Ok, let's start with some data…

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
…	…	…	…	…

We want to use the variables (Chest Pain, Good Blood Circulation, etc.)…

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
…	…	…	…	…

…to predict if someone has heart disease.

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
…	…	…	…	…

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
…	…	…	…	…

Then, when a new patient shows up…

…we can measure these variables…

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
…	…	…	…	…

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	No	No	168	

…and predict if they have heart disease or not.

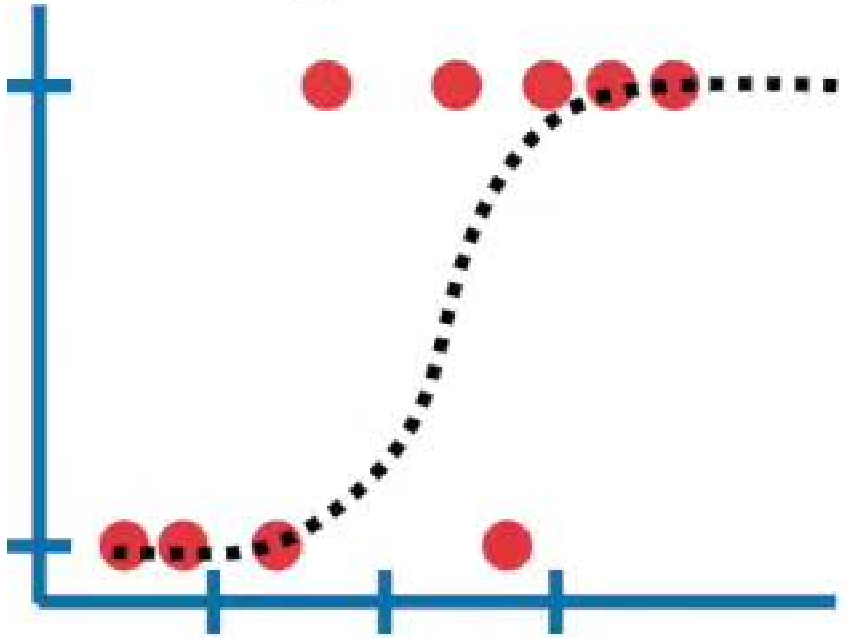
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
…	…	…	…	…

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	No	No	168	???

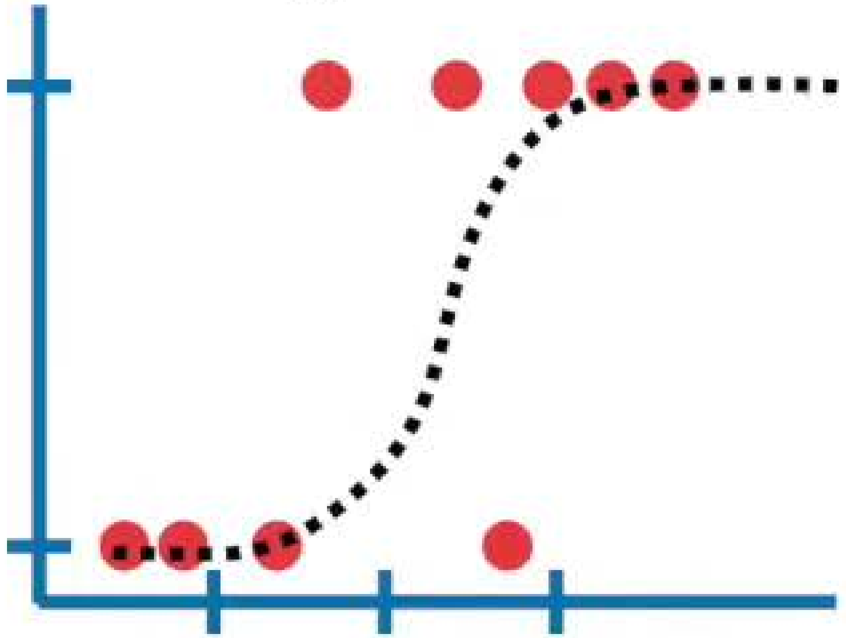
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
…	…	…	…	…

However, first we have to decide which machine learning method would be best…

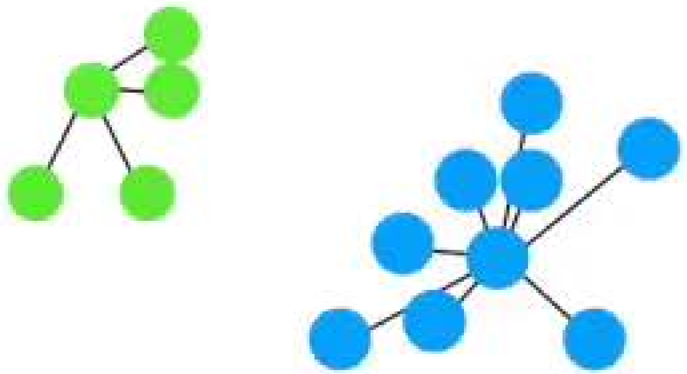
We could use Logistic Regression…



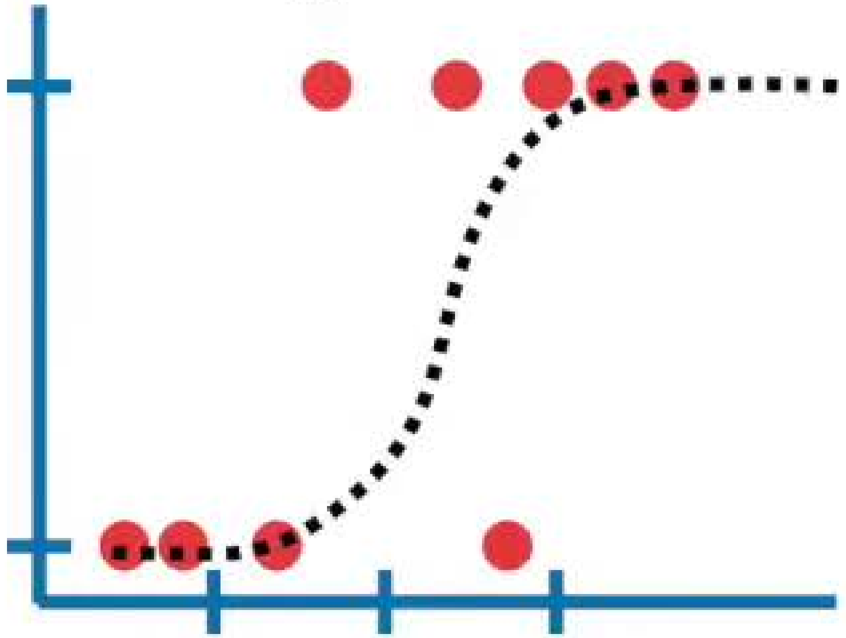
We could use Logistic Regression…



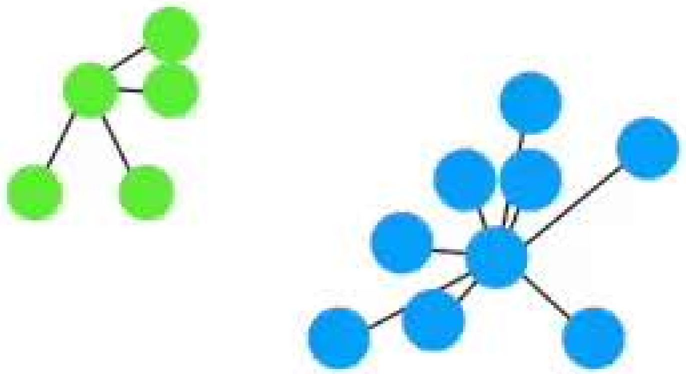
…or K-nearest neighbors…



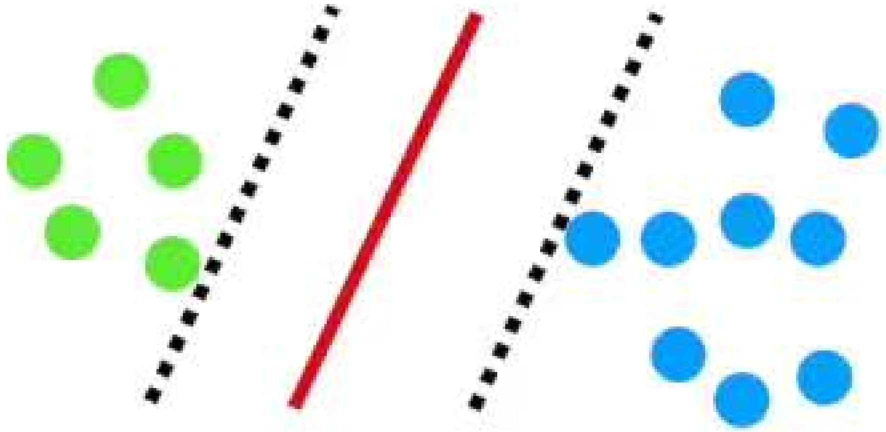
We could use Logistic Regression…



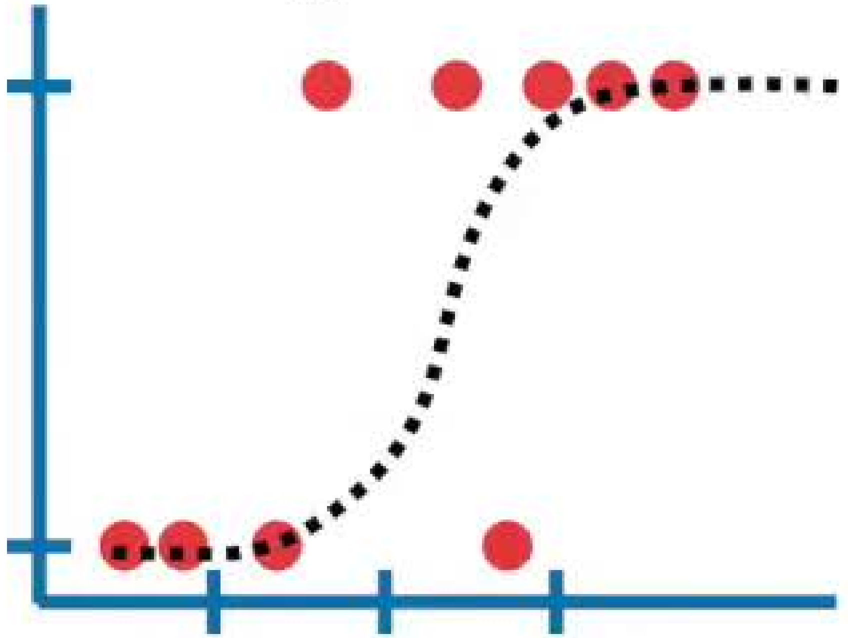
…or K-nearest neighbors…



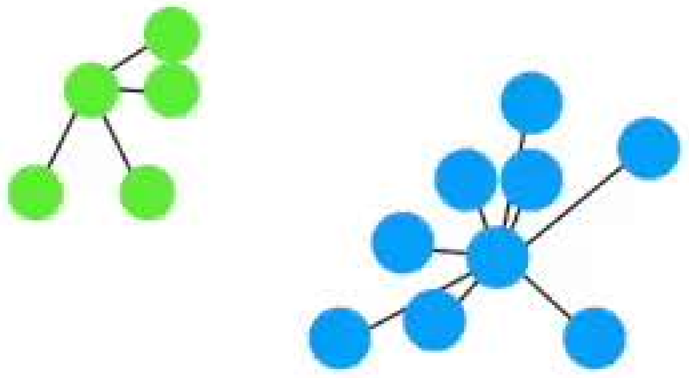
…or support vector machines (SVM)…



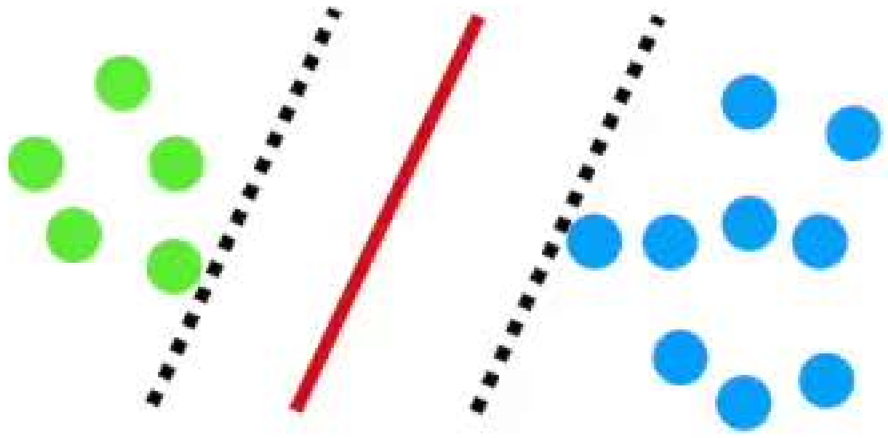
We could use Logistic Regression…



…or K-nearest neighbors…

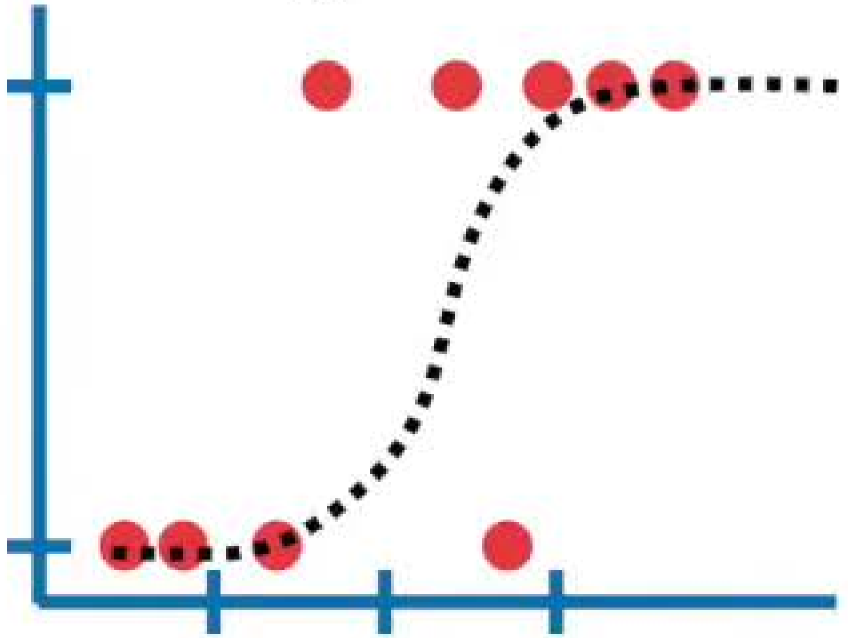


…or support vector machines (SVM)…

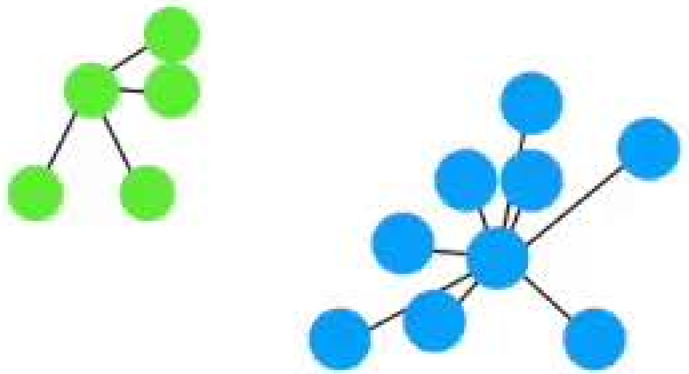


…and many more machine learning methods…

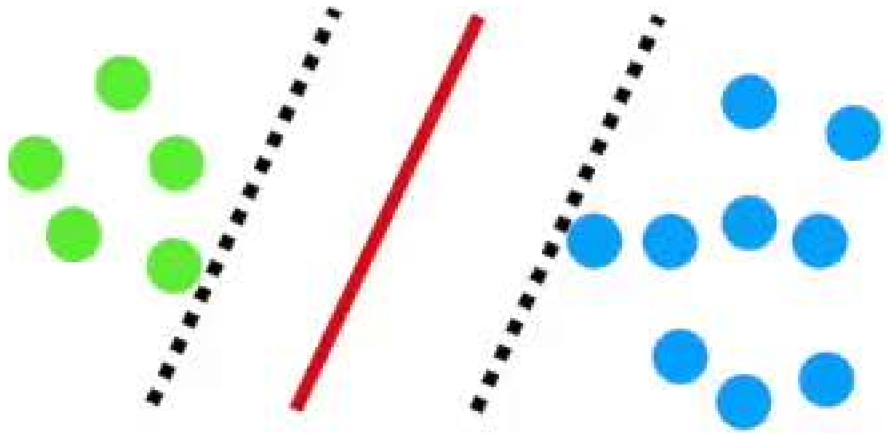
We could use Logistic Regression…



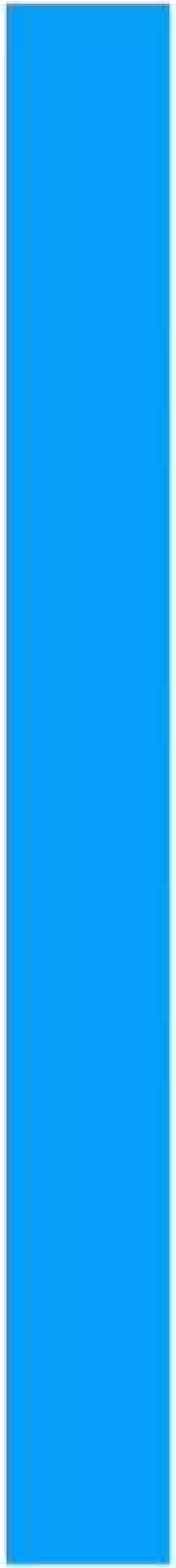
…or K-nearest neighbors…



…or support vector machines (SVM)…



Cross validation allows us to compare different machine learning methods and get a sense of how well they will work in practice.



Imagine that this **blue column** represented all of the data that we have collected about people with and without heart disease.

We need it to do two (2)
things with this data:

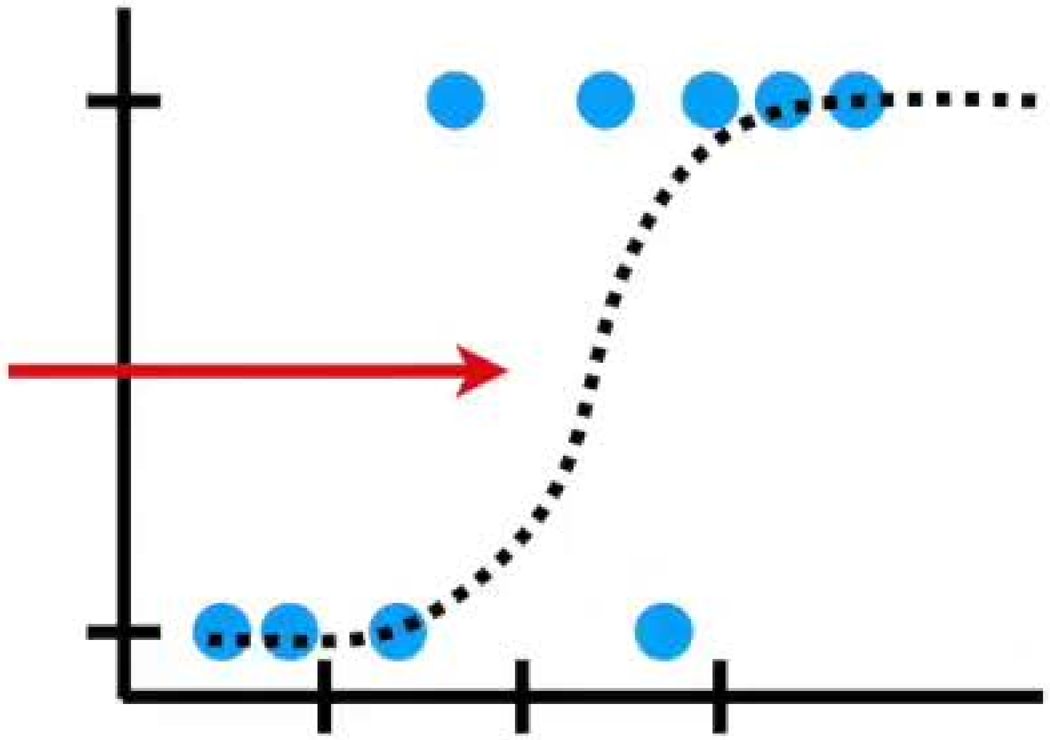
We need it to do two (2) things with this data:

1) Estimate the parameters for the machine learning methods.

We need it to do two (2) things with this data:

1) Estimate the parameters for the machine learning methods.

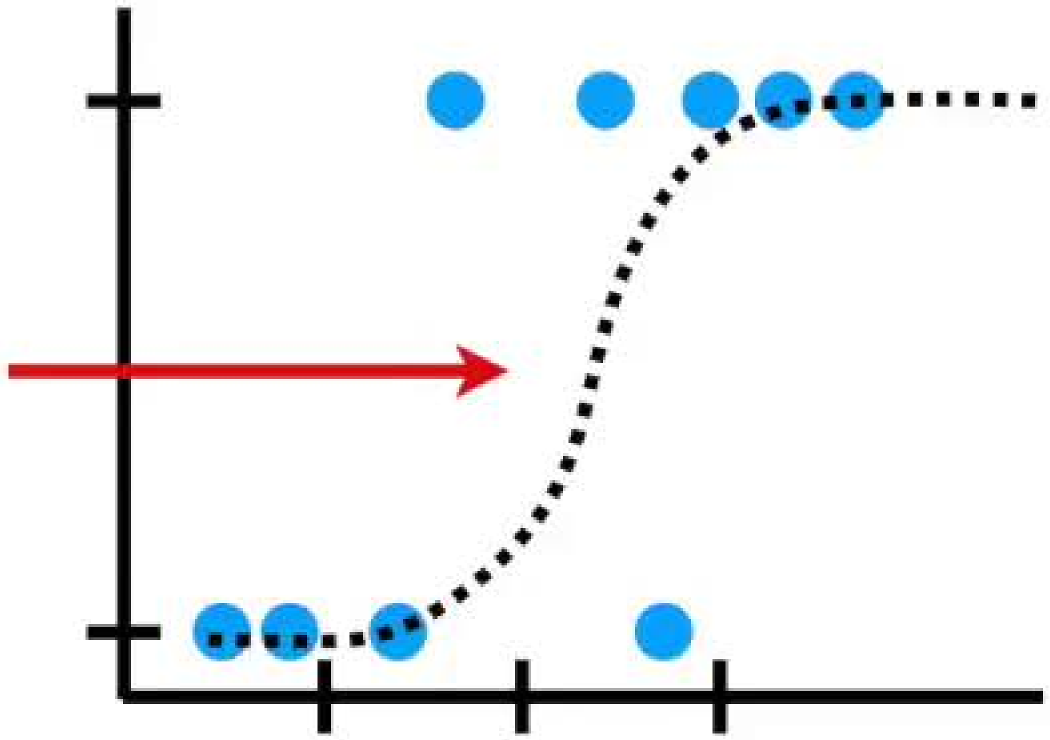
In other words, to use logistic regression, we have to use some of the data to estimate the shape of this curve…

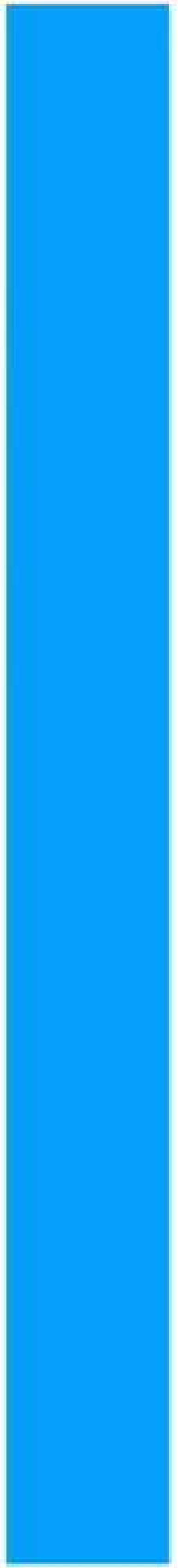


We need it to do two (2) things with this data:

1) Estimate the parameters for the machine learning methods.

In machine learning lingo, estimating parameters is called "**training** the algorithm."





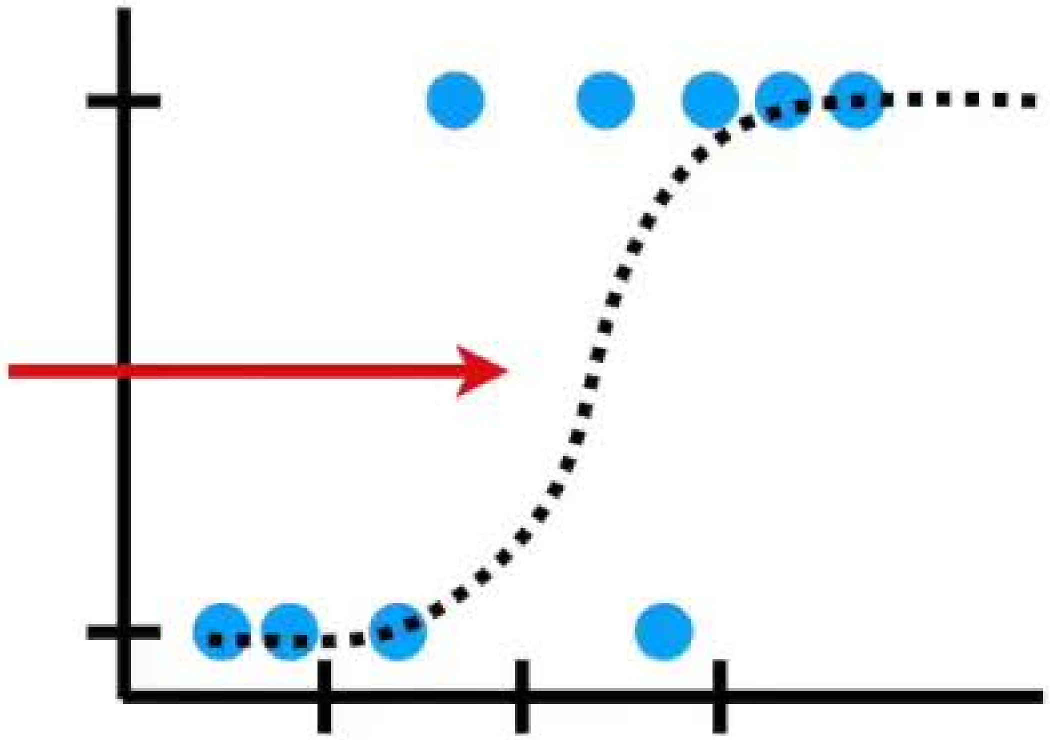
We need it to do two (2) things with this data:

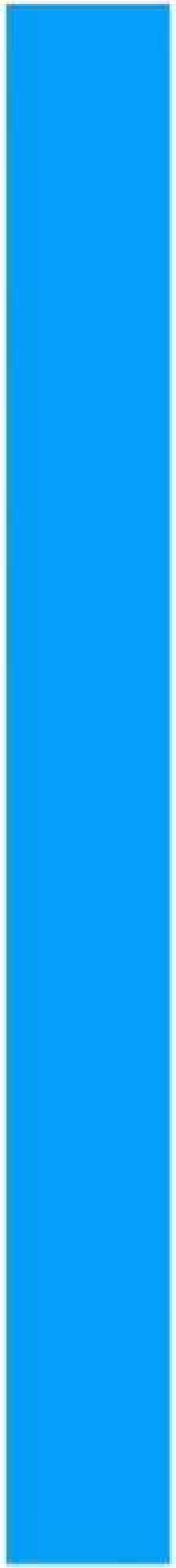
1) Estimate the parameters for the machine learning methods.
2) Evaluate how well the machine learning methods work.

We need it to do two (2) things with this data:

1) Estimate the parameters for the machine learning methods.
2) Evaluate how well the machine learning methods work.

In other words, we need to find out if this curve will do a good job categorizing new data.

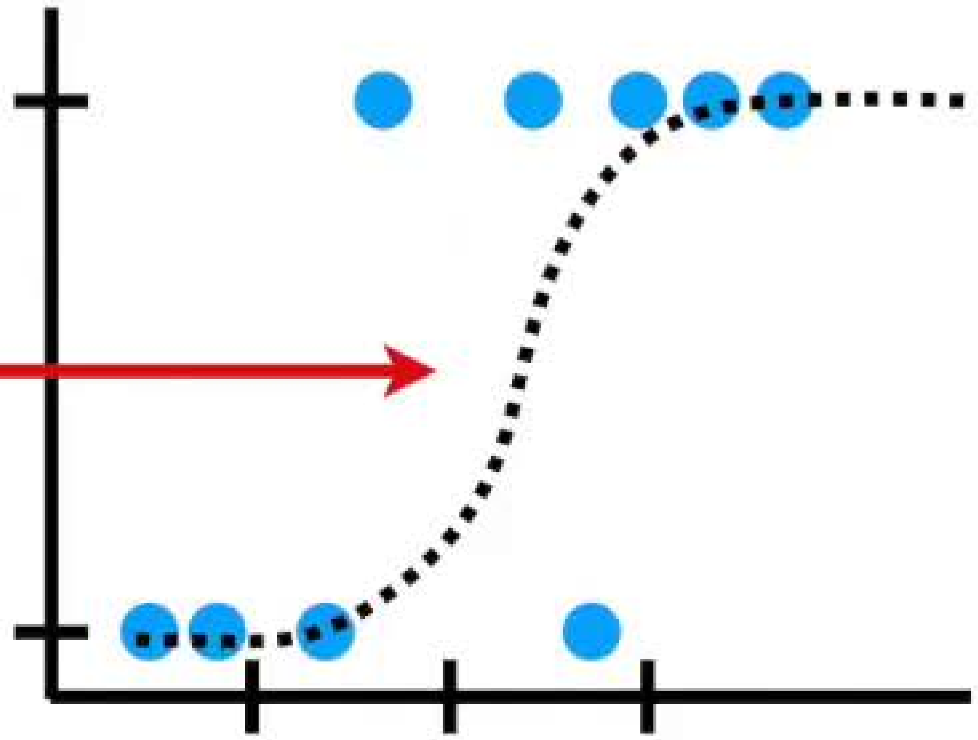




We need it to do two (2) things with this data:

1) Estimate the parameters for the machine learning methods.
2) Evaluate how well the machine learning methods work.

In machine learning lingo, evaluating a method is called "**testing** the algorithm".



Thus, using machine learning lingo, we need the data to…

Thus, using machine learning lingo, we need the data to…

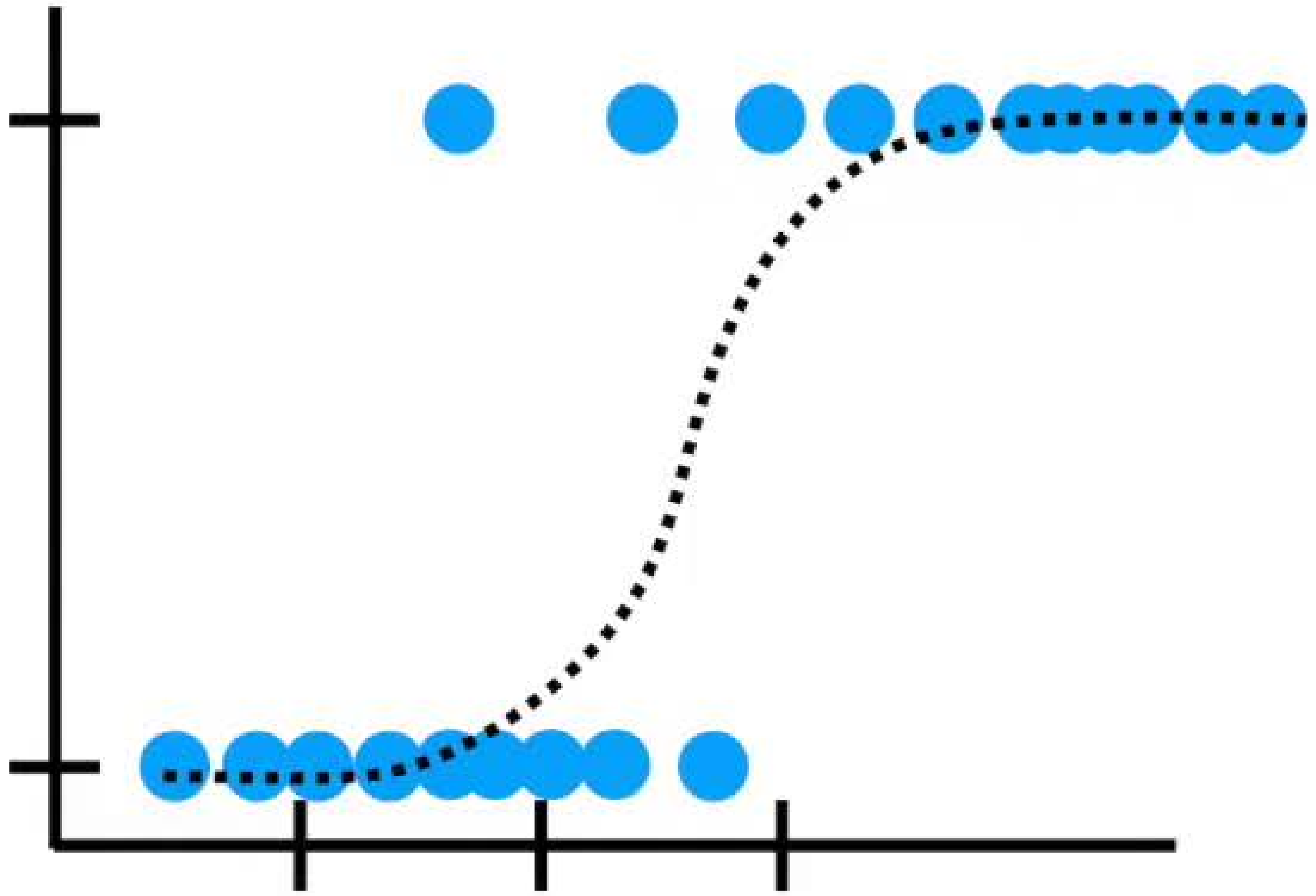
1) **Train** the machine learning methods.

Thus, using machine learning lingo, we need the data to…

1) **Train** the machine learning methods.

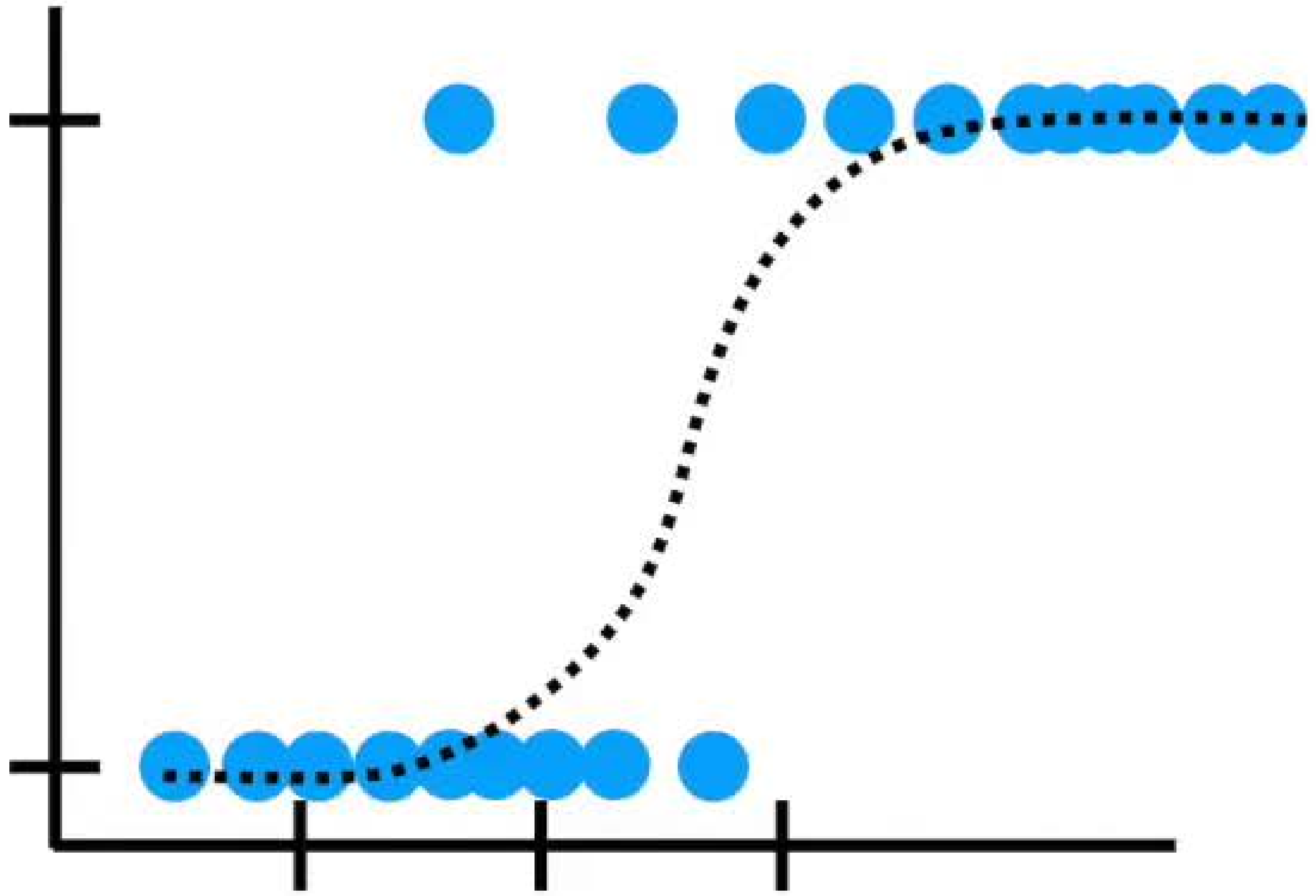
2) **Test** the machine learning methods.

A terrible approach would be to use all of the data to estimate the parameters (i.e. train the algorithm)...

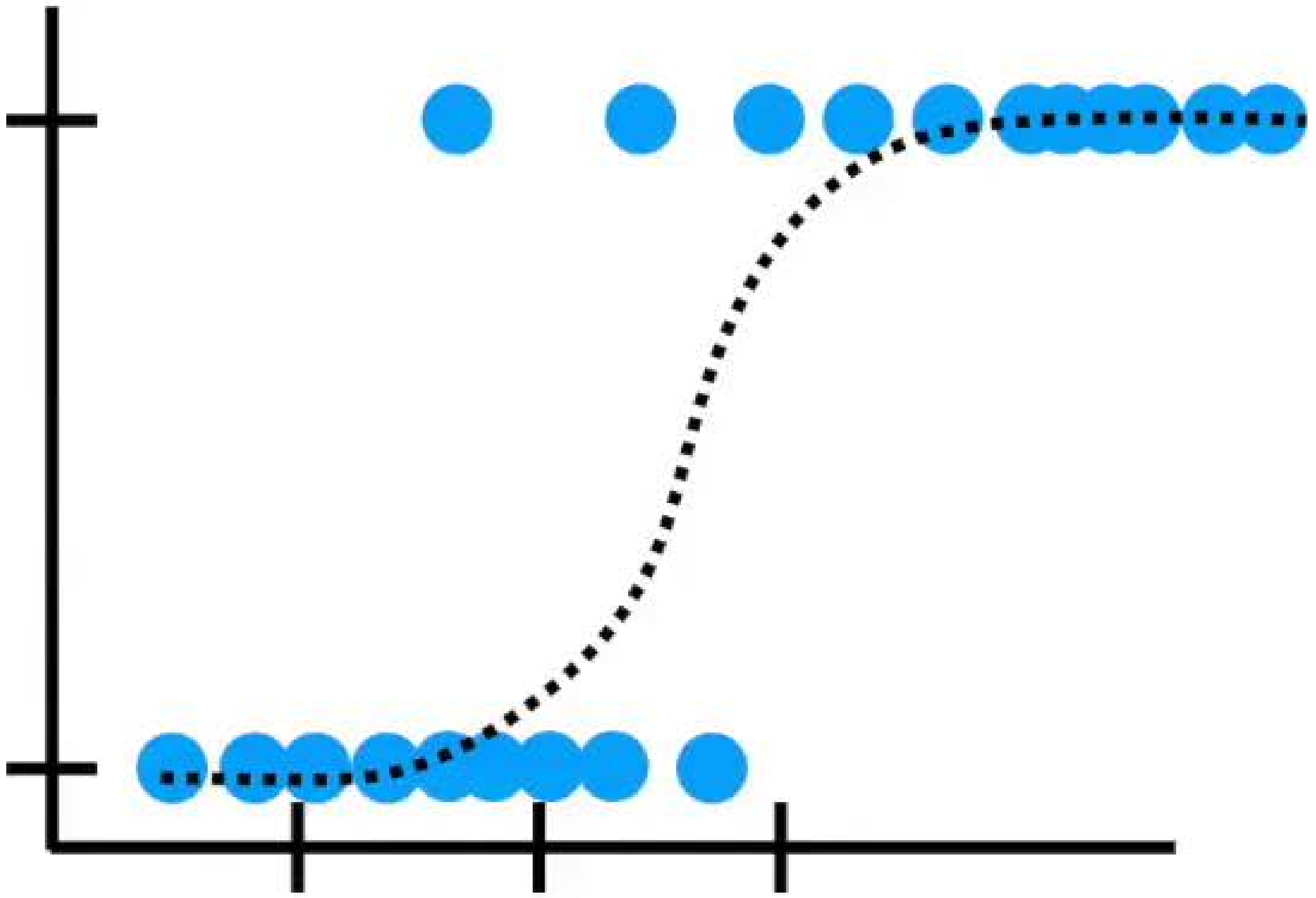


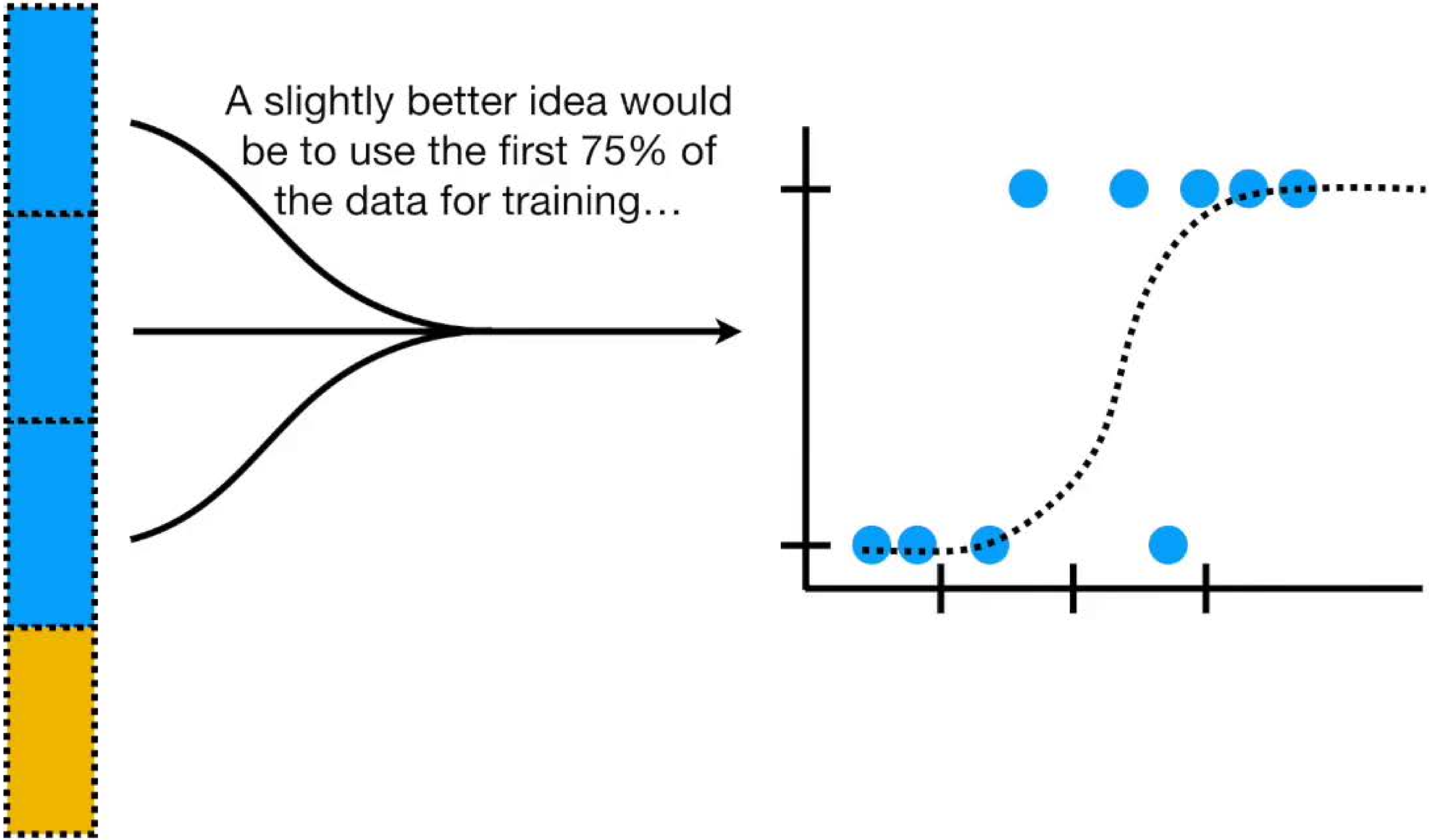
A terrible approach would be to use all of the data to estimate the parameters (i.e. train the algorithm)…

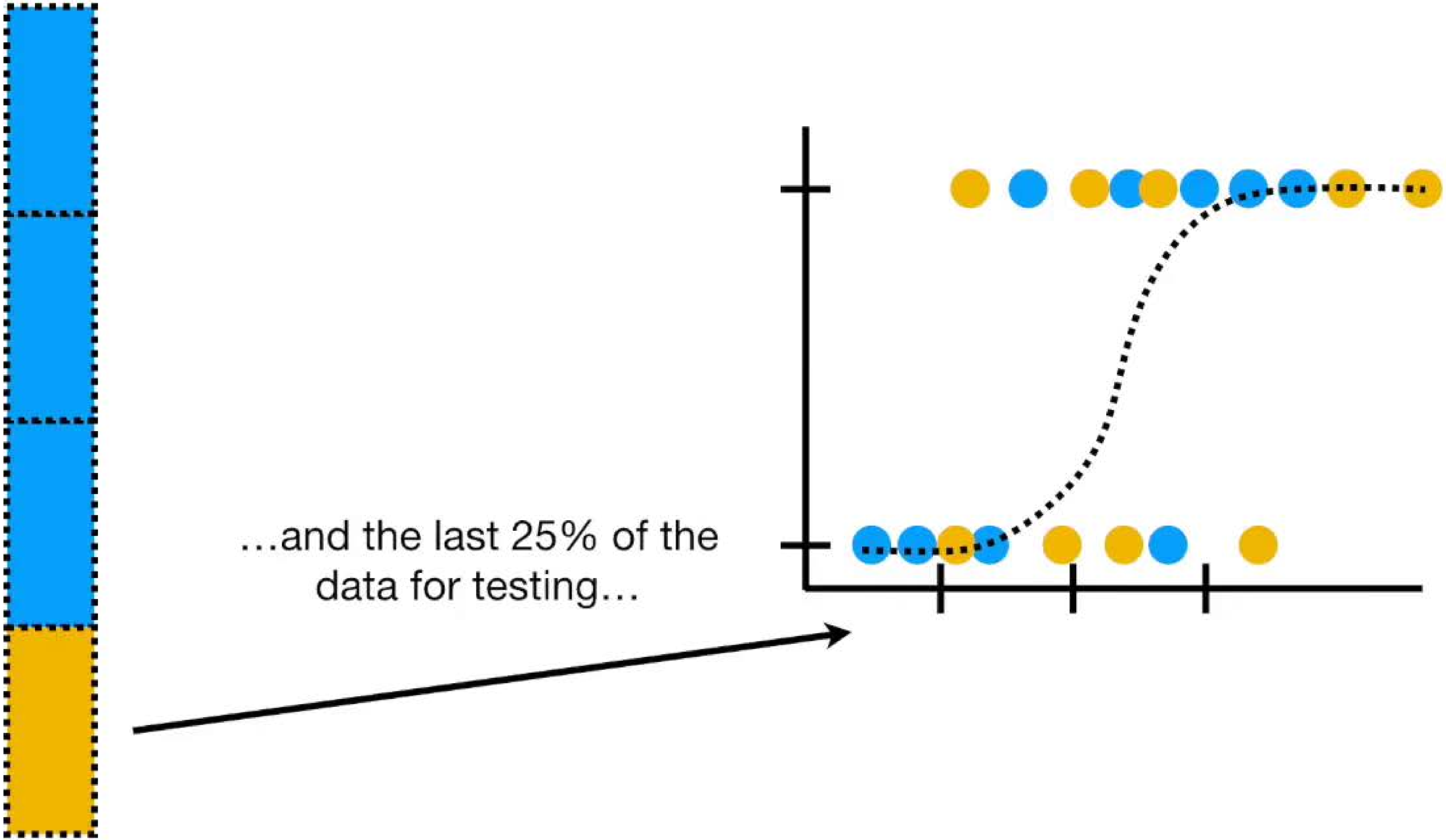
…because then there wouldn't be any data left to test the method.

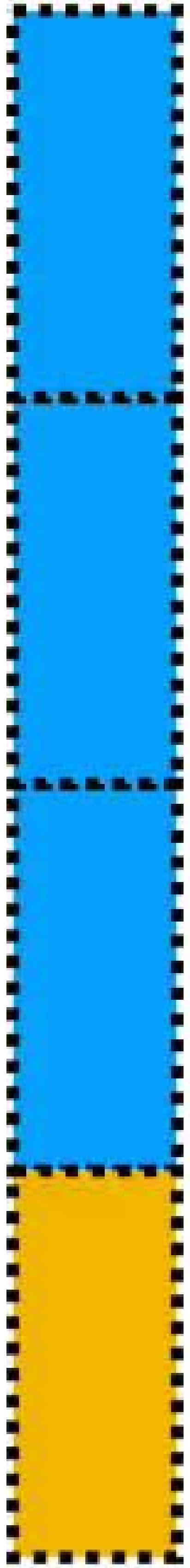


Reusing the same data for both training and testing is a bad idea because we need to know how the method will work on data it wasn't trained on.

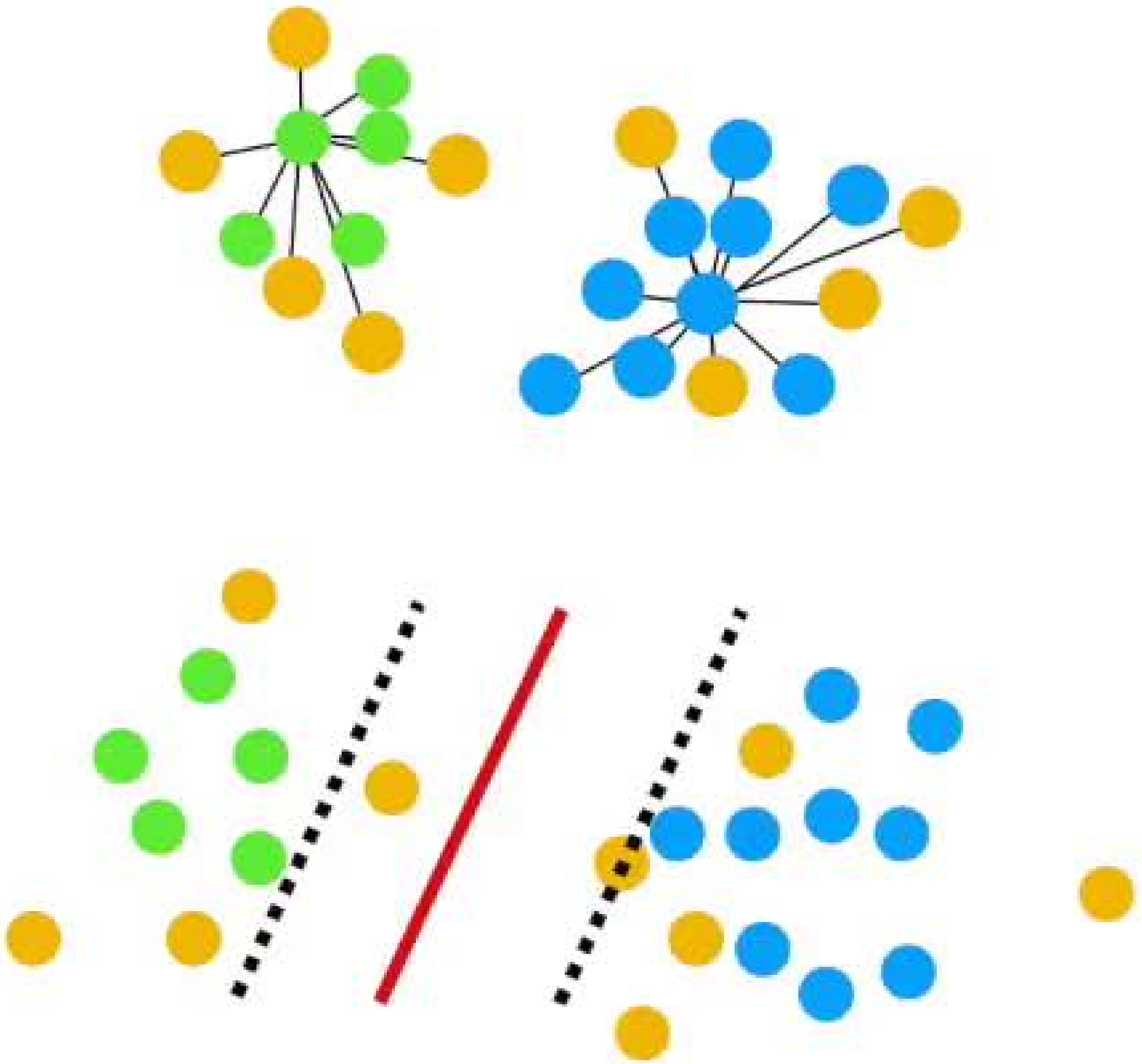


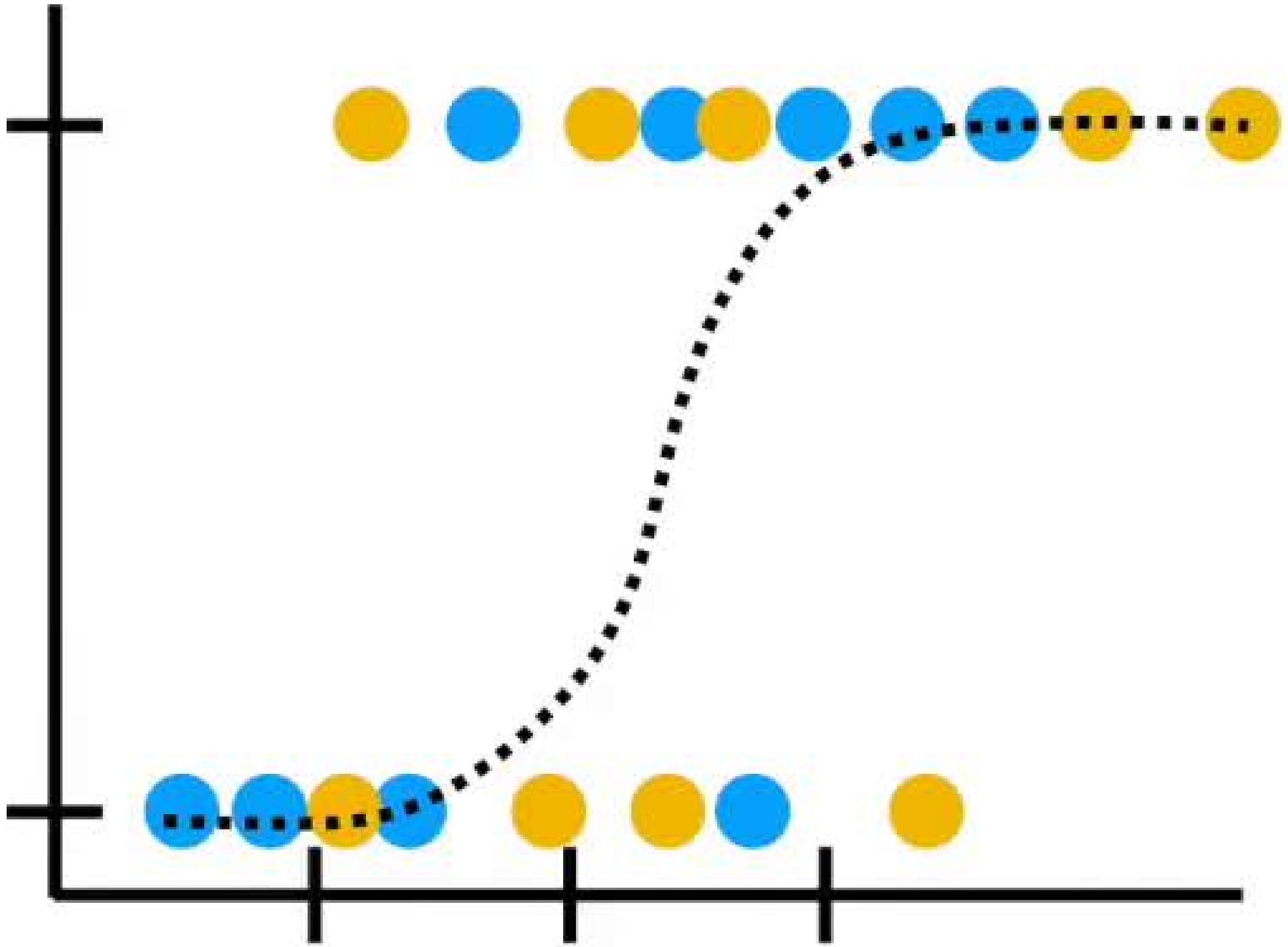

A slightly better idea would be to use the first 75% of the data for training…

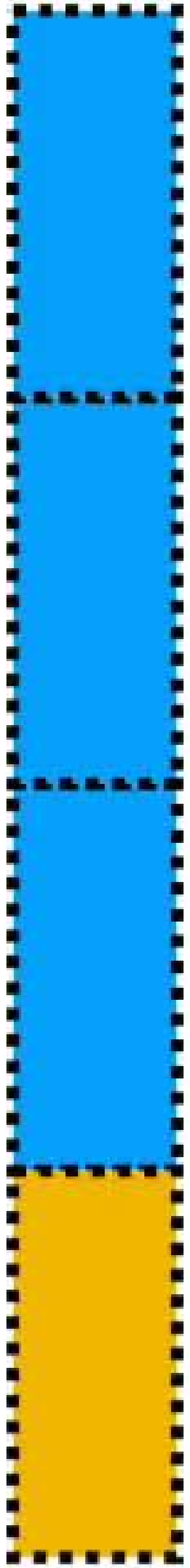

…and the last 25% of the data for testing…



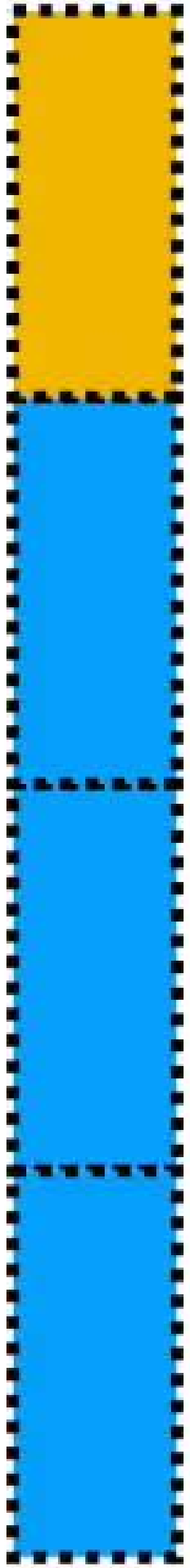
We could then compare methods by seeing how well each one categorized the test data.



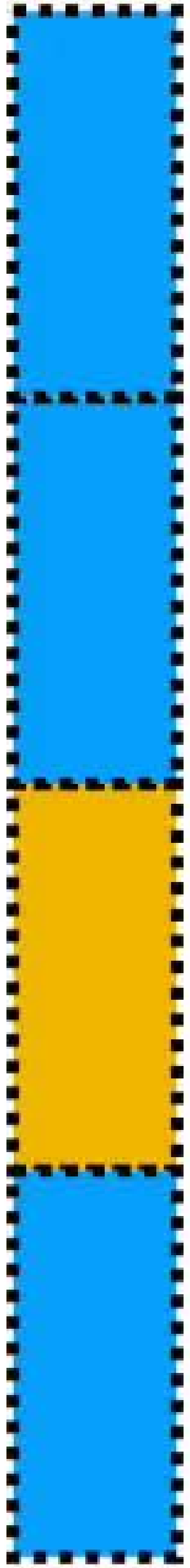




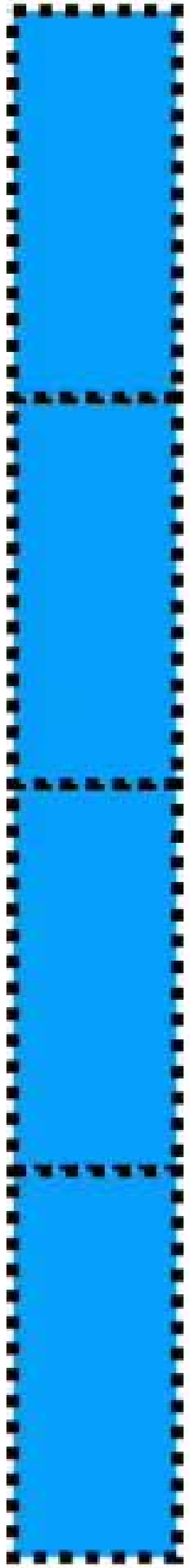
But how do we know that using the first 75% of the data for training and the last 25% of the data for testing is the best way to divide up the data?



What if we used the first 25% of the data for testing?

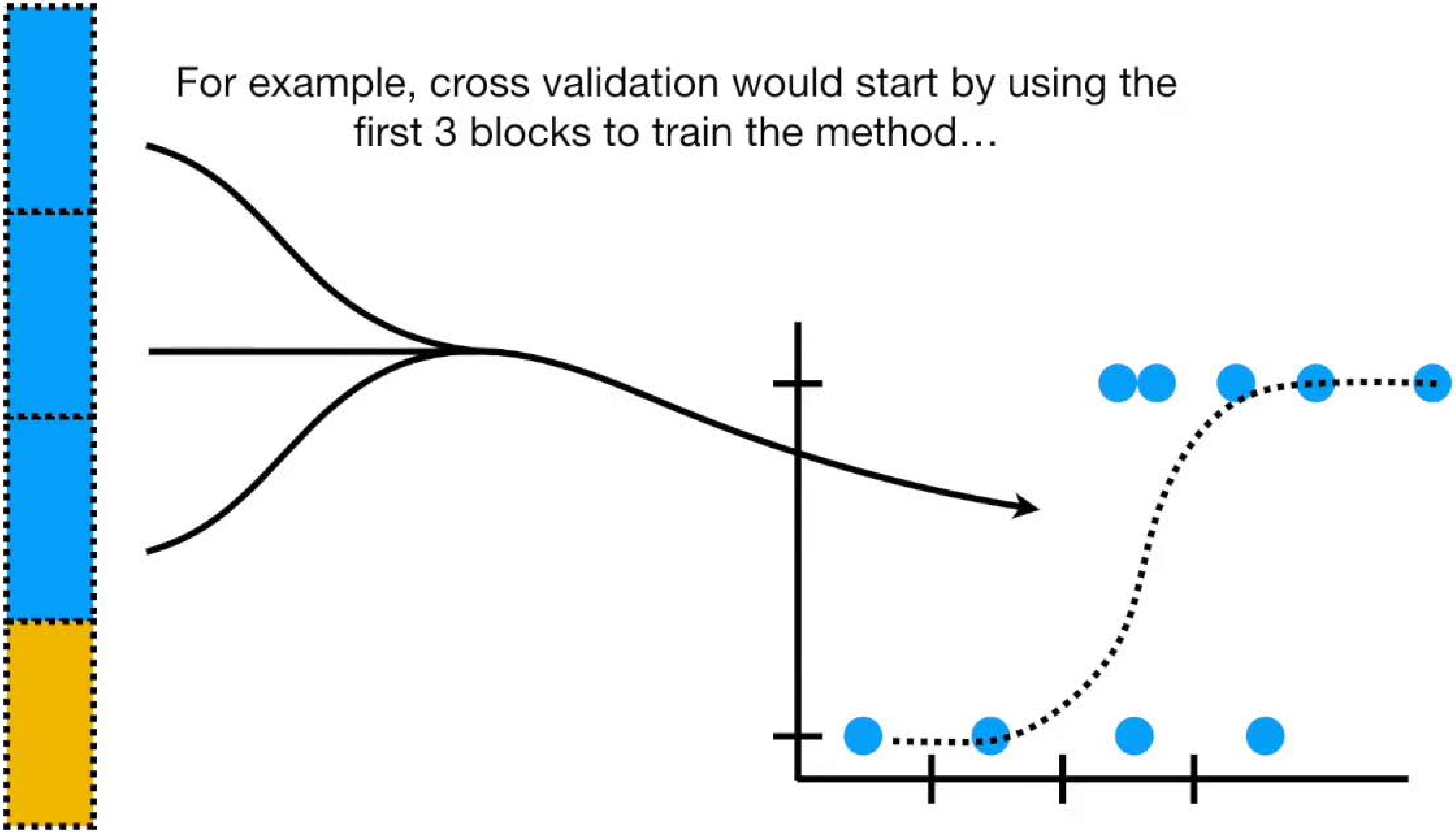


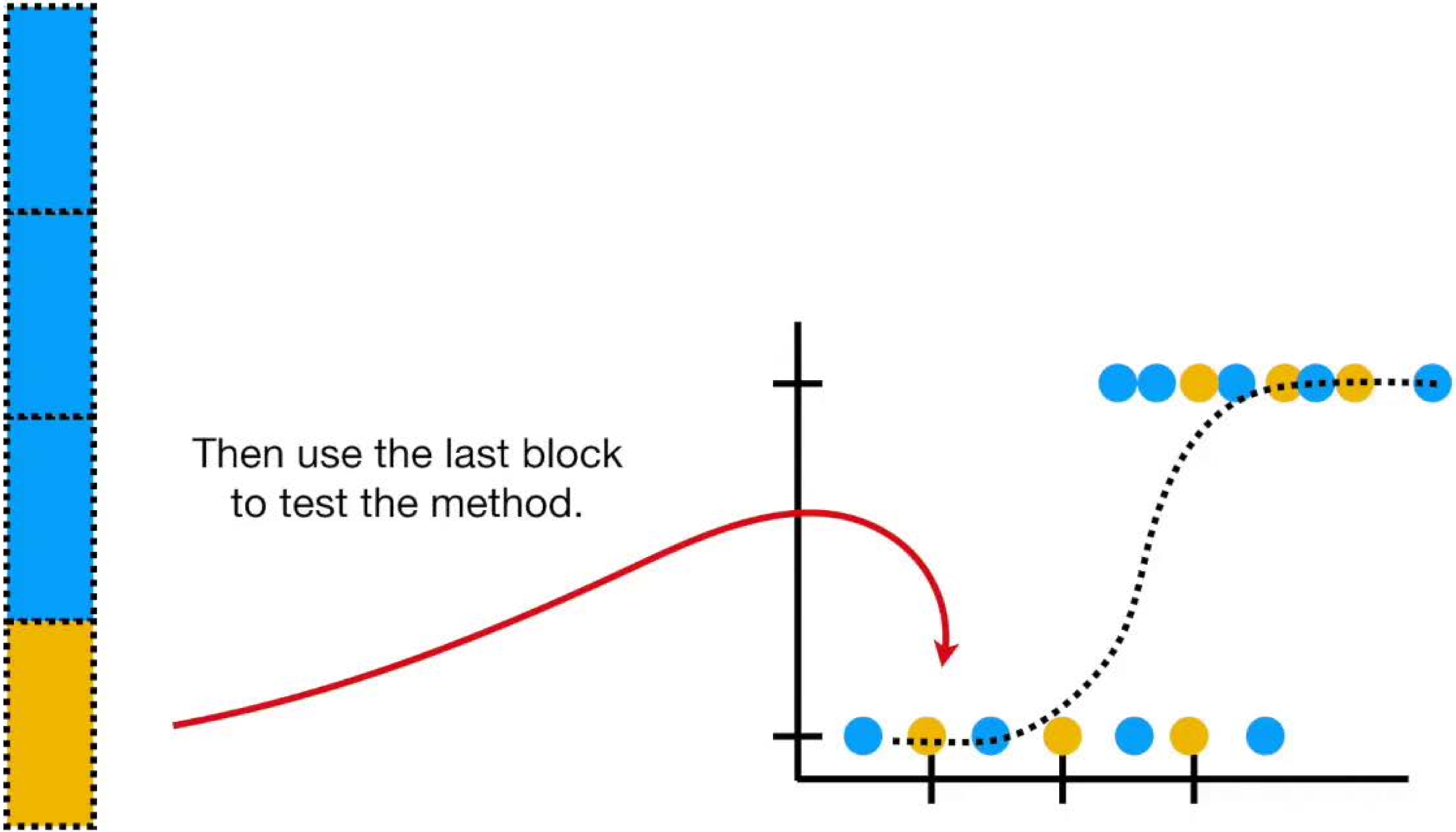
Or what about one of these
middle blocks?

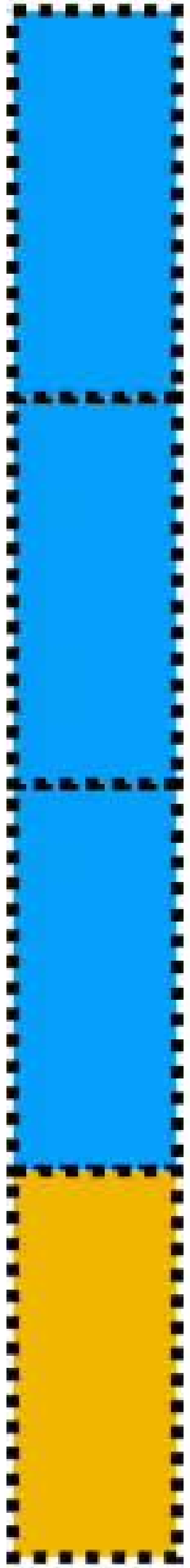


Rather than worry too much about which block would be best for testing, cross validation uses them all, one at a time, and summarizes the results at the end.

For example, cross validation would start by using the first 3 blocks to train the method…



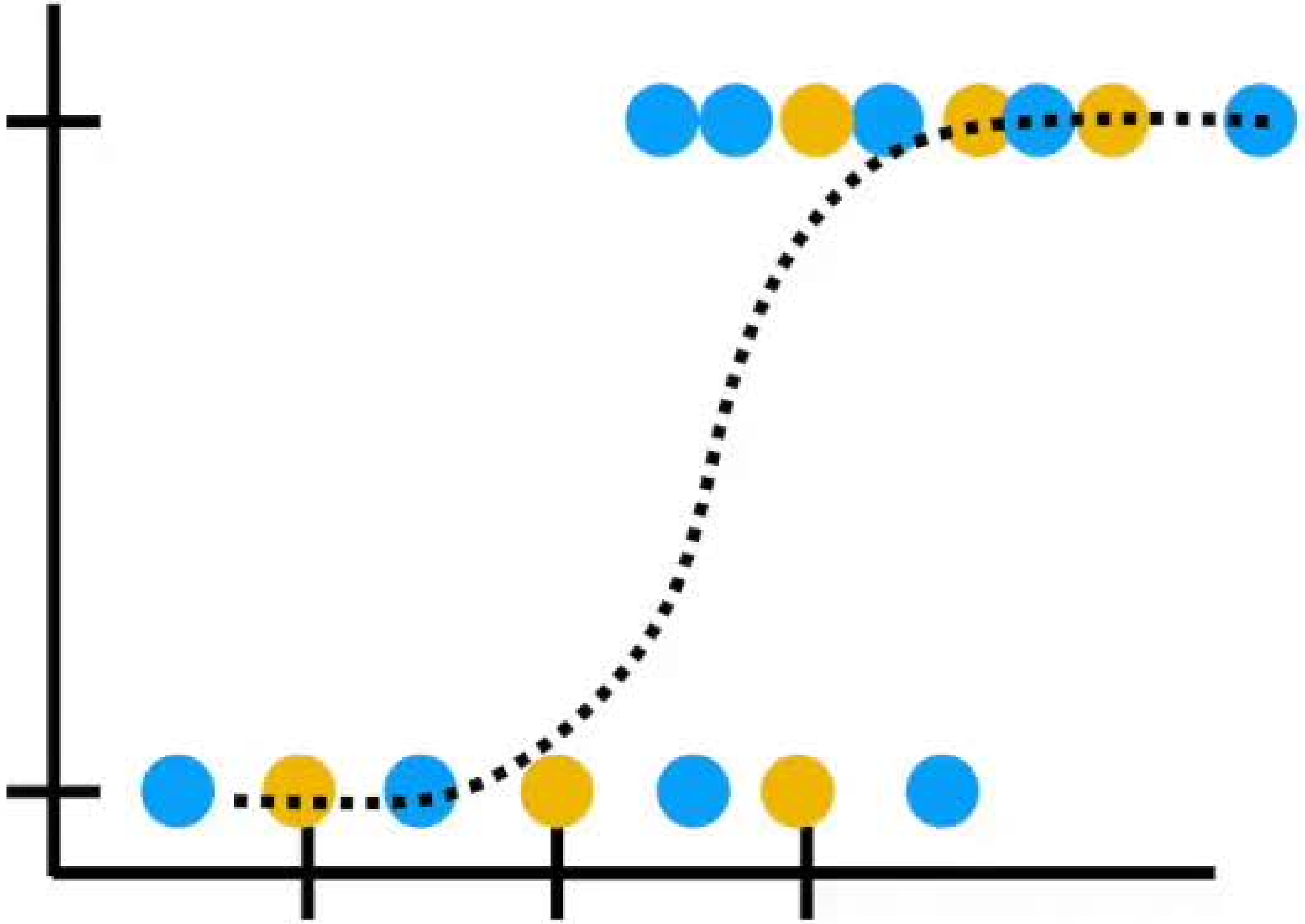

Then use the last block
to test the method.



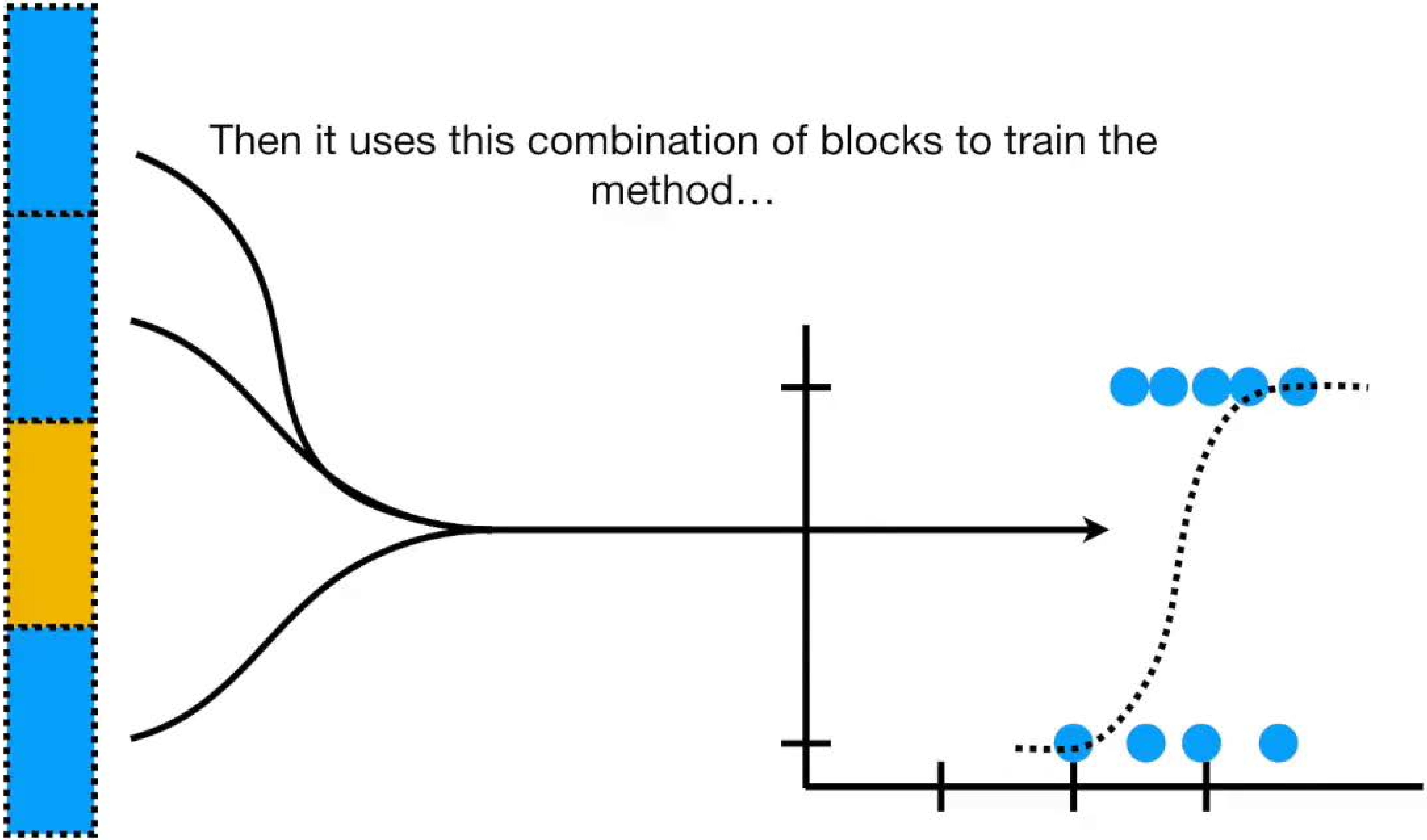
…and then it keeps track of how well the method did with the test data….

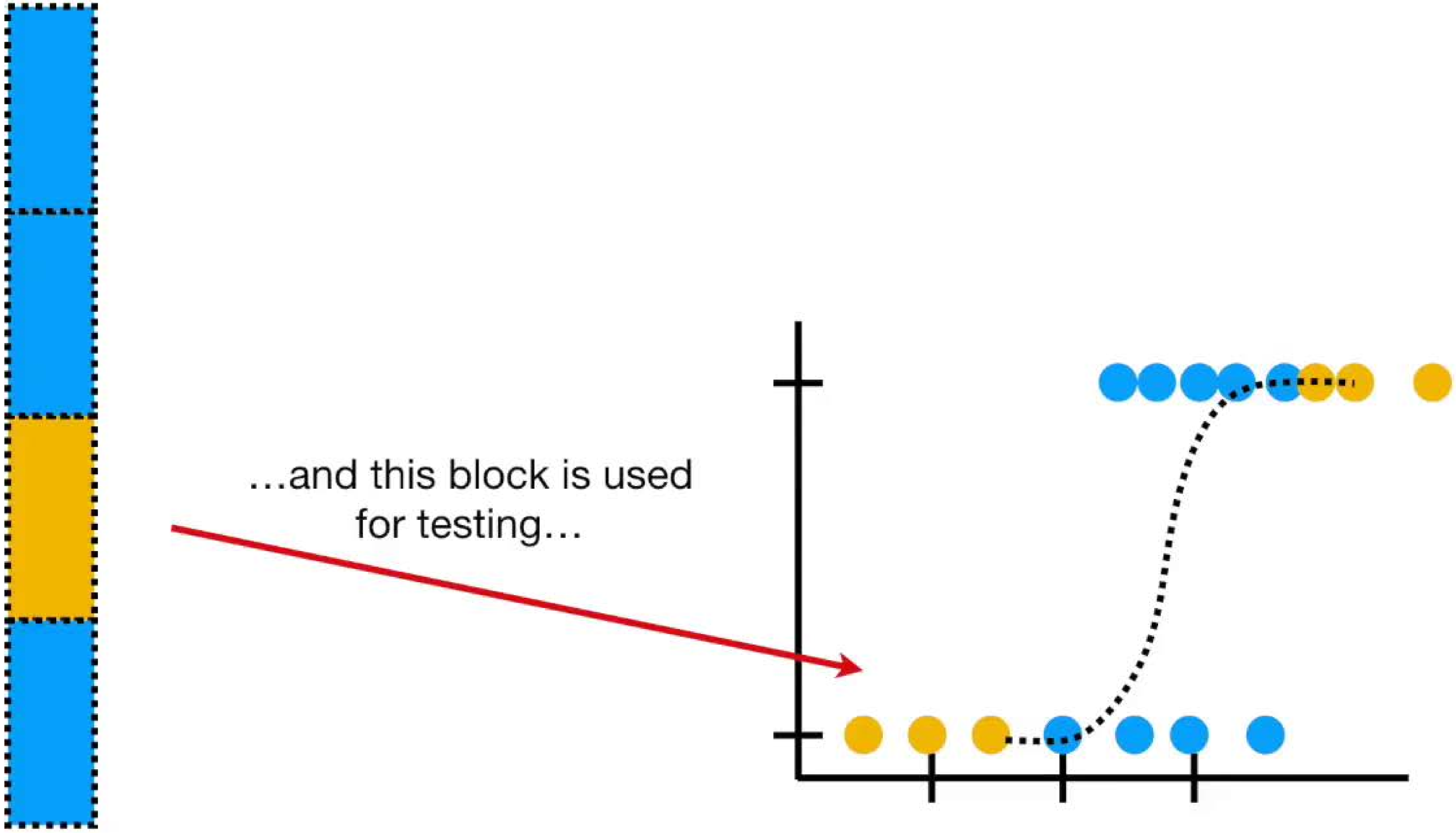
Test data categorization…

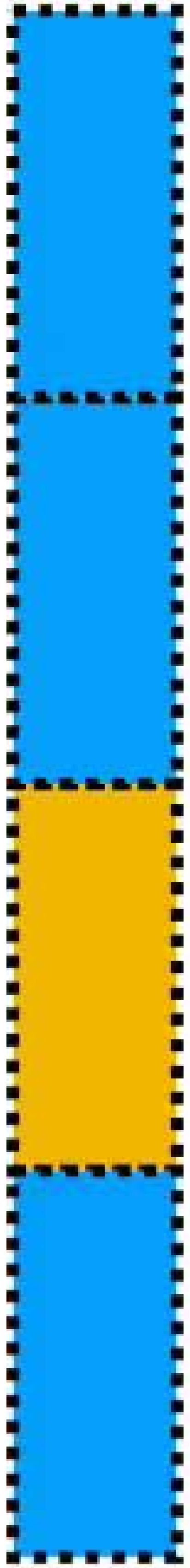
Correct	Incorrect
5	1



Then it uses this combination of blocks to train the method…



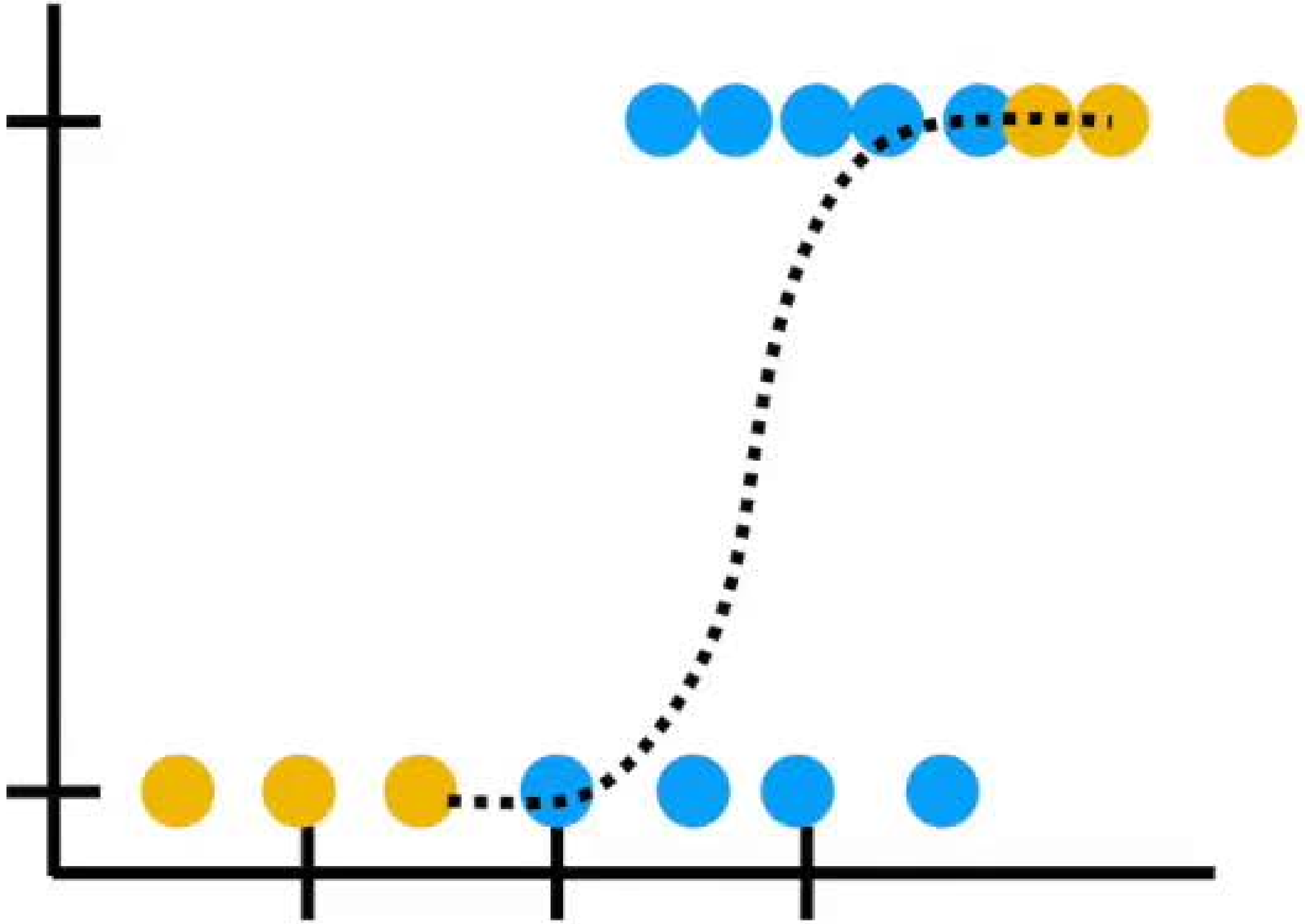

…and this block is used
for testing…



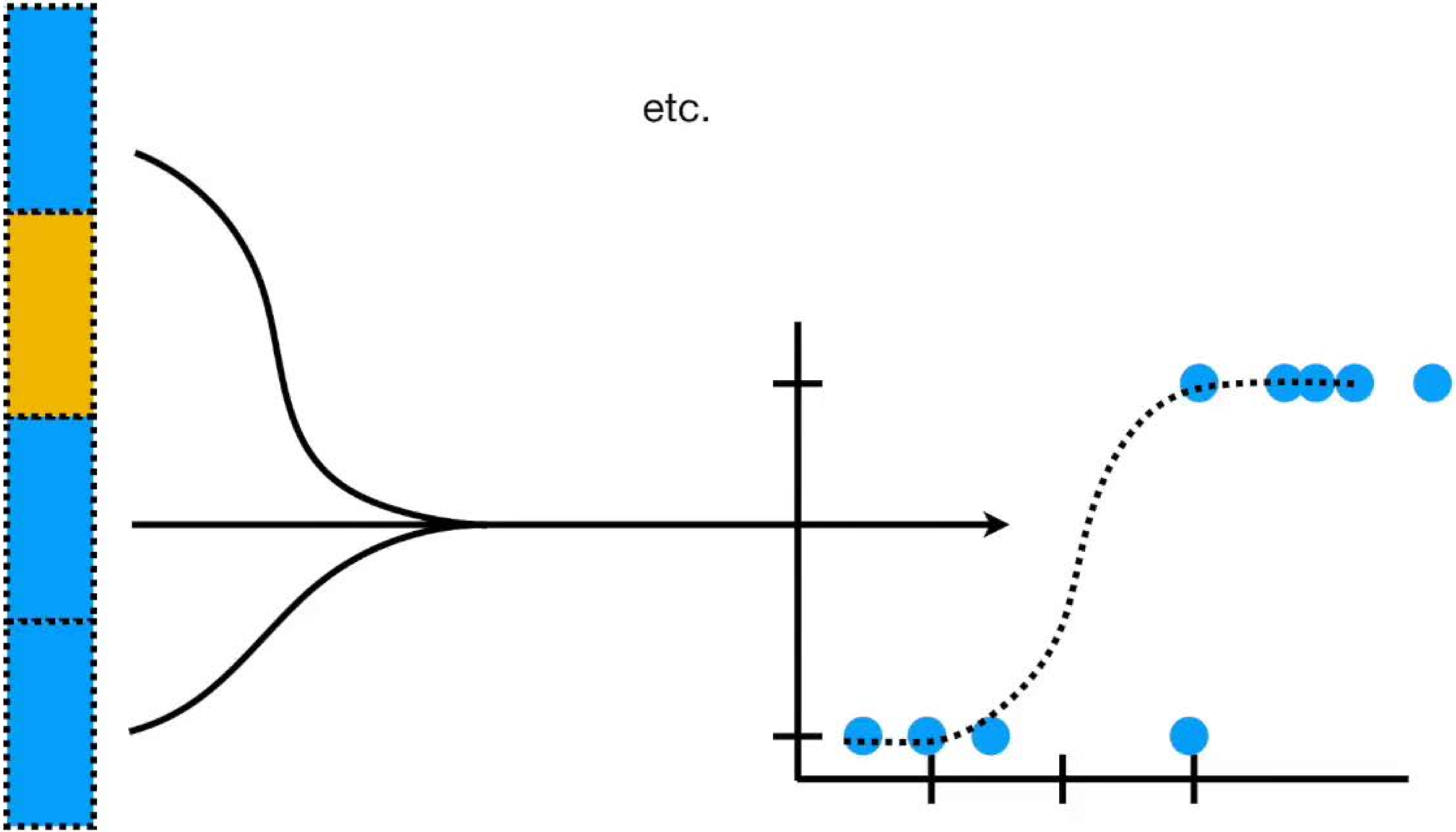
…and then it keeps track of how well the method did with the test data….

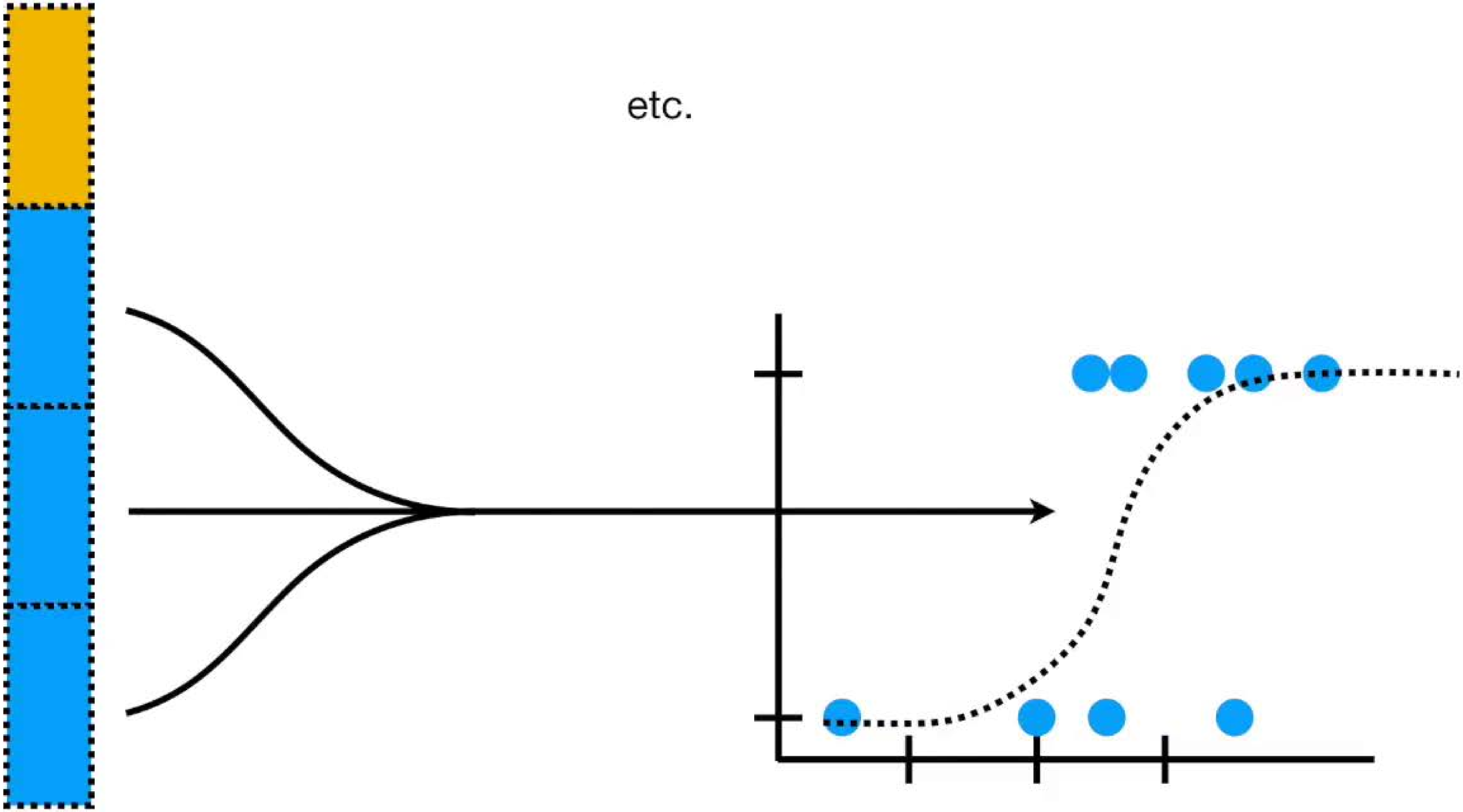
Test data categorization…

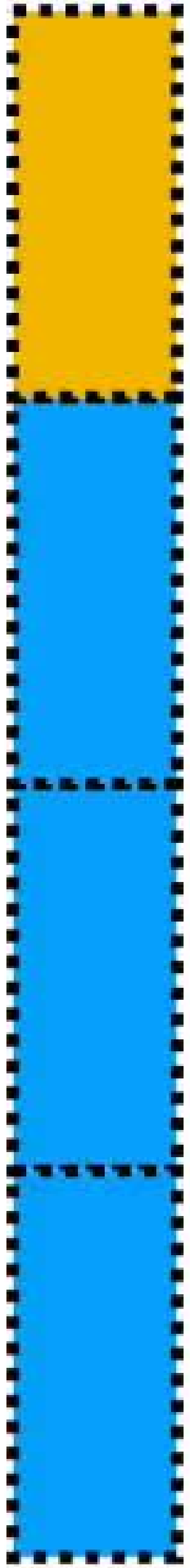
Correct	Incorrect
4	2



etc.



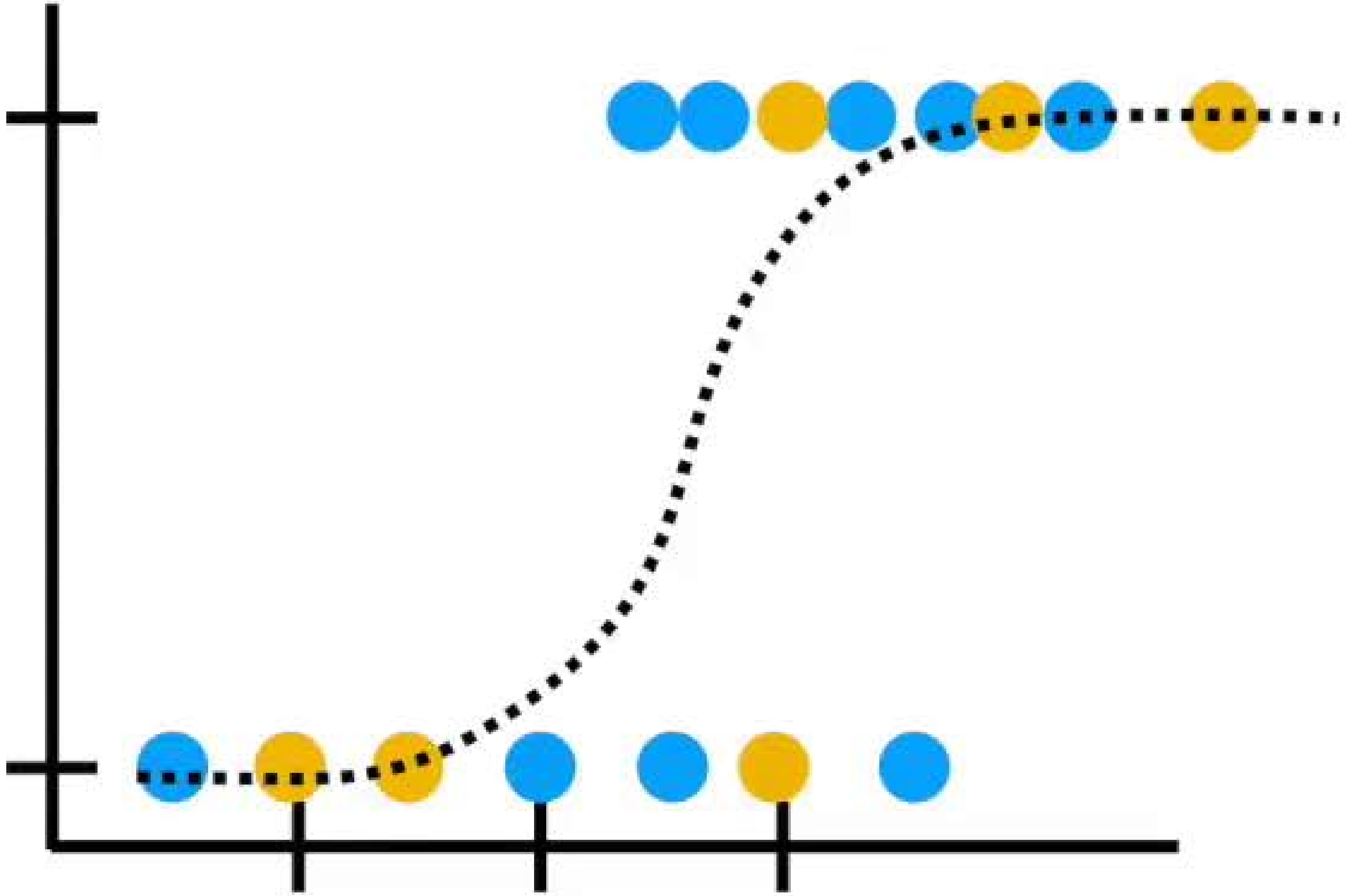

etc.



etc.

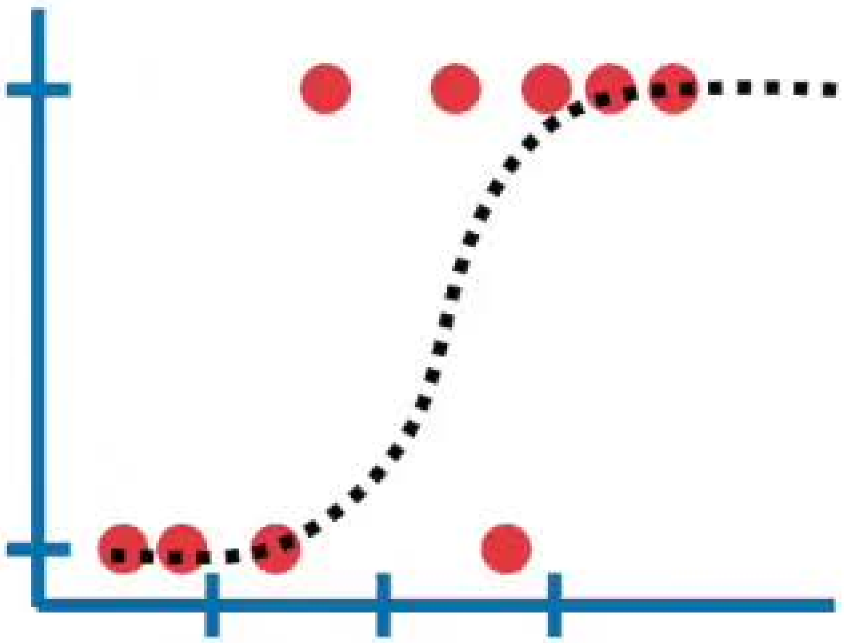
Test data categorization…

Correct	Incorrect
6	0



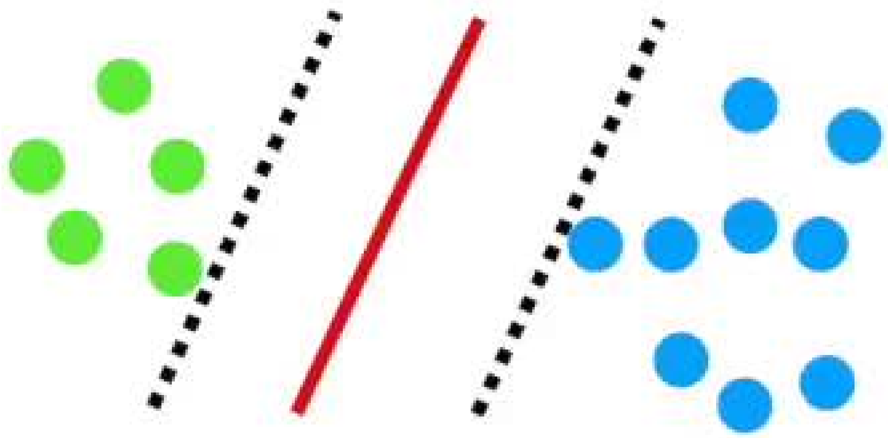
In the end, every block of data is used for testing and we can compare methods by seeing how well they performed.

Logistic Regression



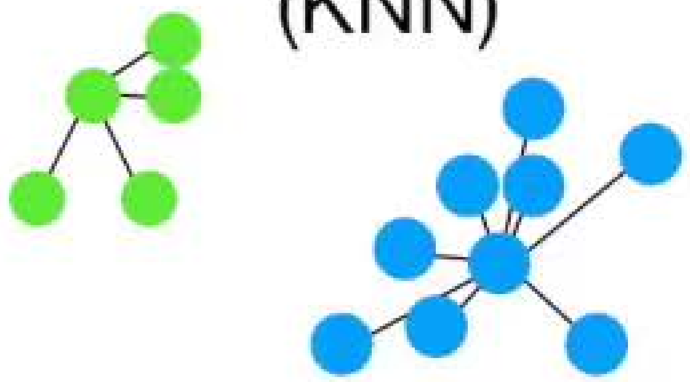
Correct	Incorrect
16	8

Support Vector machines (SVM)



Correct	Incorrect
18	6

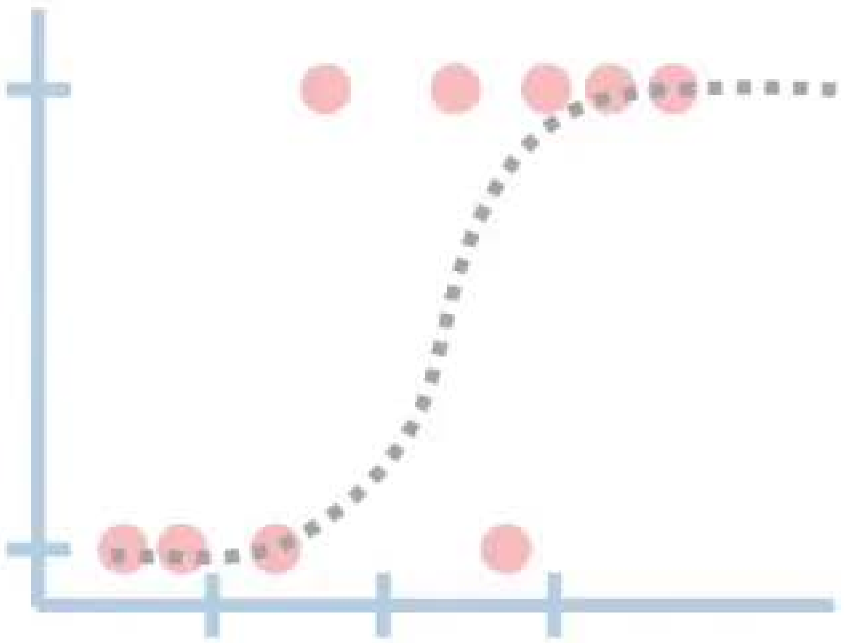
K-nearest neighbors (KNN)



Correct	Incorrect
10	12

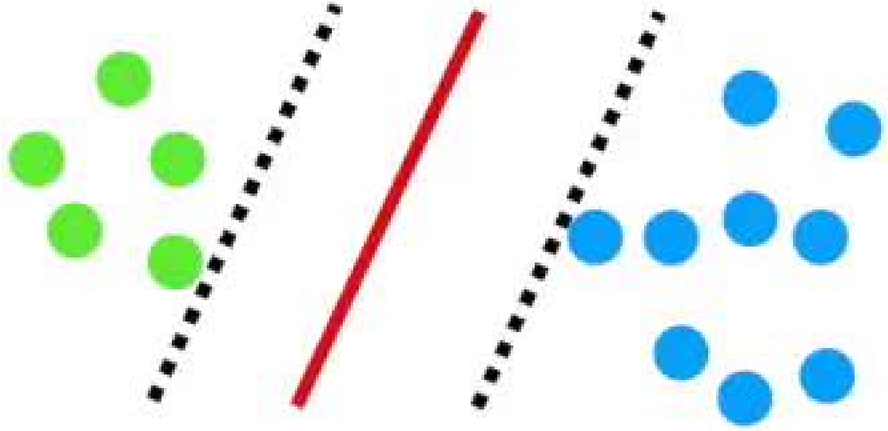
In this case, since the support vector machine did the best job classifying the test datasets, we'll use it!

Logistic Regression



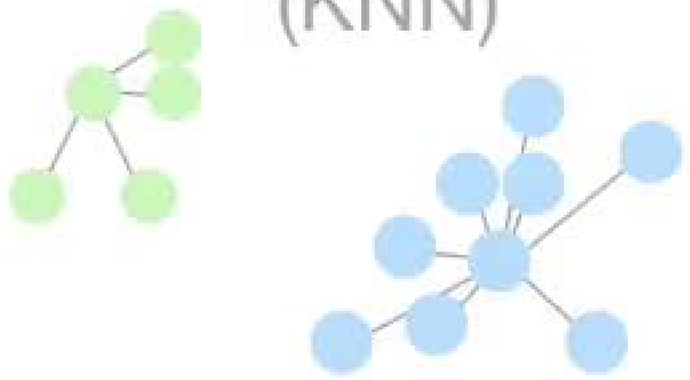
Correct	Incorrect
16	8

Support Vector machines (SVM)

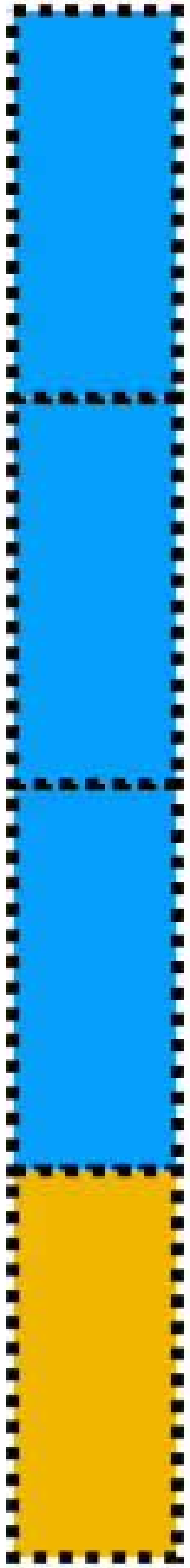


Correct	Incorrect
18	6

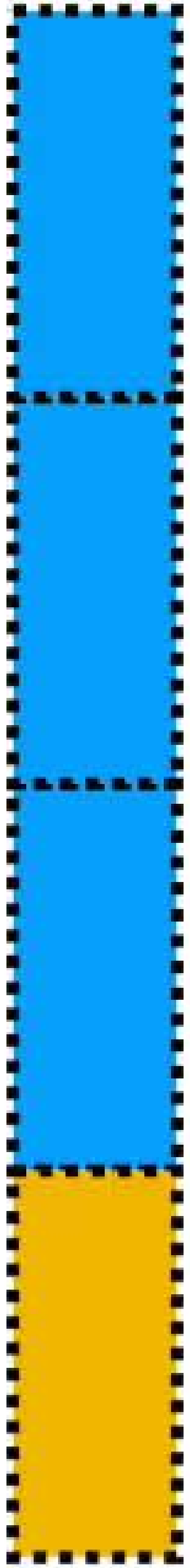
K-nearest neighbors (KNN)



Correct	Incorrect
10	12

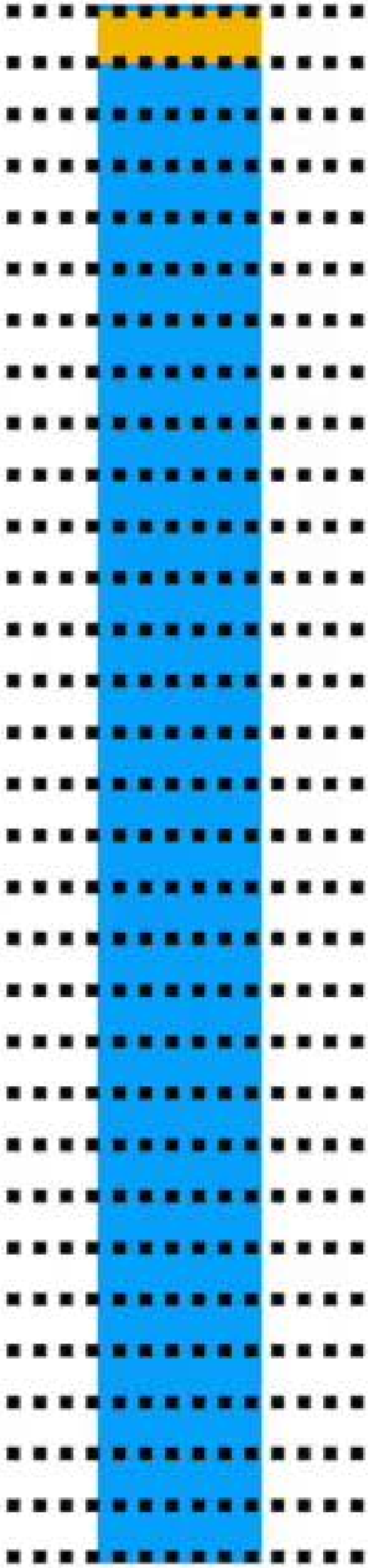


NOTE: In this example, we divided the data into 4 blocks. This is called **Four-Fold Cross Validation**.



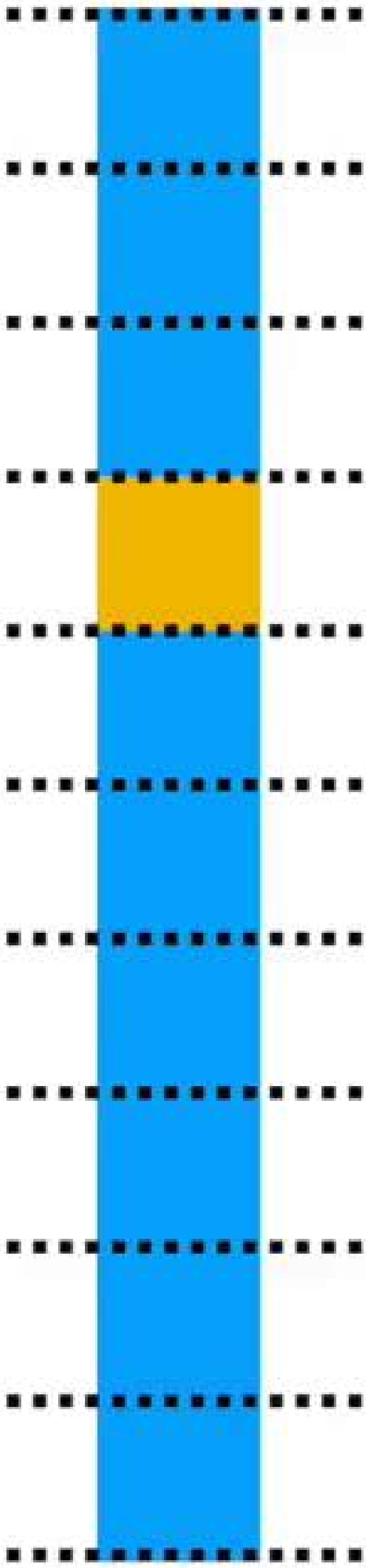
NOTE: In this example, we divided the data into 4 blocks. This is called **Four-Fold Cross Validation**.

However, the number of blocks is arbitrary.

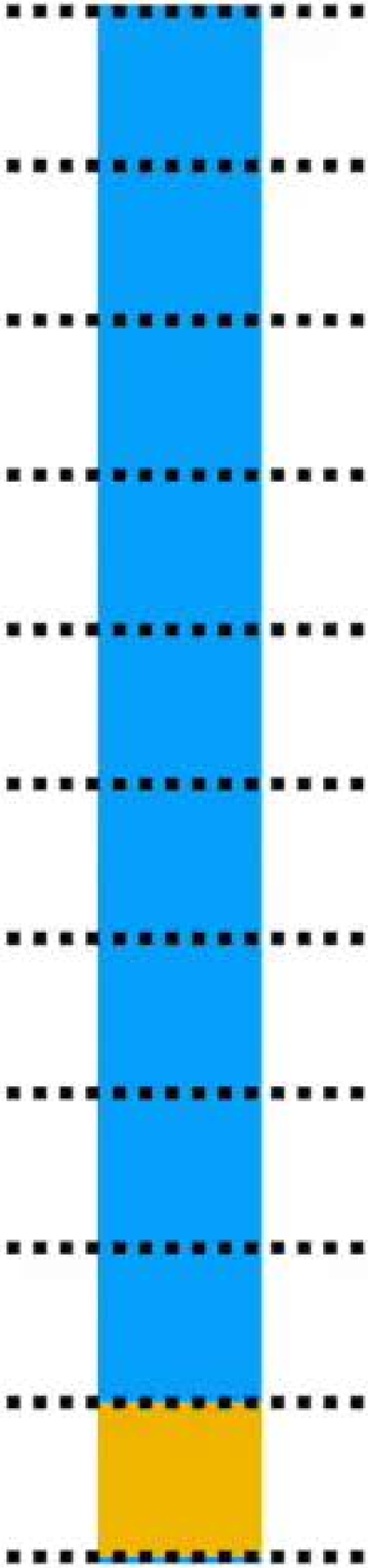


In an extreme case, we could call each individual patient (or sample) a block.

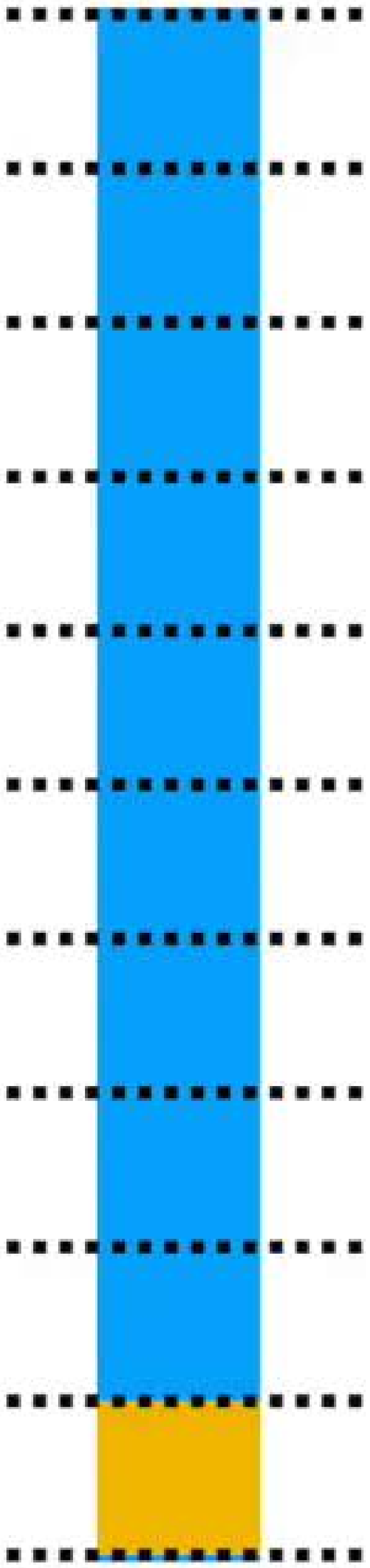
This is called “**Leave One Out Cross Validation**”



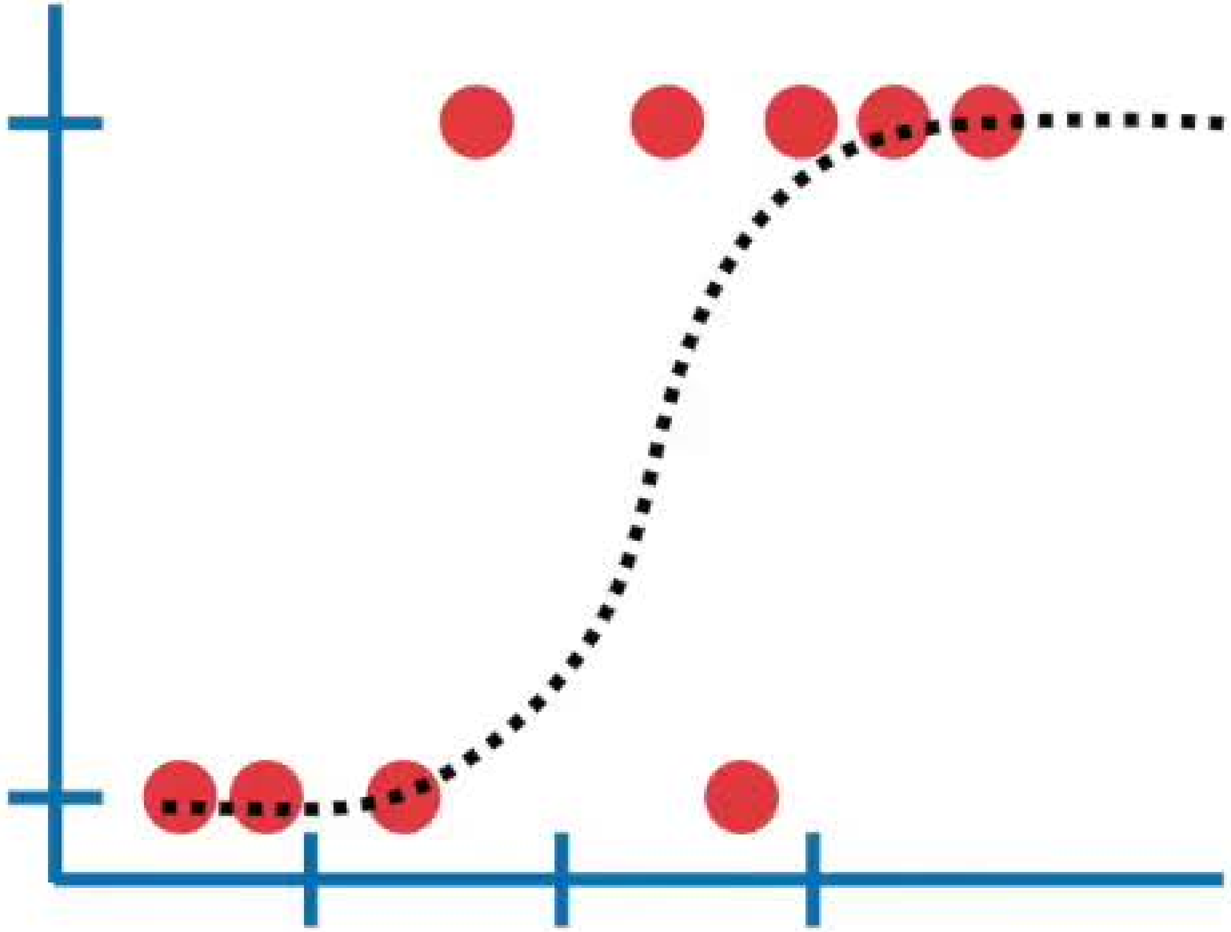
That said, in practice, it is very common to divide the data into 10 blocks. This is called **Ten-Fold Cross Validation**.

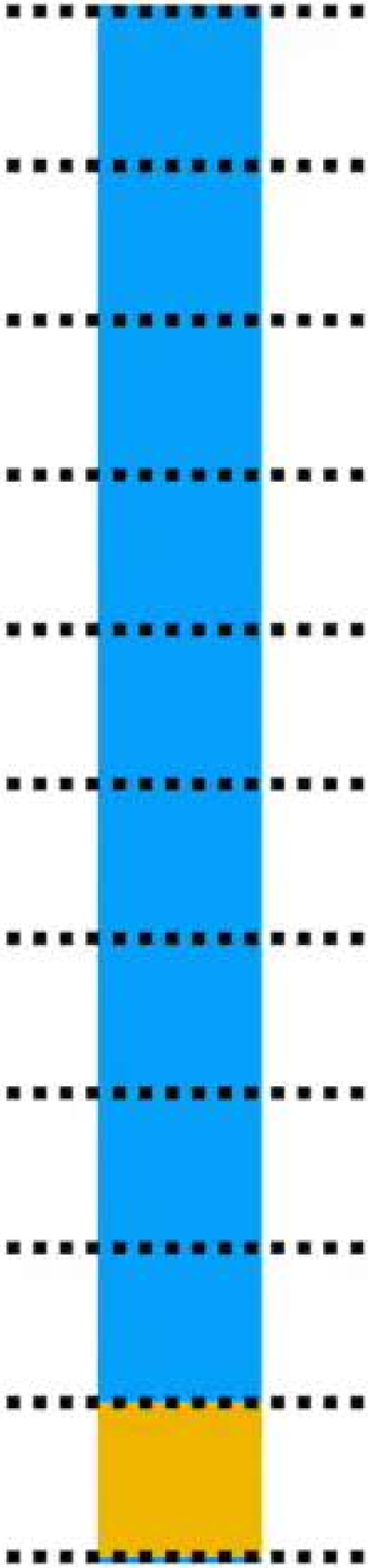


One last note before we're done…



Say like we wanted to use a method that involved a “**tuning parameter**” - a parameter that isn’t estimated, but just sort of guessed. (For example, Ridge Regression, has a tuning parameter)…





…then we could use 10-fold cross validation to help find the best value for that tuning parameter.