CSE419 – Artificial Intelligence and Machine Learning 2018

PhD Furkan Gözükara, Toros University

https://github.com/FurkanGozukara/CSE419_2018

Lecture 4

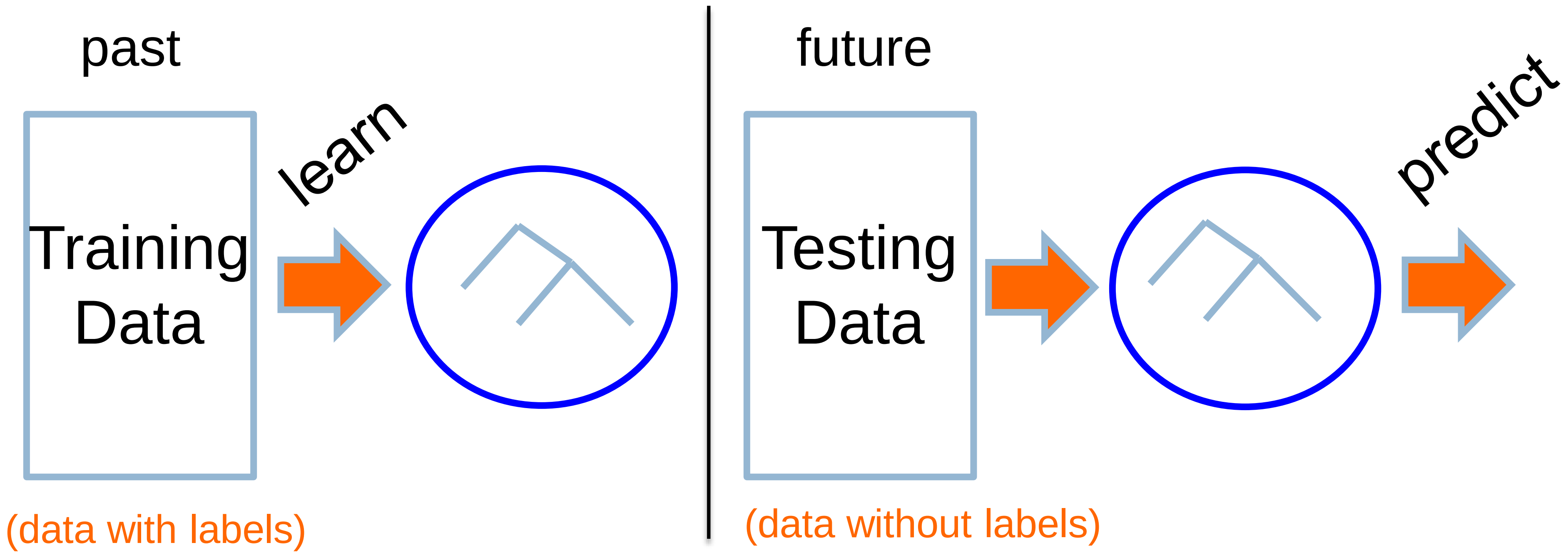
Geometric View of Data

Based on Asst. Prof. Dr. David Kauchak (Pomona College) Lecture Slides

Proper Experimentation

Experimental setup

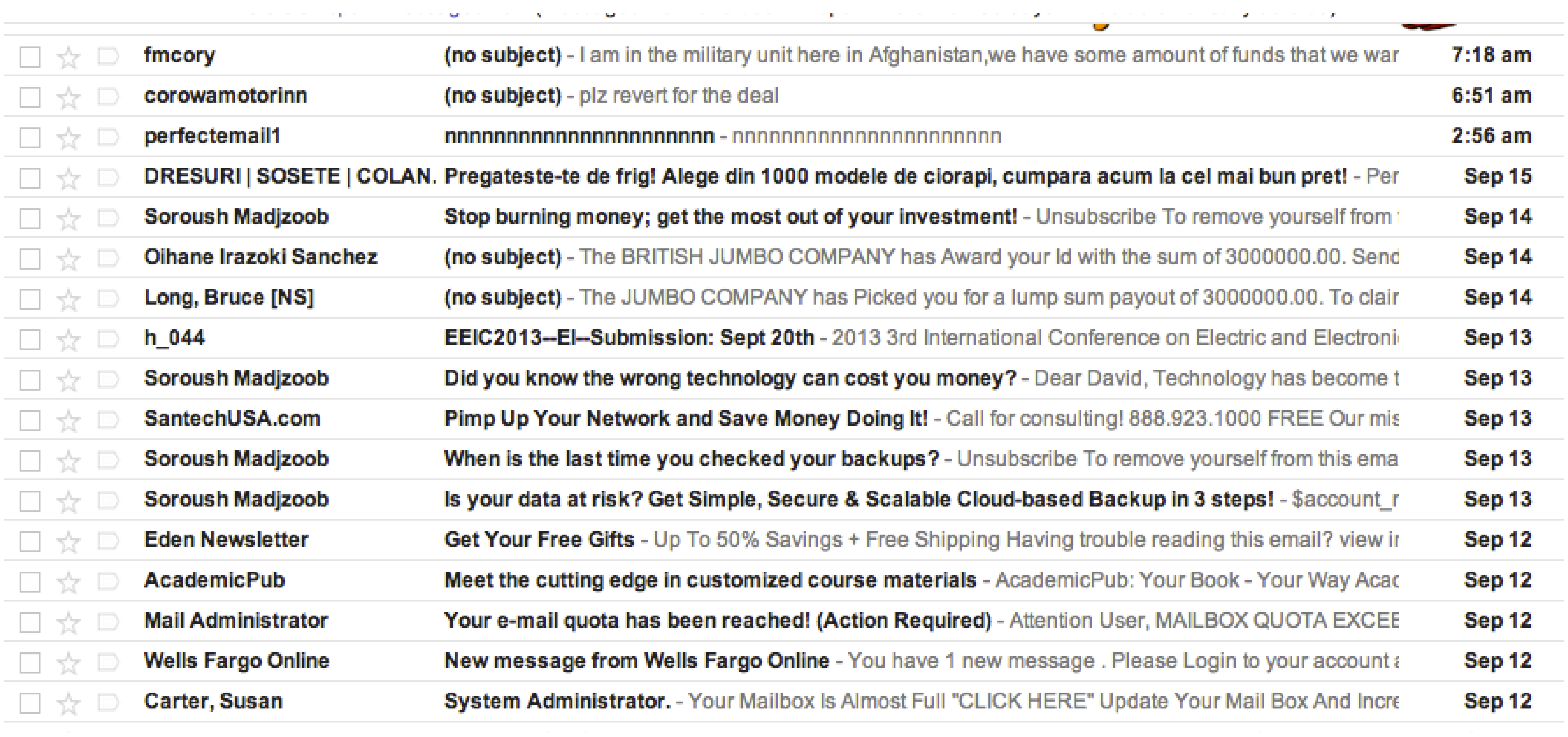
REAL WORLD USE OF ML ALGORITHMS



How do we tell how well we're doing?

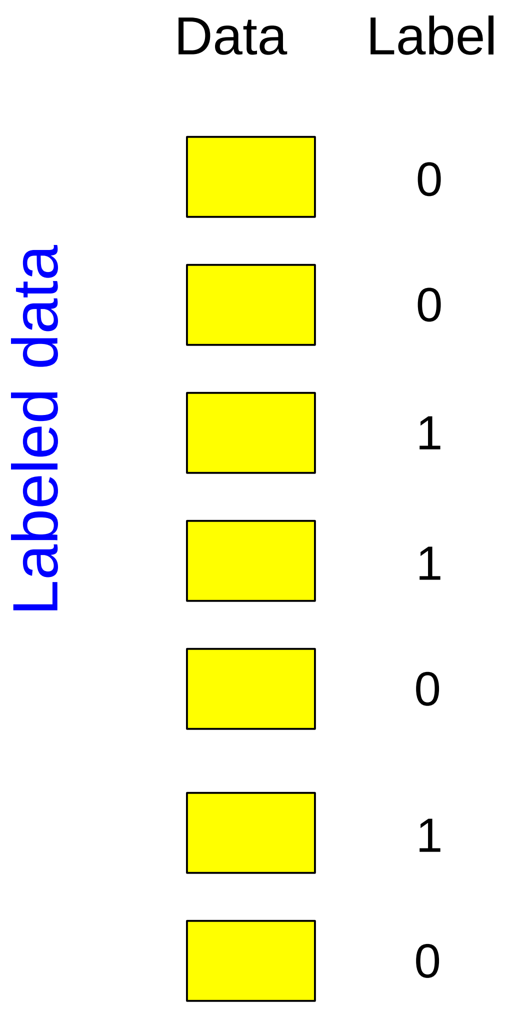
Real-world classification

Google has labeled training data, for example from people clicking the "spam" button, but when new messages come in, they're not labeled



fmcory	(no subject) - I am in the military unit here in Afghanistan,we have some amount of funds that we war	7:18 am
corowamotorinn	(no subject) - plz revert for the deal	6:51 am
perfectemail1	nnnnnnnnnnnnnnnnnnnnnn - nnnnnnnnnnnnnnnnnnnnnn	2:56 am
DRESURI | SOSETE | COLAN.	Pregateste-te de frig! Alege din 1000 modele de ciorapi, cumpara acum la cel mai bun pret! - Per	Sep 15
Soroush Madjzoob	Stop burning money; get the most out of your investment! - Unsubscribe To remove yourself from	Sep 14
Oihane Irazoki Sanchez	(no subject) - The BRITISH JUMBO COMPANY has Award your Id with the sum of 3000000.00. Send	Sep 14
Long, Bruce [NS]	(no subject) - The JUMBO COMPANY has Picked you for a lump sum payout of 3000000.00. To clair	Sep 14
h_044	EEIC2013--EI--Submission: Sept 20th - 2013 3rd International Conference on Electric and Electroni	Sep 13
Soroush Madjzoob	Did you know the wrong technology can cost you money? - Dear David, Technology has become t	Sep 13
SantechUSA.com	Pimp Up Your Network and Save Money Doing It! - Call for consulting! 888.923.1000 FREE Our mis	Sep 13
Soroush Madjzoob	When is the last time you checked your backups? - Unsubscribe To remove yourself from this ema	Sep 13
Soroush Madjzoob	Is your data at risk? Get Simple, Secure & Scalable Cloud-based Backup in 3 steps! - $account_r	Sep 13
Eden Newsletter	Get Your Free Gifts - Up To 50% Savings + Free Shipping Having trouble reading this email? view ir	Sep 12
AcademicPub	Meet the cutting edge in customized course materials - AcademicPub: Your Book - Your Way Acac	Sep 12
Mail Administrator	Your e-mail quota has been reached! (Action Required) - Attention User, MAILBOX QUOTA EXCEE	Sep 12
Wells Fargo Online	New message from Wells Fargo Online - You have 1 new message . Please Login to your account a	Sep 12
Carter, Susan	System Administrator. - Your Mailbox Is Almost Full "CLICK HERE" Update Your Mail Box And Incre	Sep 12

Classification evaluation

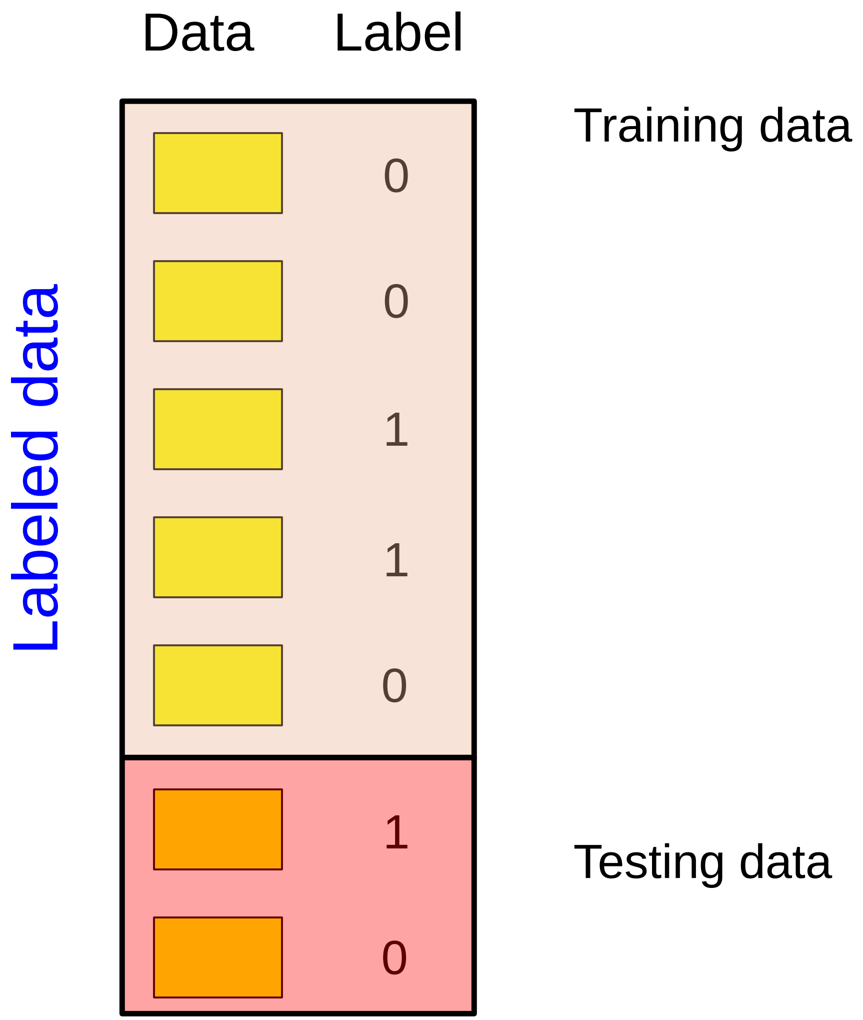


Use the labeled data we have already to create a test set with known labels!

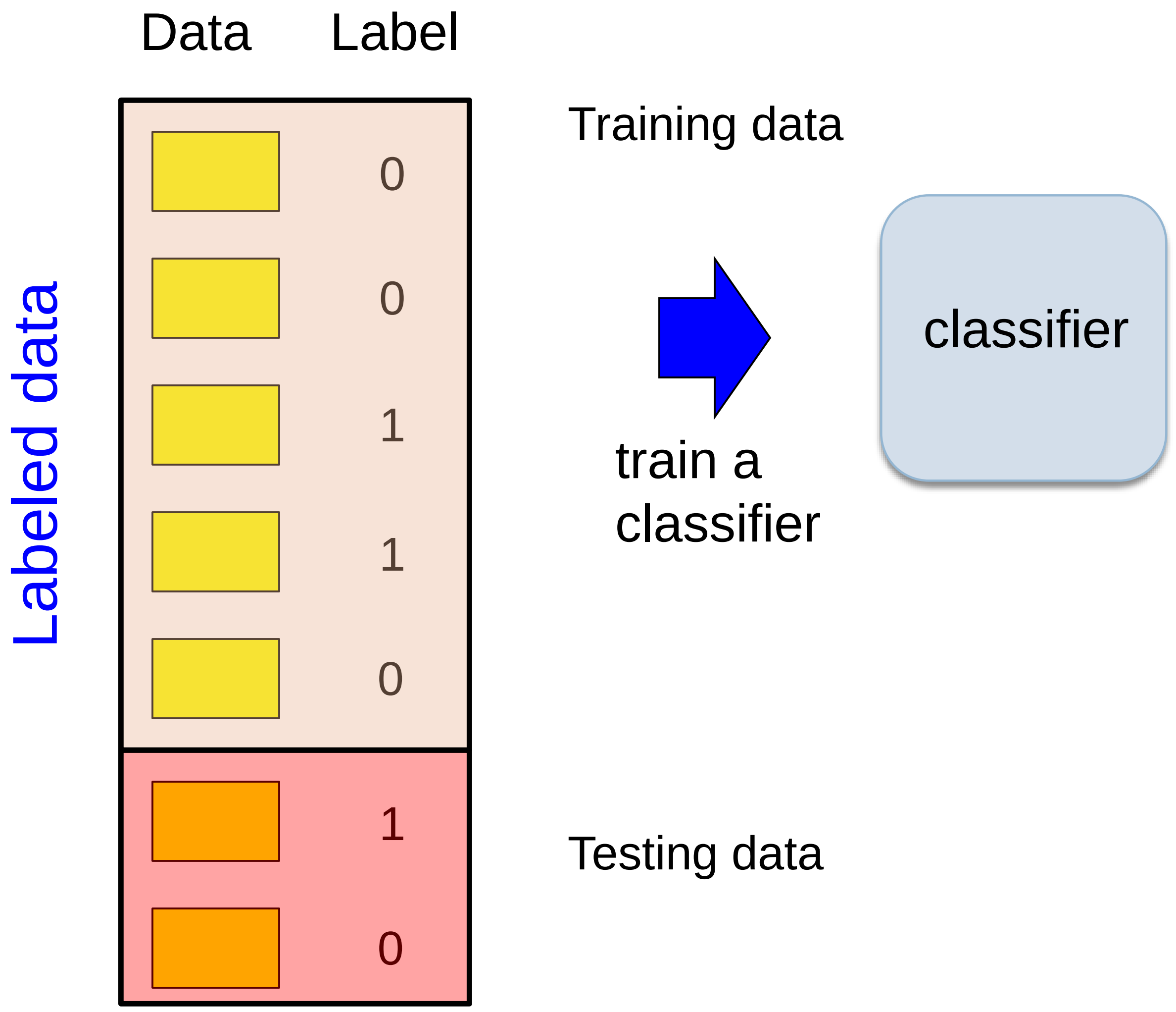
Why can we do this?

Remember, we assume there's an underlying distribution that generates both the training and test examples

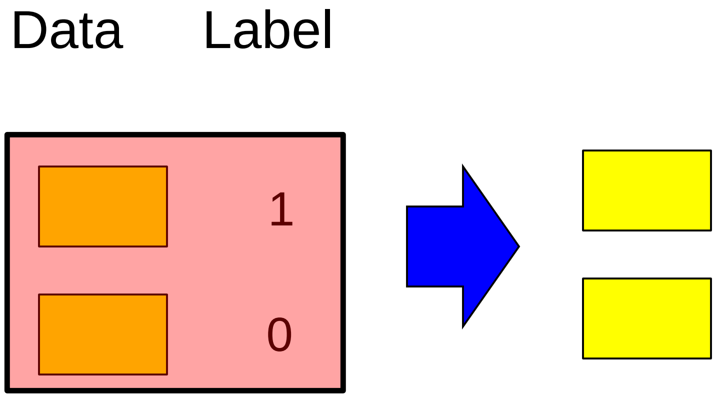
Classification evaluation



Classification evaluation

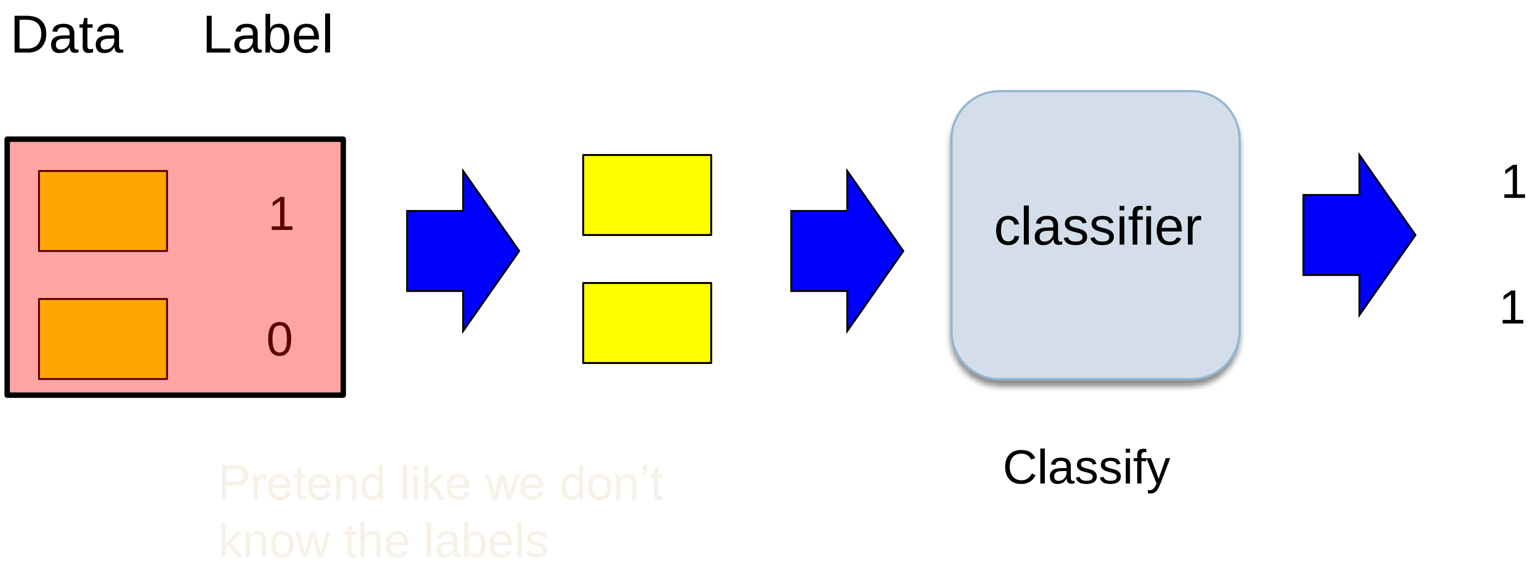


Classification evaluation

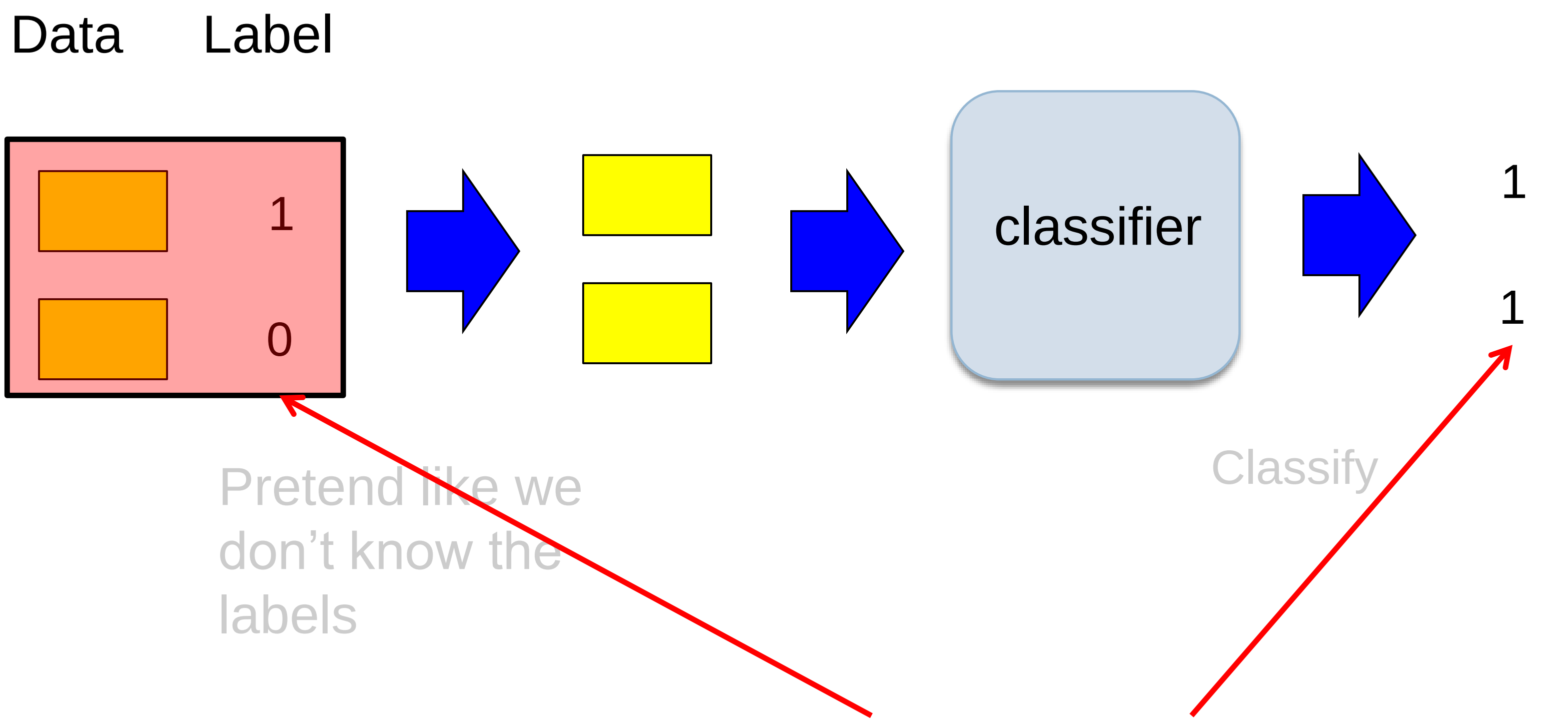


Pretend like we don’t know the labels

Classification evaluation



Classification evaluation

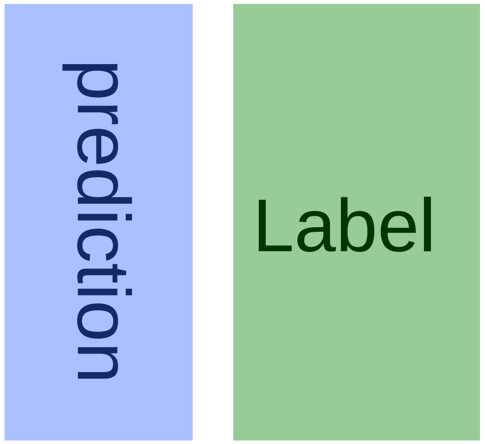


How could we score these for classification?

Compare predicted labels to actual labels

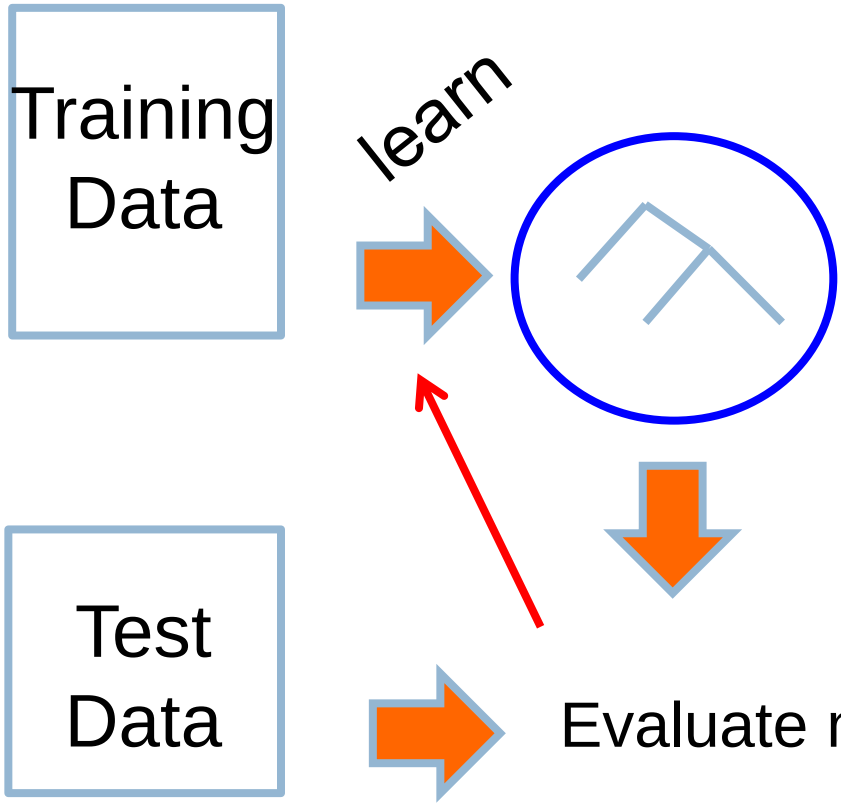
Test accuracy

To evaluate the model, compare the predicted labels to the actual labels



Accuracy: the proportion of examples where we correctly predicted the label

Proper testing



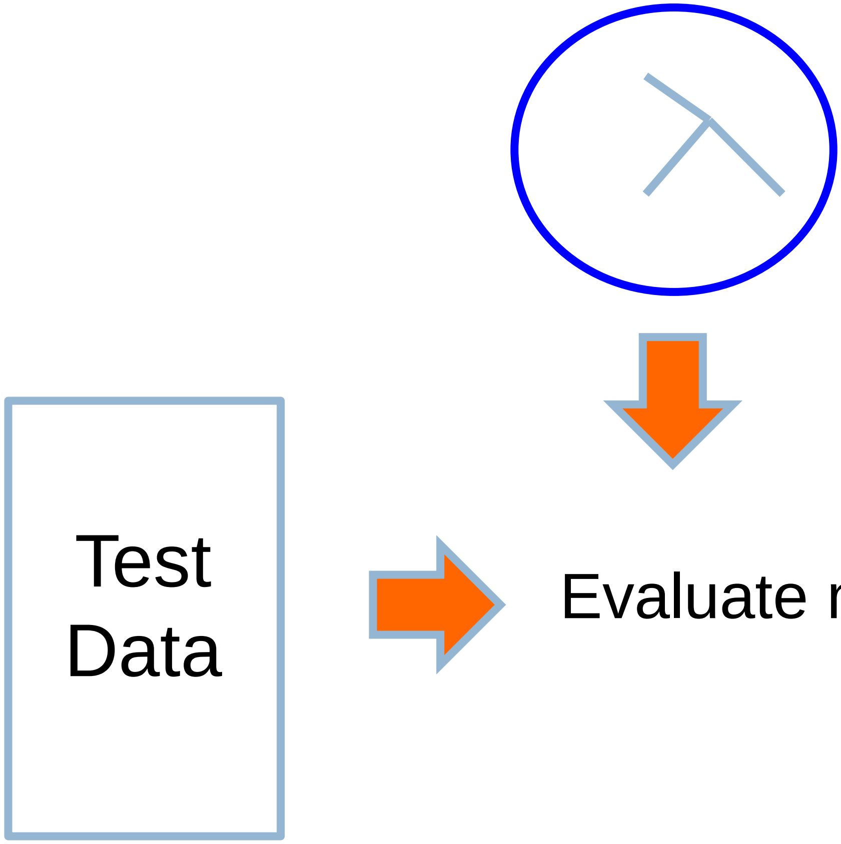
One way to do algorithm development

- try out an algorithm
- evaluated on test data
- repeat until happy with results

Is this ok?

No. Although we're not explicitly looking at the examples, we're still "cheating" by biasing our algorithm to the test data

Proper testing

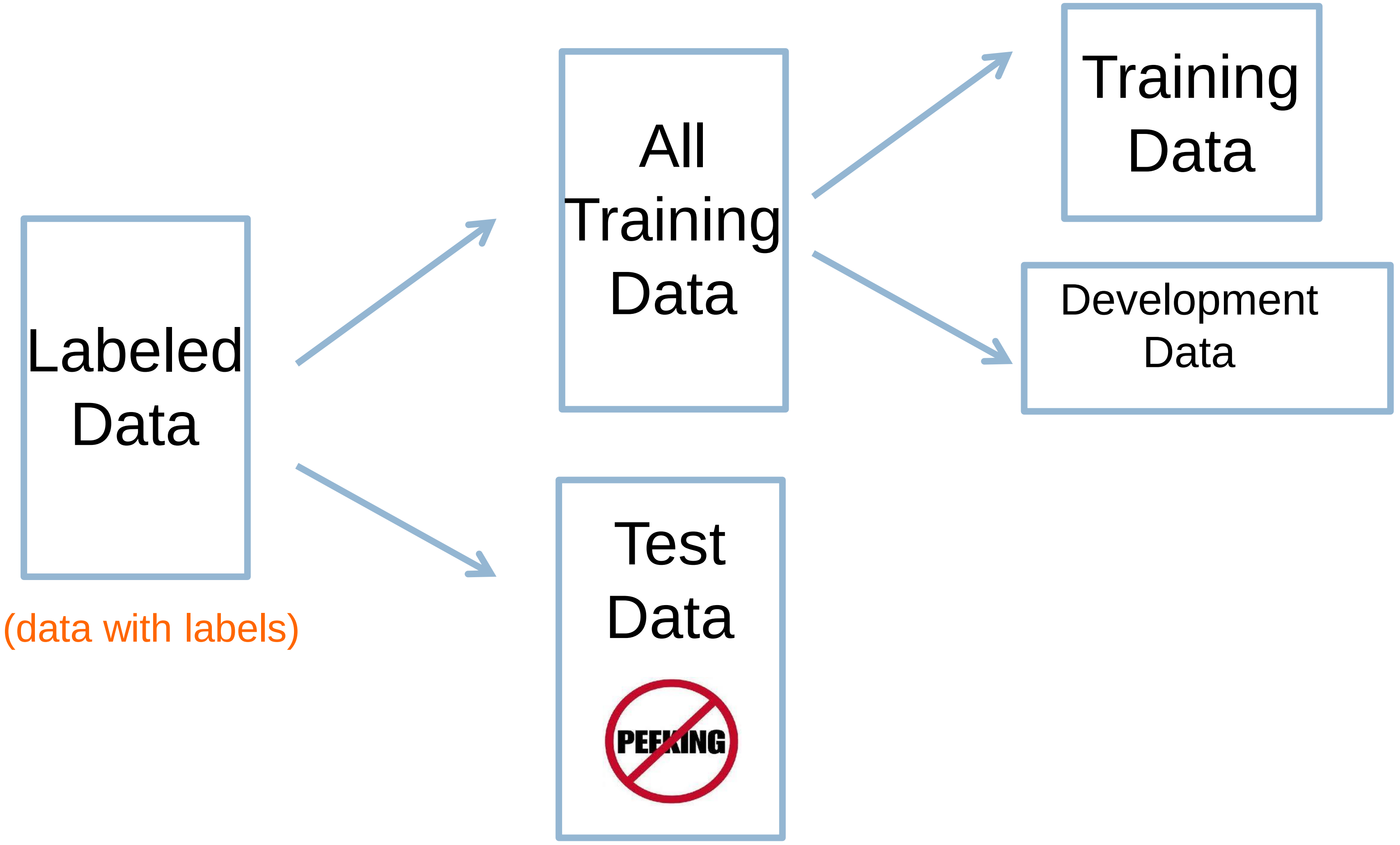


Evaluate model

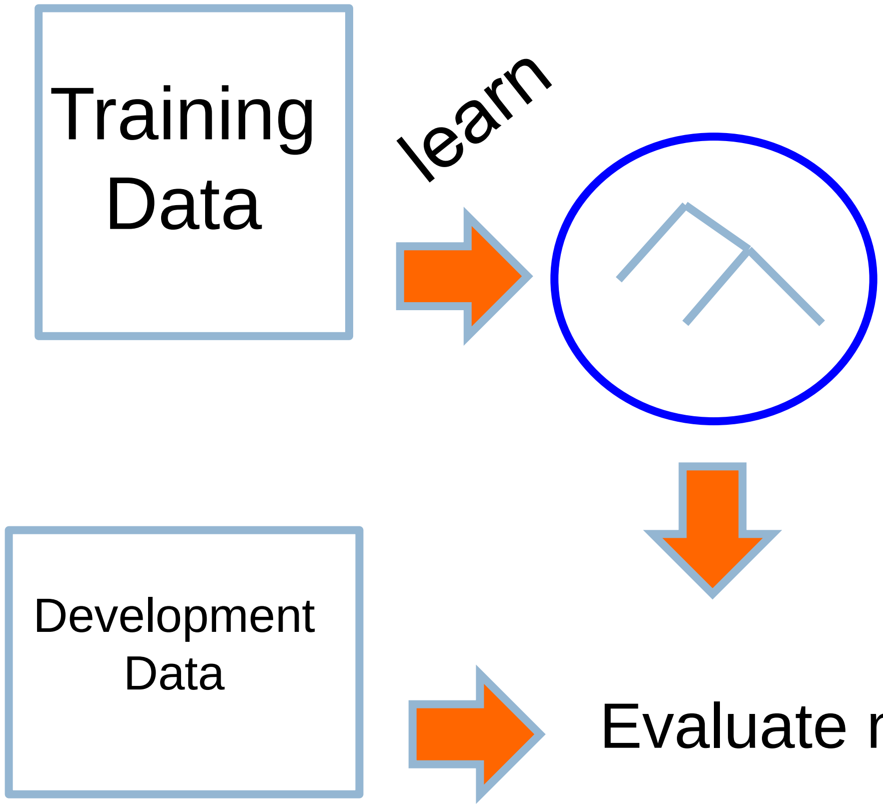
Once you look at/use test data **it is no longer test data!**

So, how can we evaluate our algorithm during development?

Development set



Proper testing

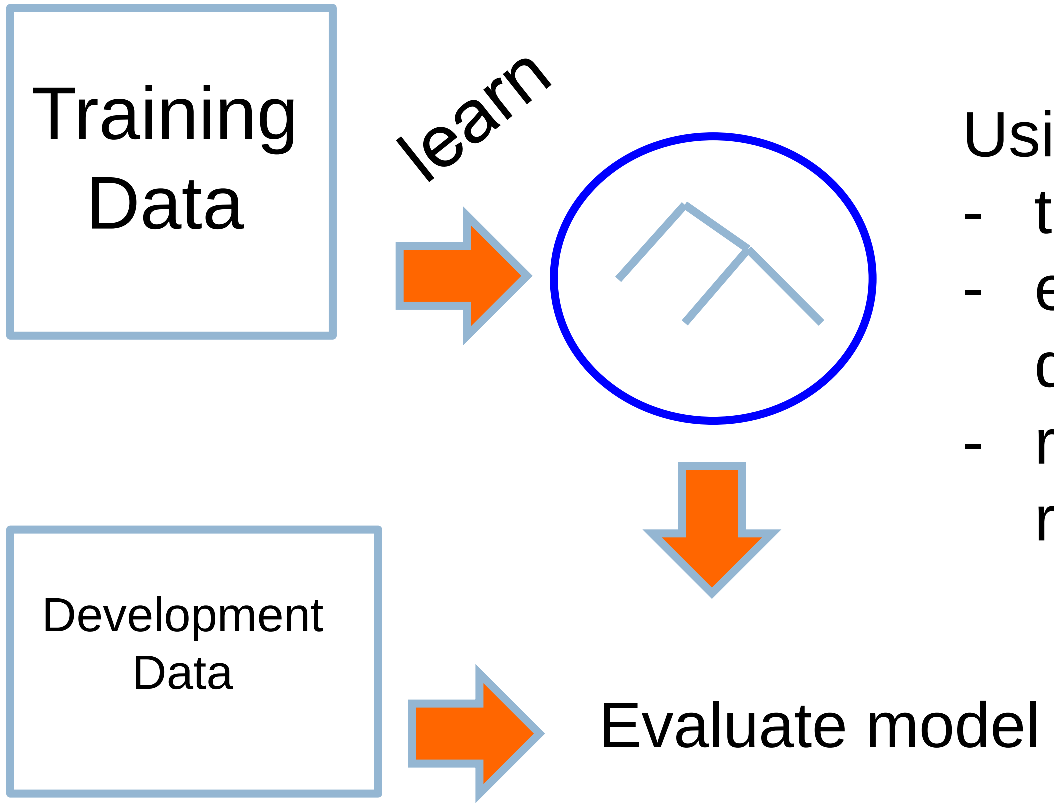


Using the **development data**:

- try out an algorithm
- evaluated on development data
- repeat until happy with results

When satisfied, evaluate on test data

Proper testing

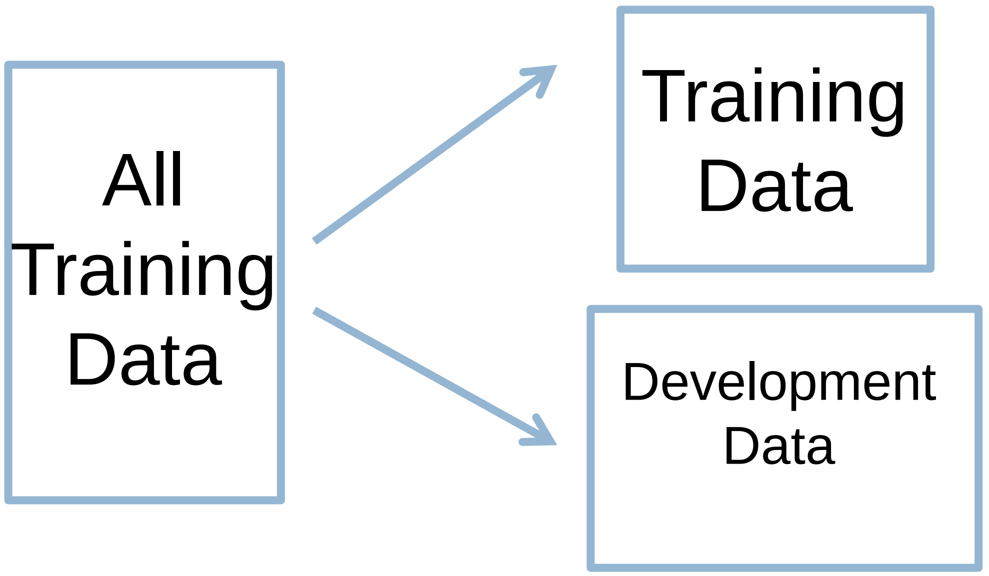


Using the development data:
- try out an algorithm
- evaluated on development data
- repeat until happy with results

Any problems with this?

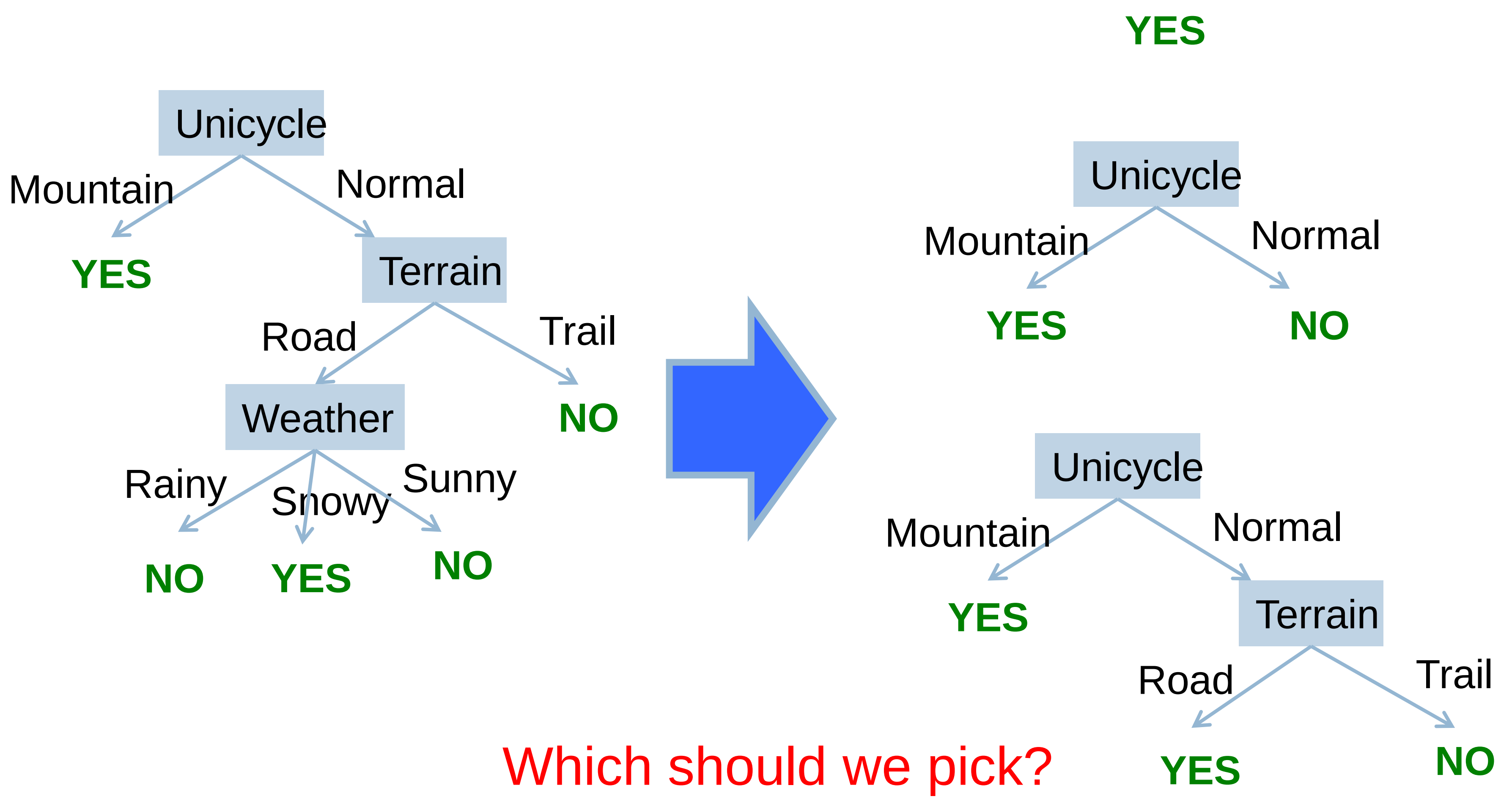
Overfitting to development data

Be careful not to overfit to the development data!



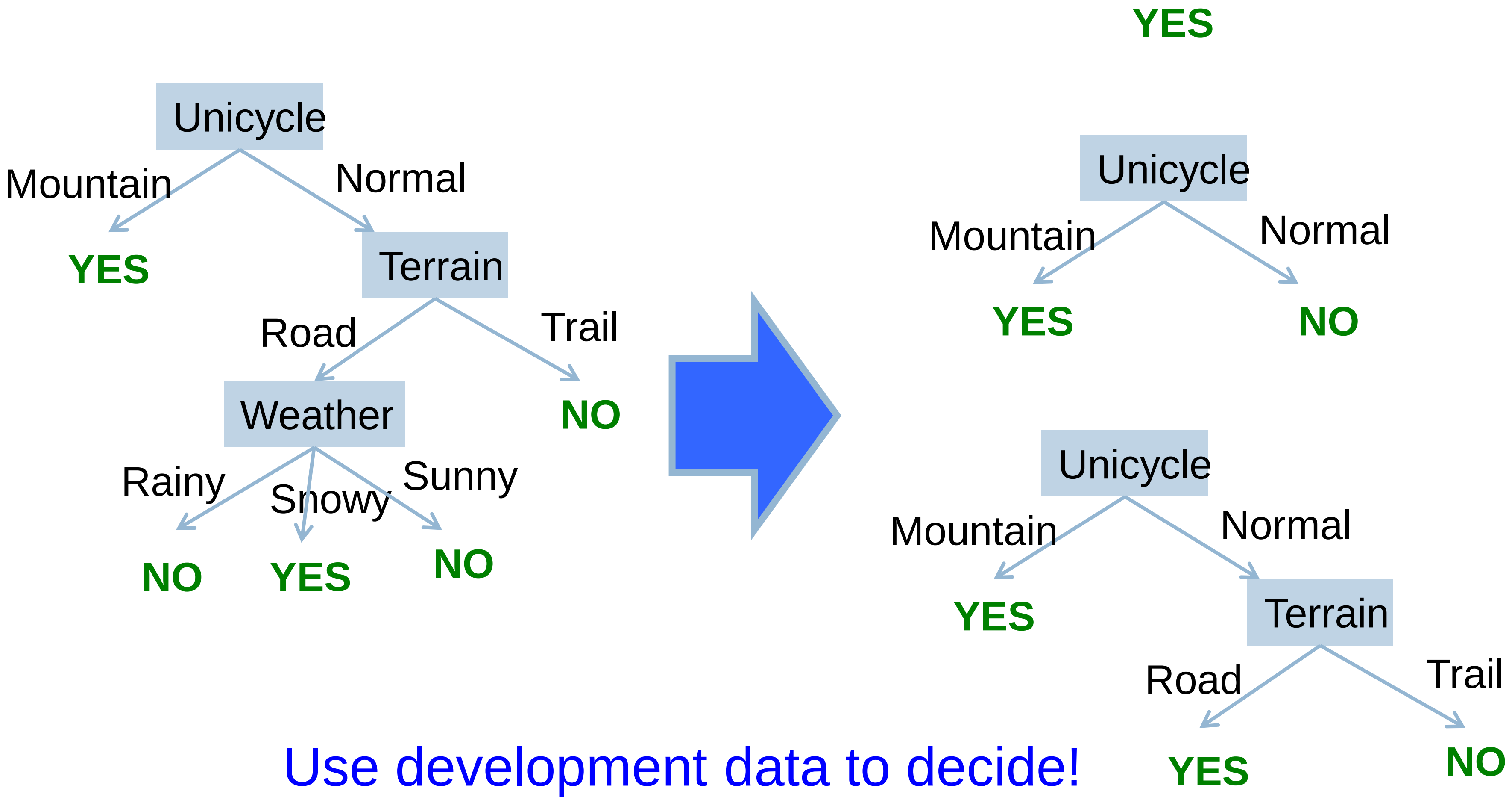
Often we'll split off development data this multiple times (in fact, on the fly), but you can still overfit, but this helps avoid it

Pruning revisited



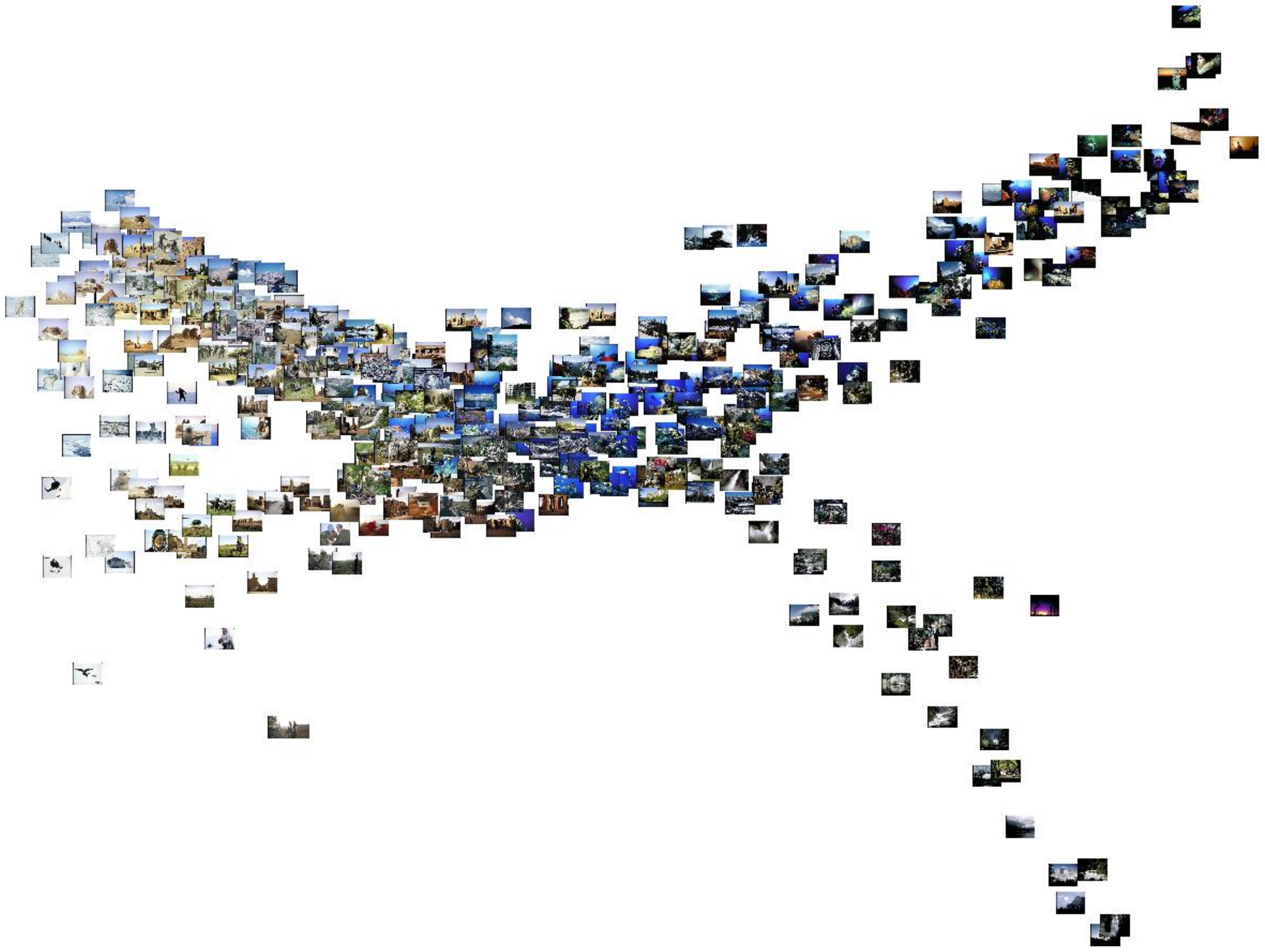
Which should we pick?

Pruning revisited



Use development data to decide!

Machine Learning: A Geometric View



Apples vs. Bananas

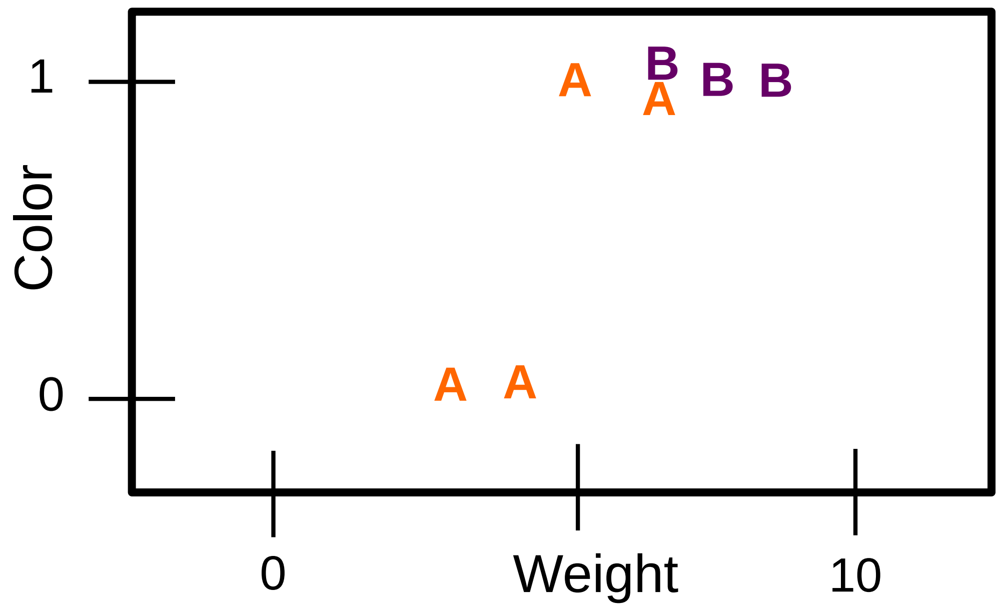
Weight	Color	Label
4	Red	Apple
5	Yellow	Apple
6	Yellow	Banana
3	Red	Apple
7	Yellow	Banana
8	Yellow	Banana
6	Yellow	Apple

Can we visualize this data?

Apples vs. Bananas

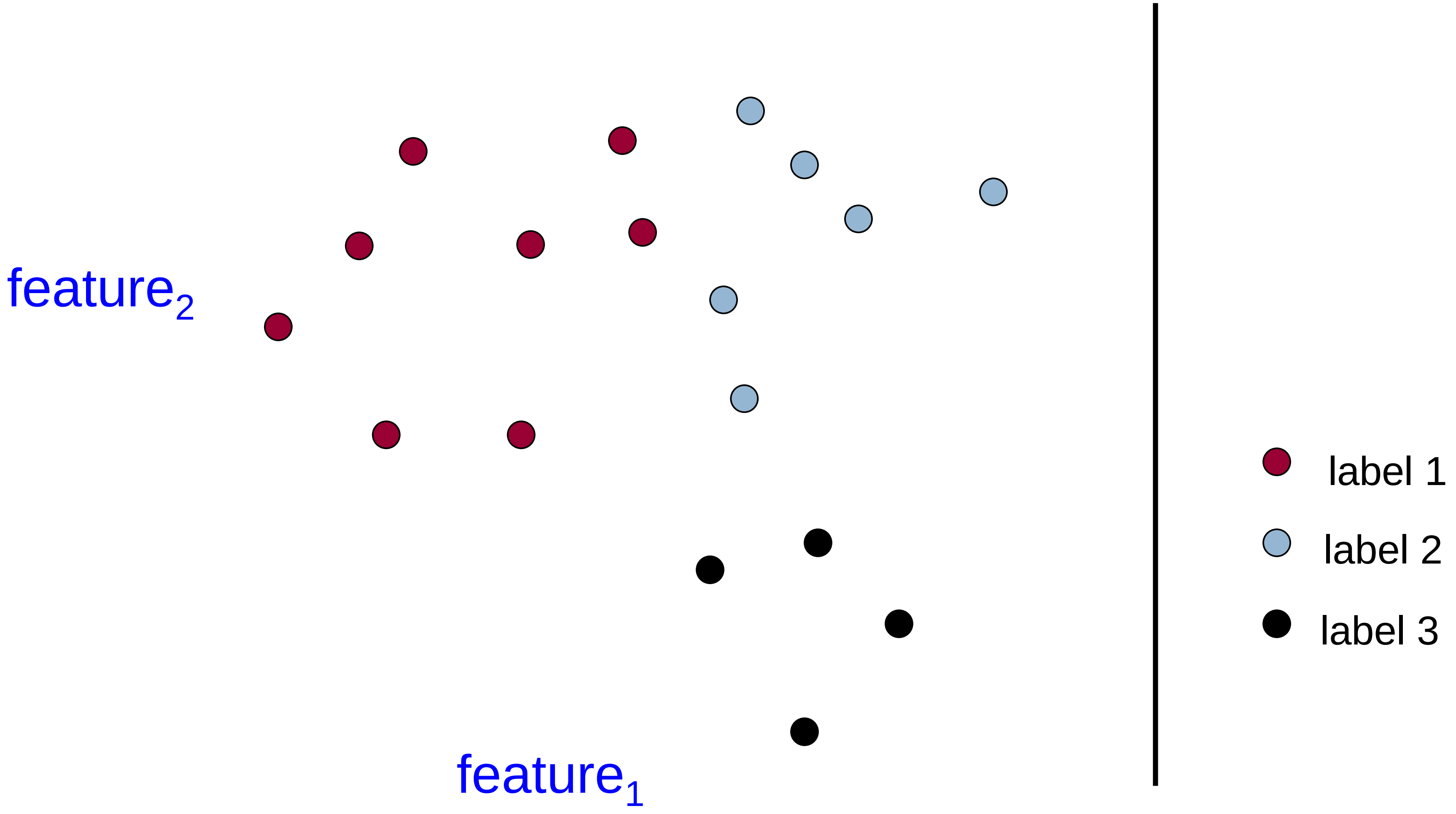
Turn features into numerical values

Weight	Color	Label
4	0	Apple
5	1	Apple
6	1	Banana
3	0	Apple
7	1	Banana
8	1	Banana
6	1	Apple

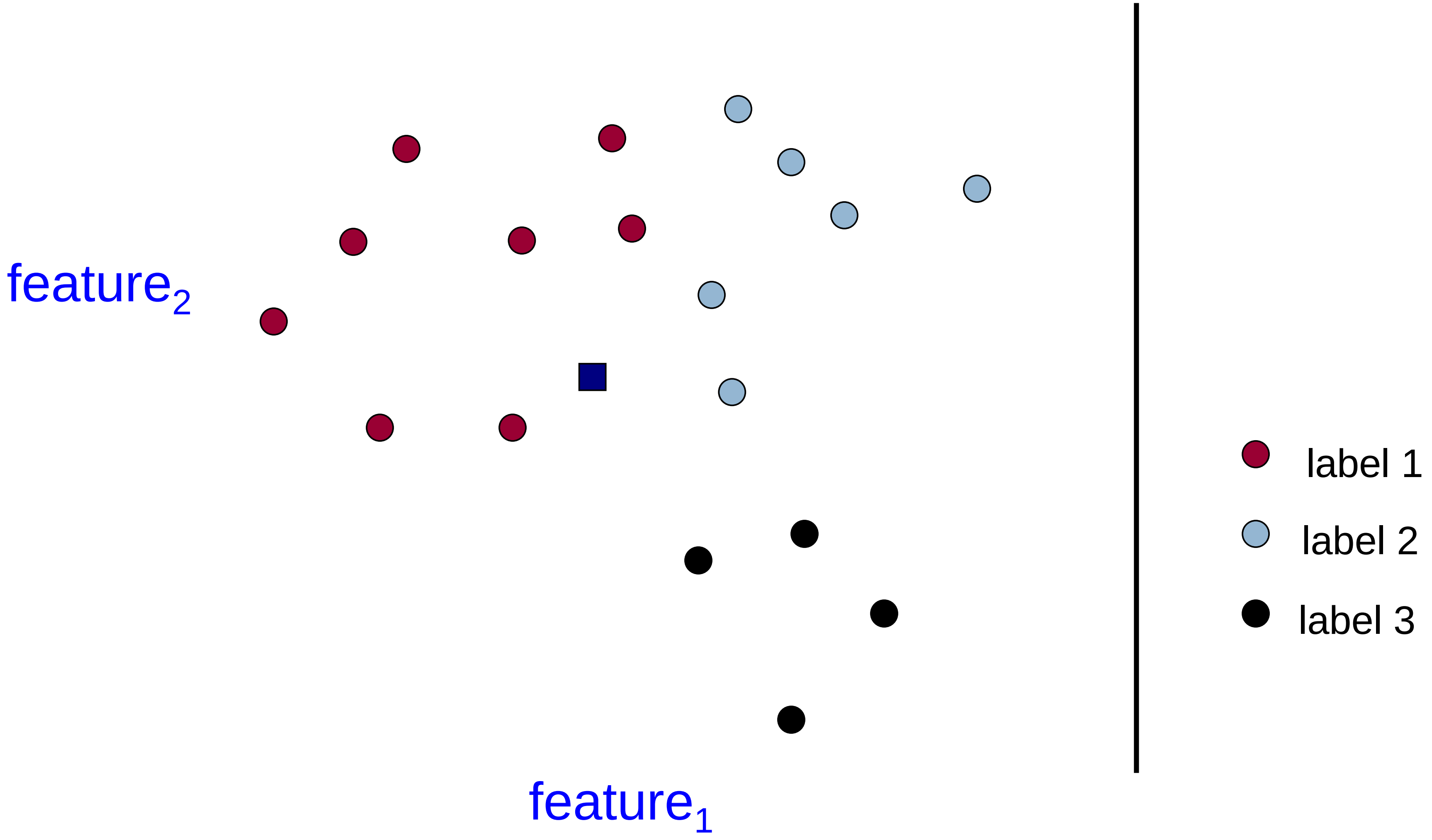


We can view examples as points in an *n*-dimensional space where *n* is the number of features

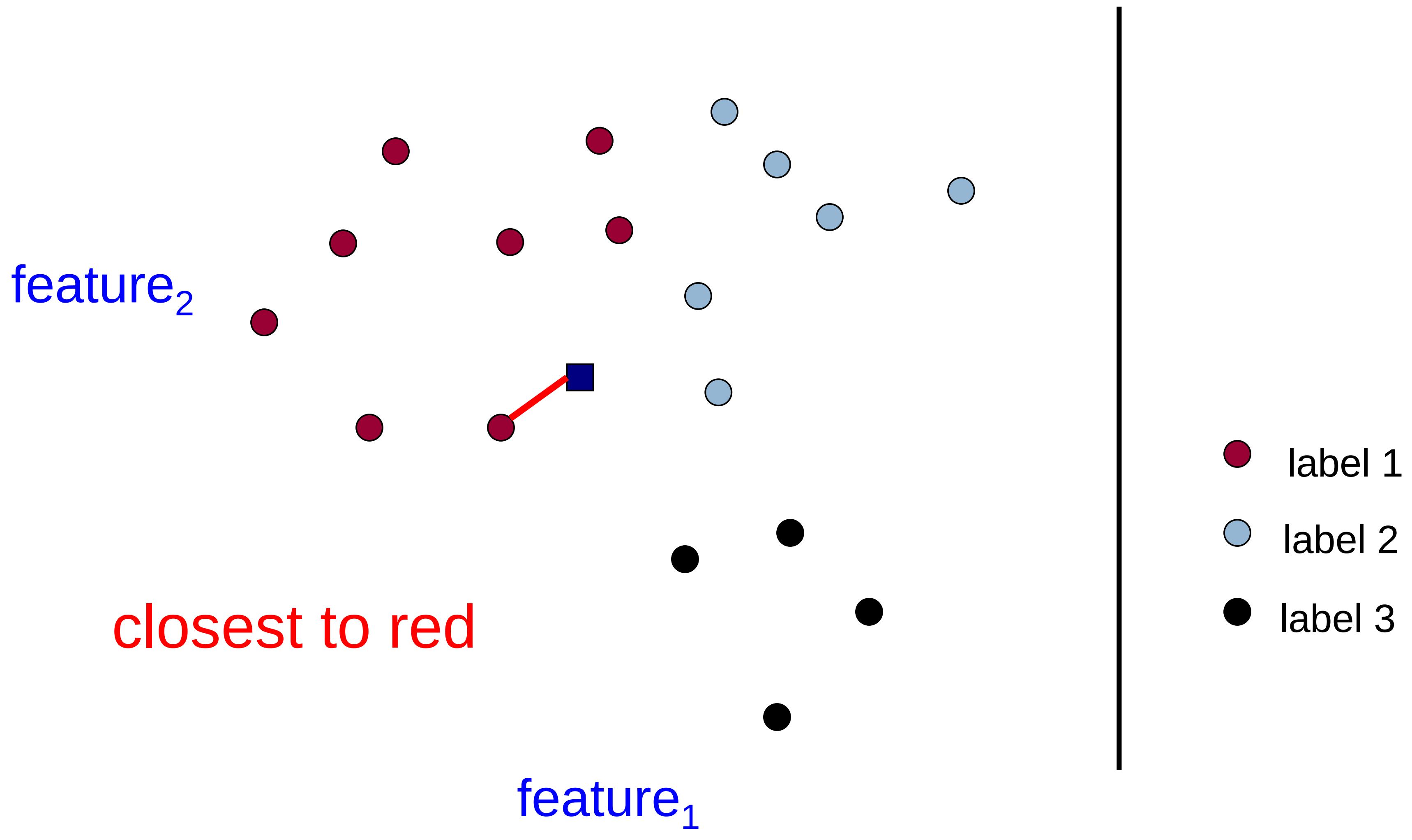
Examples in a feature space



Test example: what class?



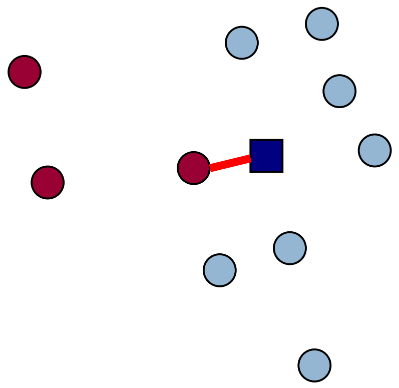
Test example: what class?



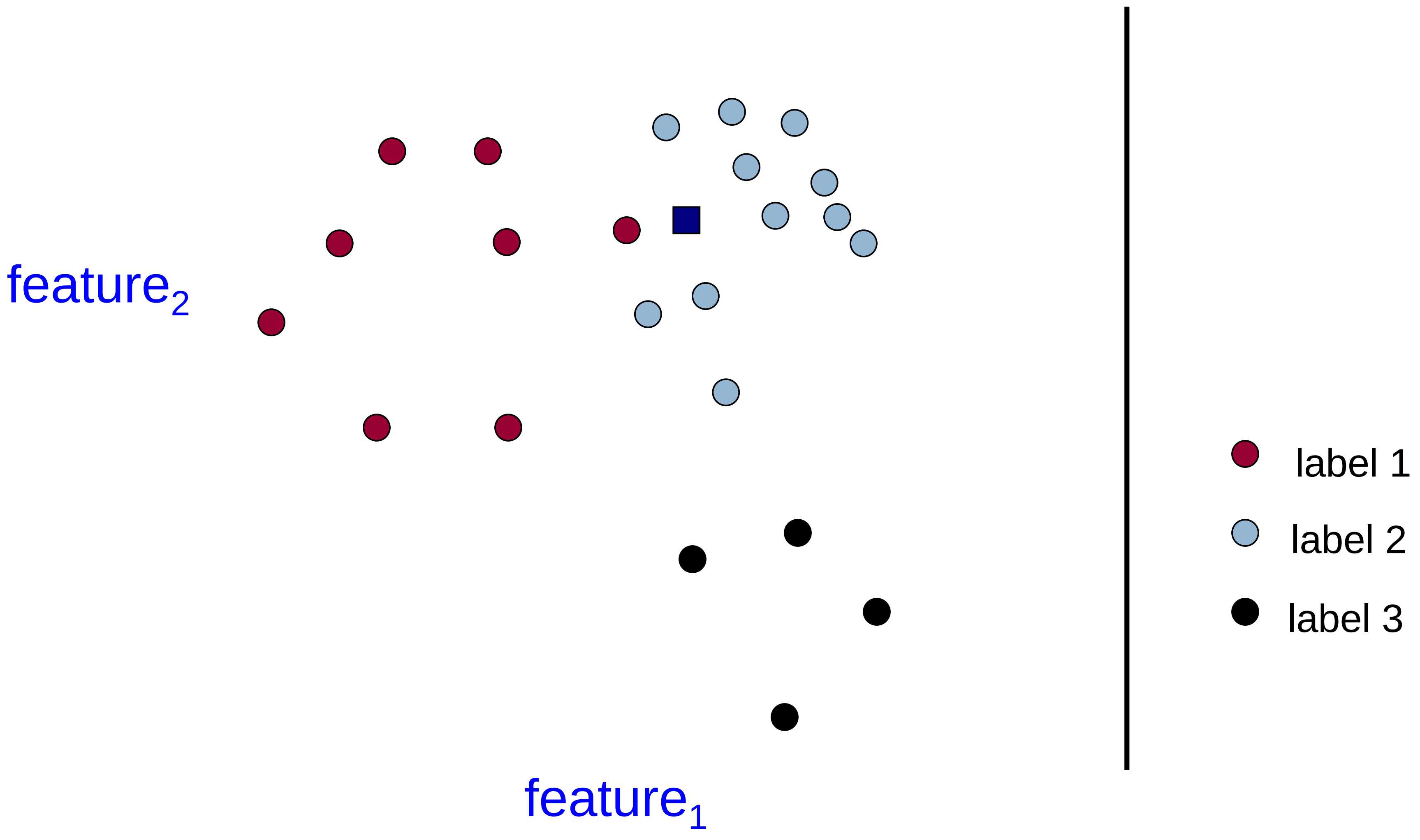
Another classification algorithm?

To classify an example ***d***:

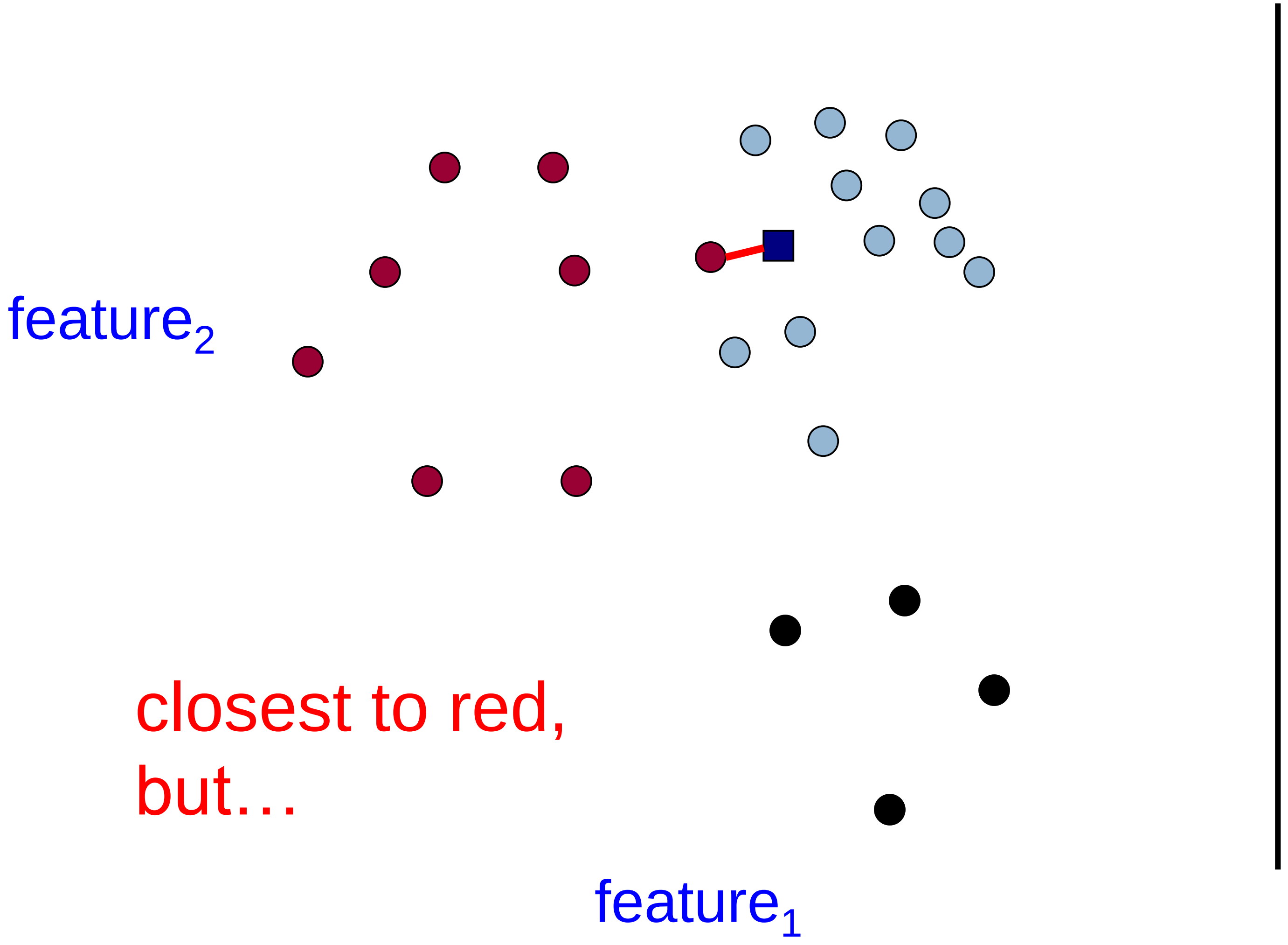
Label ***d*** with the label of the closest example to ***d*** in the training set



What about his example?

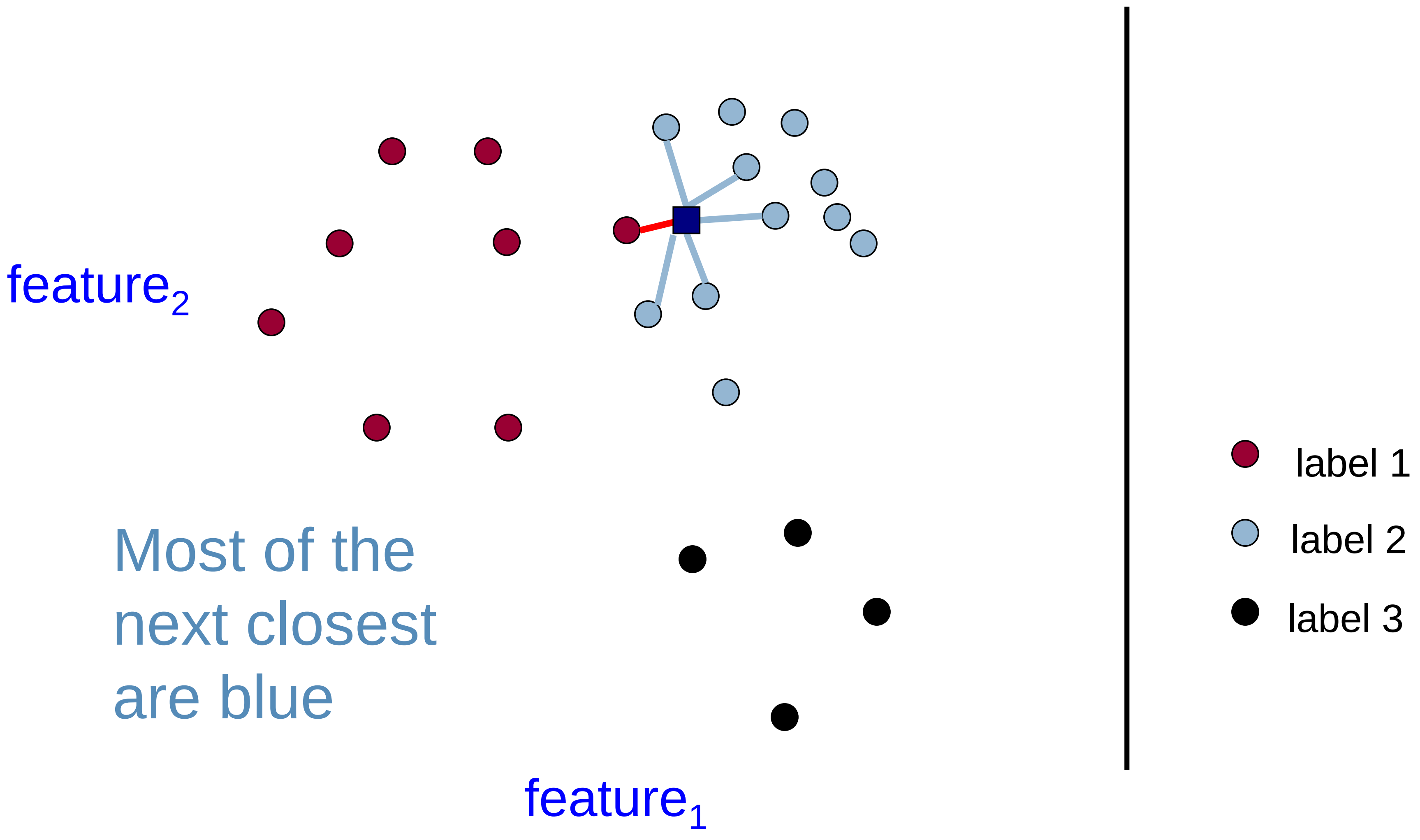


What about his example?



●	label 1
●	label 2
●	label 3

What about his example?



Most of the next closest are blue

k-Nearest Neighbor (k-NN)

To classify an example ***d***:

- Find ***k*** nearest neighbors of ***d***
- Choose as the label the majority label within the ***k*** nearest neighbors

k-Nearest Neighbor (k-NN)

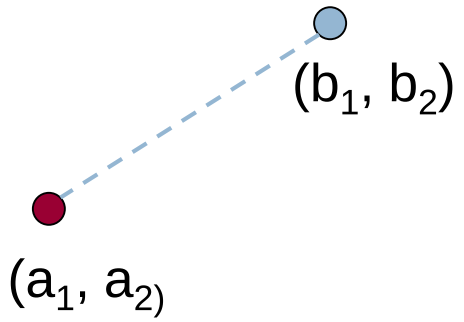
To classify an example ***d***:

- Find ***k*** *nearest* neighbors of ***d***
- Choose as the label the majority label within the ***k*** nearest neighbors

How do we measure “nearest”?

Euclidean distance

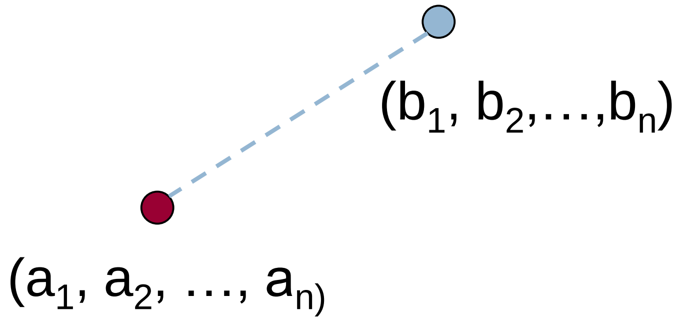
In two dimensions, how do we compute the distance?



$$D(a,b) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}$$

Euclidean distance

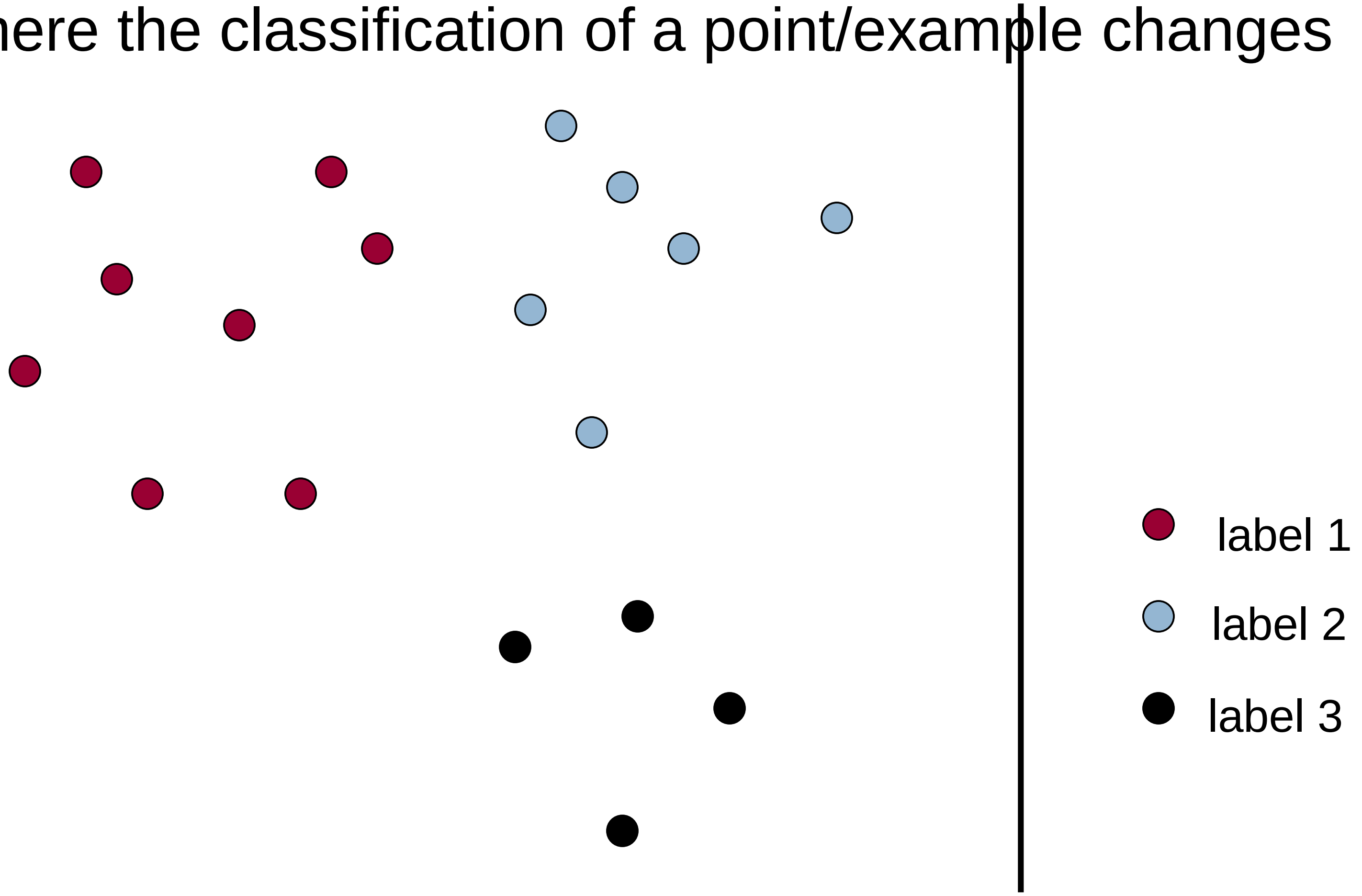
In n-dimensions, how do we compute the distance?



$$D(a,b)=\sqrt{(a_1-b_1)^2+(a_2-b_2)^2+\ldots+(a_n-b_n)^2}$$

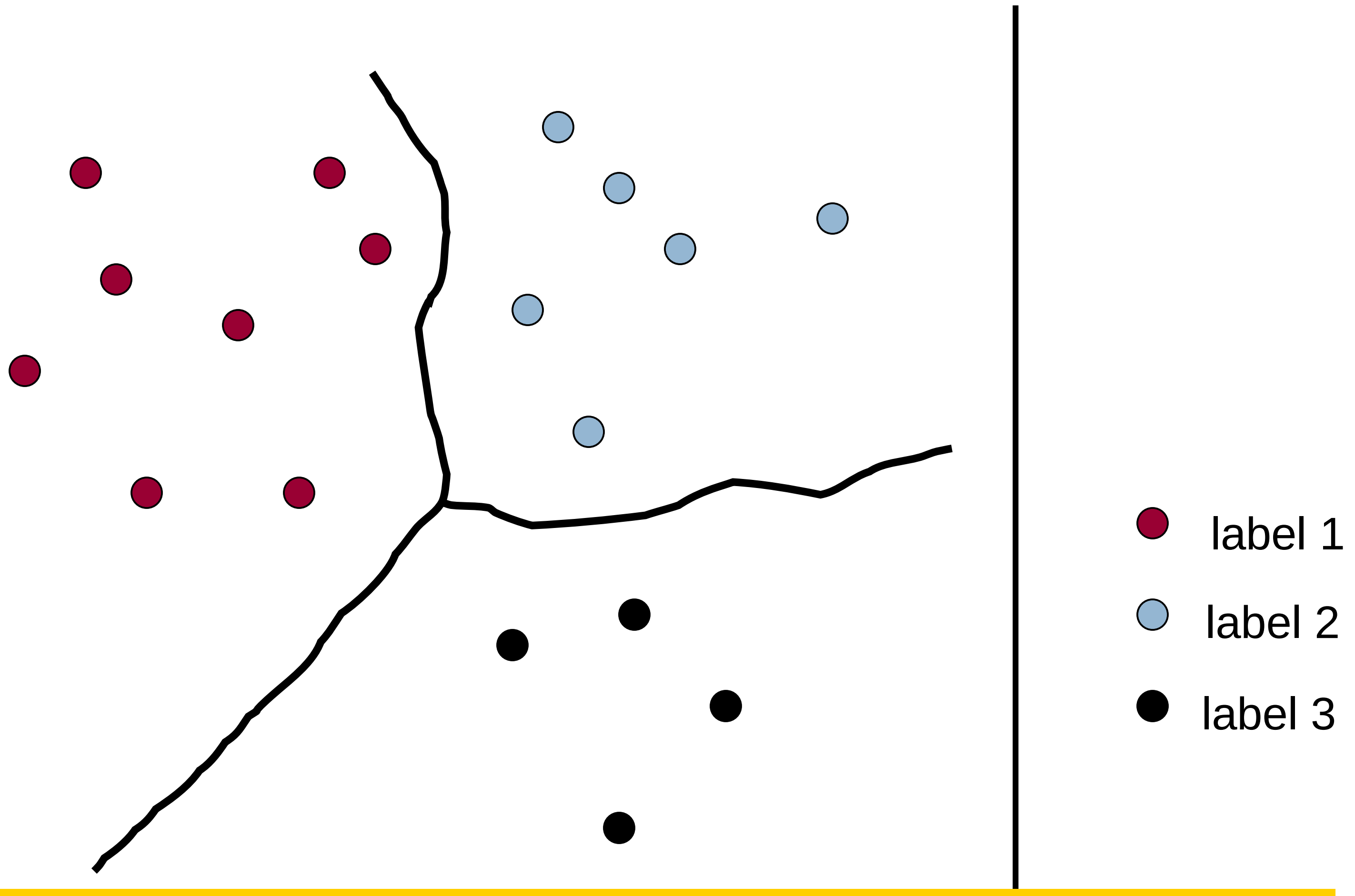
Decision boundaries

The **decision boundaries** are places in the features space where the classification of a point/example changes



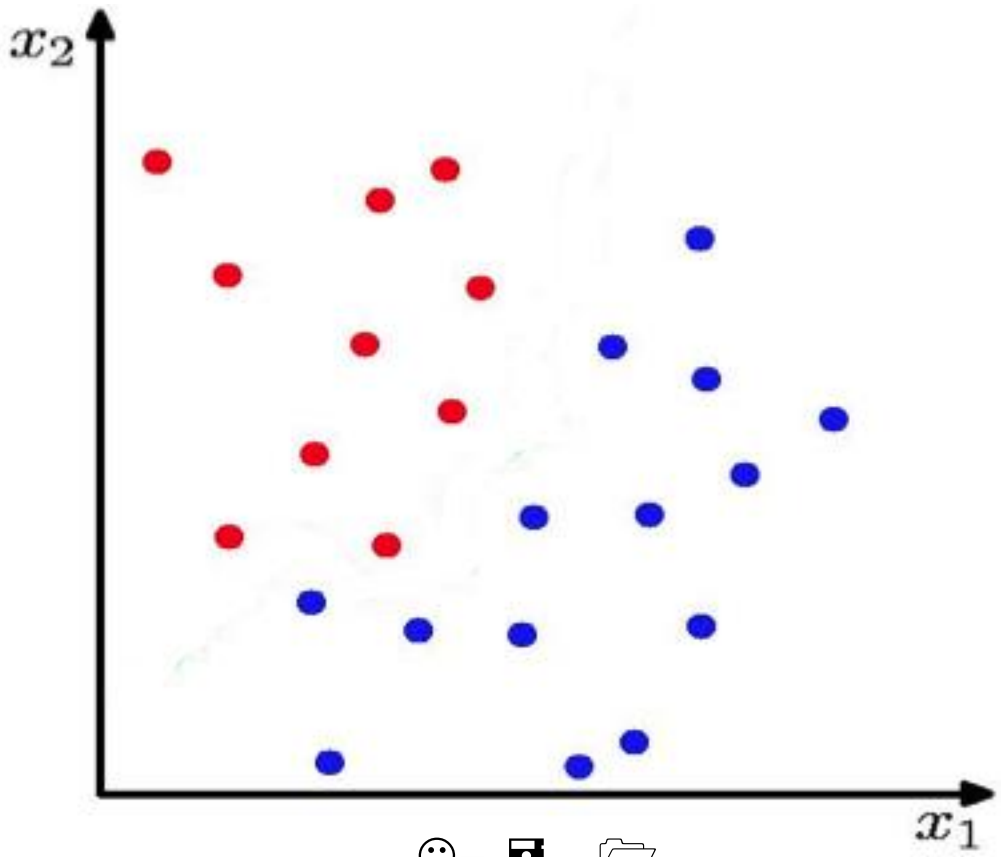
Where are the decision boundaries for k-NN?

k-NN decision boundaries



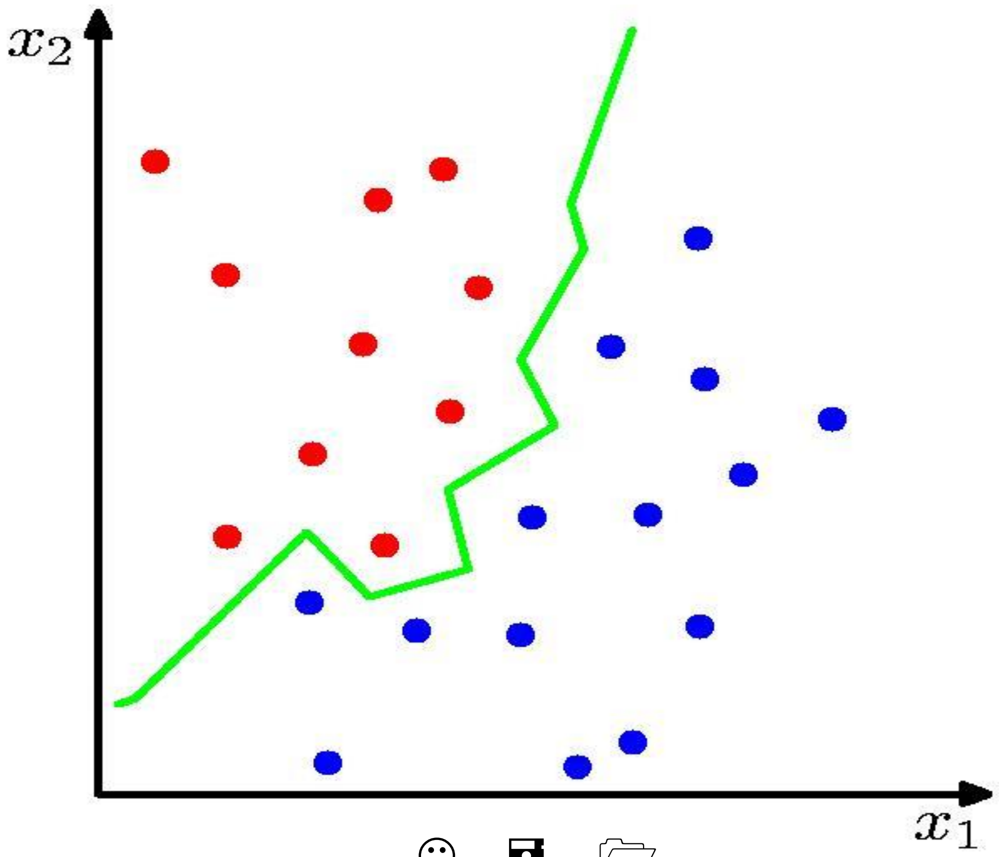
k-NN gives locally defined decision boundaries between classes

😐 Nearest Neighbour (*k*NN) Classifier



😐

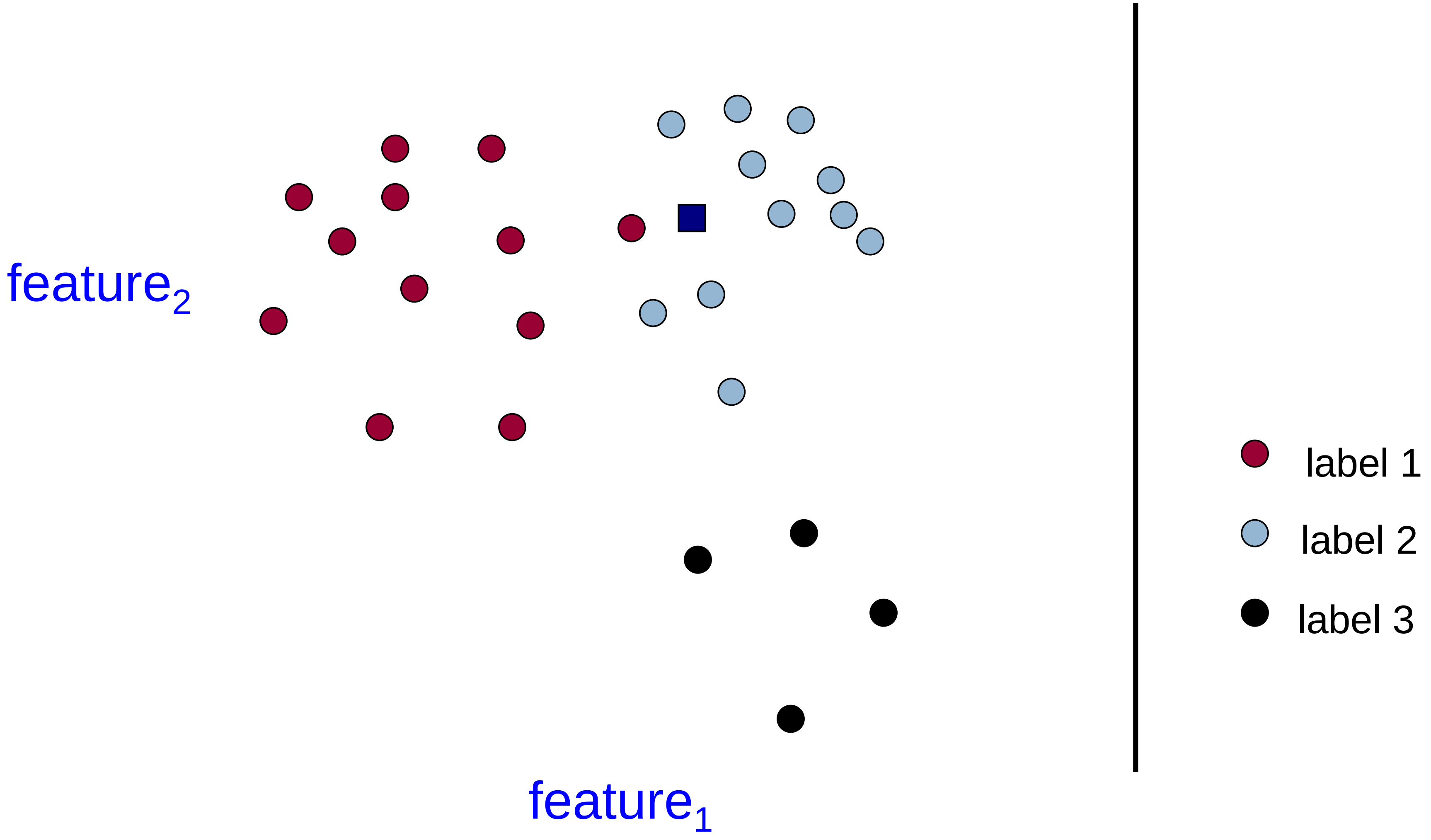
😐 Nearest Neighbour (*k*NN) Classifier



😐 🖬 🗁

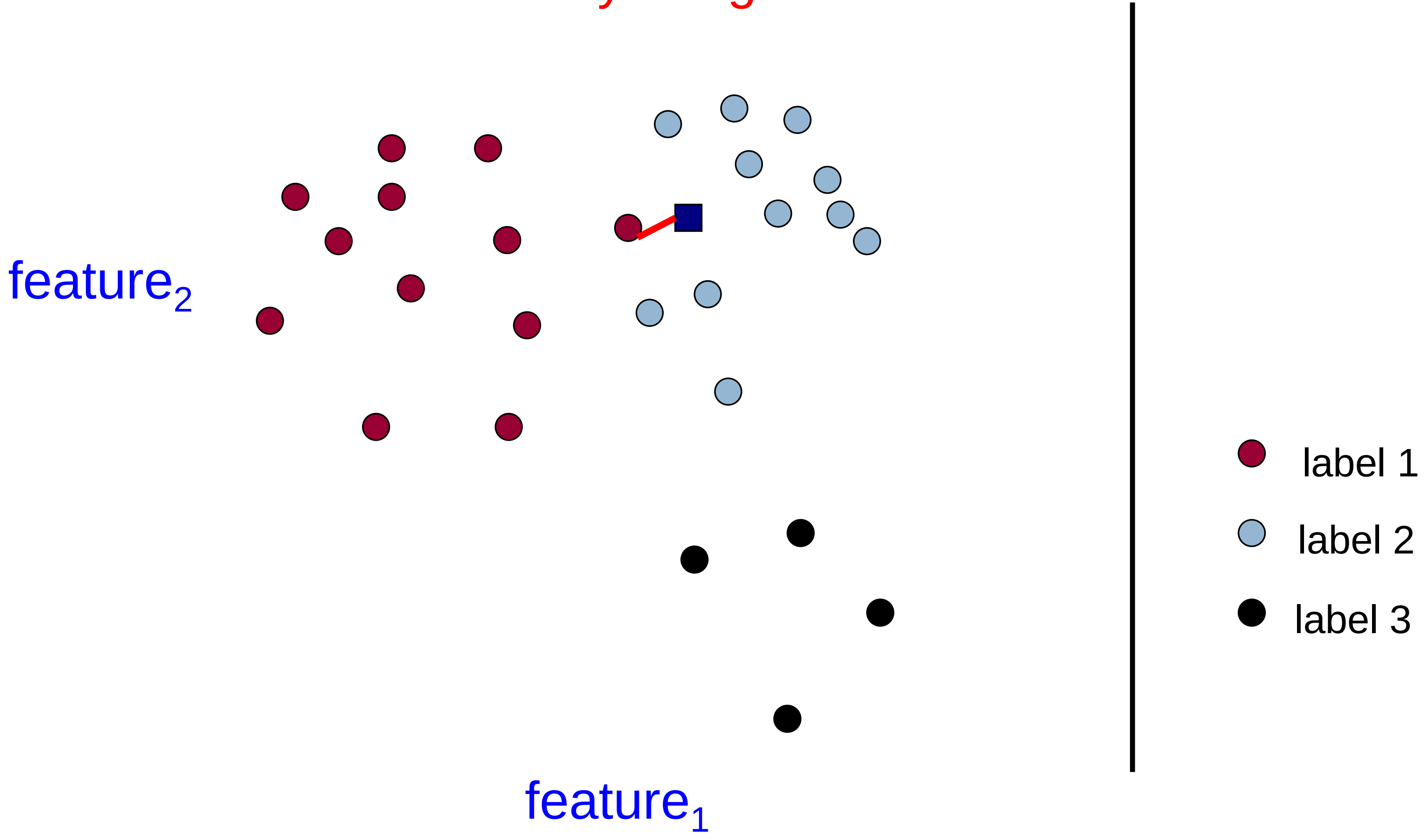
Choosing k

What is the label with k = 1?



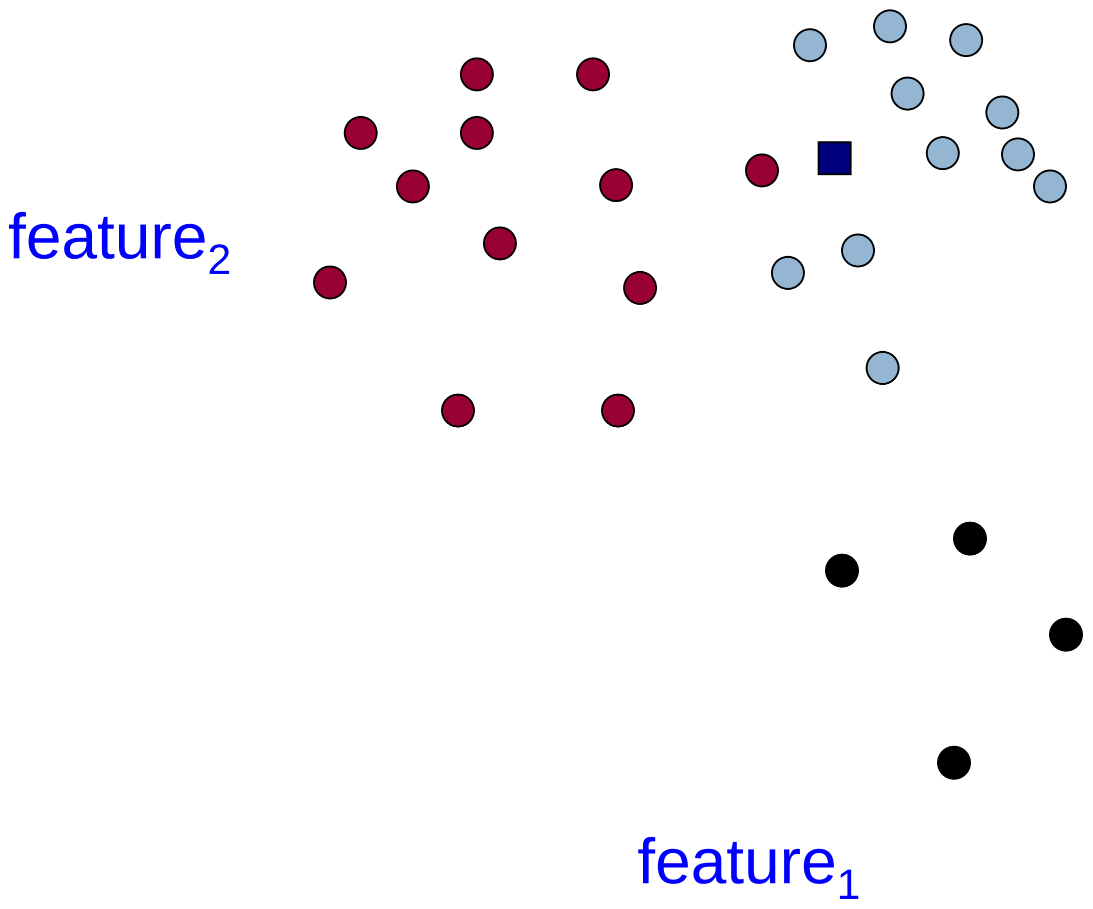
Choosing k

We'd choose red. Do you agree?



Choosing k

What is the label with k = 3?



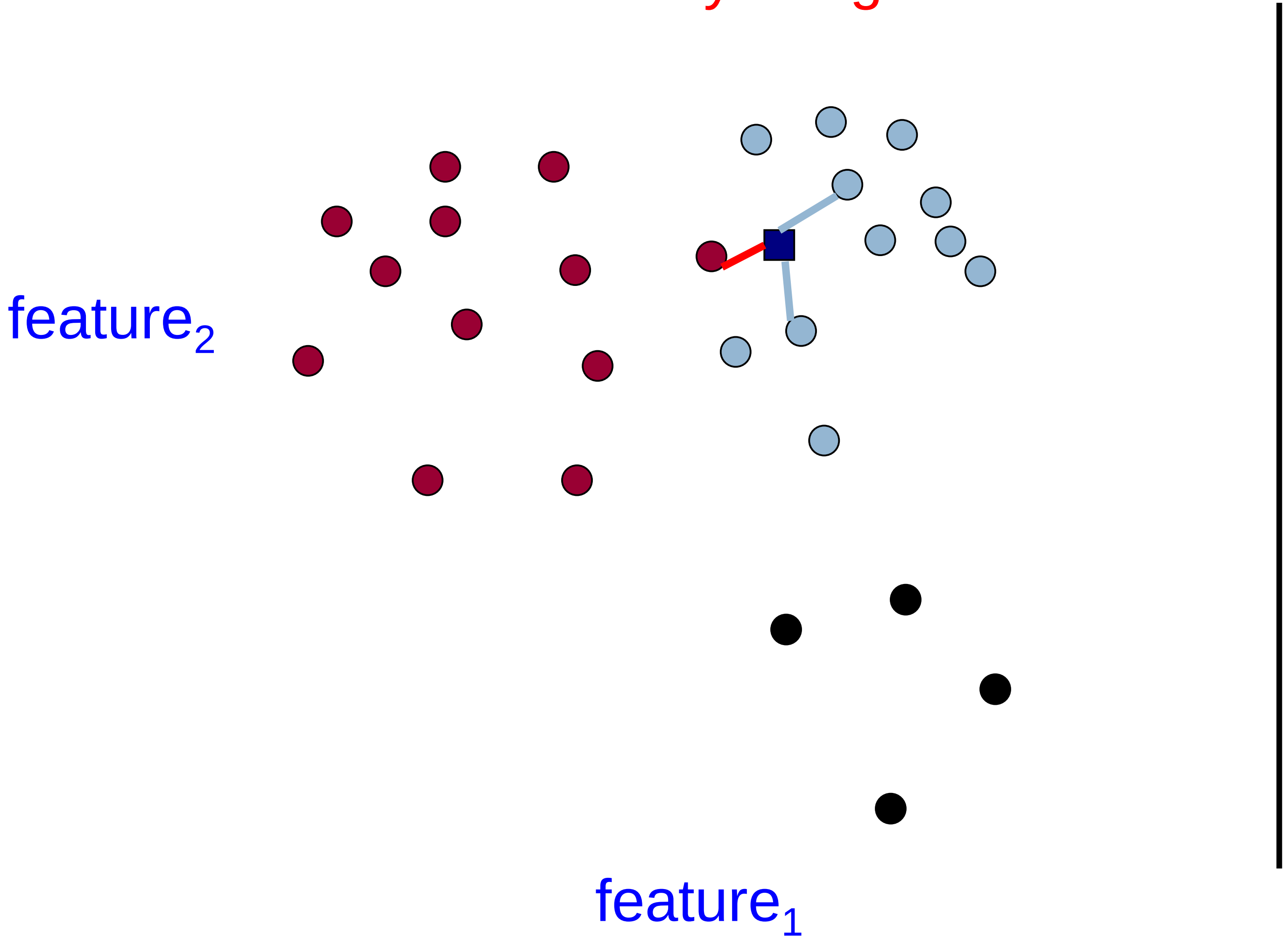
● label 1

● label 2

● label 3

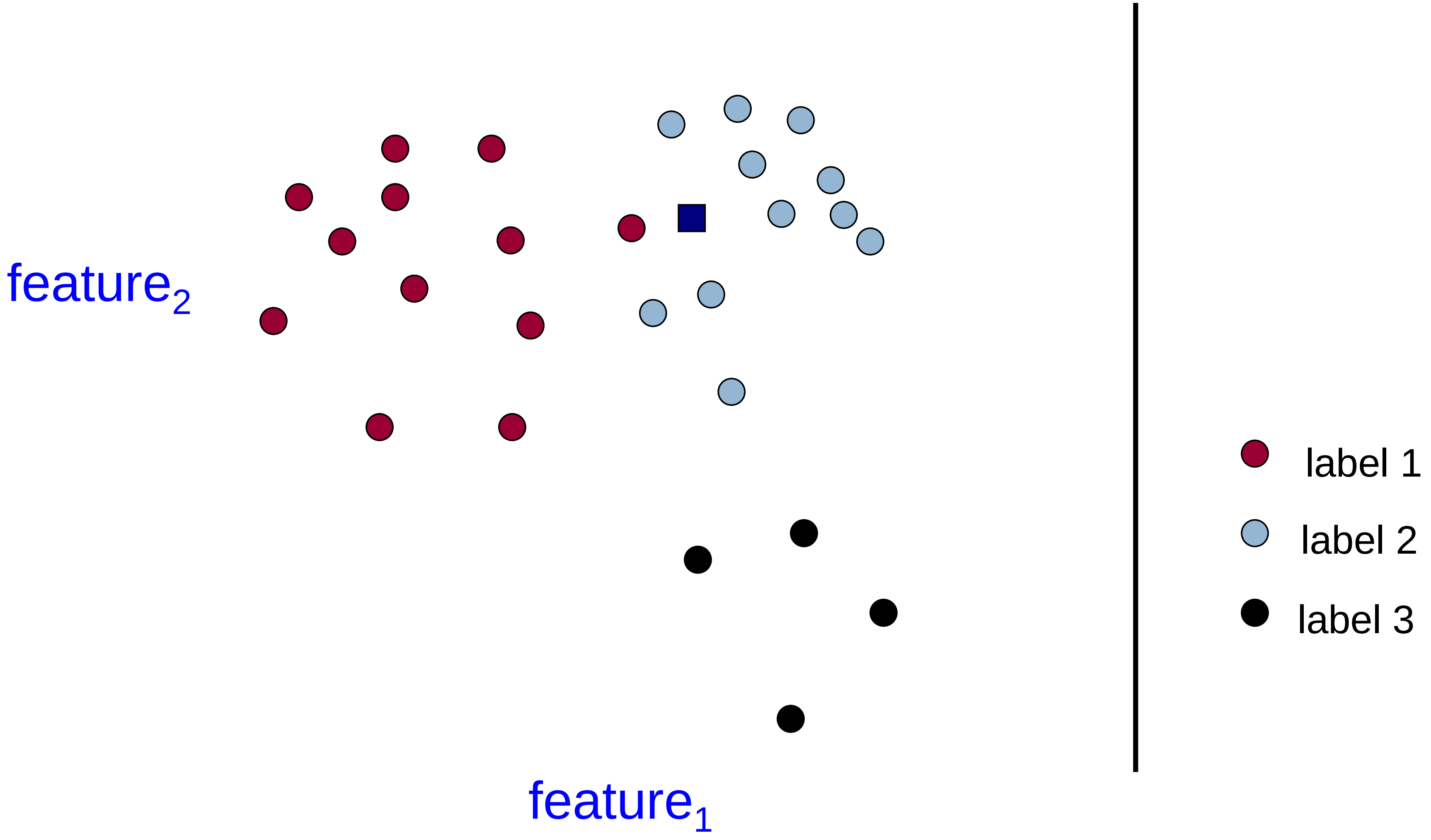
Choosing k

We'd choose blue. Do you agree?



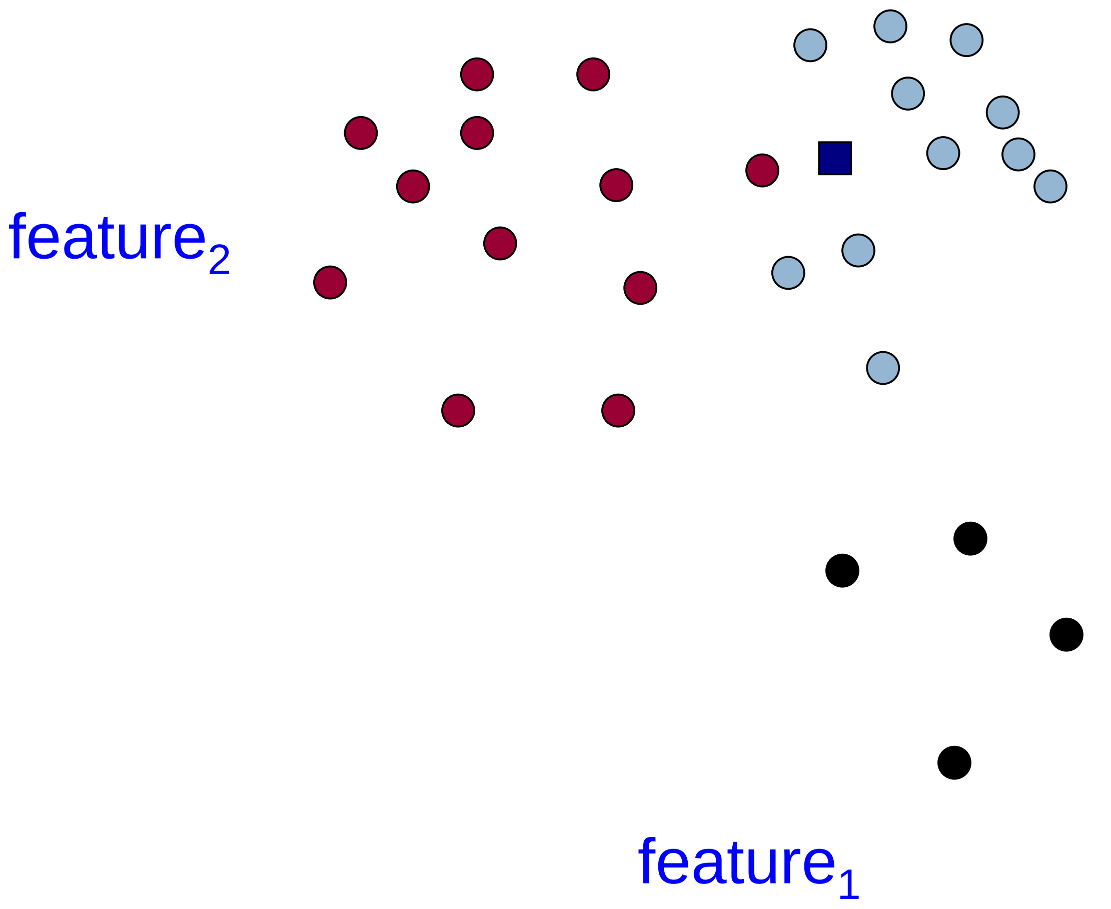
Choosing k

What is the label with k = 100?



Choosing k

We'd choose blue. Do you agree?

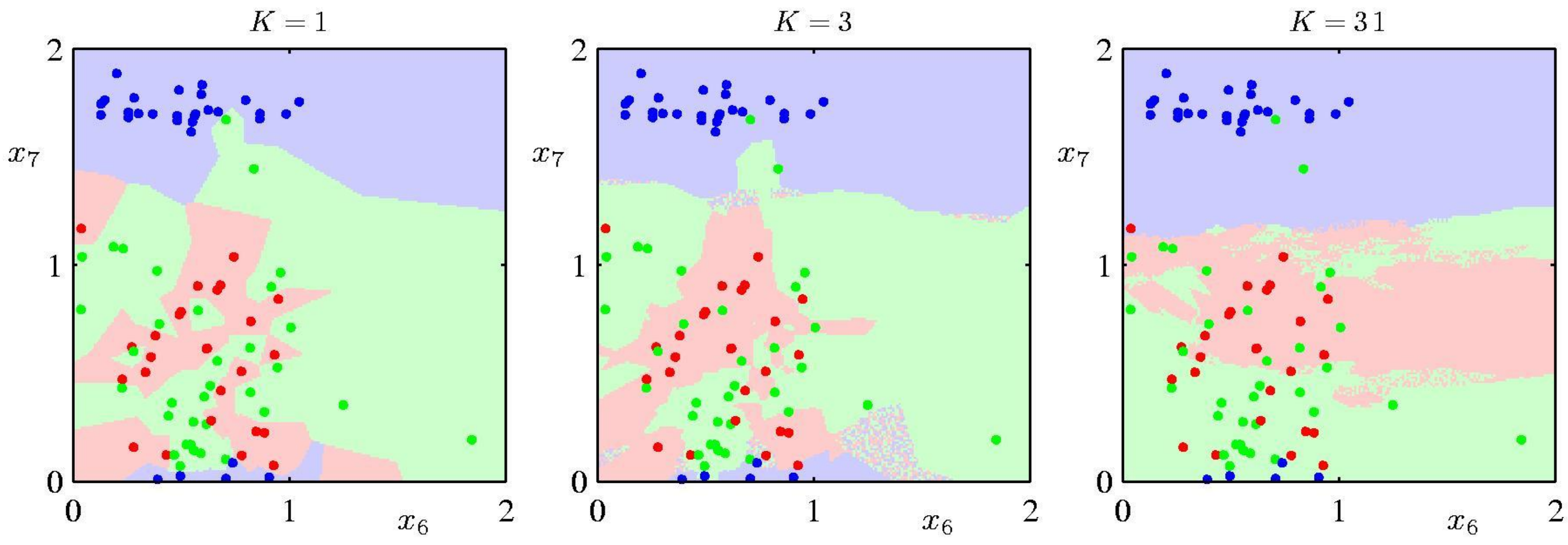


● label 1

● label 2

● label 3

The impact of k



What is the role of k?
How does it relate to overfitting and underfitting?
How did we control this for decision trees?

k-Nearest Neighbor (k-NN)

To classify an example ***d***:

- Find ***k*** nearest neighbors of ***d***
- Choose as the class the majority class within the ***k*** nearest neighbors

How do we choose *k*?

How to pick k

Common heuristics:

- often 3, 5, 7
- choose an odd number to avoid ties

Use development data

k-NN variants

To classify an example ***d***:

- Find ***k*** nearest neighbors of ***d***
- Choose as the class the majority class within the ***k*** nearest neighbors

Any variation ideas?

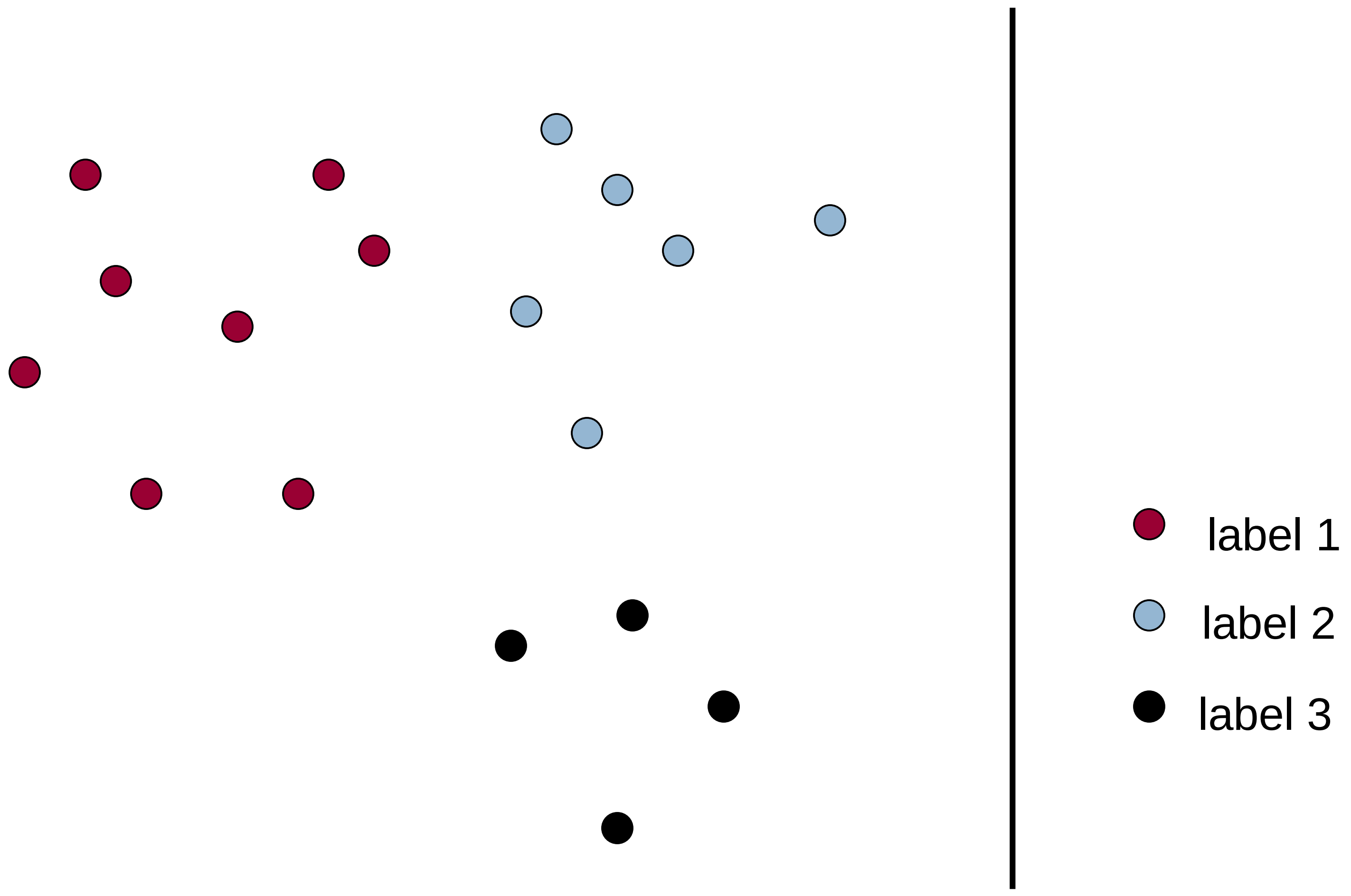
k-NN variations

Instead of *k* nearest neighbors, count majority from all examples within a fixed distance

Weighted *k*-NN:

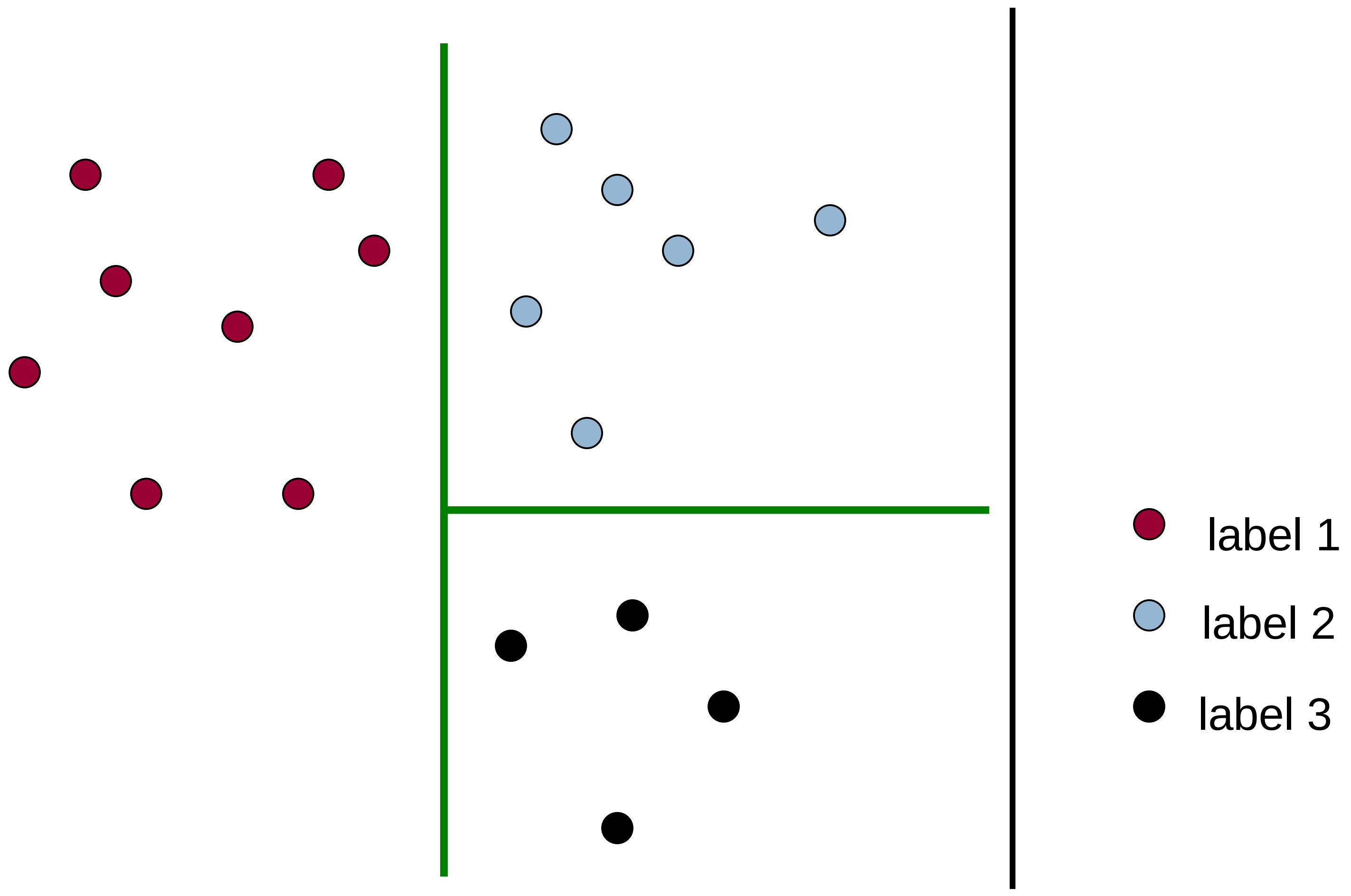
- Right now, all examples within examples are treated equally
- weight the “vote” of the examples, so that closer examples have more vote/weight
- often use some sort of exponential decay

Decision boundaries for decision trees



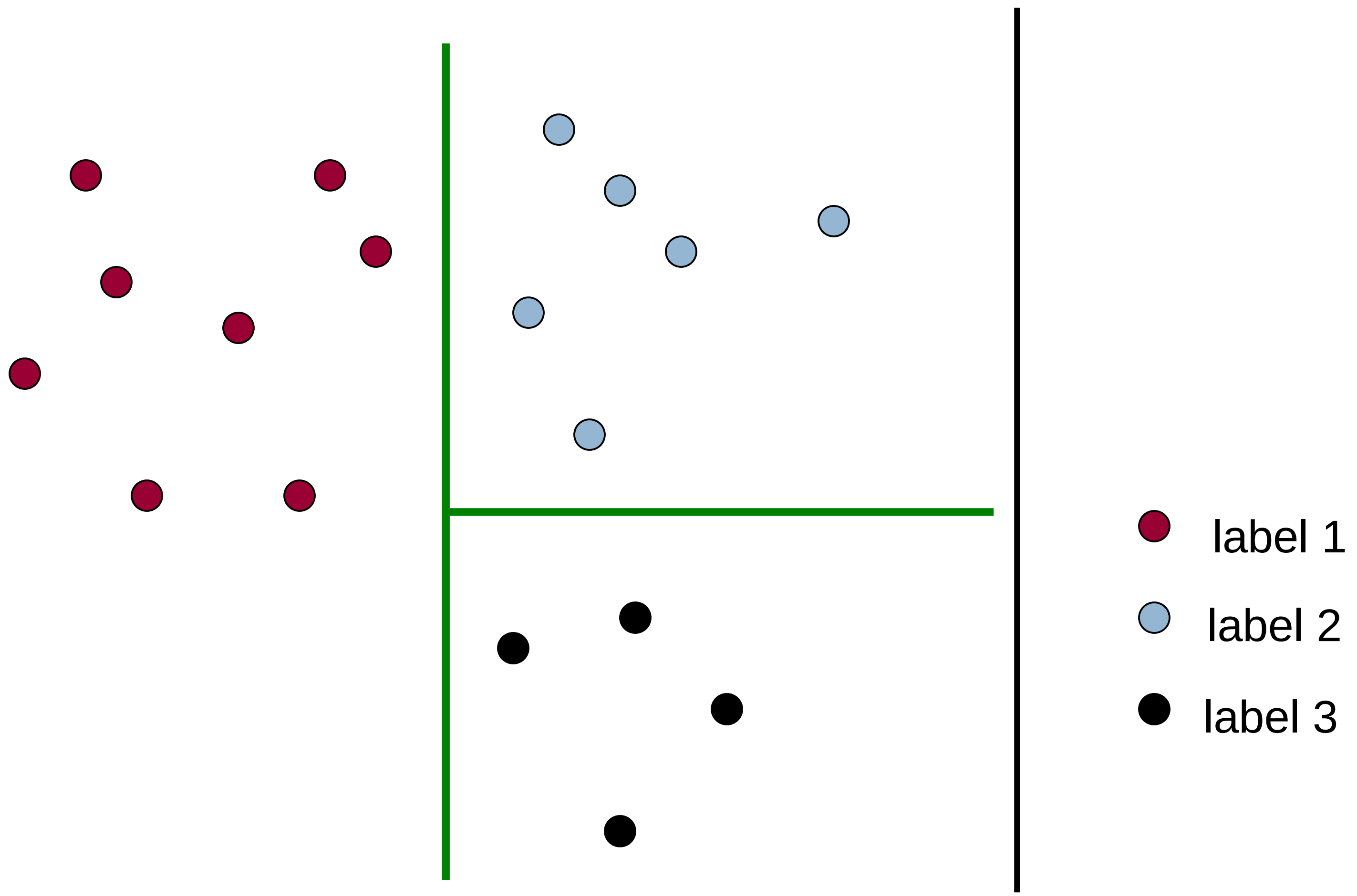
What are the decision boundaries for decision trees like?

Decision boundaries for decision trees



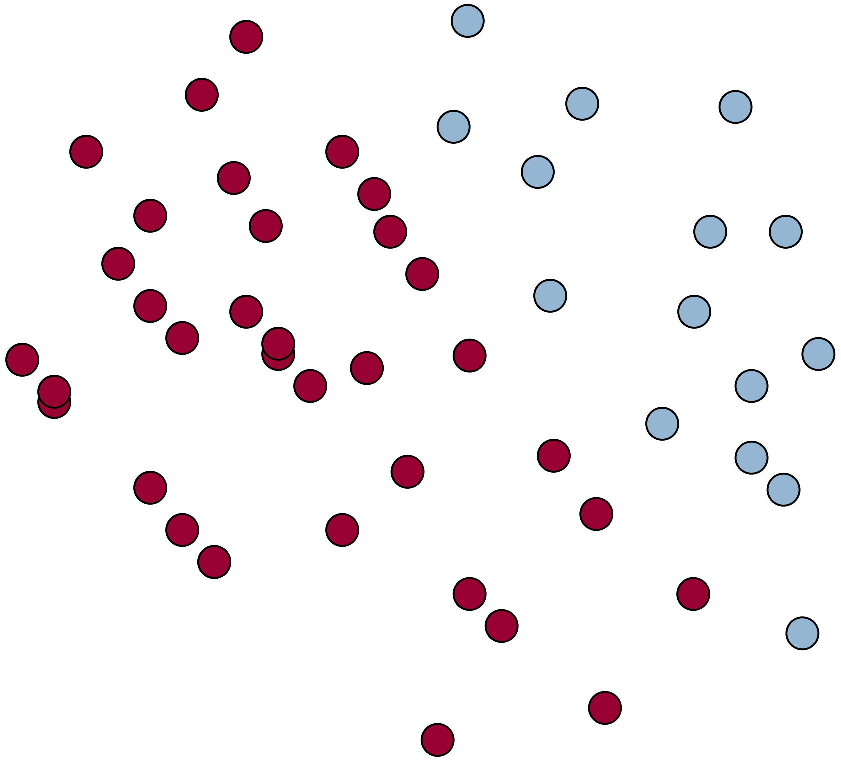
Axis-aligned splits/cuts of the data

Decision boundaries for decision trees



What types of data sets will DT work poorly on?

Problems for DT



Decision trees vs. *k*-NN

Which is faster to train?

Which is faster to classify?

Do they use the features in the same way to label the examples?

Decision trees vs. *k*-NN

Which is faster to train?

k-NN doesn't require any training!

Which is faster to classify?

For most data sets, decision trees

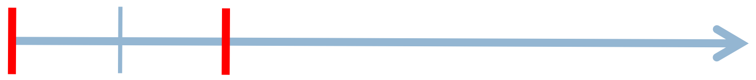
Do they use the features in the same way to label the examples?

k-NN treats all features equally! Decision trees "select" important features

A thought experiment

What is a 100,000-dimensional space like?

You're a 1-D creature, and you decide to buy a 2-unit apartment

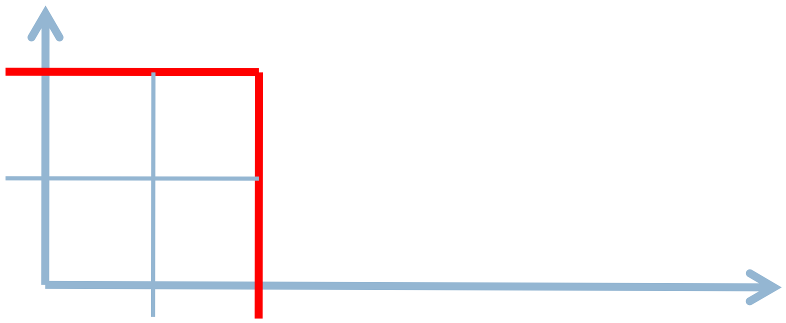


2 rooms (very, skinny rooms)

Another thought experiment

What is a 100,000-dimensional space like?

Your job's going well and you're making good money. You upgrade to a 2-D apartment with 2-units per dimension

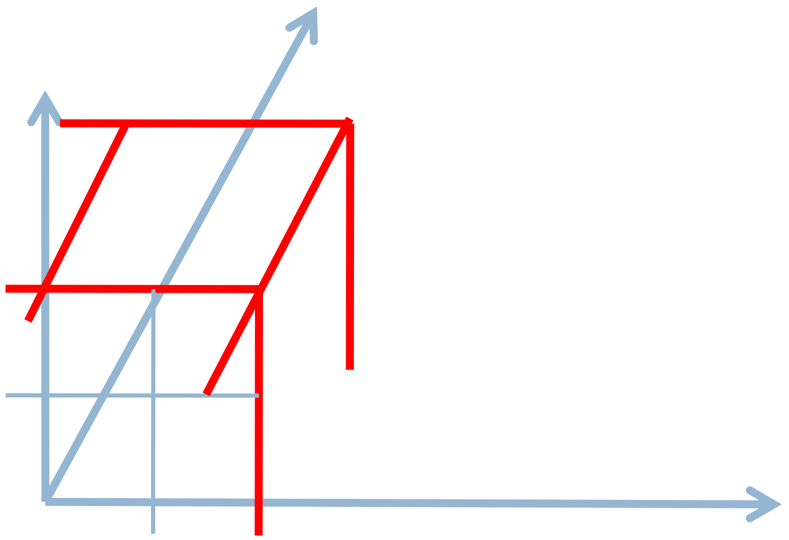


4 rooms (very, flat rooms)

Another thought experiment

What is a 100,000-dimensional space like?

You get promoted again and start having kids and decide to upgrade to another dimension.



8 rooms (very, normal rooms)

Each time you add a dimension, the amount of space you have to work with goes up exponentially

Another thought experiment

What is a 100,000-dimensional space like?

Larry Page steps down as CEO of google and they ask you if you'd like the job. You decide to upgrade to a 100,000 dimensional apartment.

How much room do you have? Can you have a big party?

$2^{100,000}$ rooms (it's very quiet and lonely…) = ~10^{30} rooms per person if you invited everyone on the planet

The challenge

Our intuitions about space/distance don’t scale with dimensions!

Important to Watch Videos

- A.I. Experiments: Visualizing High-Dimensional Space
- https://www.youtube.com/watch?v=wvsE8jm1GzE

- Neural Network 3D Simulation
- https://www.youtube.com/watch?v=3JQ3hYko51Y