SNS COLLEGE OF ENGINEERING
(Autonomous)
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

Artificial Intelligence & Machine Learning

The Curse of Dimensionality

Prepared by,
P.Ramya
Assistant Professor/ECE
SNS College of Engineering

The Curse of Dimensionality

- When confronted with a ton of data, we can use dimensionality reduction algorithms to make the data “get to the point”.

- Why we need dimensionality reduction algorithms in the first place — **The Curse of Dimensionality.**

When is Data High Dimensional and Why Might That Be a Problem?

- The Curse of Dimensionality sounds like something straight out of a pirate movie but **what it really refers to is when your data has too many features.**

Contd…

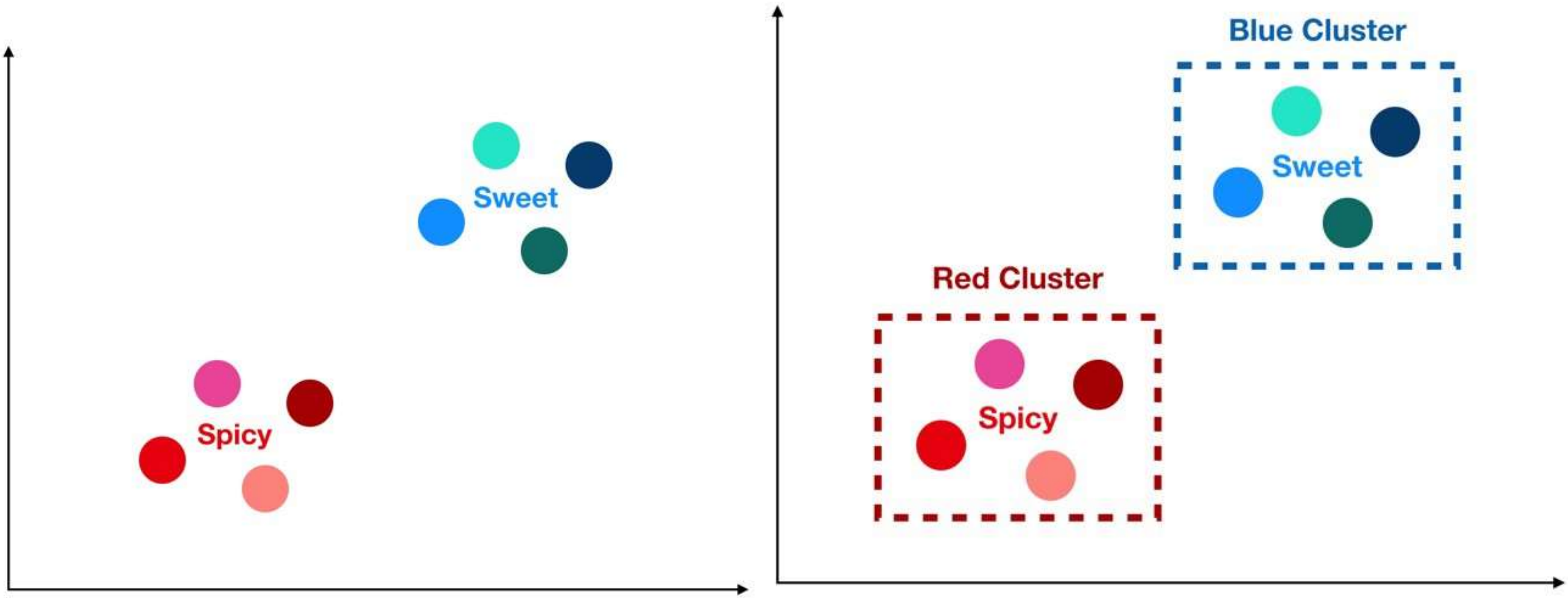
In today's big data world it can also refer to several other potential issues that arise when your data has a huge number of dimensions:

•We run the risk of massively overfitting our model — this would generally result in terrible out of sample performance.

•Too many dimensions causes every observation in your dataset to appear equidistant from all the others.

• If the distances are all approximately equal, then all the observations appear equally alike (as well as equally different), and no meaningful clusters can be formed.

A Simple Example of High Dimensional Data Cursing Us



Contd…

	Reddish	Bluish
	1	0
	1	0
	1	0
	1	0
	0	1
	0	1
	0	1
	0	1

Contd…

	Red	Maroon	Pink	Flamingo	Blue	Turquoise	Seaweed	Ocean
	1	0	0	0	0	0	0	0
	0	1	0	0	0	0	0	0
	0	0	1	0	0	0	0	0
	0	0	0	1	0	0	0	0
	0	0	0	0	1	0	0	0
	0	0	0	0	0	1	0	0
	0	0	0	0	0	0	1	0
	0	0	0	0	0	0	0	1

Contd…

Now instead of 2 categories of colors, we have 8. How would a clustering algorithm likely interpret this? It would look at each candy and make the following conclusions:

•Every candy is its own color.

•As an algorithm (without special training), I do not know the relationships between colors. For example, unlike humans, I do not know that pink is closer to red than turquoise is.

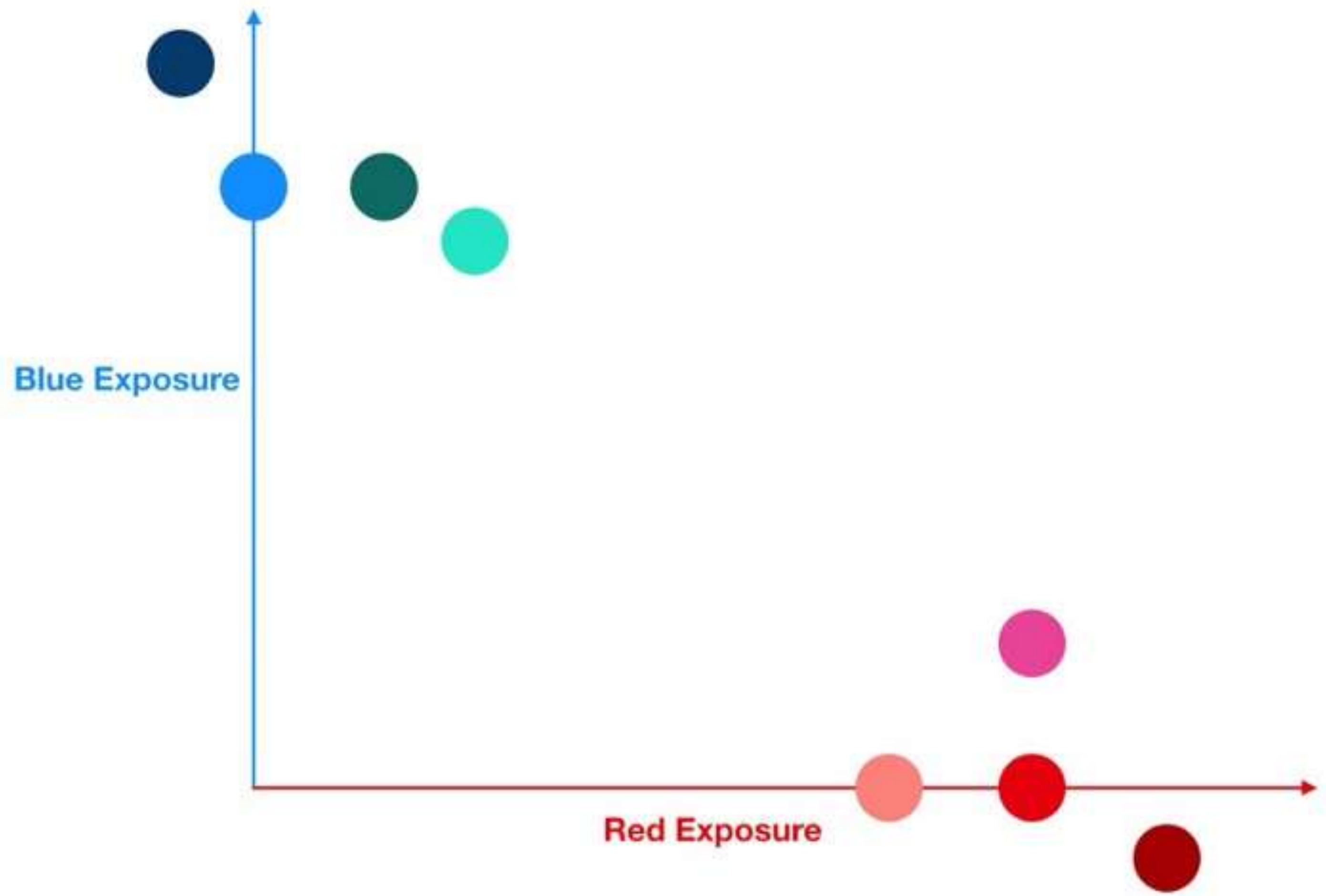
•Given this set of features, I conclude that there are 8 clusters and they are all equally similar to each other.

•I also conclude that out of my 8 clusters, 4 are spicy and 4 are sweet.

Dimensionality Reduction to the Rescue

	Red	Blue
Red	1.00	0
Maroon	1.20	-0.10
Pink	1.00	0.20
Flamingo	0.80	0
Blue	0	1.00
Turquoise	0.25	0.90
Seaweed	0.15	1.00
Ocean	-0.10	1.20

Contd…



Contd…

So our spicy/sweet model for candy would look something like the following:

- Given a new candy, **record its color.**
- Transform its color into exposures to the red feature and the blue feature — in other words, **rewrite the color in terms of our latent features.**
- Using the latent feature exposures of our new candy, **figure out whether it is more similar to the red candy cluster or the blue candy cluster** using a distance measure such as Euclidean distance.
- If our model puts it in the red cluster, we predict the new candy to be spicy (since all the red cluster candies in our original dataset were spicy). And if our model puts it in the blue cluster, we predict the new candy to be sweet (since all the blue cluster candies in our original dataset were sweet).


thank
you