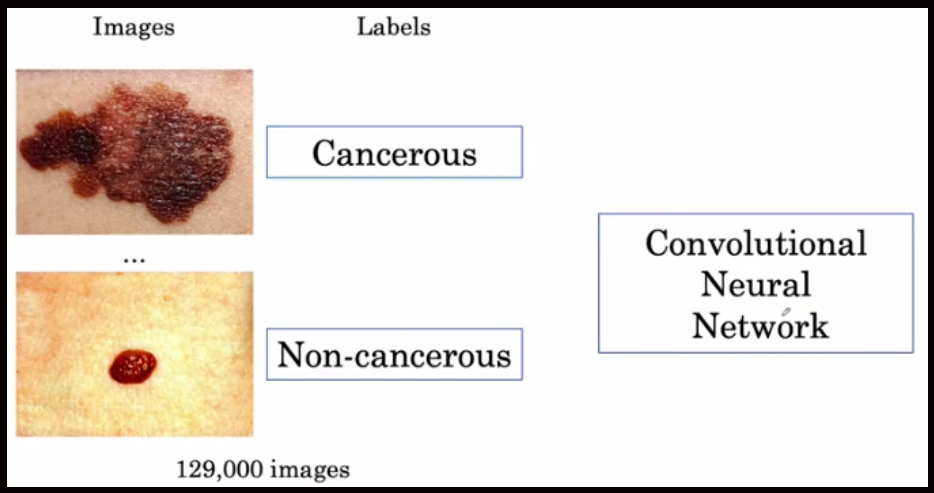
# AI for Medical Diagnosis

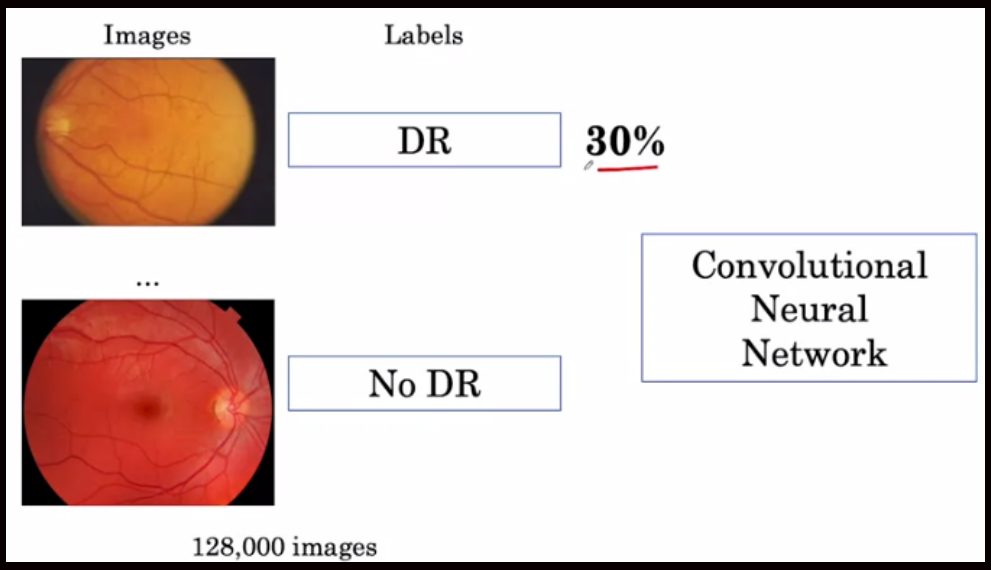
Examples :

1 Dermatology:



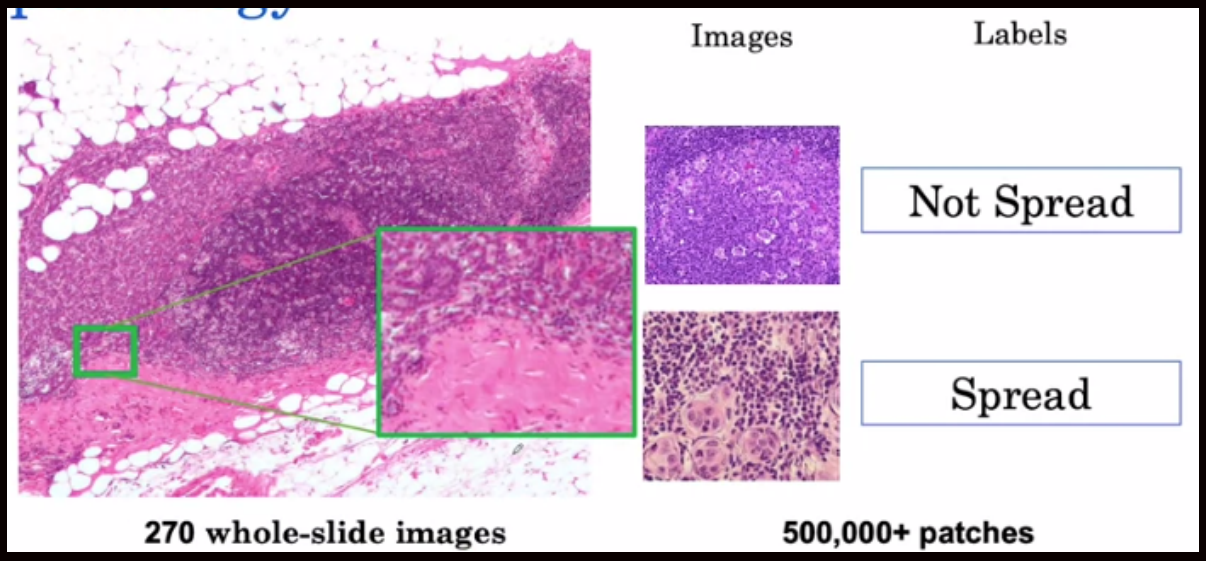

2- Ophthalmology :

Diabetic Retinopathy (DR)




* data imbalance Problem.

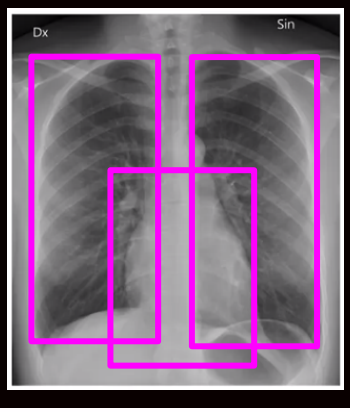
3. Histopathology

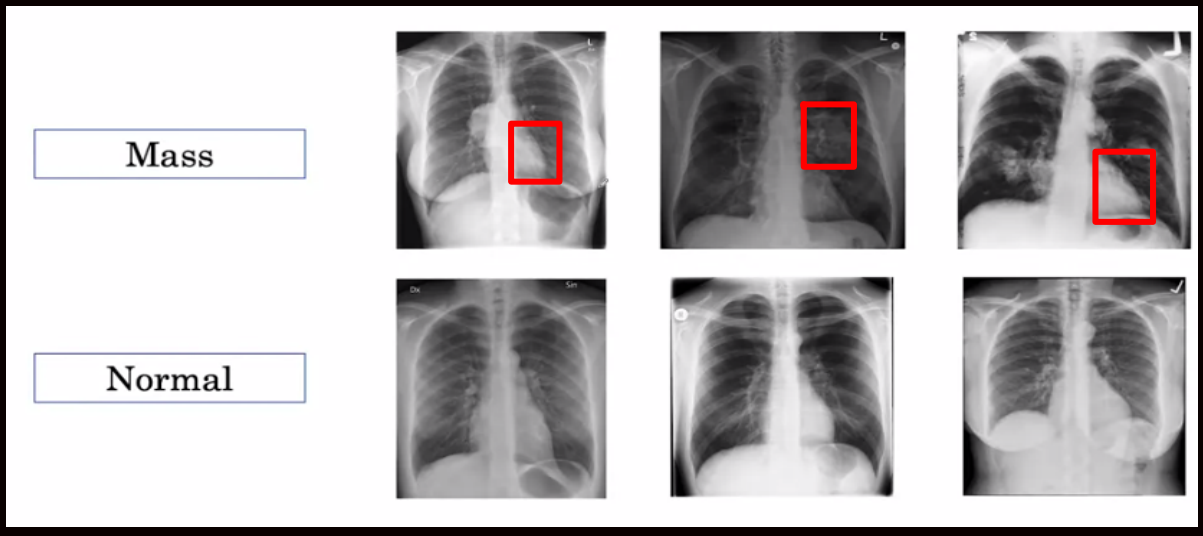



A larger image is broken down into smaller patches to get a training data

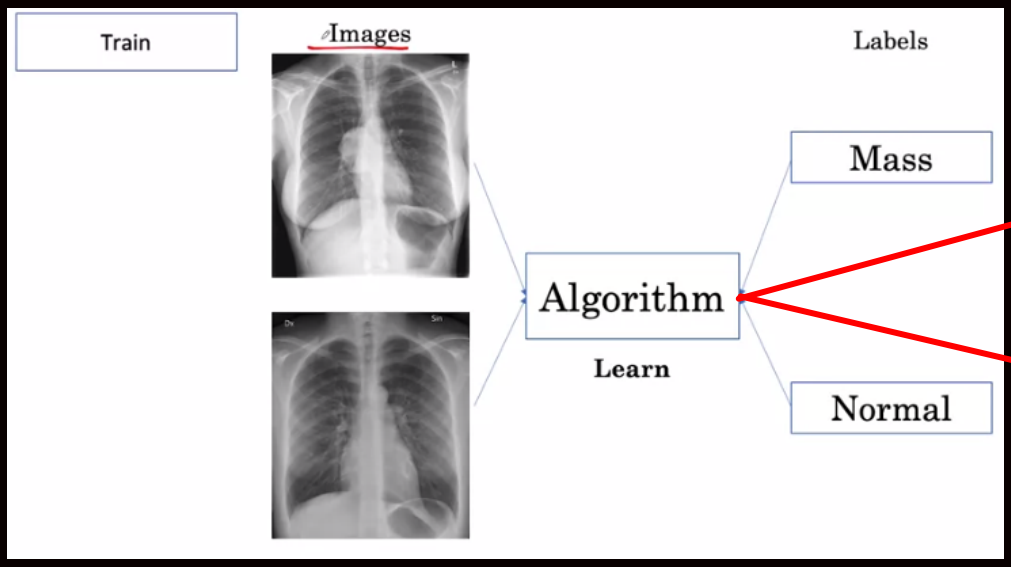
## Building and Training a Model for Medical Diagnostic.

* Critical diagnosis step of Pneumonia, lung cancer etc



# Training, Prediction and Loss




Prediction Probability

0.48 [1 = Desired]

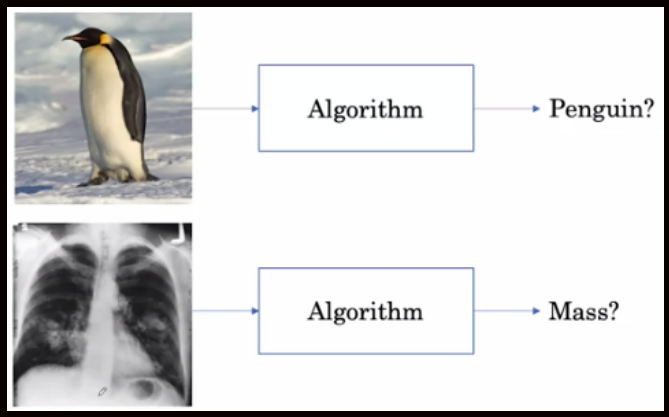
Error Loss = 0.32

Error Loss = 0.31

0.51 [0 = Desired]

Back Propagate and update the parameters to give better predictions

# Image Classification & its challenges



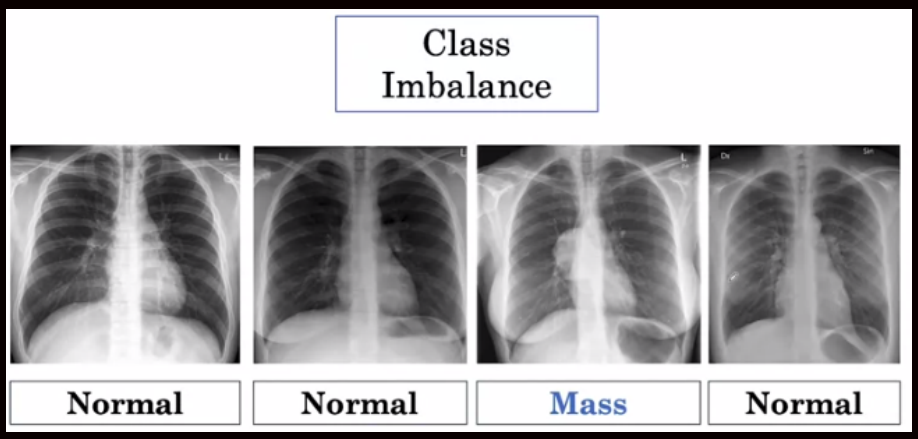

General Image

Medical Image

But medical imaging has a few key challenges,

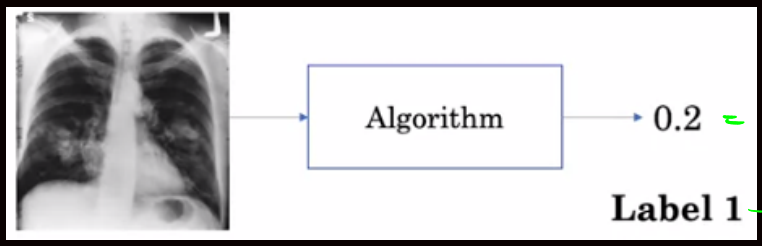
1. Class Imbalance
2. Multi-Task
3. Dataset Size

# 1. Class Imbalance Challenge



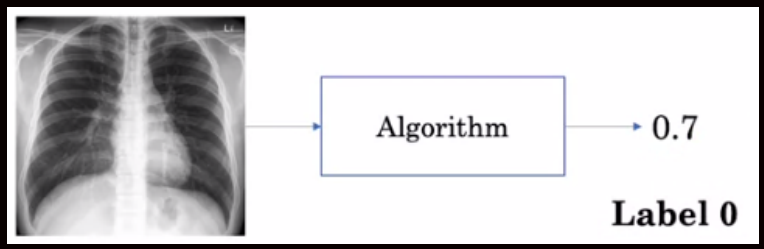

## Binary Cross Entropy Loss

$$L(x,y) = \begin{cases} -\log P(Y=1|x) & \text{if } y=1 \\ -\log P(Y=0|x) & \text{if } y=0 \end{cases}$$




$= P(Y=1|x)$

Now $L = -\log 0.2 = 0.70$




$= P(Y=1|x)$

Now, $L = -\log(1-0.7)$
$= 0.52$

When Training has not started.

$$L(X, y) = \begin{cases} -\log P(Y = 1|X) & \text{if } y = 1 \\ -\log P(Y = 0|X) & \text{if } y = 0 \end{cases}$$

| Examples | Prediction Probabilities | Loss |
|---|---|---|
| P1 Normal | 0.5 | 0.3 |
| P2 Normal | 0.5 | 0.3 |
| P3 Normal | 0.5 | 0.3 |
| P4 Mass | 0.5 | 0.3 |
| P5 Normal | 0.5 | 0.3 |
| P6 Normal | 0.5 | 0.3 |
| P7 Mass | 0.5 | 0.3 |
| P8 Normal | 0.5 | 0.3 |

Loss = $-\log(1-0.5) = 0.3$
$= -\log(0.5) = 0.3$.

* Total Loss from Mass Example = $0.3 \times 2 = 0.6$
* Total Loss from Normal Example = $0.3 \times 6 = 1.8$

Solution

Modify the loss function to calculate the mass & normal class differently.

$$L(X,y) = \begin{cases} w_p \times -\log P(Y=1|x) & \text{if } y=1 \\ w_n \times -\log P(Y=0|x) & \text{if } y=0 \end{cases}$$

$w_p \rightarrow$ weight assigned to the mass examples

$w_n \rightarrow$ weight assigned to the normal examples.

Here $w_p = 6/8$ , $w_n = 2/8$

| Labels | Probabilities | Loss |
|---|---|---|
| P1 Normal | 0.5 | 2/8 × 0.3 = 0.075 |
| P2 Normal | 0.5 | 2/8 × 0.3 = 0.075 |
| P3 Normal | 0.5 | 2/8 × 0.3 = 0.075 |
| P4 Mass | 0.5 | 6/8 × 0.3 = 0.225 |
| P5 Normal | 0.5 | 2/8 × 0.3 = 0.075 |
| P6 Normal | 0.5 | 2/8 × 0.3 = 0.075 |
| P7 Mass | 0.5 | 6/8 × 0.3 = 0.225 |
| P8 Normal | 0.5 | 2/8 × 0.3 = 0.075 |

Now

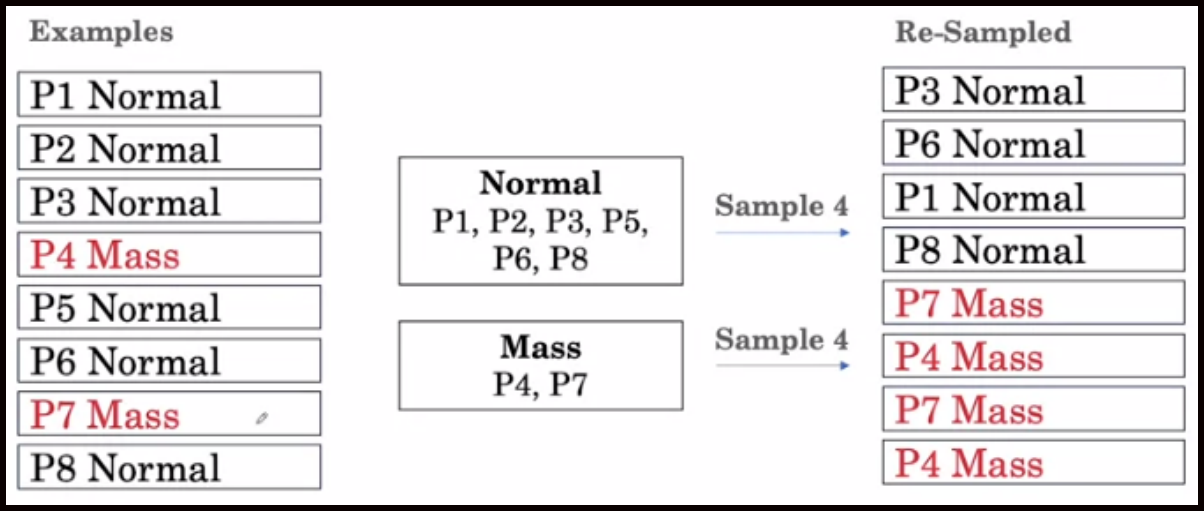
Total loss from Mass Examples = $0.225 \times 2 = 0.45$

Total loss from Normal Examples = $0.075 \times 6 = 0.45$

In general

$$w_p = \frac{\text{num negative}}{\text{num total}} \quad , \quad w_n = \frac{\text{num positive}}{\text{num total}}$$

Resampling (Solution to the data imbalance problem)



$$L(X,y) = \begin{cases} -\log P(Y=1|X) & \text{if } y=1 \\ -\log P(Y=0|X) & \text{if } y=0 \end{cases}$$

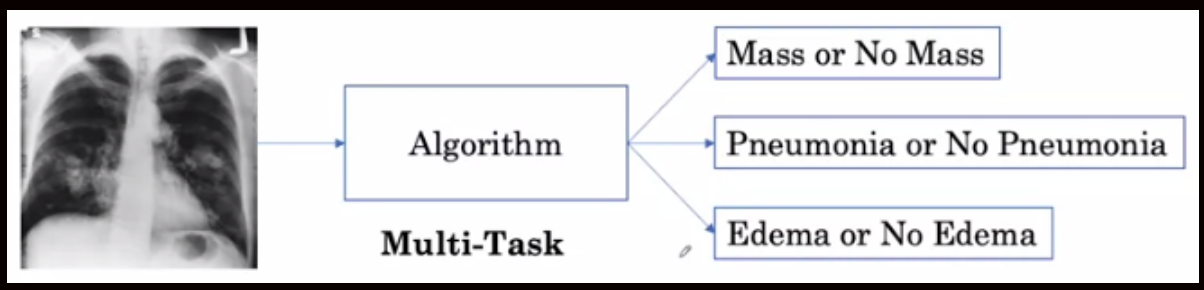
| Re-Sampled | Prediction Probabilities | Loss |
|---|---|---|
| P3 Normal | 0.5 | 0.3 |
| P6 Normal | 0.5 | 0.3 |
| P1 Normal | 0.5 | 0.3 |
| P8 Normal | 0.5 | 0.3 |
| P7 Mass | 0.5 | 0.3 |
| P4 Mass | 0.5 | 0.3 |
| P7 Mass | 0.5 | 0.3 |
| P4 Mass | 0.5 | 0.3 |

Total Loss From Mass Examples = 0.3 × 4 = 1.2
Total Loss From Normal Examples = 0.3 × 4 = 1.2

**Re-sampling methods (Undersampling, Oversampling)**

After the resampling even without the weighted loss, there is an equal contribution to the loss from the mass example & the normal example.

## ② Multitask Problem




What will be the algorithm like in such task?

| Examples (mass, pneumonia, edema) | Prediction Probabilities |
|---|---|
| P1 0, 1, 0 | 0.3, 0.1, 0.8 |
| P2 0, 0, 1 | 0.1, 0.1, 0.8 |
| P3 0, 1, 1 | 0.2, 0.2, 0.7 |
| P4 1, 0, 1 | 0.6, 0.3, 0.8 |
| P5 1, 1, 1 | 0.7, 0.7, 0.9 |
| P6 1, 0, 0 | 0.8, 0.1, 0.2 |
| P7 0, 1, 1 | 0.3, 0.9, 0.8 |
| P8 0, 0, 0 | 0.1, 0.1, 0.2 |

Here 0 → Absence of a disease.
1 → Presence of a disease

$$L(X, y) = L(X, y_{mass}) + L(X, y_{pneumonia}) + L(X, y_{edema})$$

Multi-Label / Multitask Loss

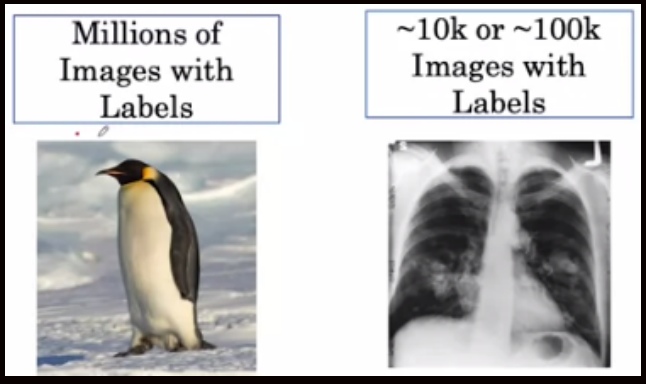
| Loss |
|---|
| 0.15 + 1.00 + 0.70 |
| 0.05 + 0.05 + 0.10 |
| 0.10 + 0.70 + 0.15 |
| 0.22 + 0.15 + 0.10 |
| 0.15 + 0.15 + 0.05 |
| 0.10 + 0.05 + 0.10 |
| 0.15 + 0.05 + 0.10 |

This is the loss calculation for the above example

Here.

$$L(X, y_{mass}) = \begin{cases} -w_{p,mass} \log P(Y=1|X) & \text{if } y=1 \\ -w_{n,mass} \log P(Y=0|X) & \text{if } y=0 \end{cases}$$

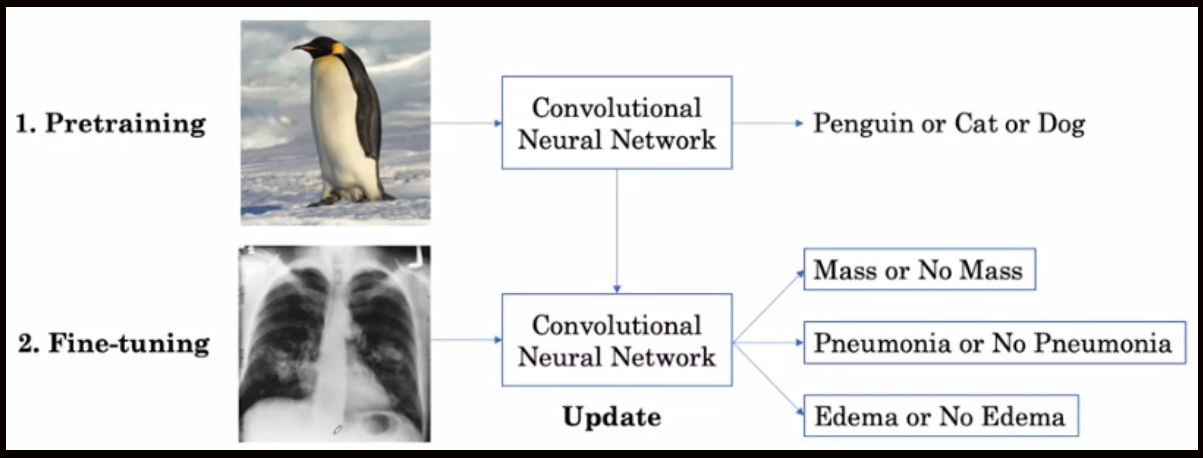
3. Dataset Choice



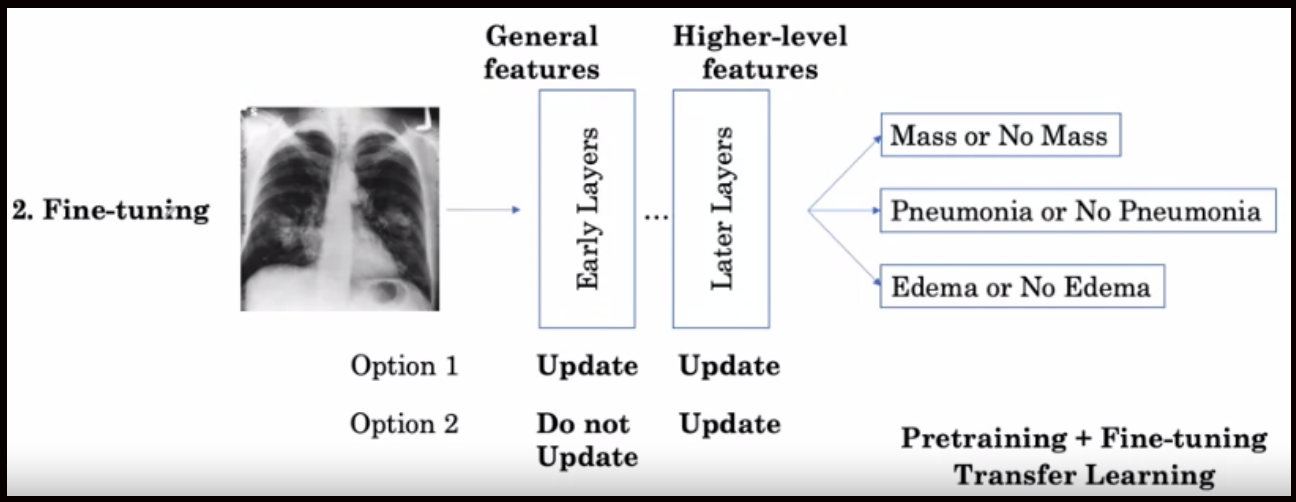

← Problem

## Solutions

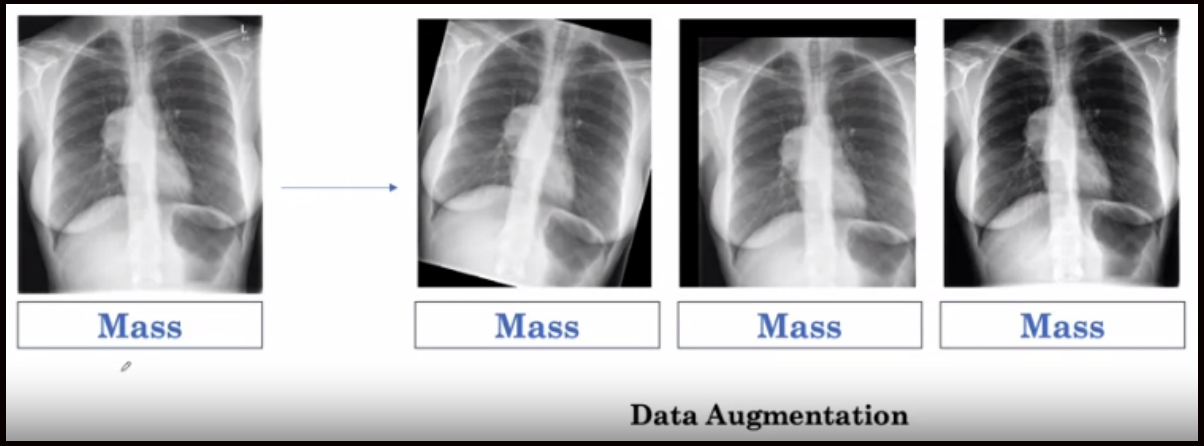
① Transfer Learning can somewhat address this issue.




* Ways of Transfer Learning

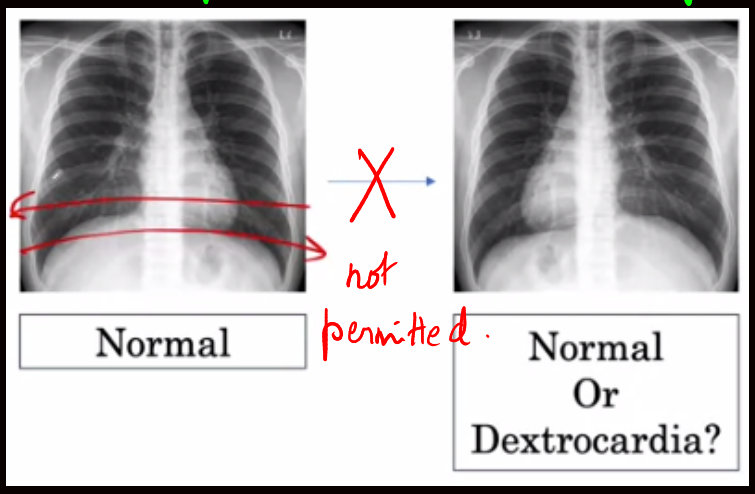



② Generating more samples

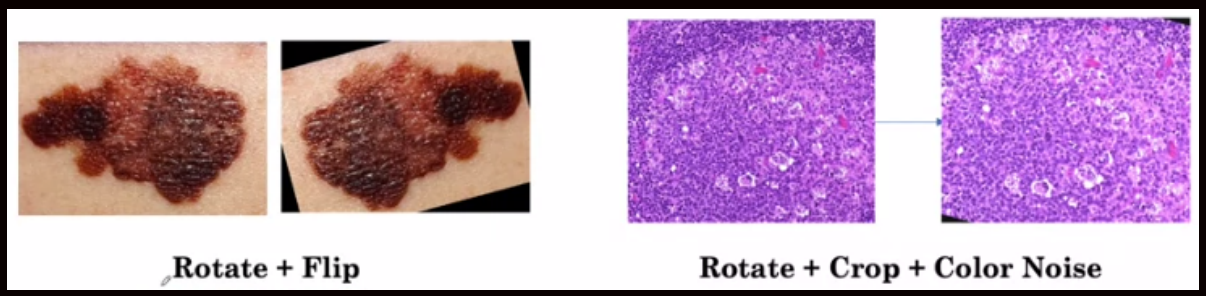


While applying data augmentation we should keep certain conditions in mind.

1. Do Augmentations reflects variations in real world ?
2. Verify if the augmentations are keeping the label same?




3.




* Possible Augmentation.

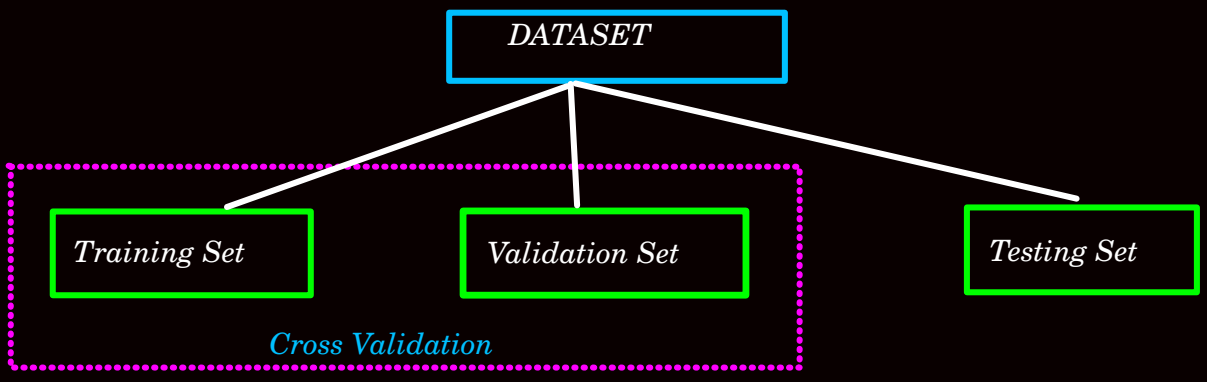
4.

Class Imbalance ⟶ Weighted Loss | Resampling

Multi-Task ⟶ Multi-Label Loss

Transfer Learning + Data Augmentation ⟶ Transfer Learning + Data Augmentation

# Model Testing

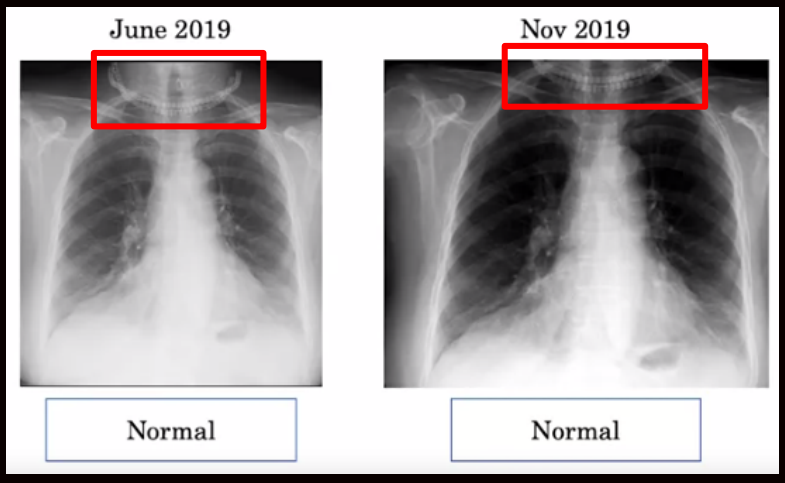



| | | | |
|---|---|---|---|
| | *Development of Models* | *Tuning and Selection of Models* | *Reporting of Results* |
| *Other Names* | *Development Set* | *Tuning or Dev Set* | *Holdout or Validation set* |

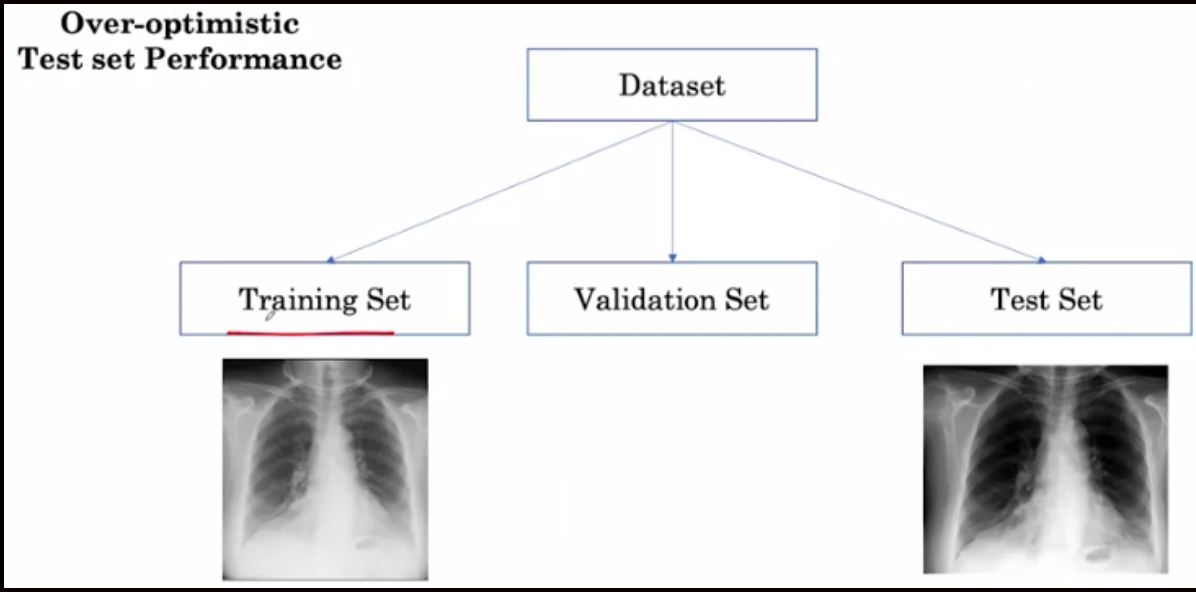
Three challenges with Ruin model testing:

1. Patient Overlap.
2. Set Sampling
3. Ground Truth

## 1. Patient Overlap



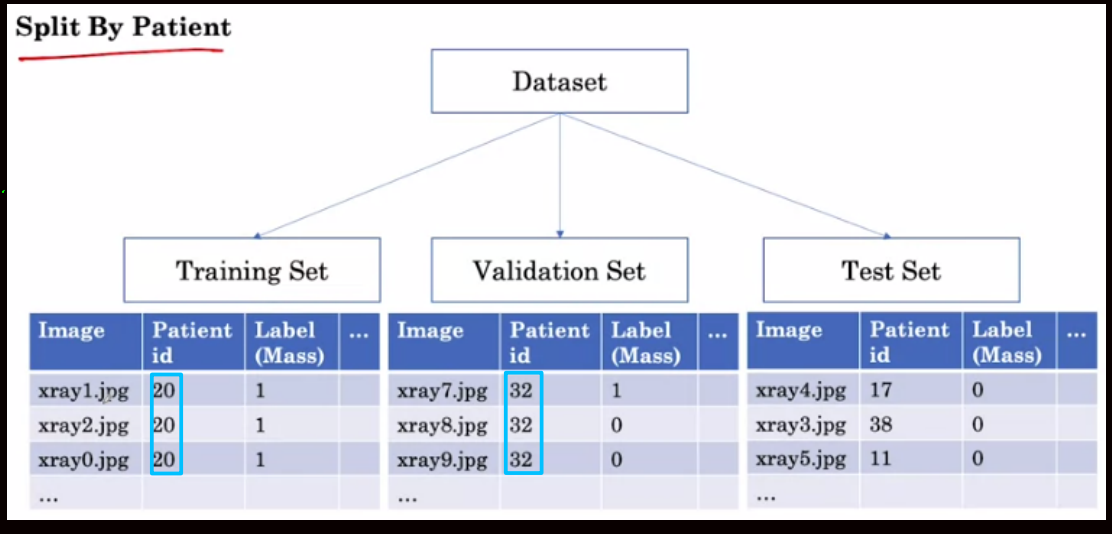

Eg. The patient wears a necklace both the times she takes an X-Ray




*One of their x-rays is sampled as part of the training set and the other as part of test. We train our deep learning model and find that it correctly predicts normal for the x-ray in the test set. The problem is that it's possible that the model actually memorized to output normal when it saw the patient with a necklace on. This is not hypothetical, deep learning models can unintentionally memorize training data, and the model could memorize rare or unique training data aspects of the patient, such as the necklace, which could help it get the right answer when testing on the same patient.*
*This would lead to an overly optimistic test set performance, where we would think that our model is better than it actually is.*

Solution:

We have to make sure that both the pictures only occur on one of the set i.e training, test or dev

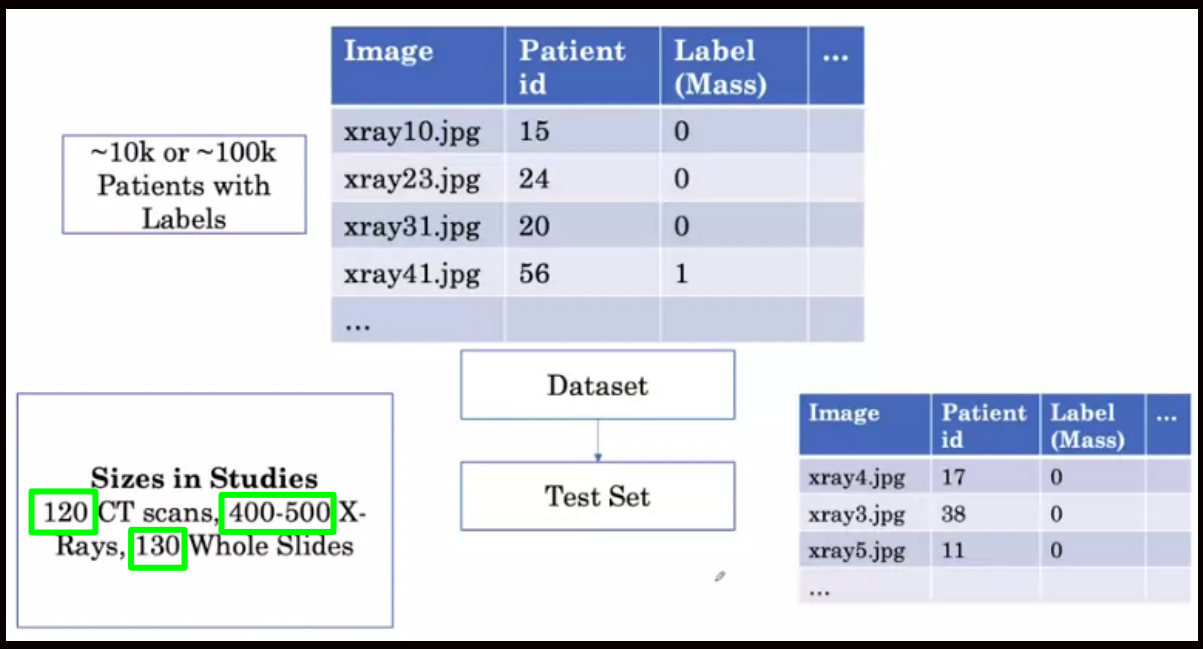



| Image | Patient id | Label (Mass) | ... |
|---|---|---|---|
| xray1.jpg | 20 | 1 | |
| xray2.jpg | 20 | 1 | |
| xray0.jpg | 20 | 1 | |
| ... | | | |

| Image | Patient id | Label (Mass) | ... |
|---|---|---|---|
| xray7.jpg | 32 | 1 | |
| xray8.jpg | 32 | 0 | |
| xray9.jpg | 32 | 0 | |
| ... | | | |

| Image | Patient id | Label (Mass) | ... |
|---|---|---|---|
| xray4.jpg | 17 | 0 | |
| xray3.jpg | 38 | 0 | |
| xray5.jpg | 11 | 0 | |
| ... | | | |

# 2. Sampling




If we sample test set randomly we might form a test set with all normal patients and with no diseases.

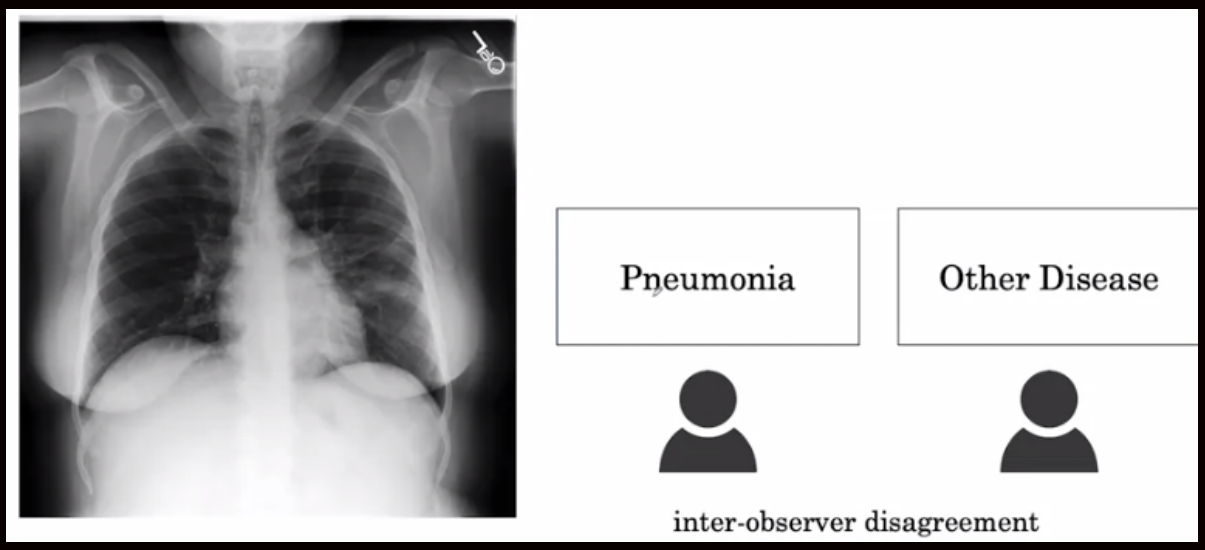
* Specially problematic for medical problems where dataset is already small.

Solution : ① Make a test set with at least X % of minority class (X = 50%). Sample test set first

② Sample validation set. The distribution of classes in the test or validation set must be same.

③ Include the remaining patients in the training set

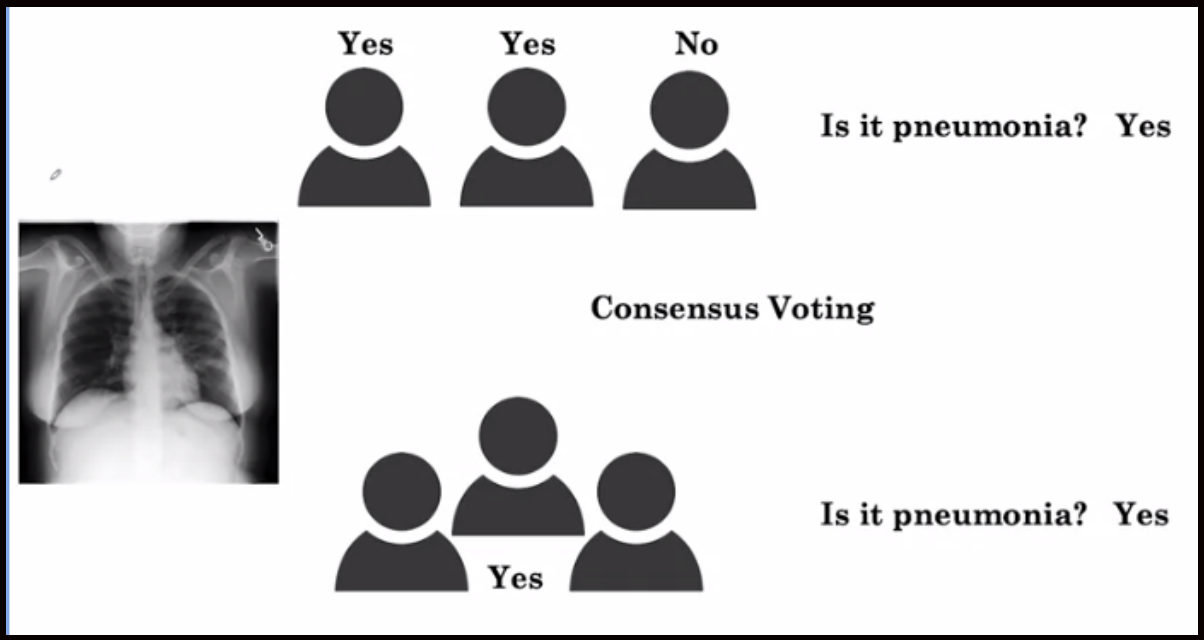
## 3. Growth Truth



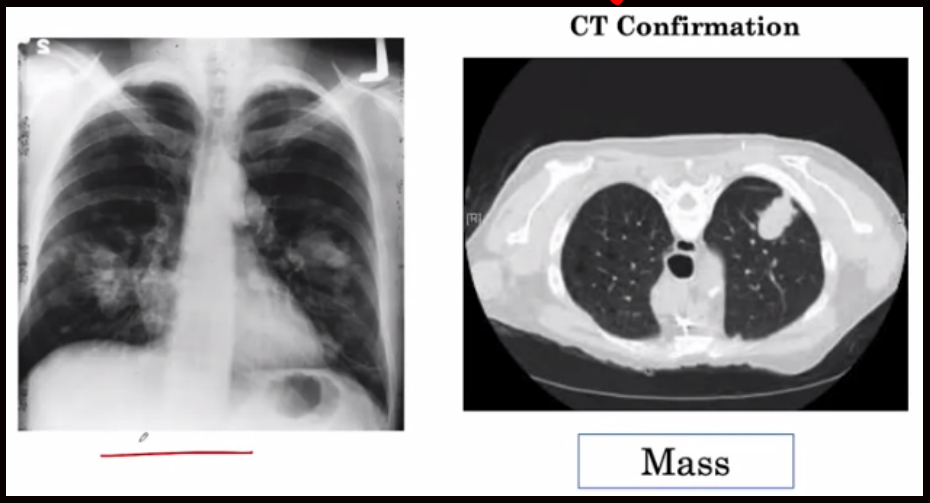

This problem of inter-observer disagreement is very common in medical imaging

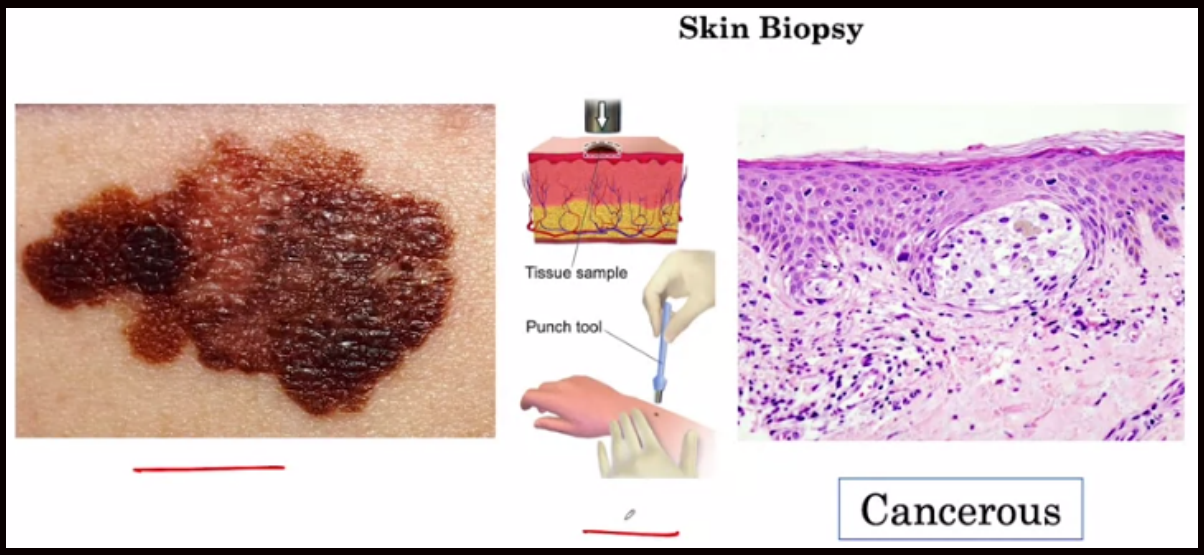
Solution:

1. Consensus Voting



## 2. Additional Medical Testing (difficult)



# WEEK-2

## Sensitivity, Specificity & Evaluation Metrics.

How good is a model :

Accuracy: $\frac{\text{Examples correctly classified}}{\text{Total no. of examples.}}$

Eg.

| Ground Truth | Model 2 | Model 1 |
|---|---|---|
| Normal | - | - |
| Normal | + | - |
| Normal | - | - |
| Normal | - | - |
| Normal | - | - |
| Disease | + | - |
| Normal | - | - |
| Disease | + | - |
| Normal | + | - |
| Normal | - | - |

$\text{Accuracy}_{\text{model1}} = \frac{8}{10}$ $\text{Accuracy}_{\text{model2}} = \frac{8}{10}$

* Although both the models have same accuracy, still the model 2 seems to be more good as it is atleast trying to distinguish bet$^{n}$ the two labels

$$\text{Accuracy} = P(\text{correct})$$

$$= P(\text{correct} \wedge \text{disease}) + P(\text{correct} \wedge \text{normal})$$

Using $P(A \wedge B) = P(A|B)\,P(B)$

$$\text{Accuracy} = P(\text{correct}|\text{disease})\,P(\text{disease}) + P(\text{correct}|\text{normal})\,P(\text{normal})$$

$$\text{Accuracy} = P(+|\text{disease})\,P(\text{disease}) + P(-|\text{normal})\,P(\text{normal})$$

$P(+|\text{disease})$: Sensitivity (true +ve rate)

$P(-|\text{normal})$: Specificity (true -ve rate)

Sensitivity: $P(+|\text{disease})$. If the patient has the disease what is the probability that the model predicts positive?

Specificity: $P(-|\text{normal})$ If the patient is normal, what is the probability that the model predicts -ve.

$$\text{Accuracy} = P(+|\text{disease})\,\underbrace{P(\text{disease})}_{\text{Prevalence}} + P(-|\text{normal})\,\underbrace{P(\text{normal})}_{1-P(\text{disease})\ =\ 1-\text{prevalence}}$$

* $$\text{Accuracy} = \text{Sensitivity} \times \text{Prevalence} + \text{Specificity} \times (1-\text{prevalence})$$

* Example

| Ground Truth | Model |
|---|---|
| Normal | - |
| Normal | - |
| Disease | + |
| Normal | - |
| Normal | - |
| Disease | - |
| Normal | - |
| Disease | + |
| Normal | + |
| Normal | - |

**Sensitivity**

P(+ | disease)

$$\frac{\#(+ \text{ and disease})}{\#(\text{disease})} = \frac{2}{3} = 0.67$$

**Specificity**

P(− | normal)

$$\frac{\#(- \text{ and normal})}{\#(\text{normal})} = \frac{6}{7} = 0.86$$

$$\text{Prevalence} = P(\text{disease}) = \frac{\#(\text{disease})}{\#(\text{total})} = \frac{3}{10} = 0.3$$

$$\therefore \text{Accuracy} = 0.67 \times 0.3 + 0.86 \times (1 - 0.3) = 0.8$$

* If a model prediction is +ve, what is the probability that a patient actually has the disease? → PPV

$P(\text{disease} \mid +)$ → PPV (Positive Predictive Value)

predicted

* If a patient is -ve, what is the probability that the patient is normal? → NPV

$P(\text{normal} \mid -)$ → NPV (Negative Predictive Value)

Example:

| Ground Truth | Model |
|---|---|
| Normal | - |
| Disease | + |
| Normal | + |
| Normal | - |
| Normal | - |
| Disease | - |
| Normal | - |
| Disease | + |
| Normal | + |
| Normal | - |

**PPV**

$$P(\text{disease} \mid +)$$

$$\frac{\#(+ \text{ and disease})}{\#(+)} = \frac{2}{4} = 0.5$$

**NPV**

$$P(\text{normal} \mid -)$$

$$\frac{\#(- \text{ and normal})}{\#(-)} = \frac{5}{6} = 0.83$$

Summary

P(disease | +) → PPV

P(+ | disease) → Sensitivity

P(normal | -) → NPV

P(- | normal) → Specificity

Confusion Matrix

| Ground Truth | Model |
|---|---|
| Normal | - |
| Disease | + |
| Normal | + |
| Normal | - |
| Normal | - |
| Disease | - |
| Normal | - |
| Disease | + |
| Normal | + |
| Normal | - |

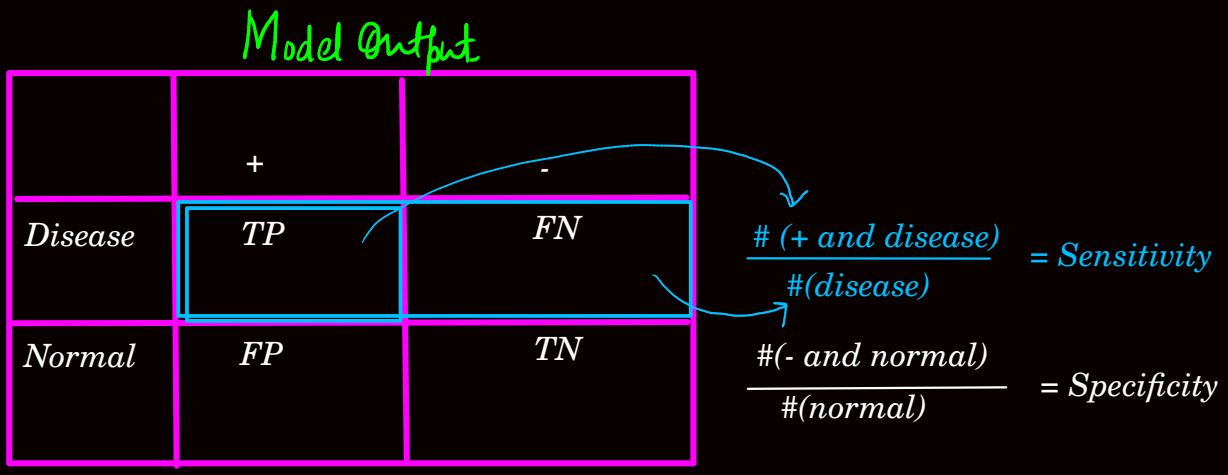
**Model Output**

| GT | + | - |
|---|---|---|
| Disease | 2 | 1 |
| Normal | 2 | 5 |

* GT → Ground Truth.

| GT | Model Output: + | Model Output: - |
|---|---|---|
| Disease | True Positive (TP) | False Negative (FN) |
| Normal | False Positive (FP) | True Negative (TN) |

| GT | Model Output: + | Model Output: - |
|---|---|---|
| Disease | 2 | 1 |
| Normal | 2 | 5 |

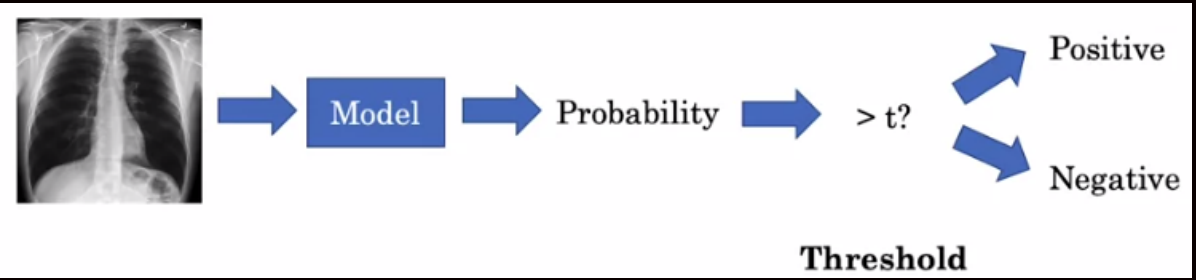



$\frac{\#(+ \text{ and disease})}{\#(+)} = PPV$ $\quad \frac{\#(- \text{ and normal})}{\#(-)} = NPV$

* $PPV = \frac{TP}{TP+FP}$ ; $NPV = \frac{TN}{TN+FN}$ ; $Sensitivity = \frac{TP}{TP+FN}$ ; $Specificity = \frac{TN}{FP+TN}$

* $PPV = \frac{\text{Sensitivity} \times \text{Prevalence}}{\text{Sensitivity} \times \text{Prevalence} + (1-\text{specificity}) \times (1-\text{prevalence})}$

# Roc Curve and Threshold



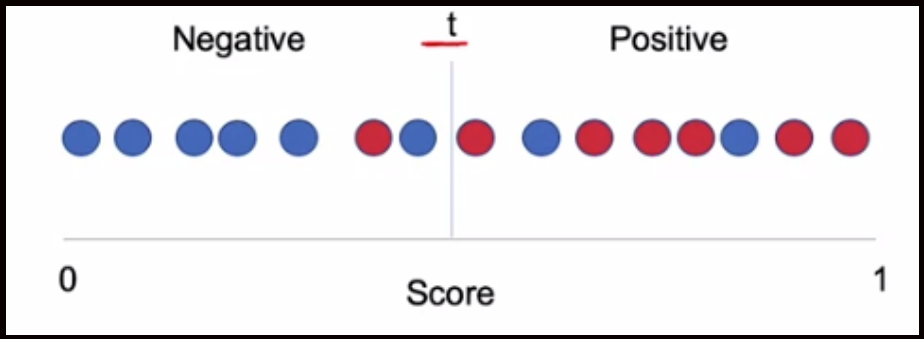

P (+ | disease)
↑
Sensitivity = A (say)

P (- | normal)
↑
Specificity = B (say)

t = 0 ; A = 1, B = 0
t = 1 ; A = 0, B = 1

Example:

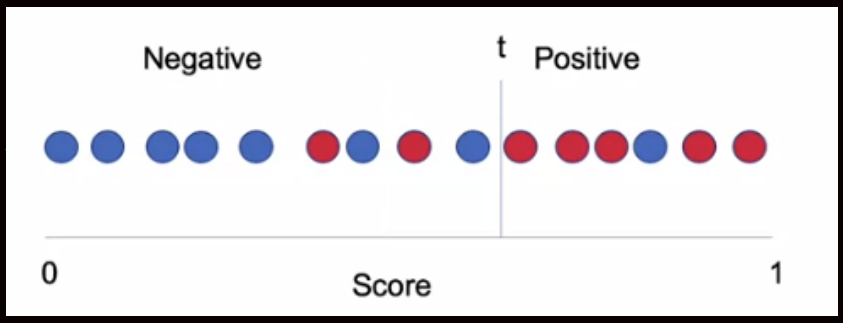
| X-Ray | Output Probability (Score) |
|---|---|
| 1 | 0.30 |
| 2 | 0.42 |
| 3 | 0.78 |
| ... | ... |
| 15 | 0.98 |



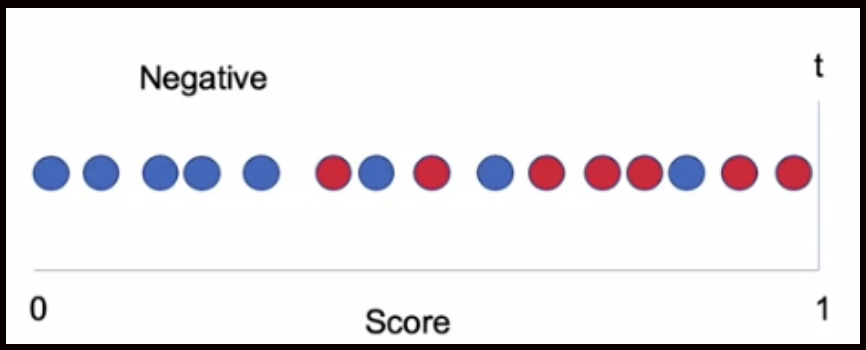

Red → Diseased.
Blue → Normal.

P (+ | disease) → Sensitivity = $\frac{6}{7}$ = 0.85

P (- | normal) → Specificity = $\frac{6}{8}$ = 0.75




Sensitivity = $\frac{5}{7}$ = 0.71 Specificity = $\frac{7}{8}$ = 0.88




Sensitivity = $\frac{0}{7}$ = 0 Specificity = $\frac{8}{8}$ = 1

## Confidence Intervals

$p$
Unknown

$\hat{p} = 0.80$, n = 100
Known

With 95% confidence, p is in the interval [0.72, 0.88]

# Interpretation of 95% Confidence Interval

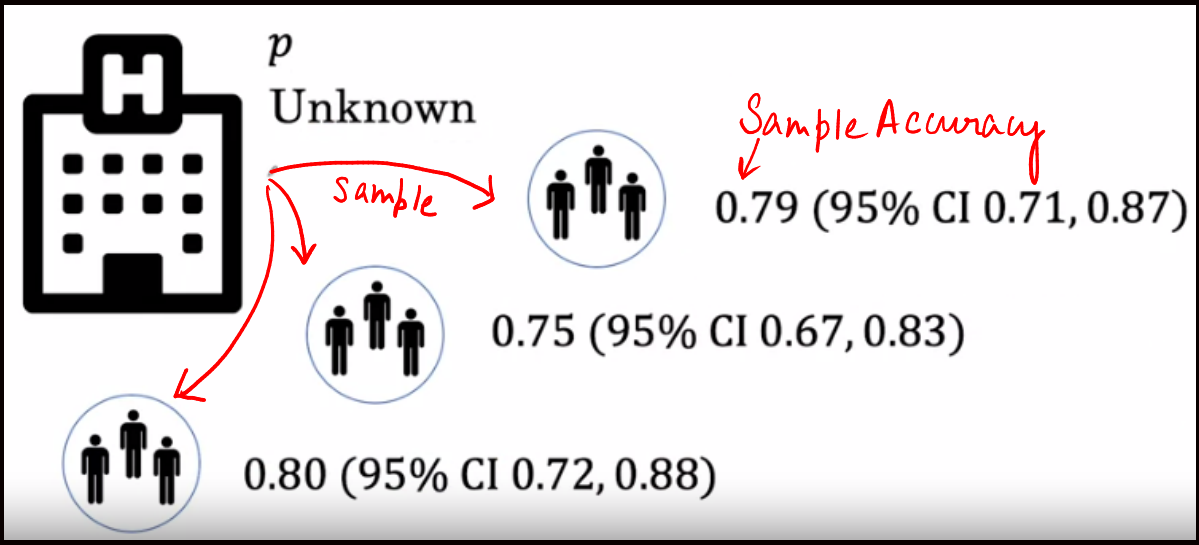
Within 95% confidence, p in the interval [0.72, 0.88]

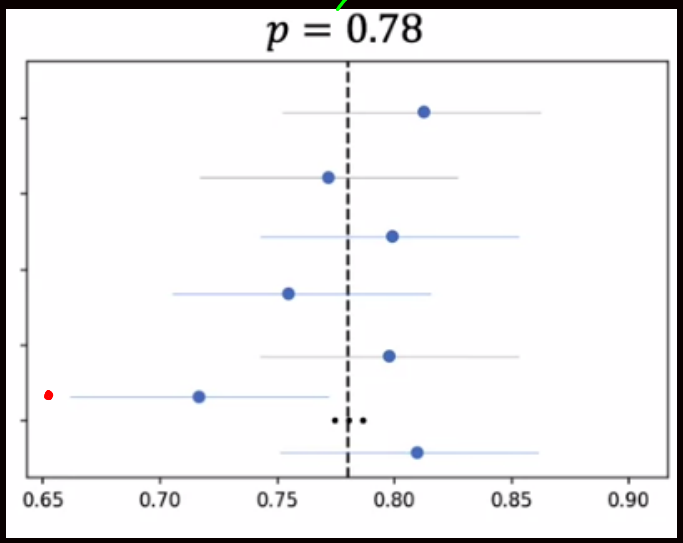
## Wrong Interpretations. ※

✗ There is 95% probability that p lies within the interval [0.72, 0.88]

✗ 95% of the sample accuracies lie within the interval [0.72, 0.88]

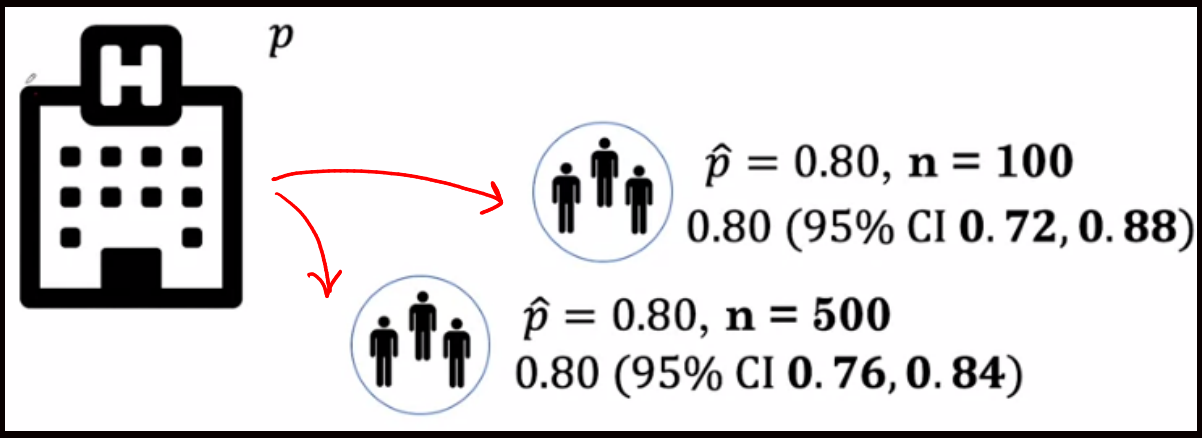
✱ True Interpretation.



True Interpretation: In repeated sampling, this method produces intervals that will include the population accuracy in 95% of samples

* *In practice, we don't compute the confidence intervals for many samples. We only compute our model performance on one sample. For our sample, the computed confidence interval may or may not contain p. However, we can be 95% confident that it does.*




Although in the above example for both the sample the accuracy is the same, the confidence intervals (CI's) tightens where the sample size(n) is high

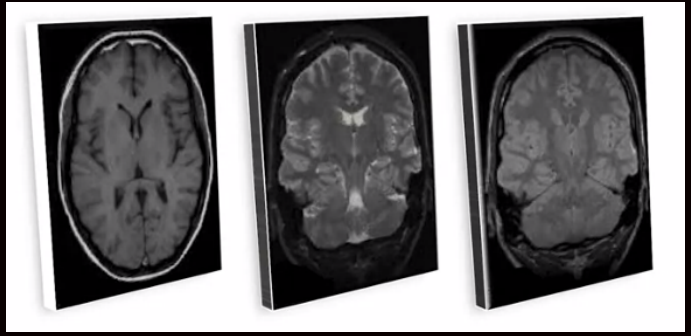
WEEK-3

MRI Data & Image Registration

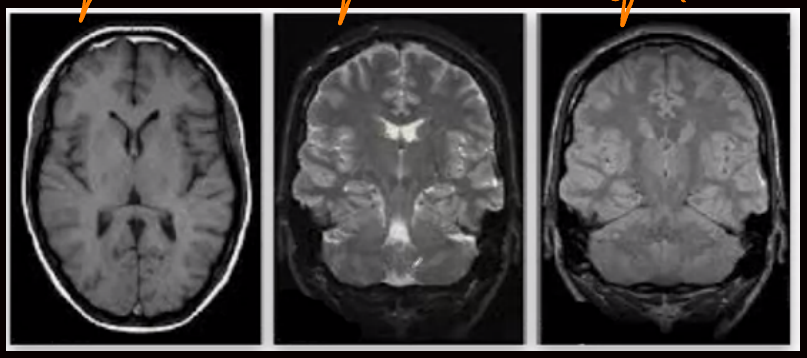
1. MRI data in Coronal.

2. MRI consists of multiple imaging sequences

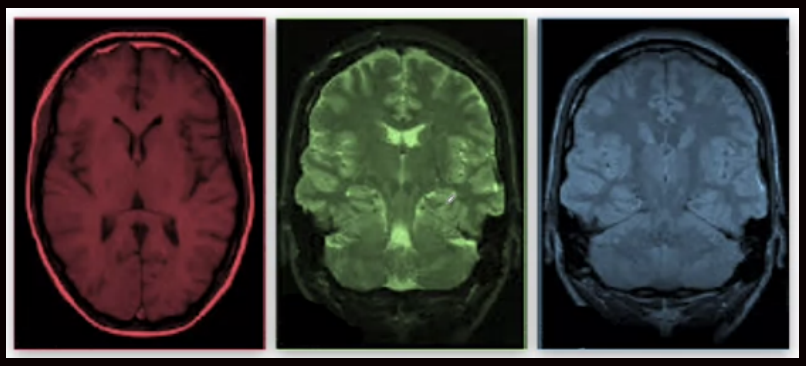
i.e multiple 3D Volumes into 1 3D Volume



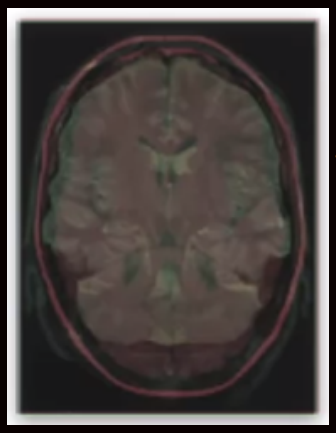
3. Pick a Slice




4. Treat the slices as different channels

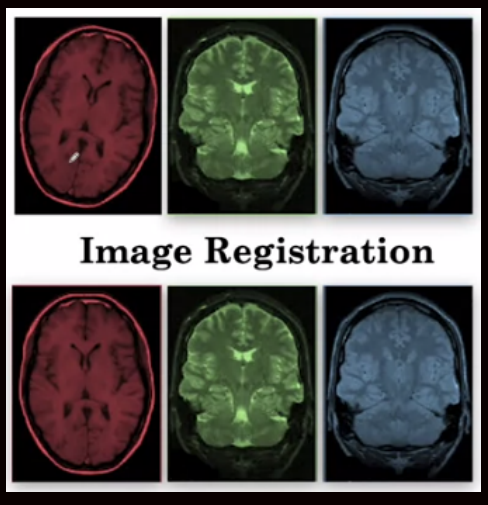


5 Combine the three images.



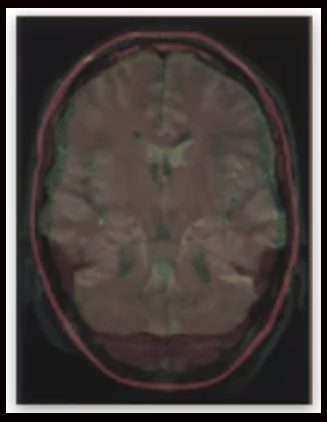
* But sometimes if the three channels are not aligned properly the combination would cause misalignment problem

* Solution →

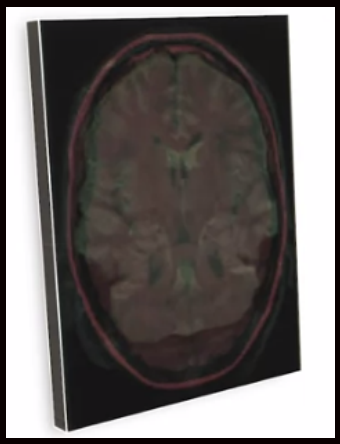


# Segmentation

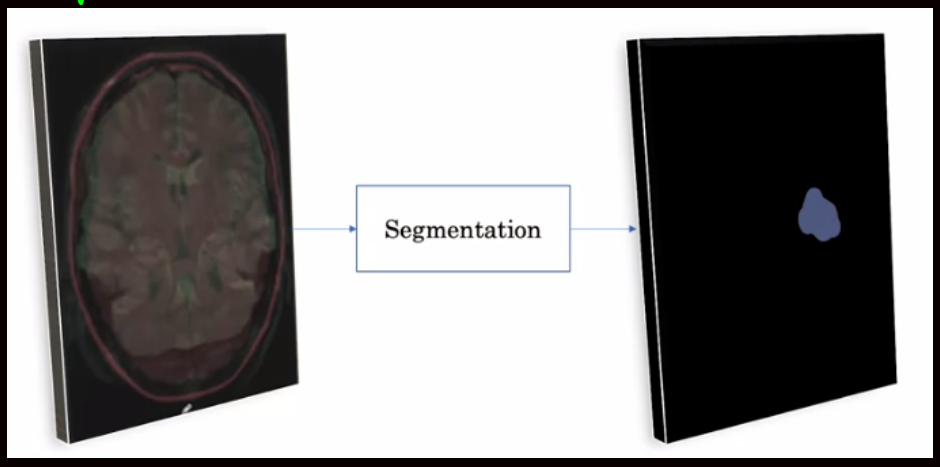
* Take the combined sequence.



* Apply to all slices. & combine information of many different sequences



* Segment out the indivisual tissues.

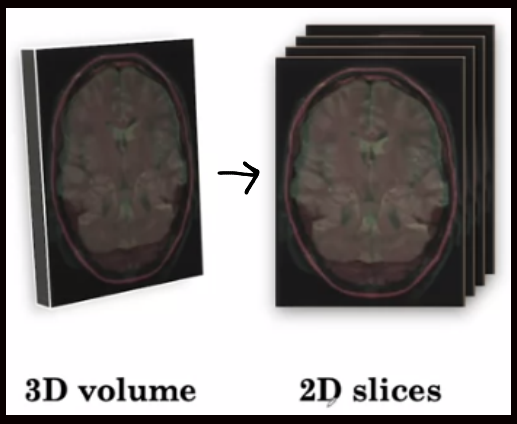



There are two approaches adopted in image segmentation

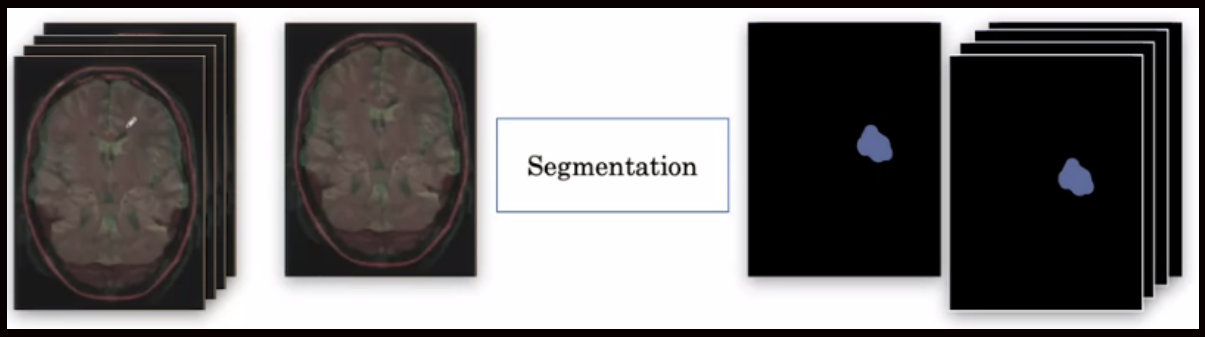
1. 2D Approach
2. 3D Approach.

## 2D Approach

1.




2.

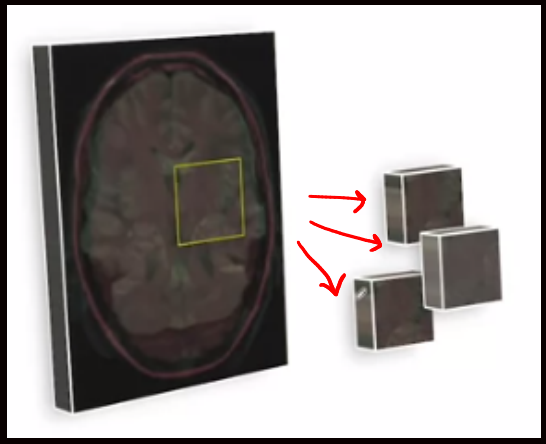



3. The 2D slices can then be combined to form the output 3D volume

* Drawback: We might lose imp. 3D context.

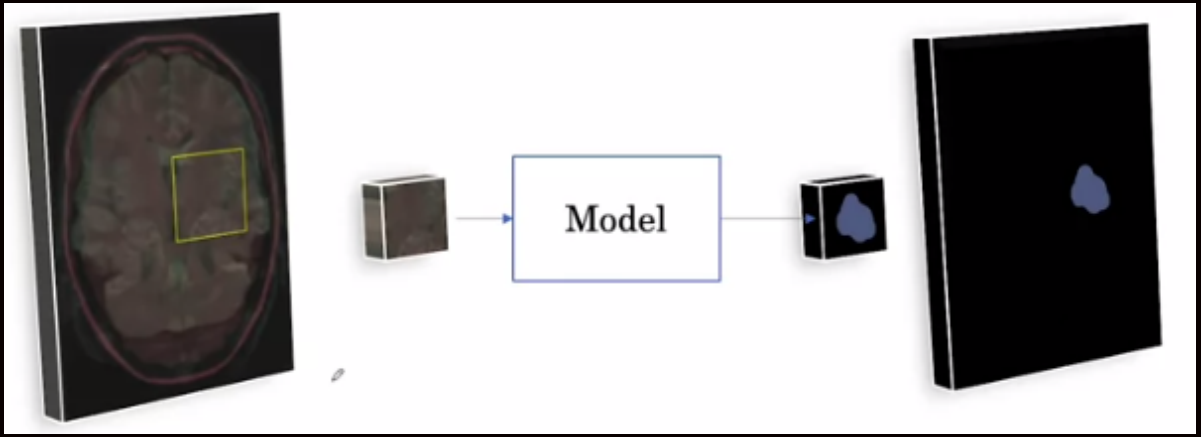
## 3D Approach

Step 1:



Break the 3D volumes into many subvolumes

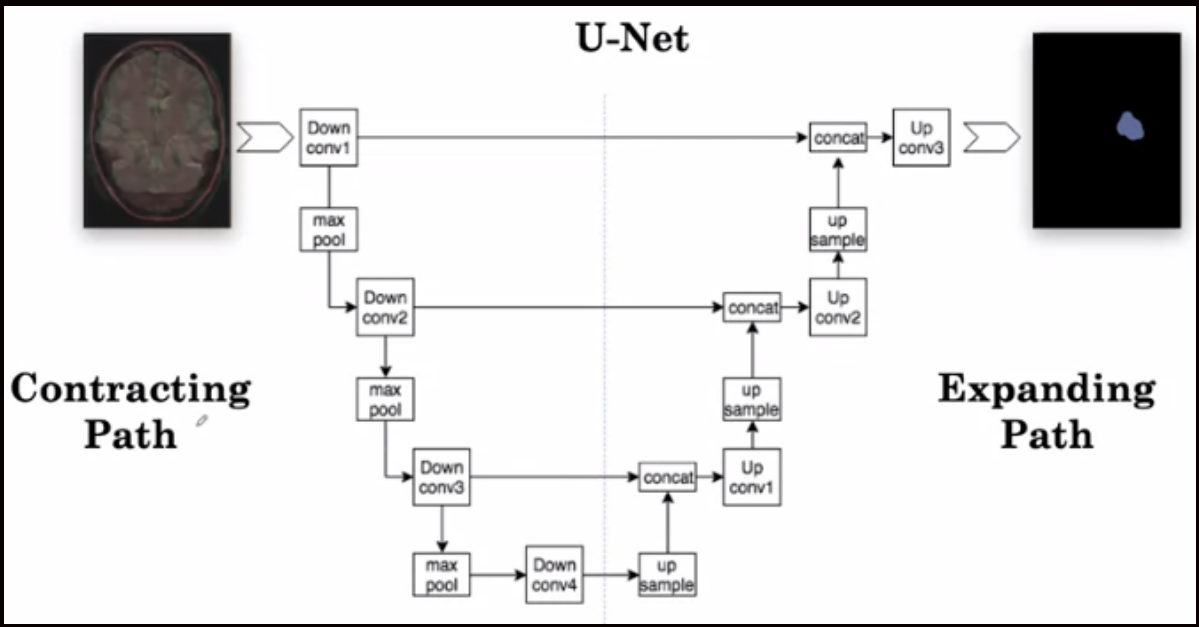
2.



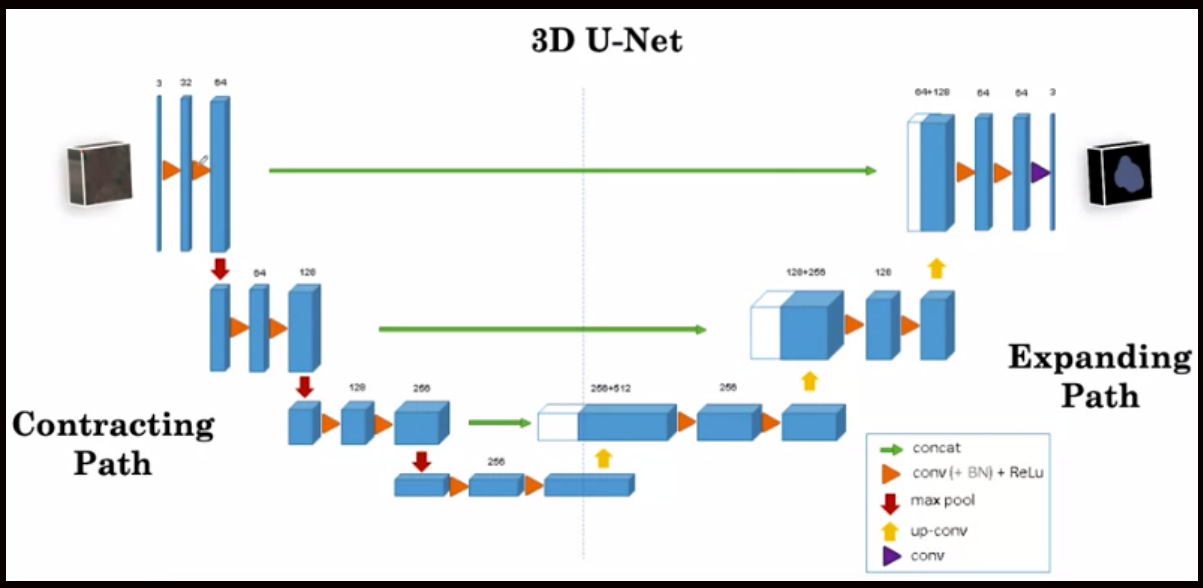

* Drawback: We might lose some spatial information.

## U-Net Architechture

This is the 2D Approach where 2D slices are used

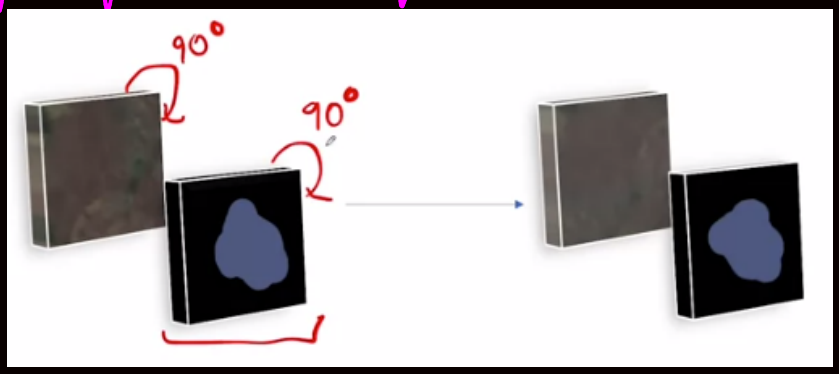


## 3D U-Net (3D Subvolumes are used)




## Data Augmentation for Segmentation.

* If we rotate the i/p we also need to rotate the output segmentation image by the same degree.




* Transformations are needed to be applied to the entire 3D volume

Ex:

| P | | | G | | | |
|---|---|---|---|---|---|---|
| $p_1$ 0.1 | $p_2$ 0.1 | $p_3$ 0.1 | $g_1$ 0 | $g_2$ 0 | $g_3$ 0 | 1 (Tumor) |
| $p_4$ 0.8 | $p_5$ 0.9 | $p_6$ 0.9 | $g_4$ 0 | $g_5$ 1 | $g_6$ 1 | 0 (Normal Brain Tissue) |
| $p_7$ 0.1 | $p_8$ 0.4 | $p_9$ 0.1 | $g_7$ 0 | $g_8$ 1 | $g_9$ 0 | |

P = Predicted Segmented Image

G = Ground Truth.

| i | p | g |
|---|---|---|
| 1 | 0.1 | 0 |
| 2 | 0.1 | 0 |
| 3 | 0.1 | 0 |
| 4 | 0.8 | 0 |
| 5 | 0.9 | 1 |
| 6 | 0.9 | 1 |
| 7 | 0.1 | 0 |
| 8 | 0.4 | 1 |
| 9 | 0.1 | 0 |

## Soft Dice Loss

* Commonly used for image segmentation
* Works well for imbalanced data

$$L(P,G) = 1 - \left[ \frac{2 \sum_i^n p_i g_i}{\sum_i^n p_i^2 + \sum_i^n g_i^2} \right]$$

↳ Overlap (needs to be large)

| i | p | g | $p_i g_i$ | $p_i^2$ | $g_i^2$ |
|---|---|---|---|---|---|
| 1 | 0.1 | 0 | 0 | 0.01 | 0 |
| 2 | 0.1 | 0 | 0 | 0.01 | 0 |
| 3 | 0.1 | 0 | 0 | 0.01 | 0 |
| 4 | 0.8 | 0 | 0 | 0.64 | 0 |
| 5 | 0.9 | 1 | 0.9 | 0.81 | 1 |
| 6 | 0.9 | 1 | 0.9 | 0.81 | 1 |
| 7 | 0.1 | 0 | 0 | 0.01 | 0 |
| 8 | 0.4 | 1 | 0.4 | 0.16 | 1 |
| 9 | 0.1 | 0 | 0 | 0.01 | 0 |
| | | | 2.2 | 2.47 | 3 |

$$\therefore L = 1 - \frac{2 \times 2.2}{2.47 + 3}$$

$$= 1 - \frac{4.4}{5.47}$$

$$= 0.2$$

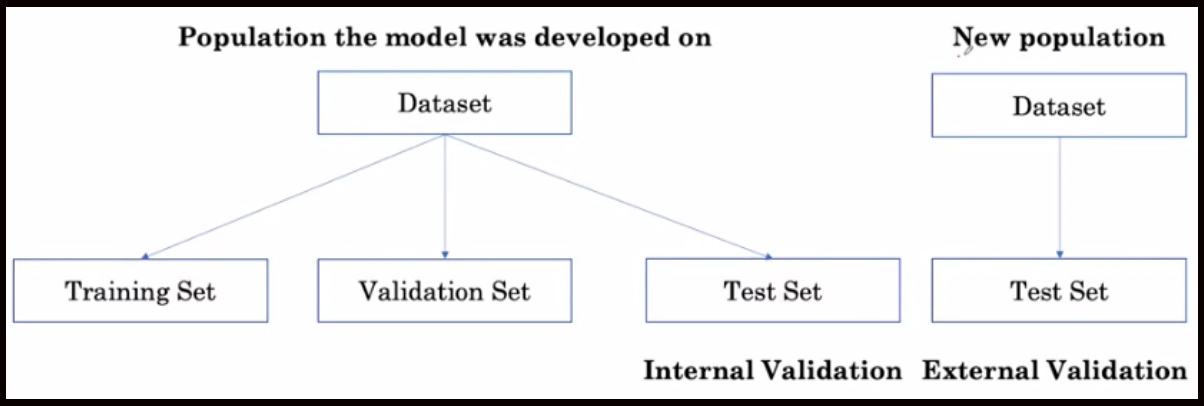
## Different Population & Diagnostic Technology

*One of the main challenges with applying AI algorithms in the clinic, is achieving reliable generalization. Generalization can be hard due to a variety of reasons.*

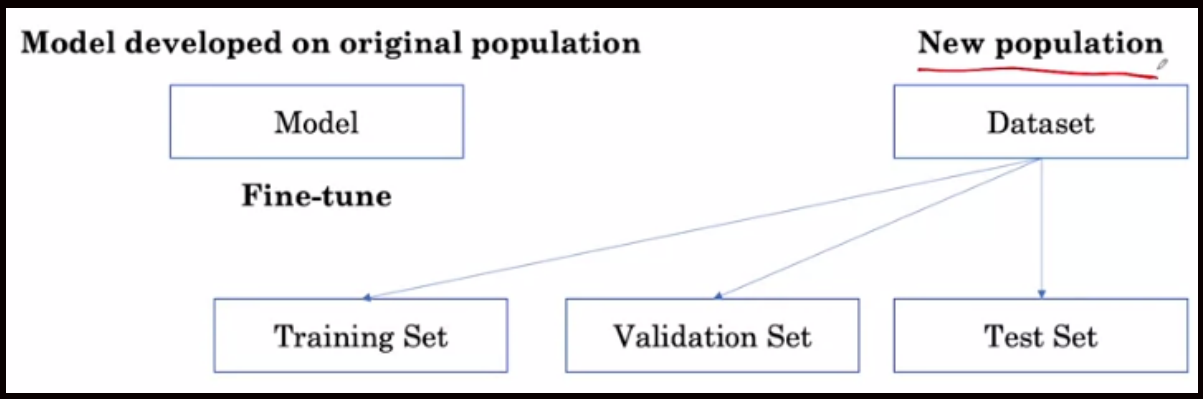
*1. We developed our chest x-ray model on US data, and we wanted to apply it to a hospi--tal in a different country, say India. In India, the patient population might have x-rays that l ooked different than what the model has been trained on.*

*2. We've been able to measure the model performance on data on brain tumor segmentation collected from a few countries over a few years, but MRI technology is not standard across the globe and across time. The latest scanners have much higher resolution than older scanners. Before we apply the segmentation model in a new hospital, we'd want to make sure that the model is able to generalize to the resolution of the scanner at the hospital.*

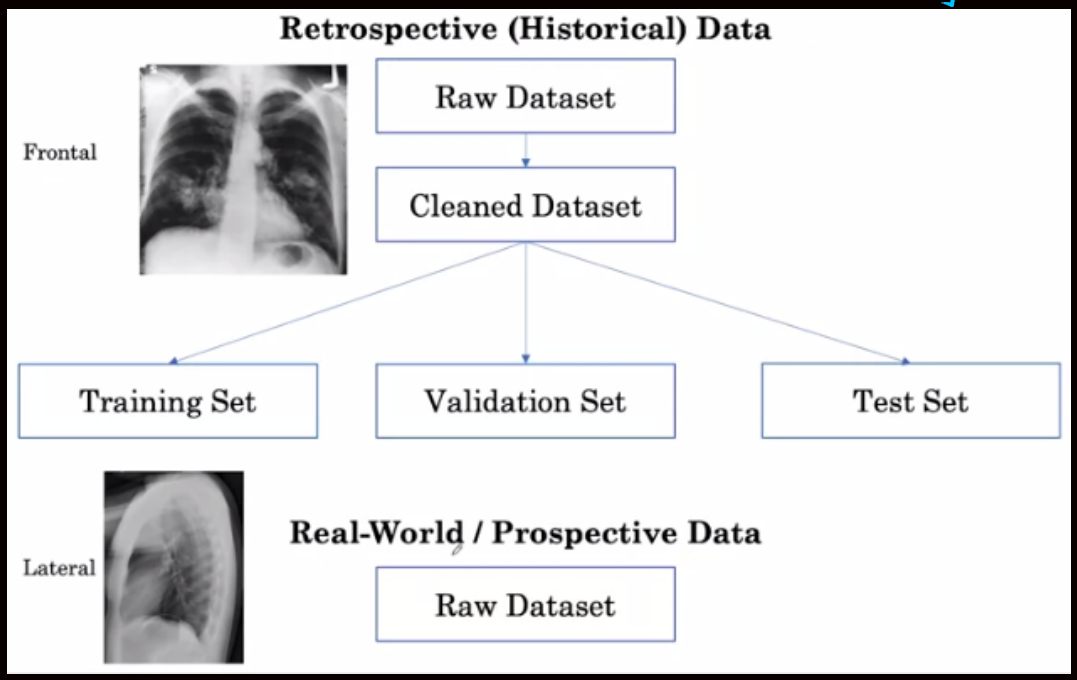
External Validation




If the performance is not good on the new population we can get a few more samples from the new dataset




* Problem with Real World Data. [Lateral View, Raw]

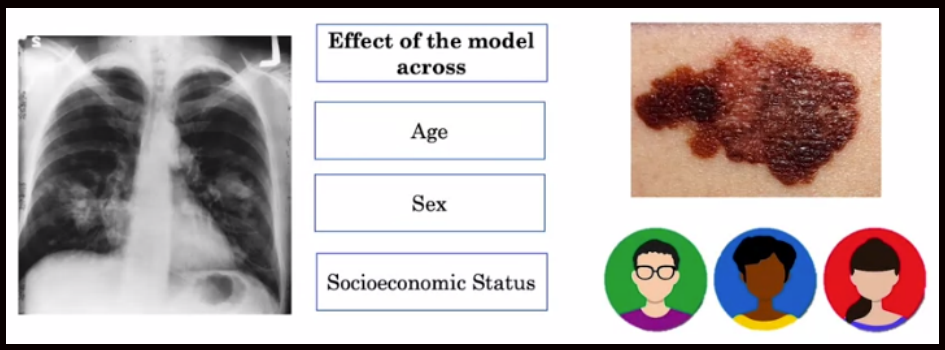


# Measuring Patient Outcomes

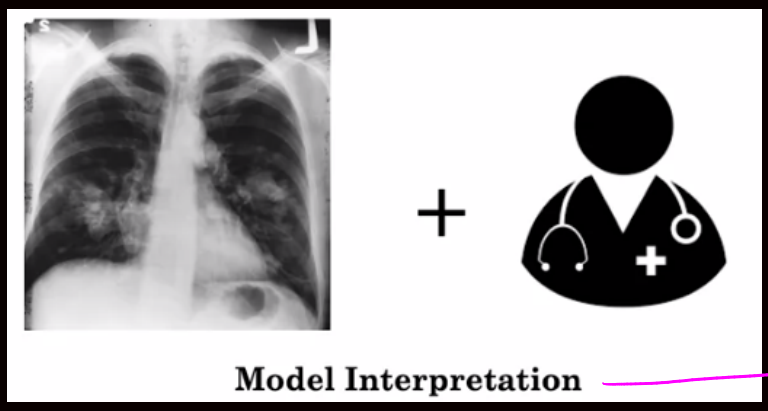
* Decision Curve Analysis
* Randomized Controlled Trails

} Will be covered later.

*




*




→ Is difficult for AI models as we are unaware on what basis the output is predicted