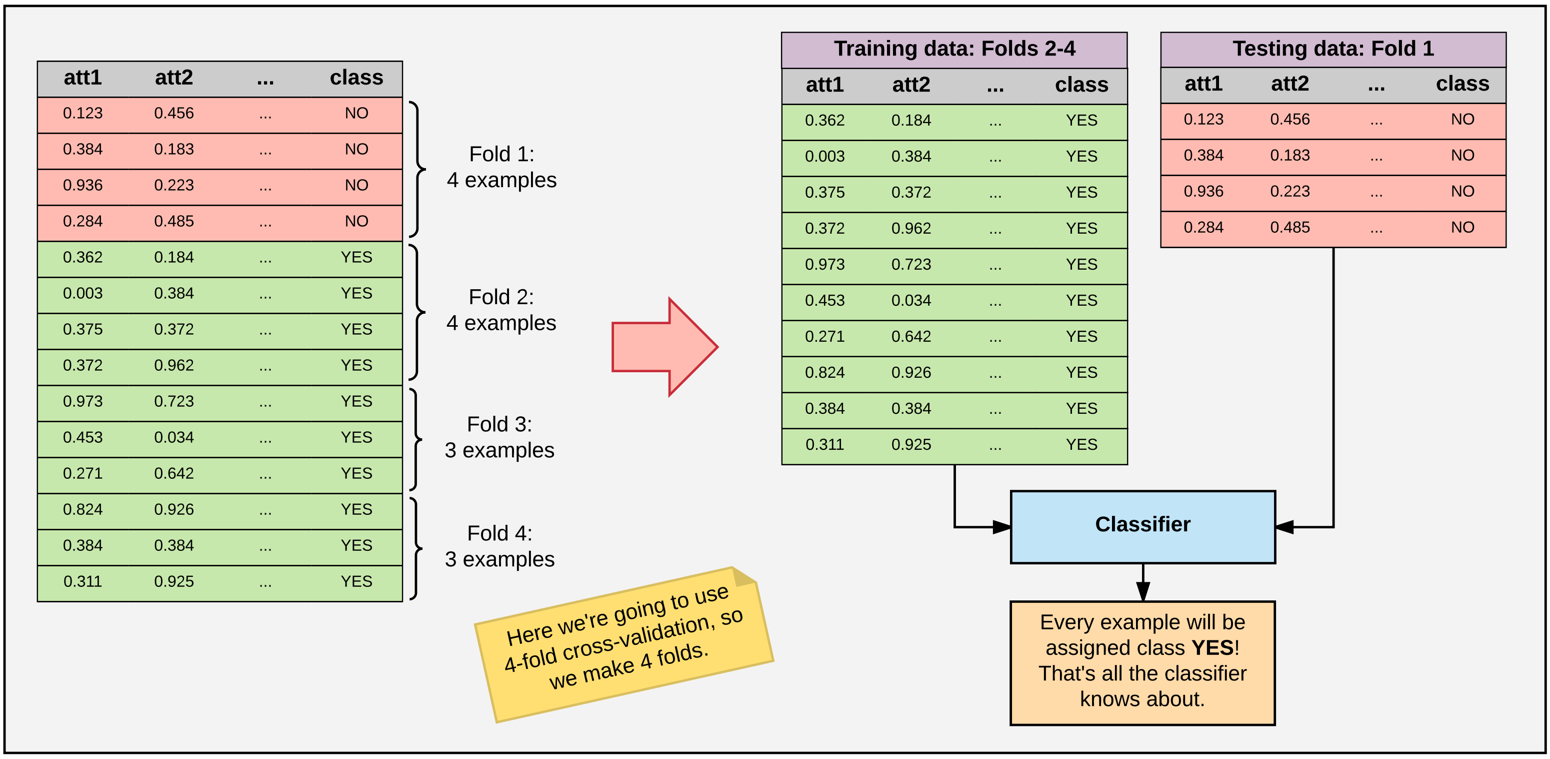
Why 'stratify' our folds?



att1	att2	...	class
0.123	0.456	...	NO
0.384	0.183	...	NO
0.936	0.223	...	NO
0.284	0.485	...	NO
0.362	0.184	...	YES
0.003	0.384	...	YES
0.375	0.372	...	YES
0.372	0.962	...	YES
0.973	0.723	...	YES
0.453	0.034	...	YES
0.271	0.642	...	YES
0.824	0.926	...	YES
0.384	0.384	...	YES
0.311	0.925	...	YES

Fold 1:
4 examples

Fold 2:
4 examples

Fold 3:
3 examples

Fold 4:
3 examples

Here we're going to use 4-fold cross-validation, so we make 4 folds.

Training data: Folds 2-4

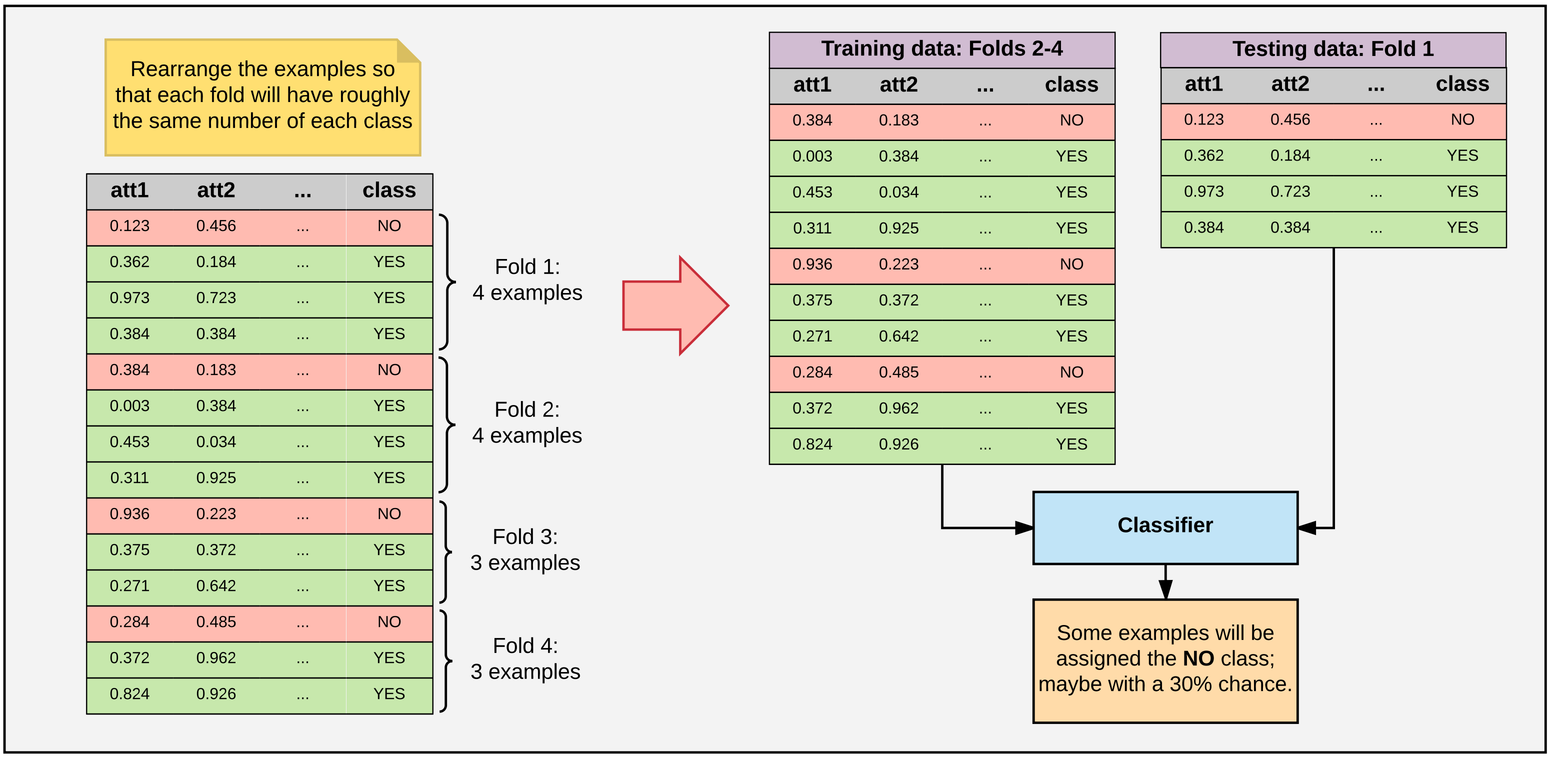
att1	att2	...	class
0.362	0.184	...	YES
0.003	0.384	...	YES
0.375	0.372	...	YES
0.372	0.962	...	YES
0.973	0.723	...	YES
0.453	0.034	...	YES
0.271	0.642	...	YES
0.824	0.926	...	YES
0.384	0.384	...	YES
0.311	0.925	...	YES

Testing data: Fold 1

att1	att2	...	class
0.123	0.456	...	NO
0.384	0.183	...	NO
0.936	0.223	...	NO
0.284	0.485	...	NO

Every example will be assigned class **YES**! That's all the classifier knows about.

Ideally each fold would have 3.5 (14/4) examples in it, each with 2.5 (10/4) **YES** examples and 1 (4/4) **NO** example.



Rearrange the examples so that each fold will have roughly the same number of each class

att1	att2	...	class
0.123	0.456	...	NO
0.362	0.184	...	YES
0.973	0.723	...	YES
0.384	0.384	...	YES
0.384	0.183	...	NO
0.003	0.384	...	YES
0.453	0.034	...	YES
0.311	0.925	...	YES
0.936	0.223	...	NO
0.375	0.372	...	YES
0.271	0.642	...	YES
0.284	0.485	...	NO
0.372	0.962	...	YES
0.824	0.926	...	YES

Fold 1:
4 examples

Fold 2:
4 examples

Fold 3:
3 examples

Fold 4:
3 examples

Training data: Folds 2-4

att1	att2	...	class
0.384	0.183	...	NO
0.003	0.384	...	YES
0.453	0.034	...	YES
0.311	0.925	...	YES
0.936	0.223	...	NO
0.375	0.372	...	YES
0.271	0.642	...	YES
0.284	0.485	...	NO
0.372	0.962	...	YES
0.824	0.926	...	YES

Testing data: Fold 1

att1	att2	...	class
0.123	0.456	...	NO
0.362	0.184	...	YES
0.973	0.723	...	YES
0.384	0.384	...	YES

Some examples will be assigned the **NO** class; maybe with a 30% chance.