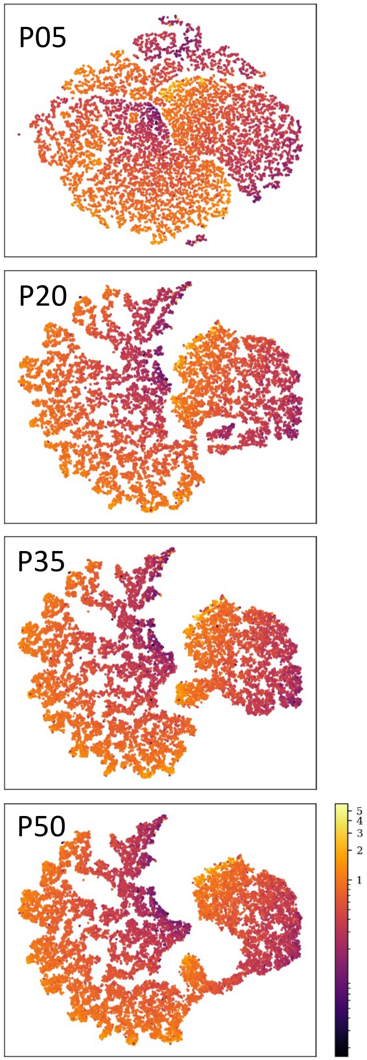


COSMIC DAWN CENTER
DAWN

ROC Curve for Catastrophic Errors

Hovis-Afflerbach, Steinhardt et al. 2020


TPR
FPR
P05: ΣROC=0.889
P20: ΣROC=0.884
P35: ΣROC=0.897
P50: ΣROC=0.912


P05
P20
P35
P50
5
4
3
2
1