GENERAL ASSEMBLY

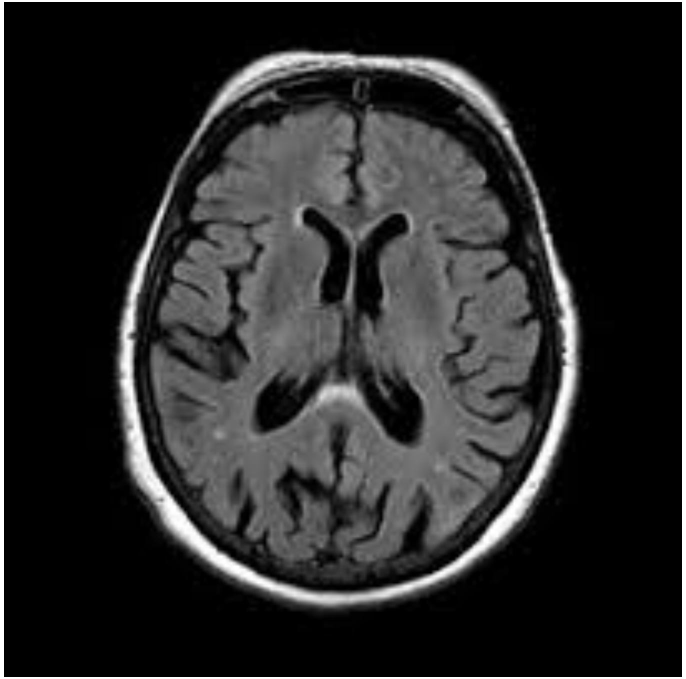
INTRO TO DATA SCIENCE

ADVANCED TOPIC: IMBALANCED CLASSES

REMEMBER THIS…?

Cancer Screen => classify cancer scans for doctor to review

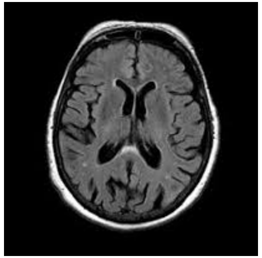
No Cancer



Cancer

ISSUE: Not all errors are equal…

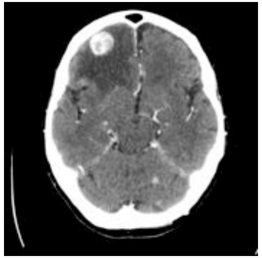
Error 1



Classifier Label: Cancerous

Permissable, because a physician will review it

Error 2

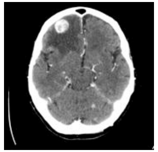


Classifier Label: Non-Cancerous

Not permissable, because this data will be discarded

To deal with issue 2 we need a more sophisticated definition of error rates in a binary classification problem

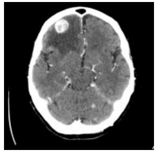
True Positive: An Example that is **positive** and is classified as **positive**



Label: **positive**

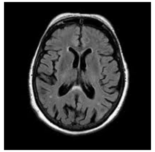
To deal with issue 2 we need a more sophisticated definition of error rates in a binary classification problem

True Positive: An Example that is **positive** and is classified as **positive**



Label: **positive**

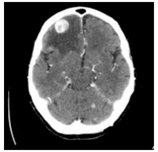
True Negative: An Example that is **negative** and is classified as **negative**



Label: **negative**

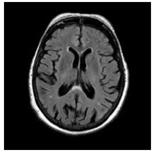
To deal with issue 2 we need a more sophisticated definition of error rates in a binary classification problem

True Positive: An Example that is **positive** and is classified as **positive**



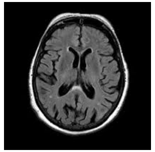
Label: **positive**

True Negative: An Example that is **negative** and is classified as **negative**



Label: **negative**

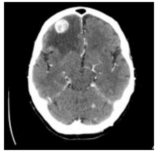
False Positive: An Example that is **negative** and is classified as **positive**



Label: **positive**

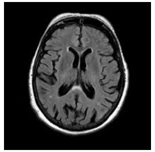
To deal with issue 2 we need a more sophisticated definition of error rates in a binary classification problem

True Positive: An Example that is **positive** and is classified as **positive**



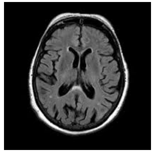
Label: **positive**

True Negative: An Example that is **negative** and is classified as **negative**



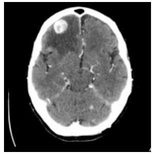
Label: **negative**

False Positive: An Example that is **negative** and is classified as **positive**



Label: **positive**

False Negative: An Example that is **positive** and is classified as **negative**



Label: **negative**

Confusion Matrix

	Condition Positive	Condition Negative
Test Positive	TRUE POSITIVE	FALSE POSITIVE (Type I error)
Test Negative	FALSE NEGATIVE (Type II error)	TRUE NEGATIVE

Confusion Matrix

n = 165	*Condition Positive*	*Condition Negative*
Test Positive	*100*	*10*
Test Negative	*5*	*50*

How many classes are there?
How many patients?
How many times is disease predicted?
How many patients actually have the disease?

Confusion Matrix

		Condition (as determined by "Gold standard")			
	Total population	Condition positive	Condition negative	Prevalence $= \frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	
Test outcome	Test outcome positive	**True positive**	**False positive** (Type I error)	Positive predictive value (PPV), Precision $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Test outcome positive}}$	False discovery rate (FDR) $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Test outcome positive}}$
	Test outcome negative	**False negative** (Type II error)	**True negative**	False omission rate (FOR) $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Test outcome negative}}$	Negative predictive value (NPV) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Test outcome negative}}$
	Accuracy (ACC) $= \frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	True positive rate (TPR), Sensitivity, Recall $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR), Fall-out $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$
		False negative rate (FNR), Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	True negative rate (TNR), Specificity (SPC) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$	Negative likelihood ratio (LR−) $= \frac{FNR}{TNR}$	

n = 165	Condition Positive	Condition Negative
Test Positive	100	10
Test Negative	5	50

Precision:
When test is positive, how often is prediction correct?
TP / test yes = 100/110 = 0.91

Sensitivity/Recall/TPR:
When actual value is positive, how often is prediction correct?
TP / actual yes = 100/105 = 0.95

Accuracy:
Overall, how often is it **correct**?

(TP + TN) / total = 150/165 = 0.91

Specificity/TNR:
When actual value is negative, how often is prediction correct?
TN / actual no = 50/60 = 0.83

n = 165	Condition Positive	Condition Negative
Test Positive	100	10
Test Negative	5	50

Precision:
When test is positive, how often is prediction correct?
TP / test yes = 100/110 = 0.91

Sensitivity/Recall/TPR:
When actual value is positive, how often is prediction correct?
TP / actual yes = 100/105 = 0.95

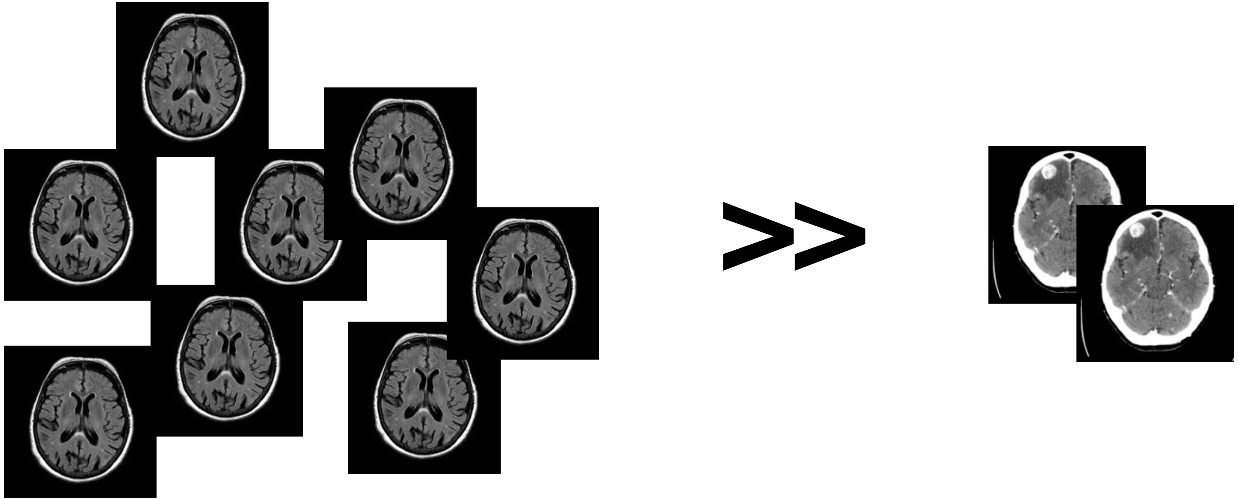
F score

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

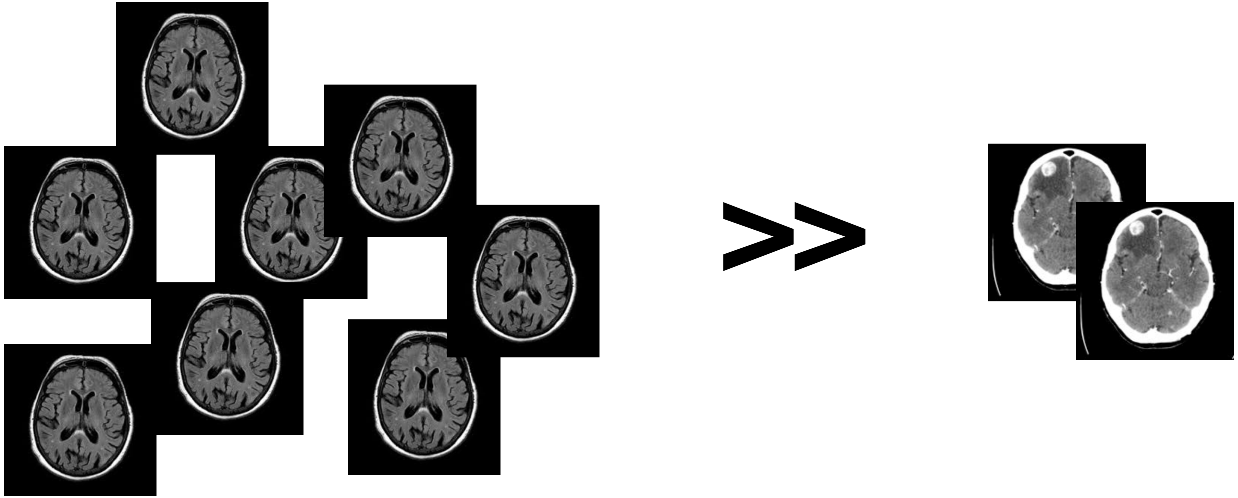
IMBALANCED CLASSES

ISSUE: Many more healthy brain scans

- Imbalance confuses classifiers => only perform well on dominant class
- Situation is very common in other fields (e.g. fraud detection)



Imbalanced classes can be re-balanced in several ways



Imbalanced classes can be re-balanced in several ways

1. **Undersampling** the dominant class - remove some the majority class so it has less weight

Imbalanced classes can be re-balanced in several ways

1. **Undersampling** the dominant class - remove some the majority class so it has less weight

2. **Oversampling** the minority class - add more of the minority class so it has more weight.

Imbalanced classes can be re-balanced in several ways

1. **Undersampling** the dominant class - remove some the majority class so it has less weight

2. **Oversampling** the minority class - add more of the minority class so it has more weight.

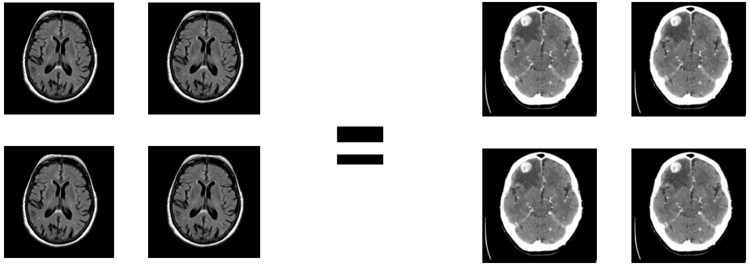
3. **Hybrid** - doing both

Imbalanced classes can be re-balanced in several ways

1. **Undersampling** the dominant class - remove some the majority class so it has less weight

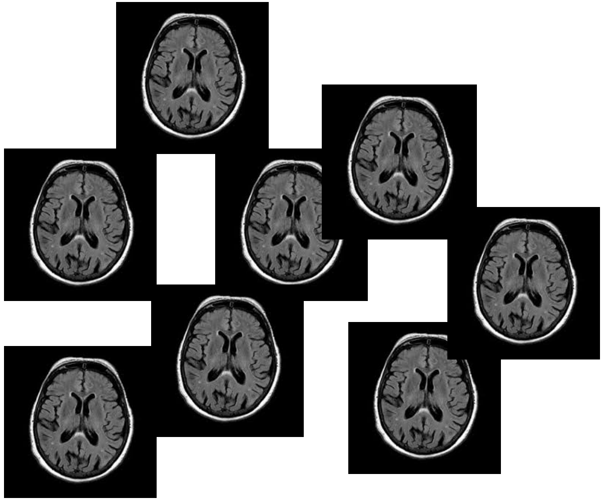
2. **Oversampling** the minority class - add more of the minority class so it has more weight.

3. **Hybrid** - doing both



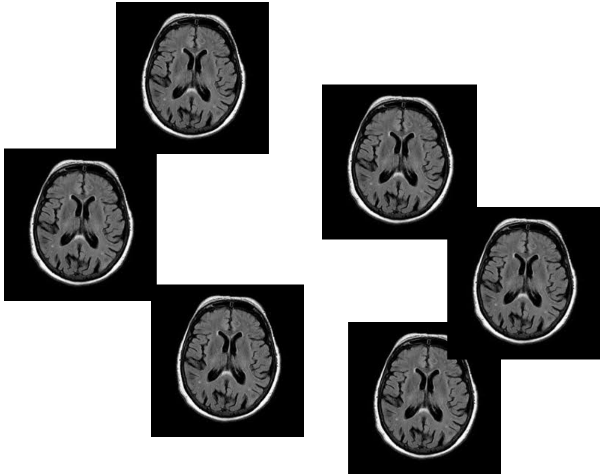
Undersampling

Randomly remove elements from the majority class.



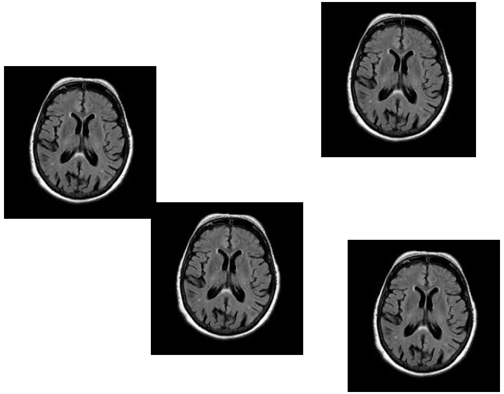
Undersampling

Randomly remove elements from the majority class.



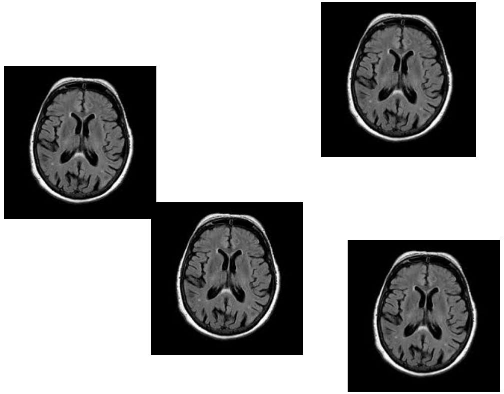
Undersampling

Randomly remove elements from the majority class.



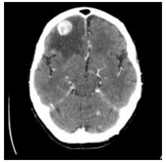
Undersampling

Randomly remove elements from the majority class.



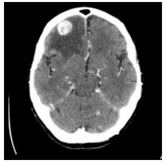
Drawback: Removing data points could lose important information

Oversampling

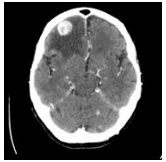


Duplicate elements of your minority class

Oversampling

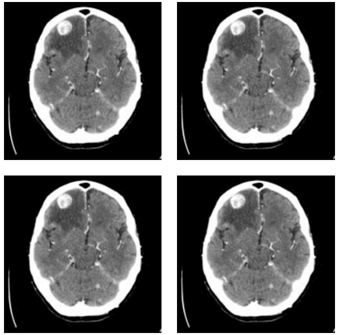


Duplicate elements of your minority class



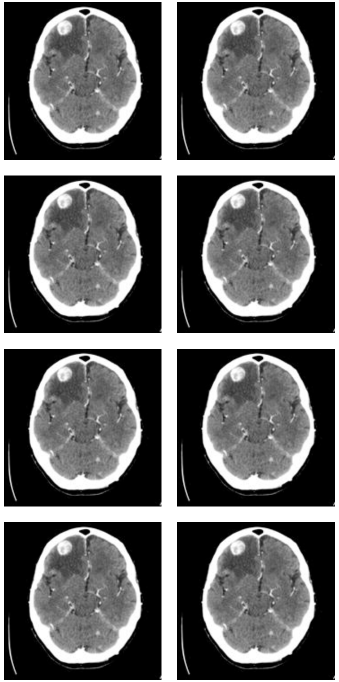
Oversampling

Duplicate elements of your minority class



Oversampling

Duplicate elements of your minority class



Oversampling

Duplicate elements of your minority class

Drawback: Just replicating randomly minority classes could cause overfit