Department of Computing

Imperial College of Science, Technology and Medicine

# Causality and Generative Models

*Author: Anton Zhitomirsky*

## Contents

# 1 Trustworthy AI/ML

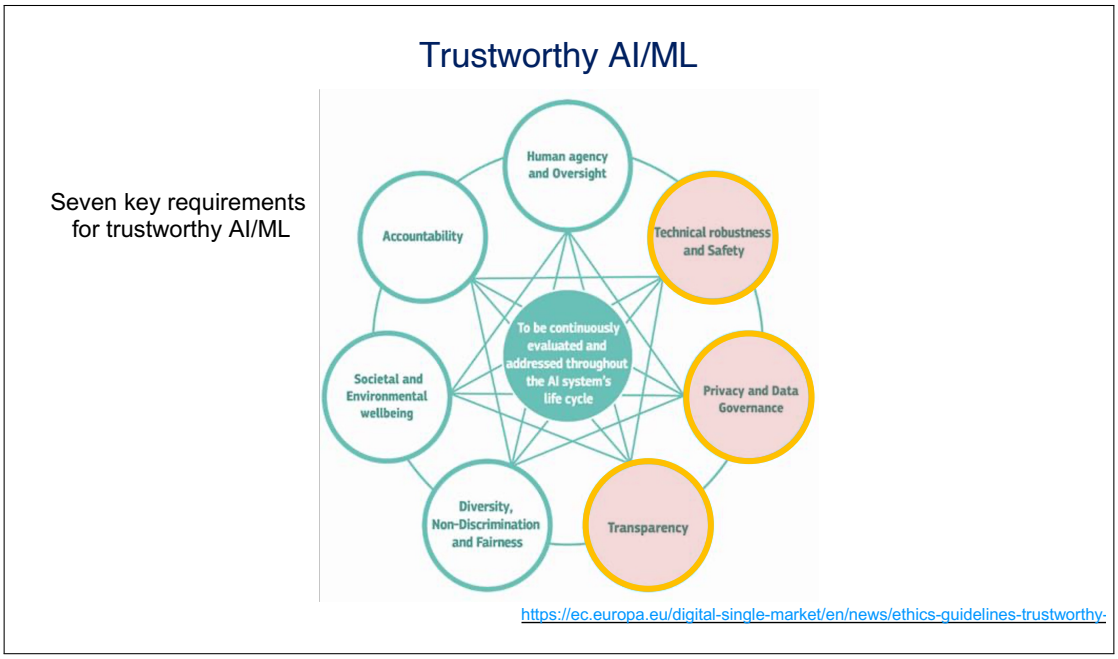



**Figure 1:** caption

## 1.1 The need for data

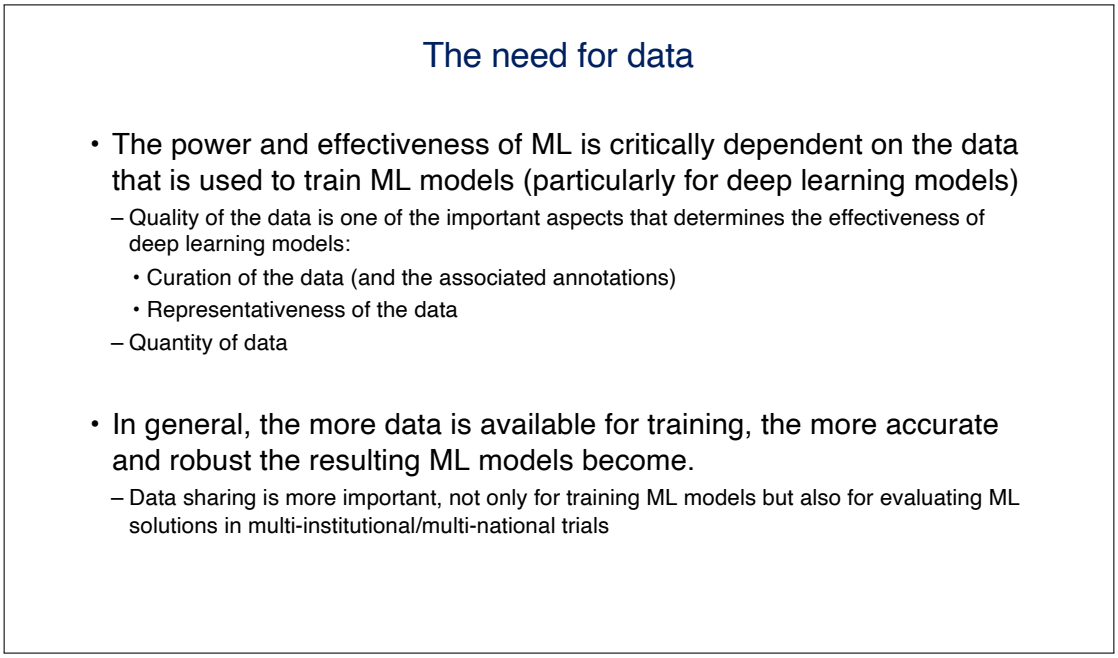



**Figure 2:** caption

## 1.2 What are the hurdles to getting more data?

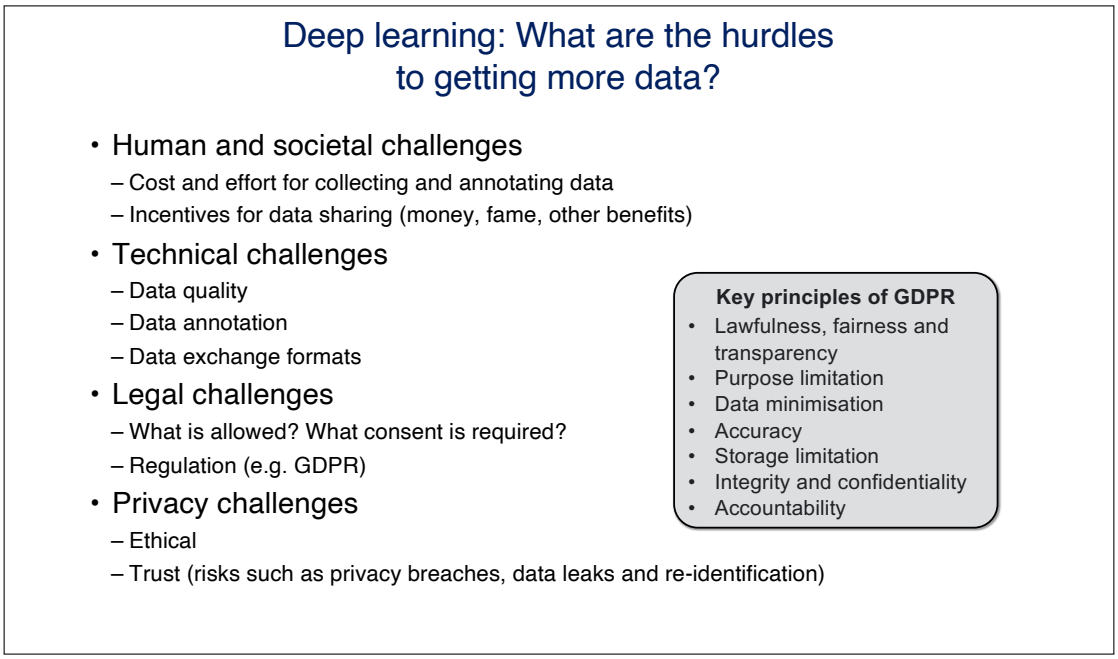



**Figure 3:** caption

## 1.3 Secure and Privacy-aware ML

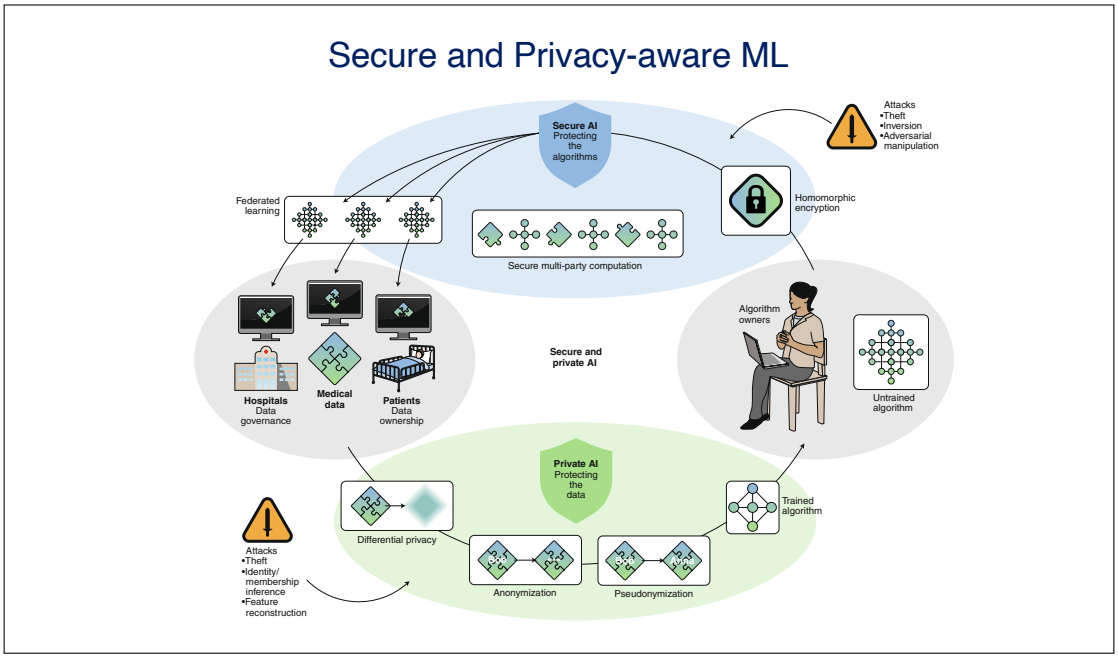



**Figure 4:** caption

## 1.4 Secure and Privacy-preserving ML

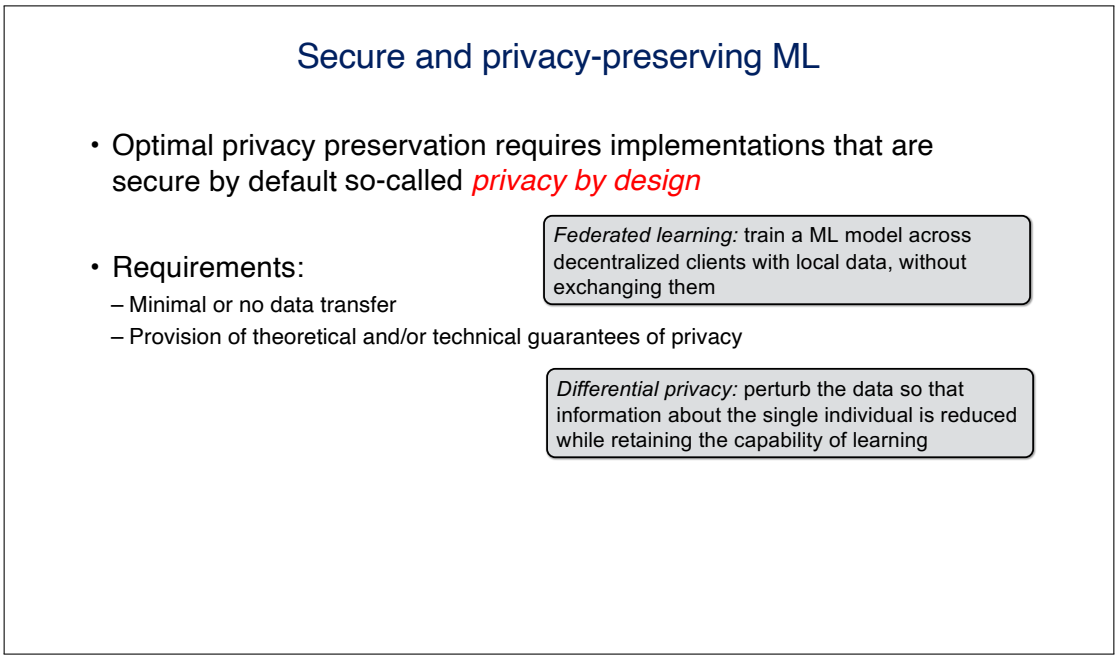



**Figure 5:** caption

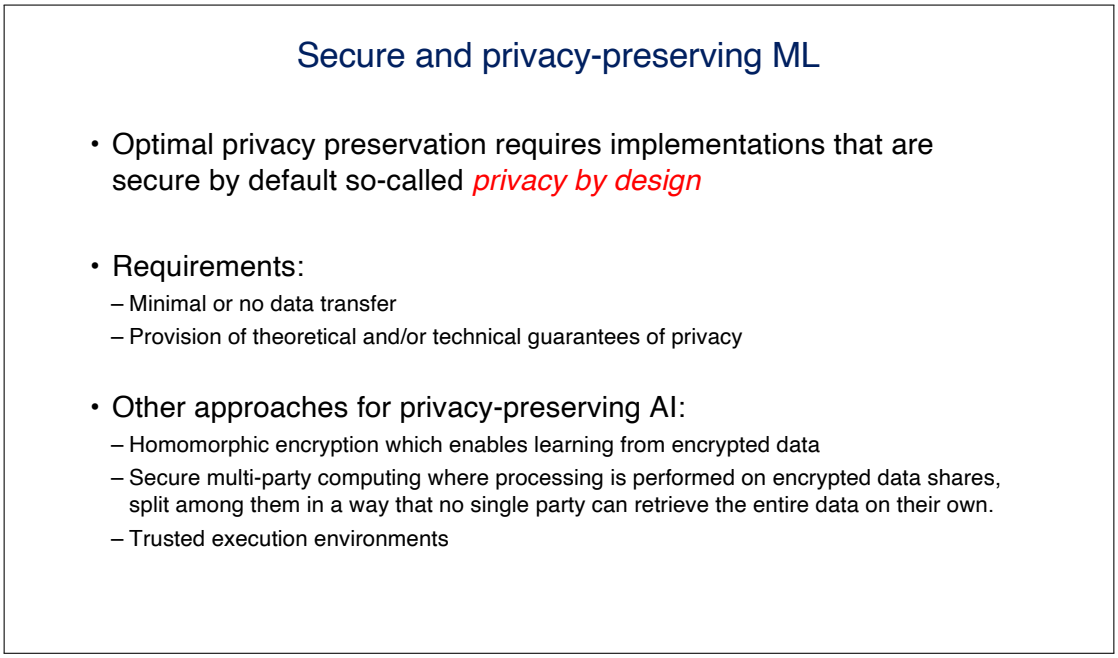



**Figure 6:** caption

## 1.5 Federated learning

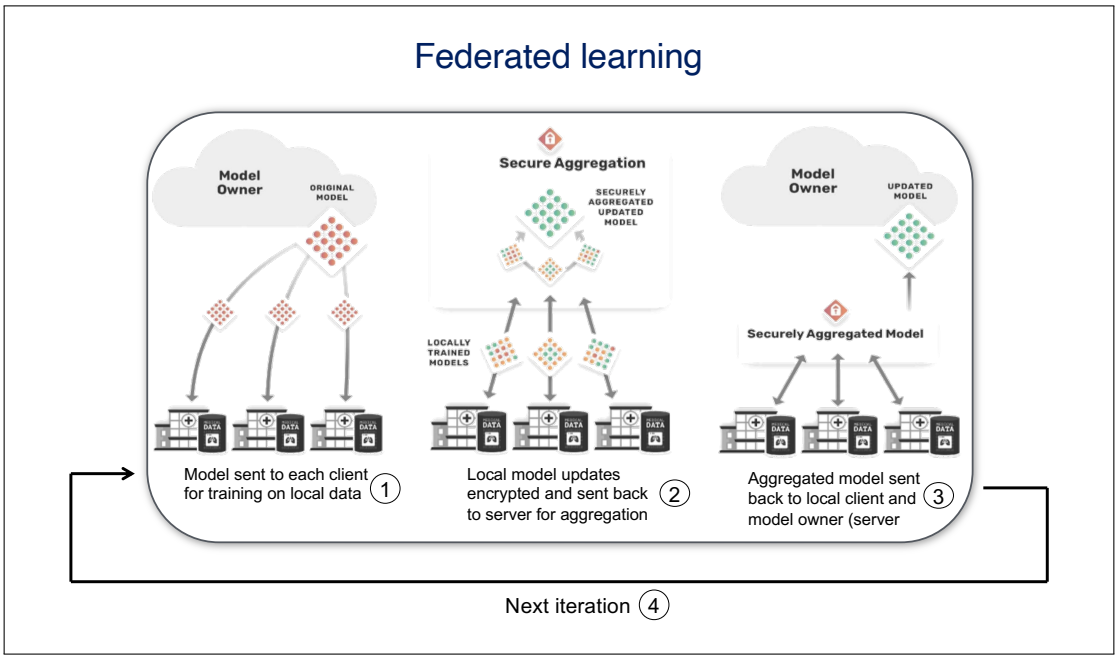



**Figure 7:** caption

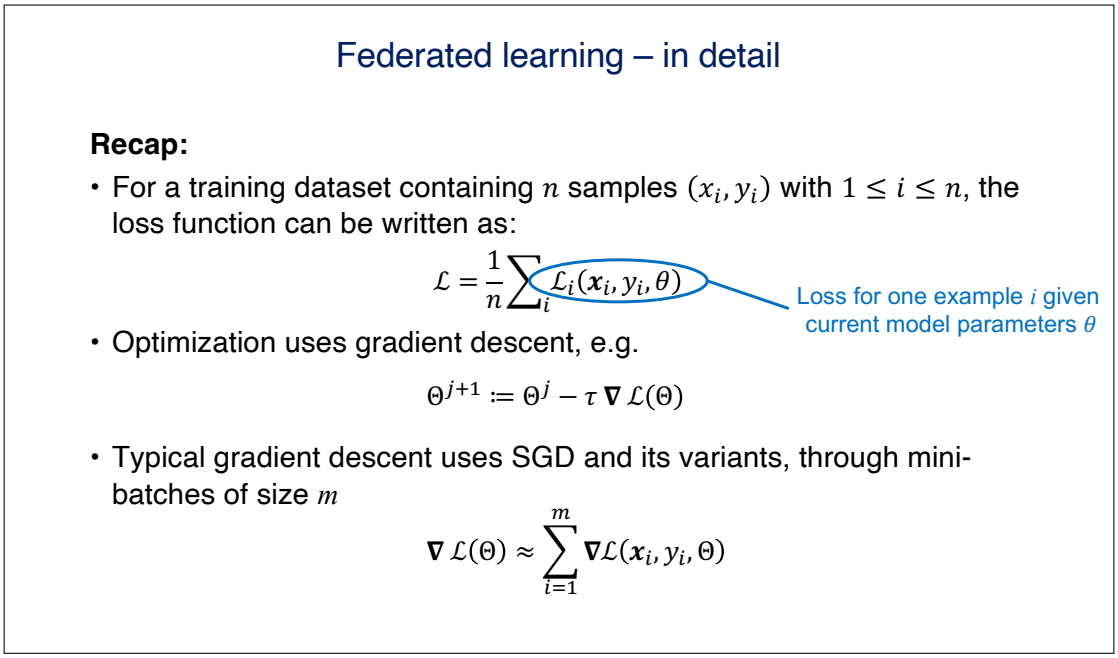



**Figure 8:** caption

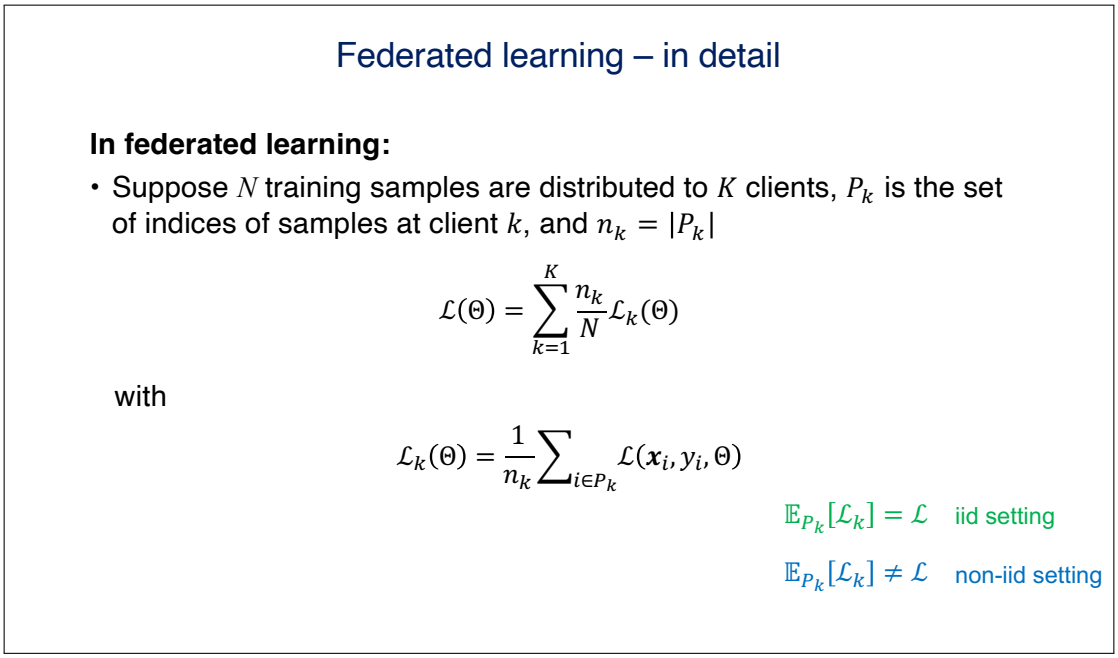



**Figure 9:** caption

### 1.5.1 Federated learning - SGD

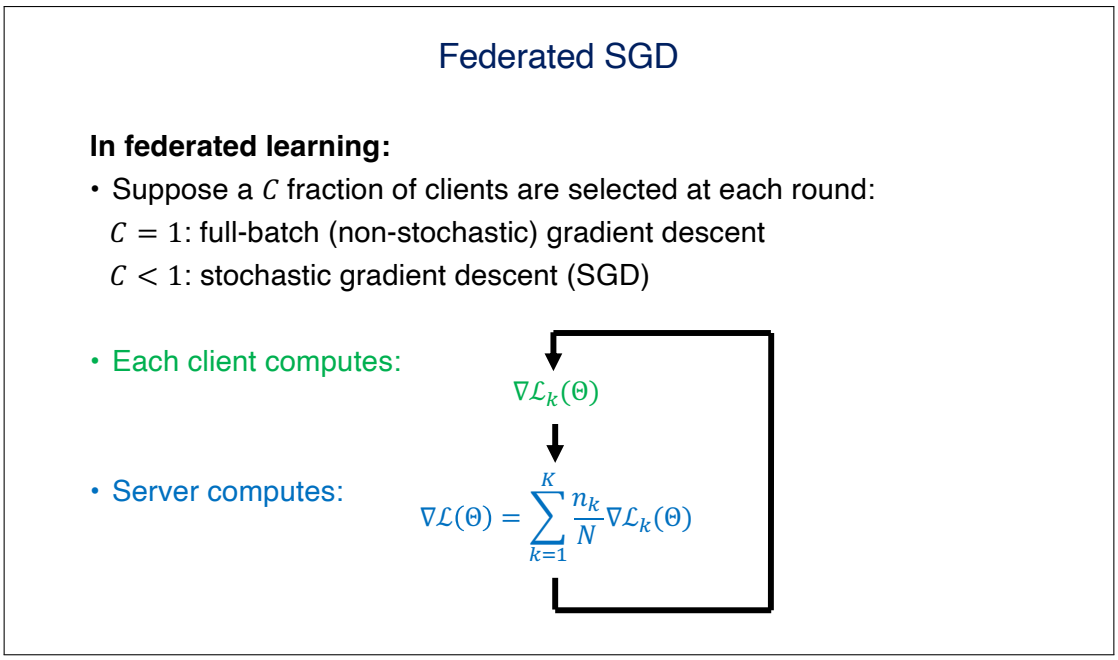



**Figure 10:** caption

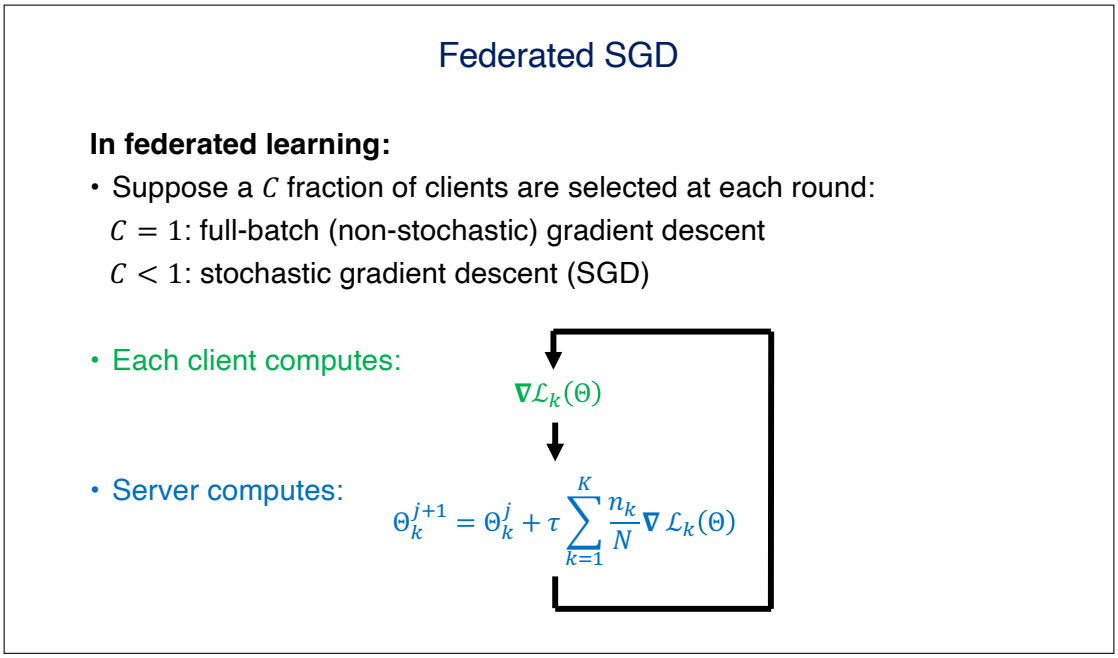



**Figure 11:** caption

### 1.5.2 Federated Averaging

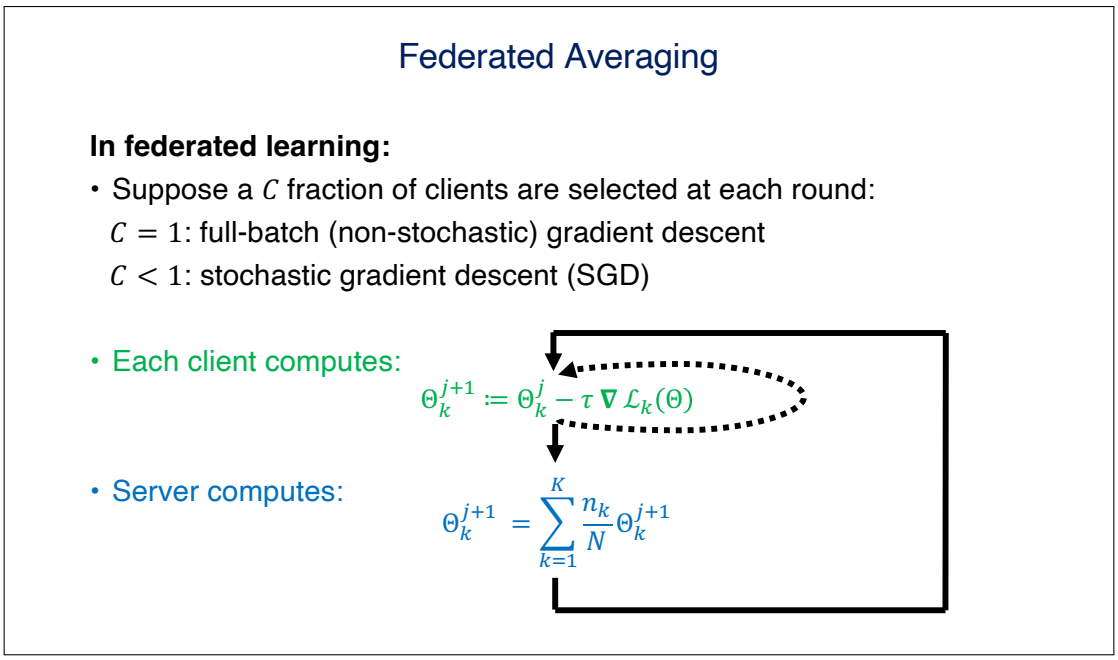



**Figure 12:** caption

## Federated Averaging

**Algorithm 1** `FederatedAveraging`. The $K$ clients are indexed by $k$; $B$ is the local minibatch size, $E$ is the number of local epochs, and $\eta$ is the learning rate.

**Server executes:**
initialize $w_0$
**for** each round $t = 1, 2, \dots$ **do**
  $m \leftarrow \max(C \cdot K, 1)$
  $S_t \leftarrow$ (random set of $m$ clients)
  **for** each client $k \in S_t$ **in parallel do**
    $w_{t+1}^k \leftarrow \text{ClientUpdate}(k, w_t)$
  $w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_{t+1}^k$

**ClientUpdate**$(k, w)$: *// Run on client $k$*
$\mathcal{B} \leftarrow$ (split $\mathcal{P}_k$ into batches of size $B$)
**for** each local epoch $i$ from 1 to $E$ **do**
  **for** batch $b \in \mathcal{B}$ **do**
    $w \leftarrow w - \eta \nabla \ell(w; b)$
return $w$ to server

- First, model is randomly initialized on the central server
- For each round t:
  - *A random set of clients are chosen;*
  - *Each client performs local gradient descent steps;*
  - *The server aggregates model parameters submitted by the clients.*

**Figure 13:** caption

### 1.5.3 Challenges

## Challenges for federated learning

- Non-IID data
  - Training data for a given client is typically site specific, hence the site's local dataset will not be representative of the distribution of training samples.
- Unbalanced data
  - Sites may have a lot or little training data, leading to varying amounts of local training data across different sites.
- Massively distributed data
  - There may be extreme scenarios where each site only has very training samples (in the limiting case one example)
- Communication costs
  - Communication between clients and servers occurs communication overheads. The amount of overhead will depend on the number of clients and the frequency of updates from/to server.

**Figure 14:** caption

## 1.6 Homomorphic Encryption

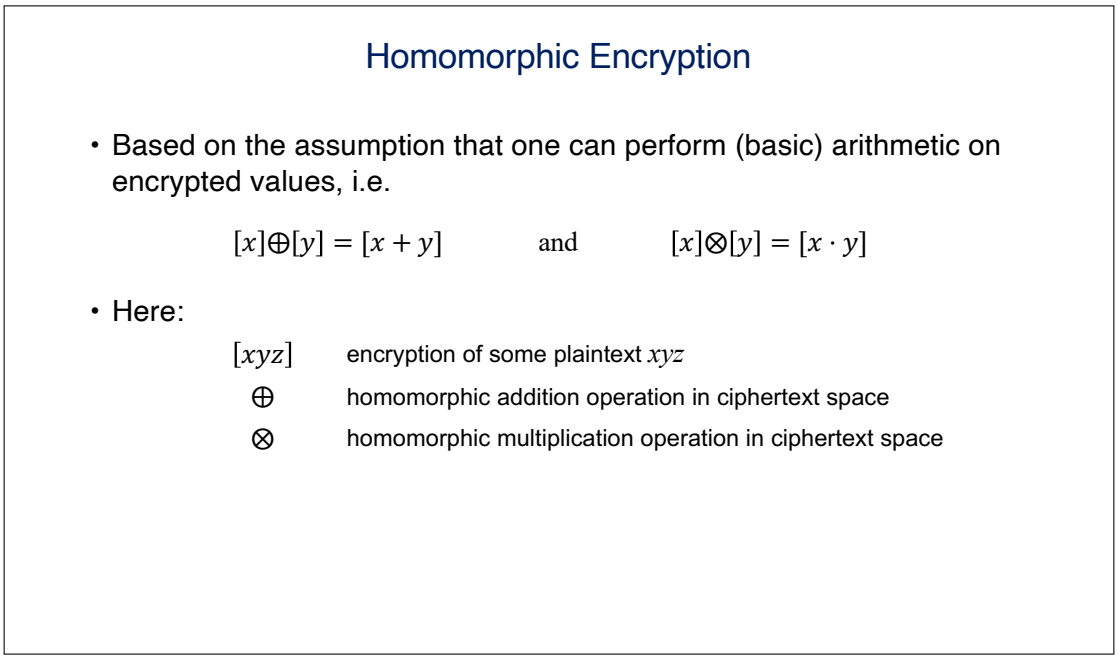



**Figure 15:** caption

### 1.6.1 ML in standard setting

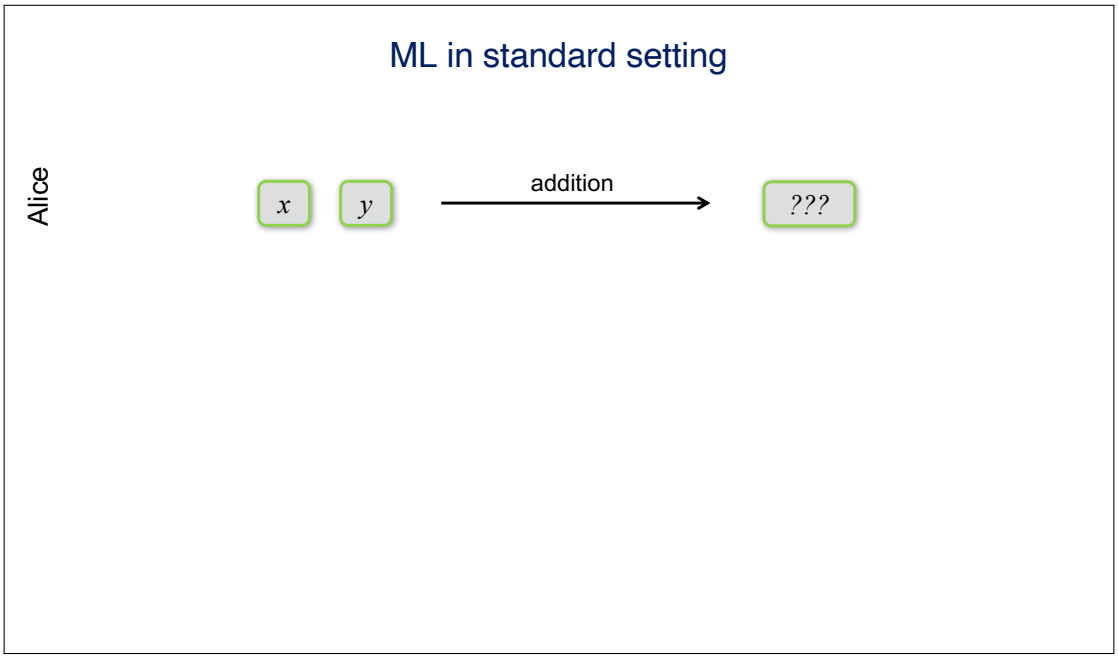



**Figure 16:** caption

### 1.6.2 ML in clinet/server setting

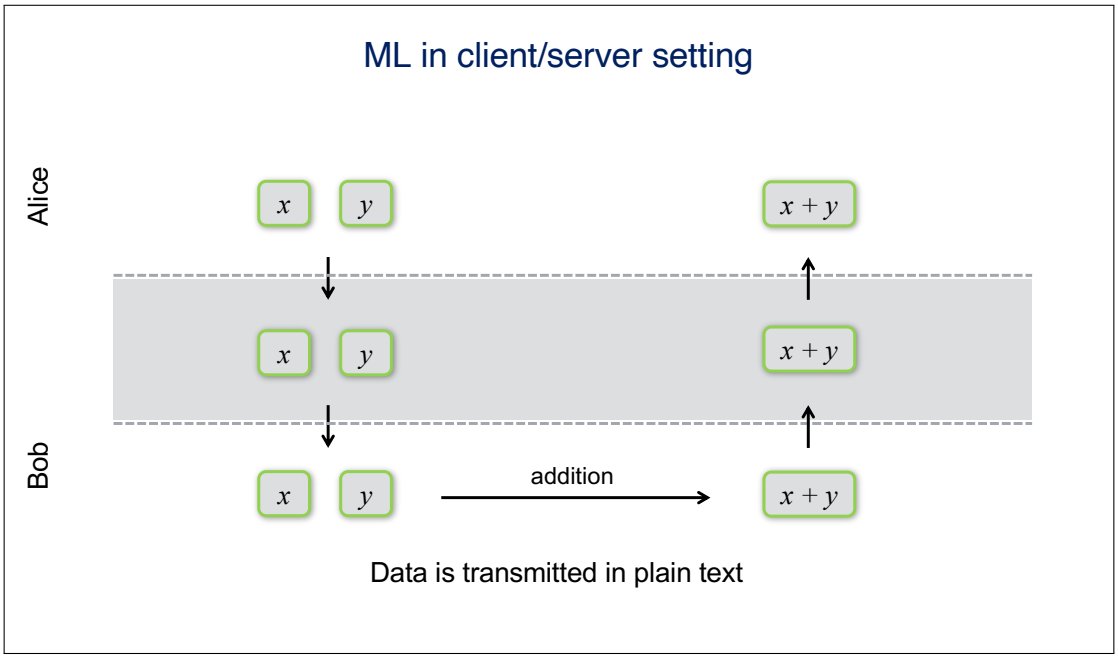



**Figure 17:** caption

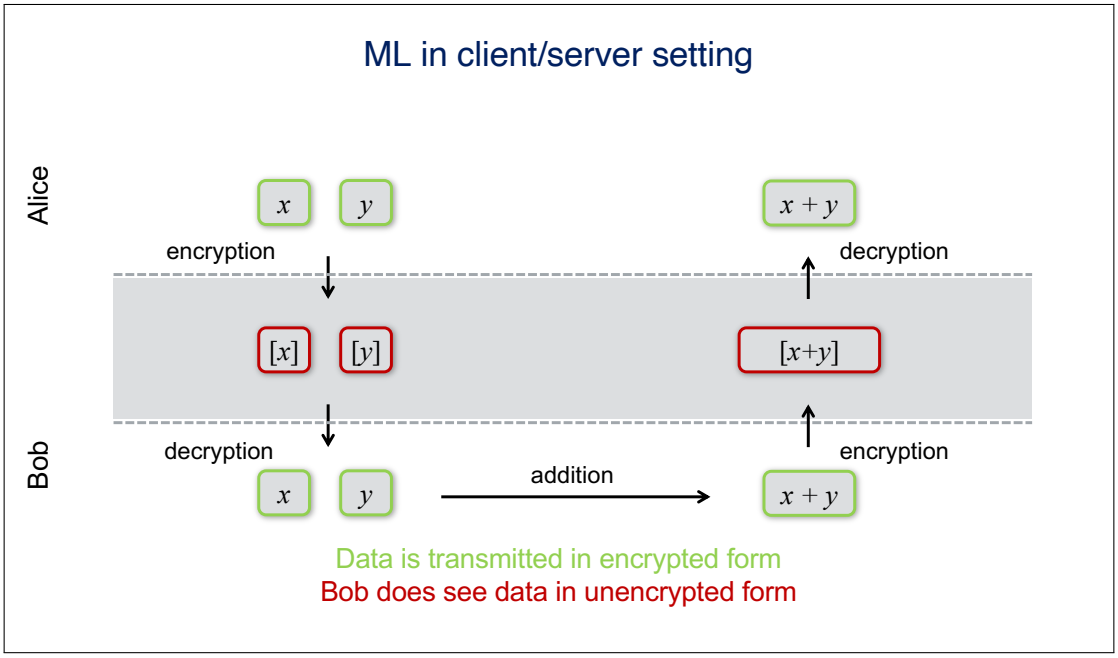



**Figure 18:** caption

## 1.7 Homomorphic Encrypion

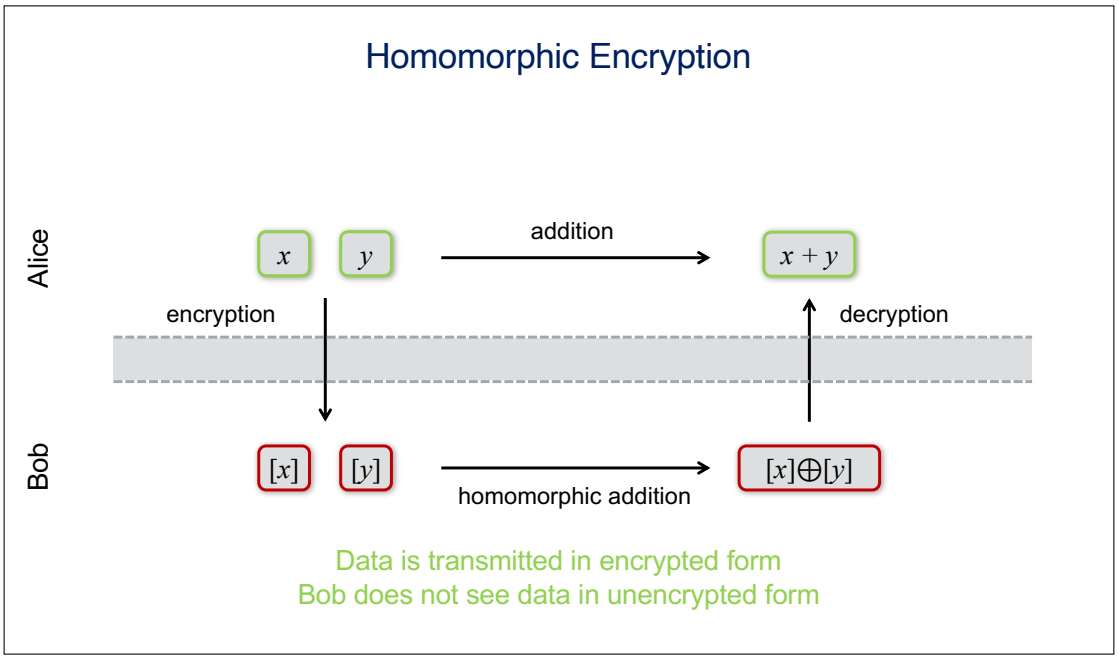



**Figure 19:** caption

## 1.8 ML and Homomorphic Encrypion

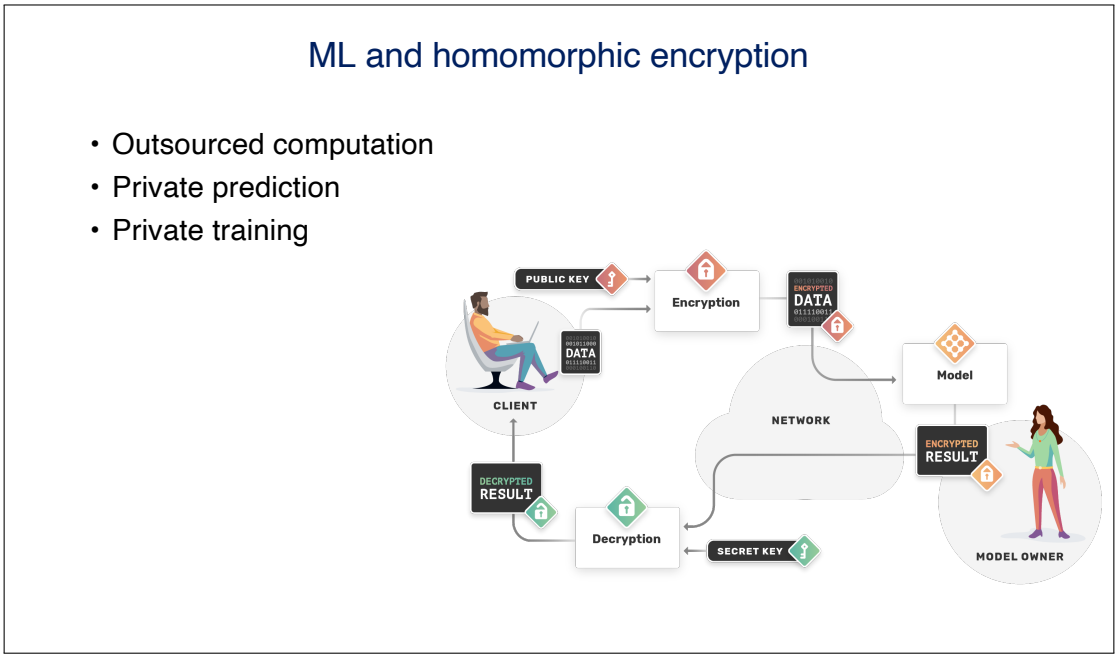



**Figure 20:** caption

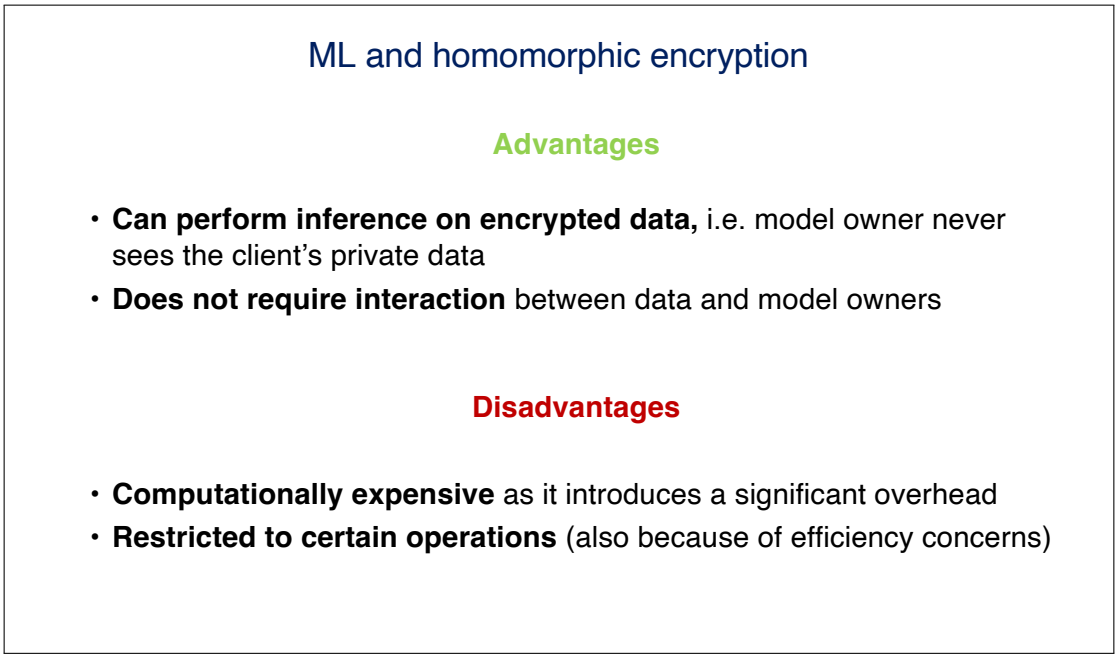



**Figure 21:** caption

## 1.9 Secure Multi-party Computation

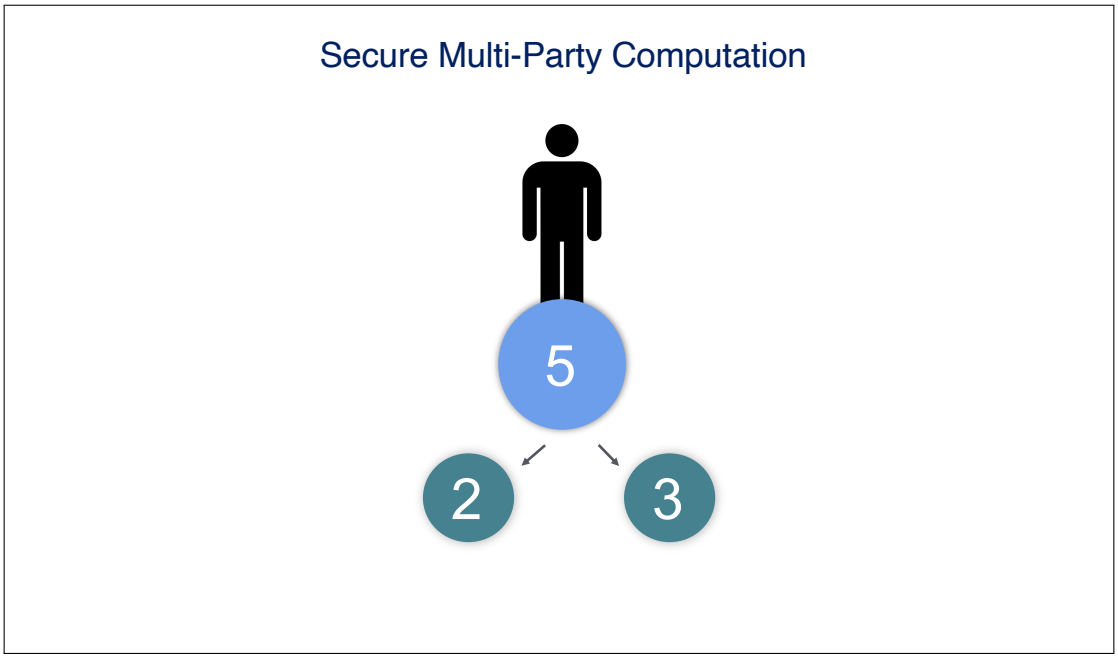



**Figure 22:** caption

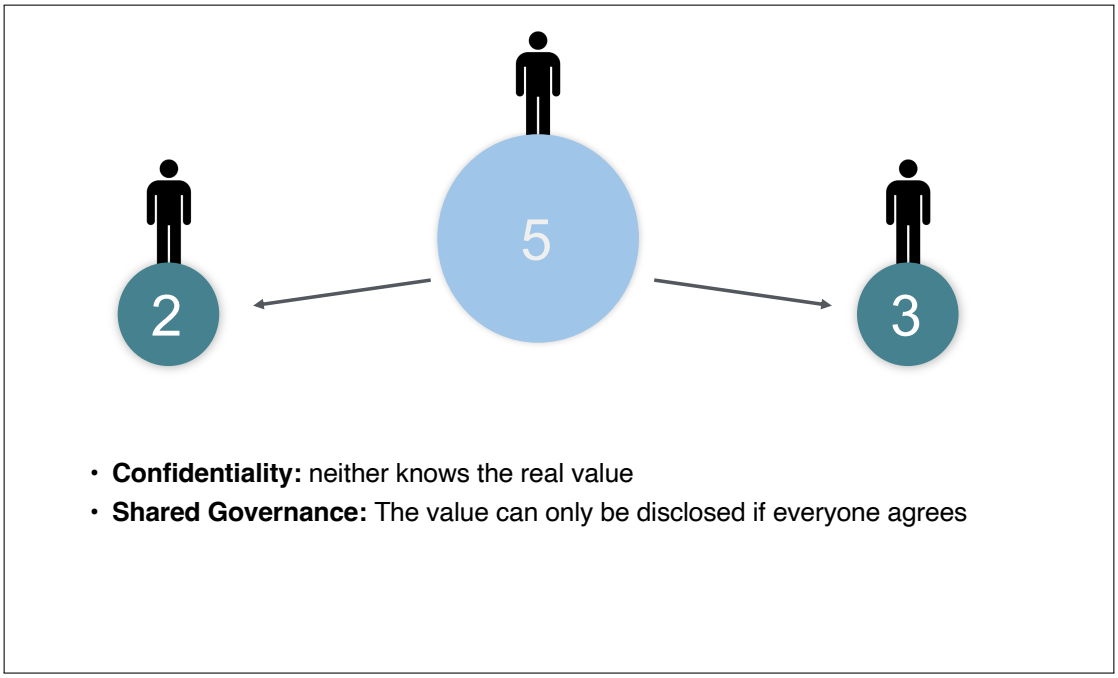



**Figure 23:** caption

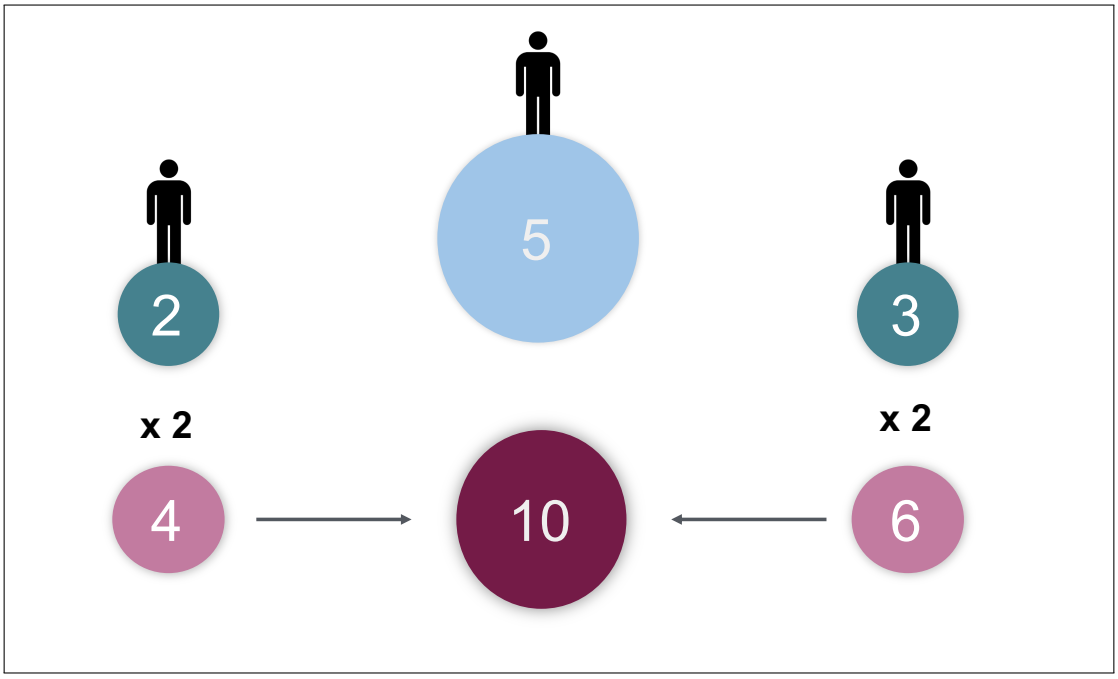



**Figure 24:** caption

## 1.10 Secure Multi-Party Computation

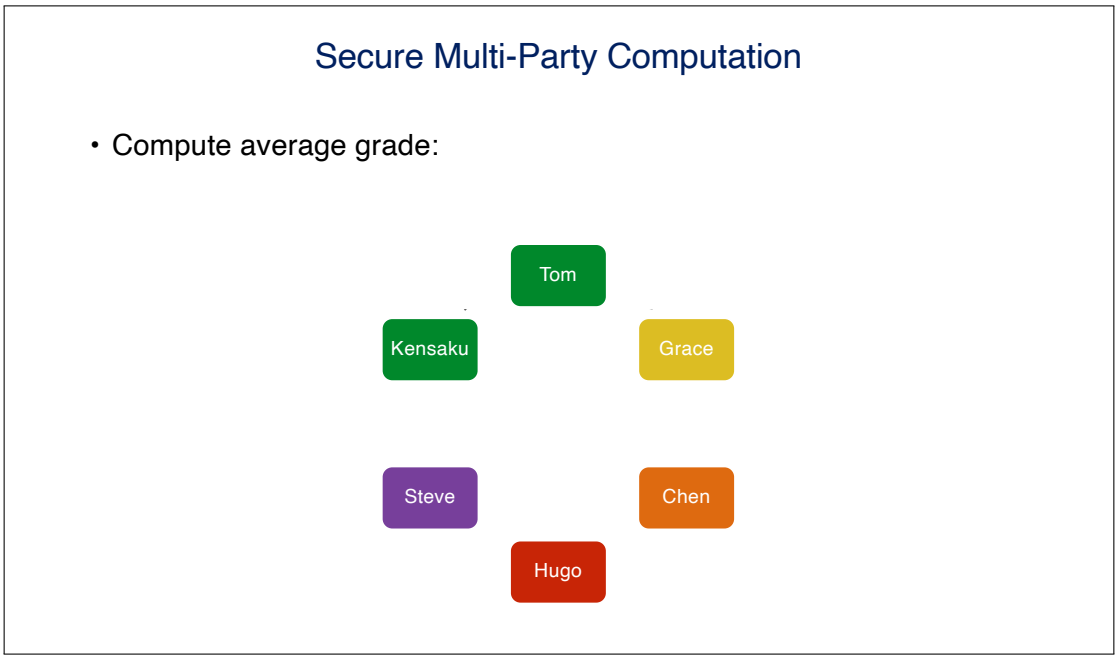



**Figure 25:** caption

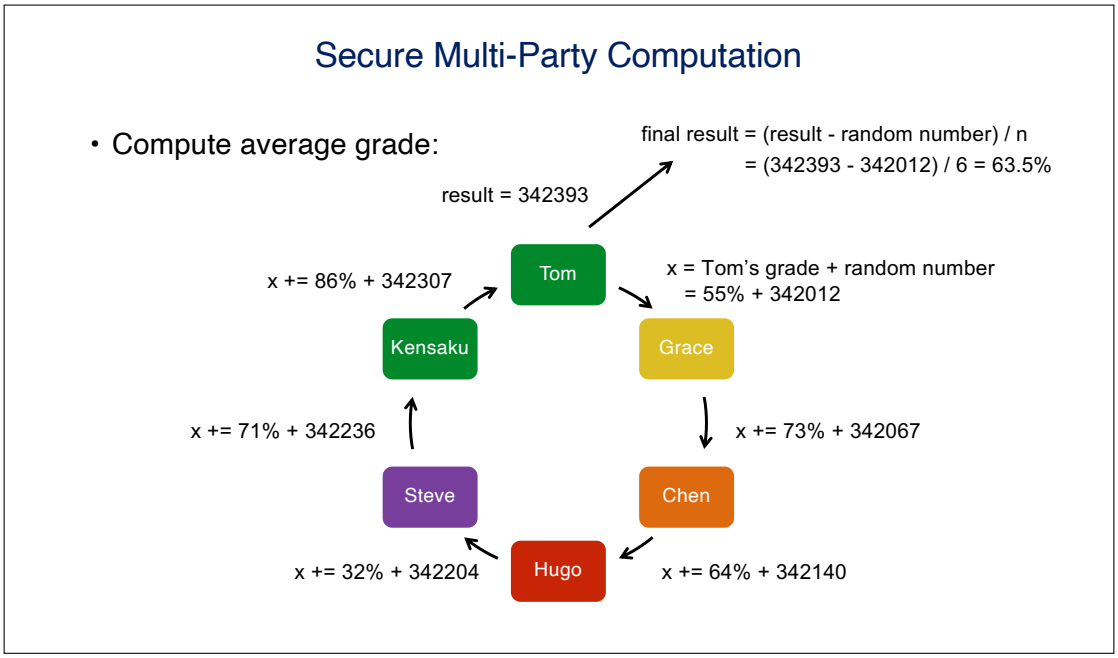



**Figure 26:** caption

## 1.11 Trusted Execution Environments

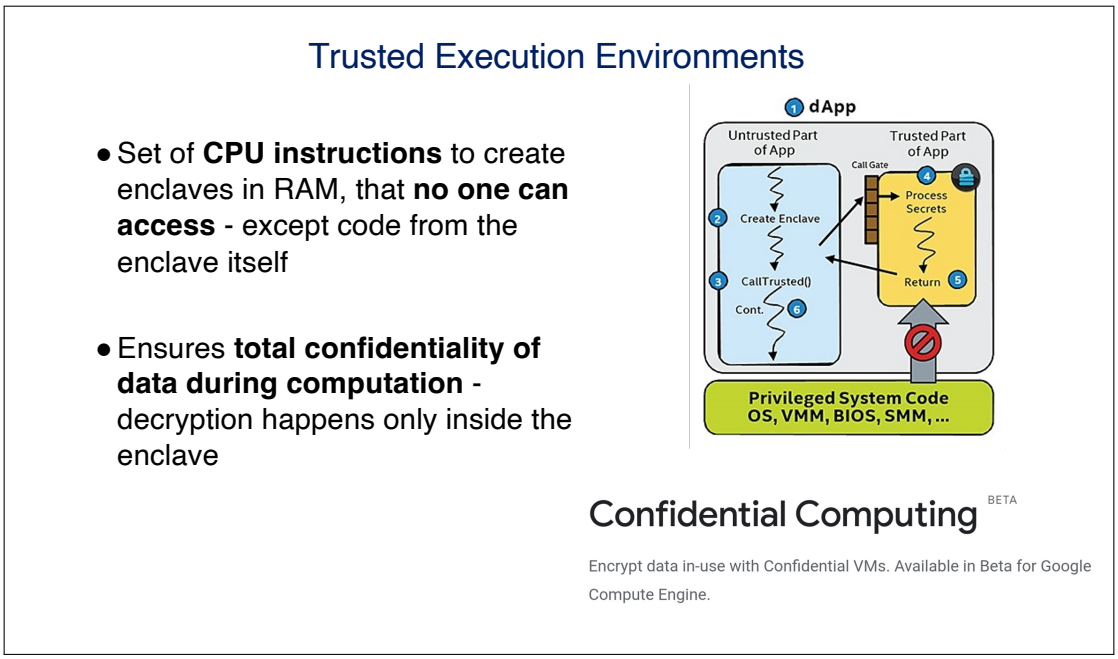



**Figure 27:** caption

## 1.12 What is Privacy?

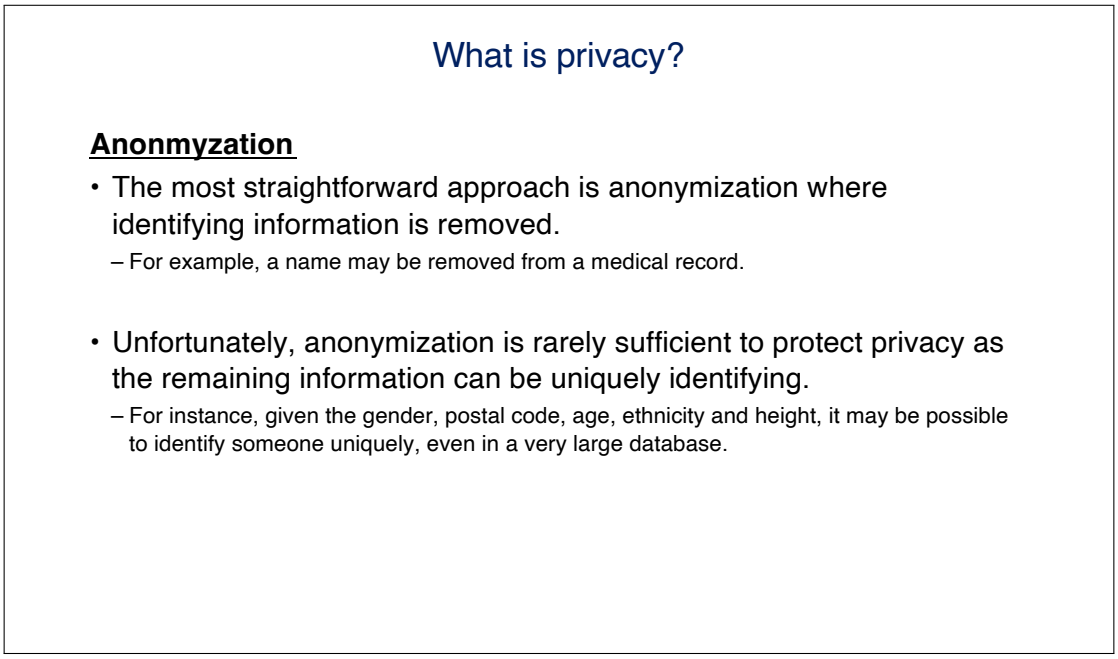



**Figure 28:** caption

## 1.13 k-anonymity

### $k$-anonymity

- One approach to prevent linkage attacks is $k$-anonymity.
  - A dataset is said to be $k$ *anonymous* if, for any person's record in a dataset, there are at least $k$ –1 other records which are indistinguishable.
  - So if a dataset is $k$-anonymous, then the best a linkage attack could ever do is identify a group of $k$ records which could belong to the person of interest.
  - Even if a dataset isn't inherently $k$-anonymous, it could be made so by removing entire fields of data (like names and addresses) and selectively censoring fields of individual people who are particularly unique.

**Figure 29:** caption

### $k$-anonymity: Example

Original Database to Disclose

| | IDENTIFYING VARIABLE | | QUASI-IDENTIFIERS | |
|---|---|---|---|---|
| ID | Name | Gender | Year of Birth | Test Result |
| 1 | John Smith | Male | 1959 | +ve |
| 2 | Alan Smith | Male | 1962 | -ve |
| 3 | Alice Brown | Female | 1955 | -ve |
| 4 | Hercules Green | Male | 1959 | -ve |
| 5 | Alicia Freds | Female | 1942 | -ve |
| 6 | Gill Stringer | Female | 1975 | -ve |
| 7 | Marie Kirkpatrick | Female | 1966 | +ve |
| 8 | Leslie Hall | Female | 1987 | -ve |
| 9 | Bill Nash | Male | 1975 | -ve |
| 10 | Albert Blackwell | Male | 1978 | -ve |
| 11 | Beverly McCulsky | Female | 1964 | -ve |
| 12 | Douglas Henry | Male | 1959 | +ve |
| 13 | Freda Shields | Female | 1975 | -ve |
| 14 | Fred Thompson | Male | 1967 | -ve |

2-Anonymization

| | QUASI-IDENTIFIERS | | |
|---|---|---|---|
| ID | Gender | Decade of Birth | Test Result |
| 1 | Male | 1950-1959 | +ve |
| 2 | Male | 1960-1969 | -ve |
| 4 | Male | 1950-1959 | -ve |
| 6 | Female | 1970-1979 | -ve |
| 7 | Female | 1960-1969 | +ve |
| 9 | Male | 1970-1979 | -ve |
| 10 | Male | 1970-1979 | -ve |
| 11 | Female | 1960-1969 | -ve |
| 12 | Male | 1950-1959 | +ve |
| 13 | Female | 1970-1979 | -ve |
| 14 | Male | 1960-1969 | -ve |

Disclosed (k-Anonymized) Database ($\varepsilon$)

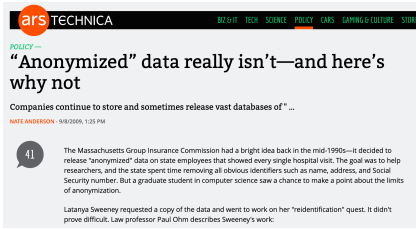
K. El Emam et al. Protecting privacy using k-anonymity. J Am Med Inform Assoc. 200
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC252802

**Figure 30:** caption

## $k$-anonymity

**Anonmyzation and linkage attacks**

- Famous example:
  - After an insurance group released health records which had been stripped of personal information like patient name and address, a CS student was able to "deanonymize" which records belonged to politicians (including the Governor of Massachusetts) by cross referencing with public voter registers.
  - This is an example of a **linkage attack**, where connections to other sources of information work to deanonymize a dataset.
  - Linkage attacks have been successful on a range of anonymized datasets including the Netflix challenge and genome data.




https://arstechnica.com/tech-policy/2009/09/your-secrets-live-online-in-databases-of-ruin/

**Figure 31:** caption

## $k$-anonymity

- Unfortunately, $k$-anonymity isn't sufficient for anything but very large datasets with only small numbers of simple fields for each record.
- Intuitively, the more fields and the more possible entries there are in those fields, the more unique a record can be and the harder it is to ensure that there are $k$ equivalent records.

**Figure 32:** caption

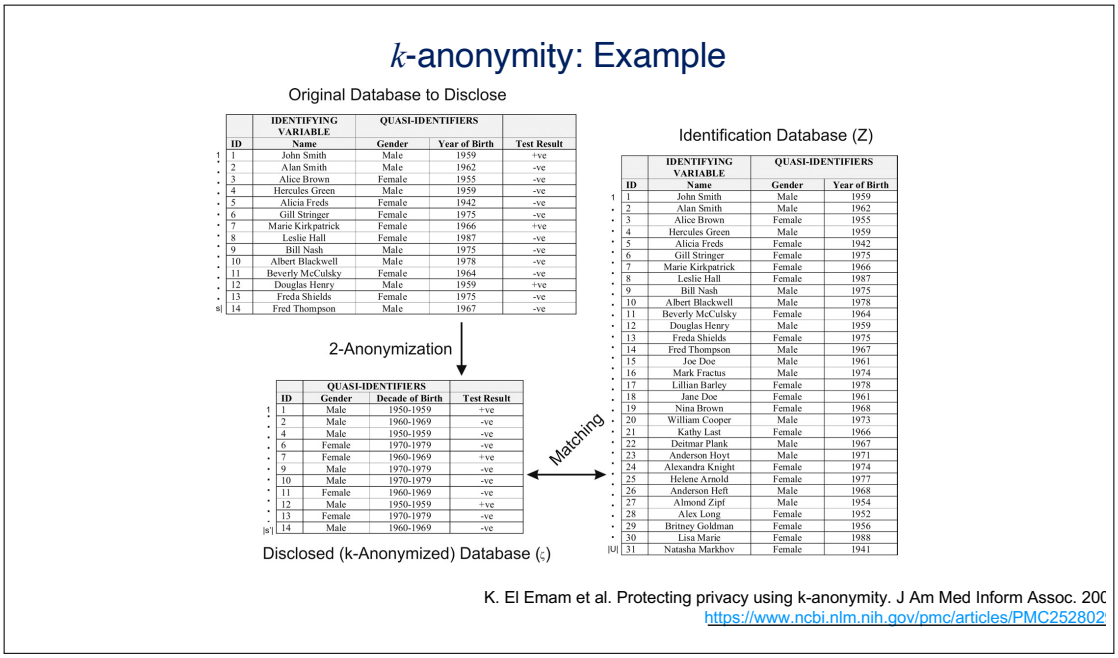



**Figure 33:** caption

## 1.14 Privacy attacks

### 1.14.1 Model inversion

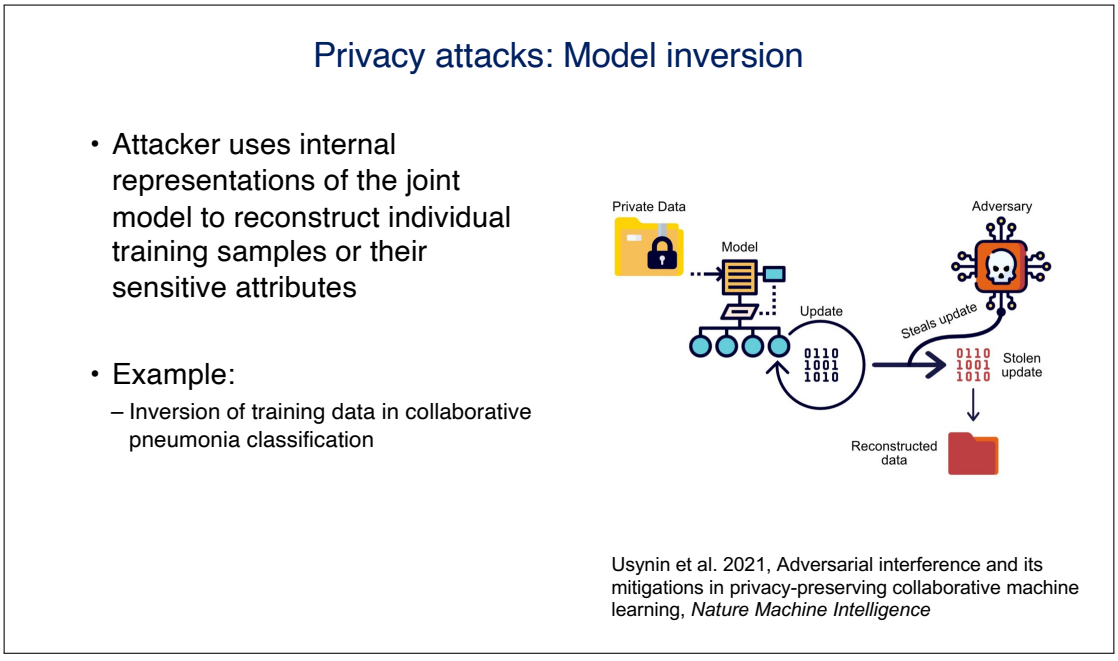



**Figure 34:** caption

### 1.14.2 Membership inference

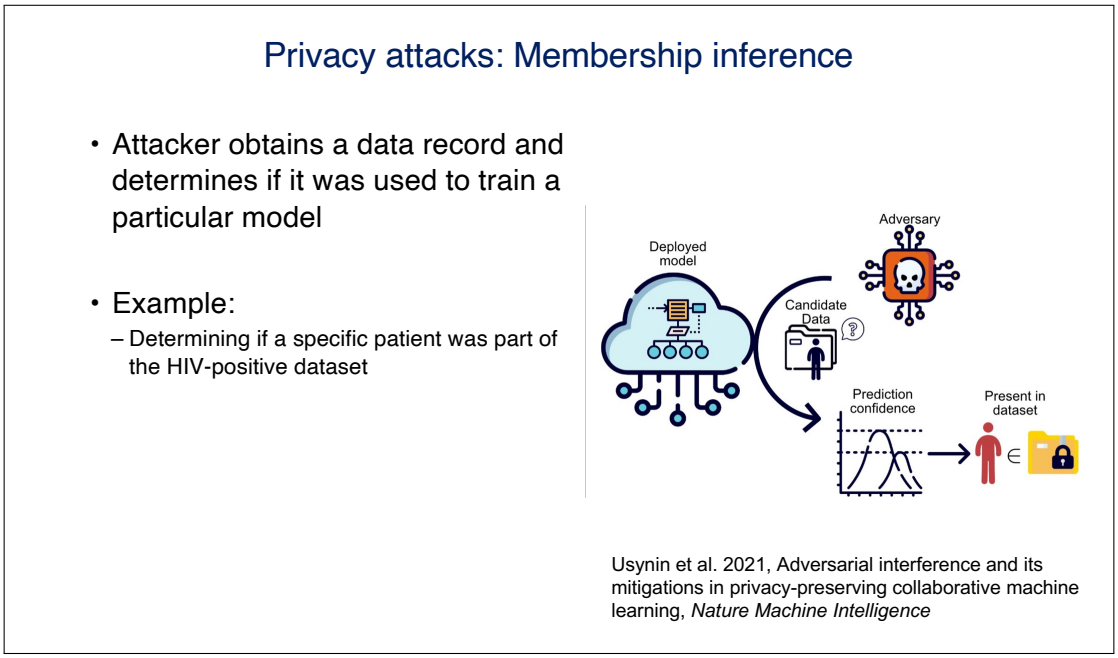



**Figure 35:** caption

### 1.14.3 Attribute inference

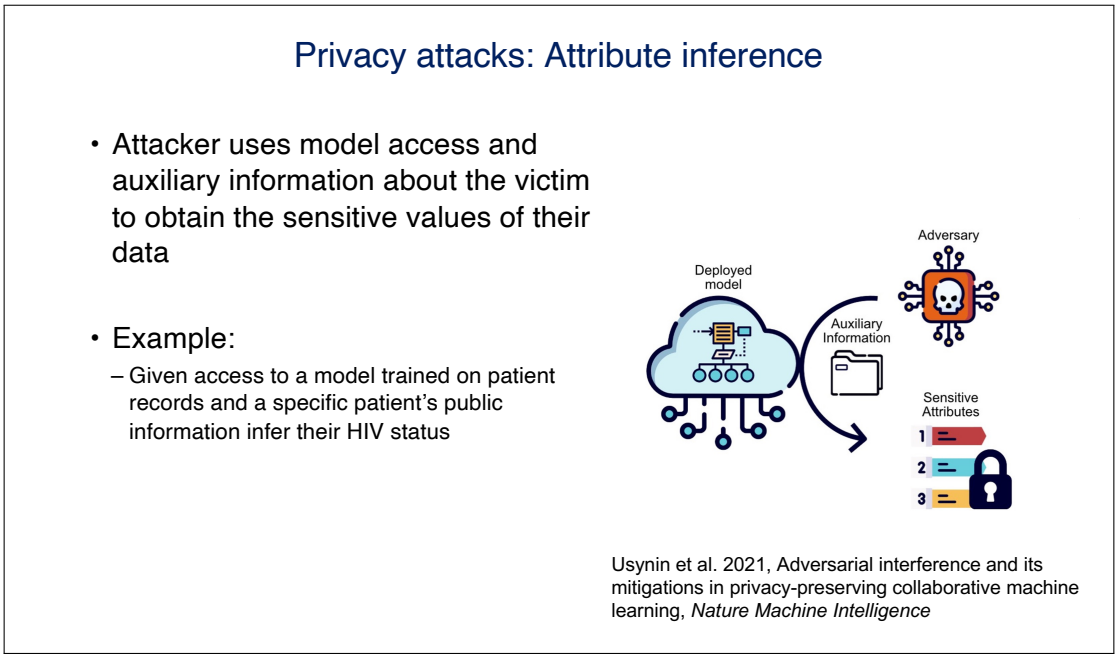



**Figure 36:** caption

## 1.15 Differential Privacy

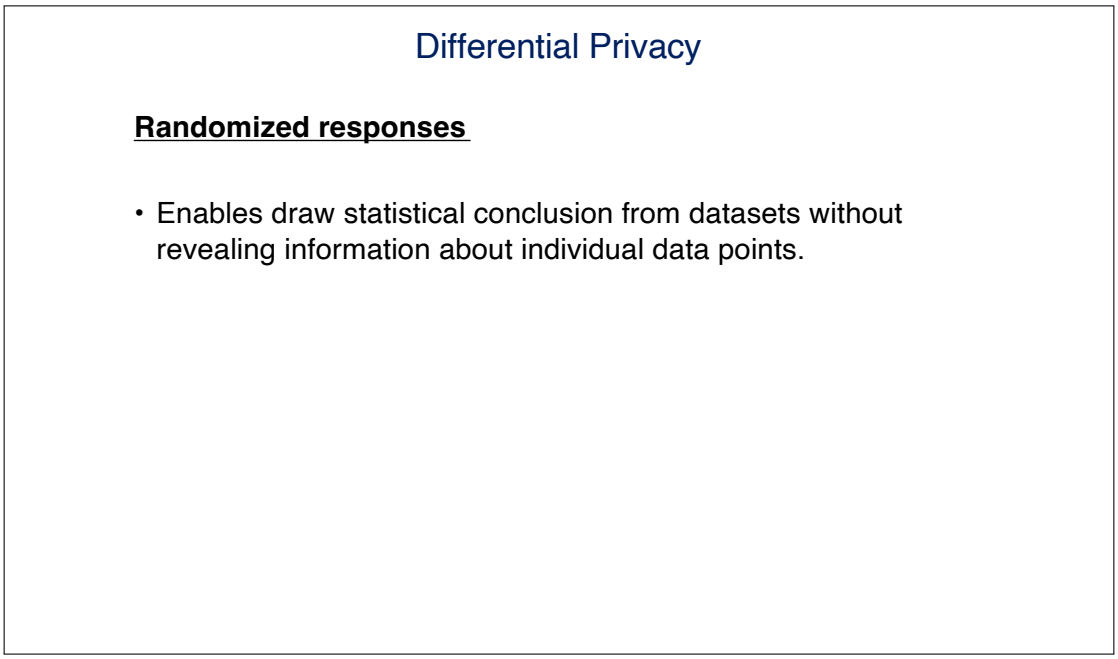



**Figure 37:** caption

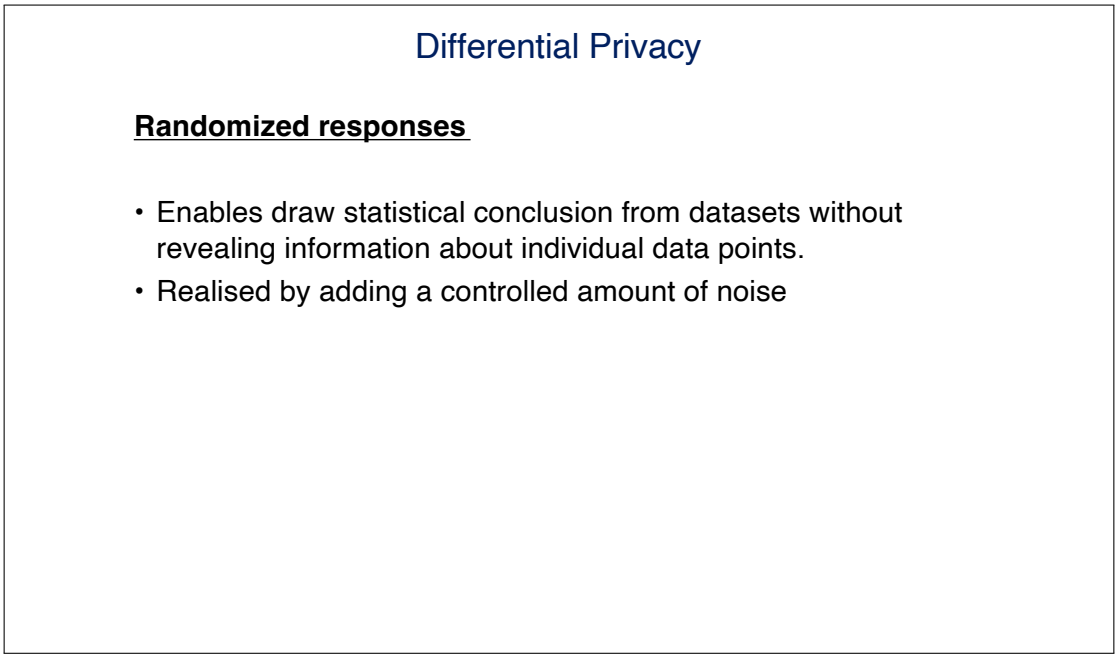



**Figure 38:** caption

### 1.15.1 Randomized responses


**Figure 39:** caption

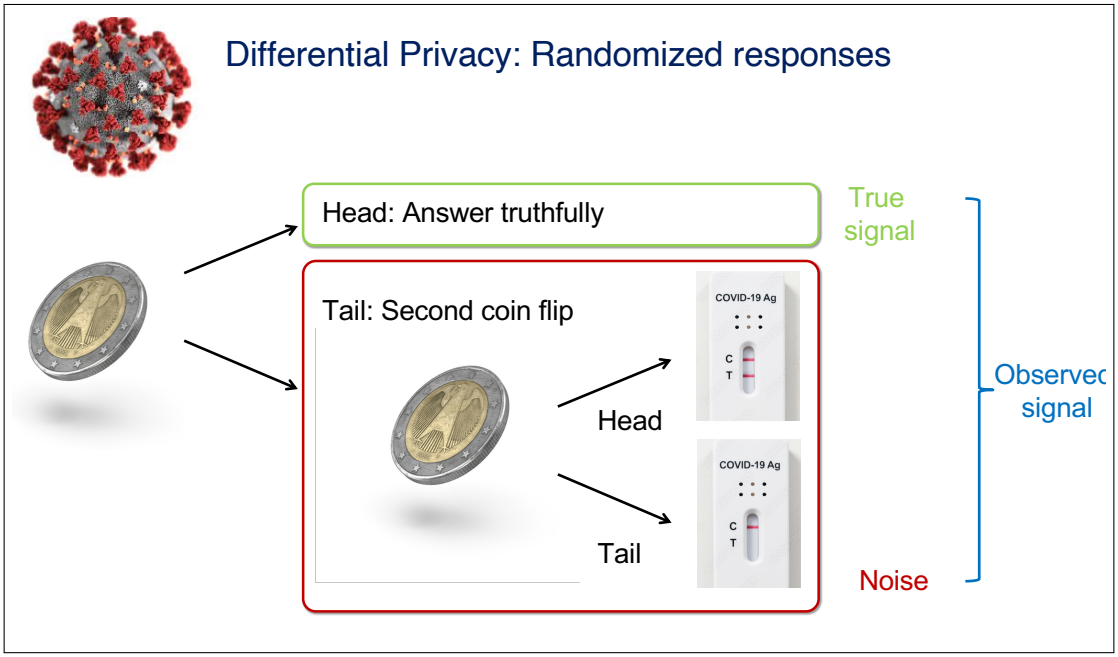



**Figure 40:** caption

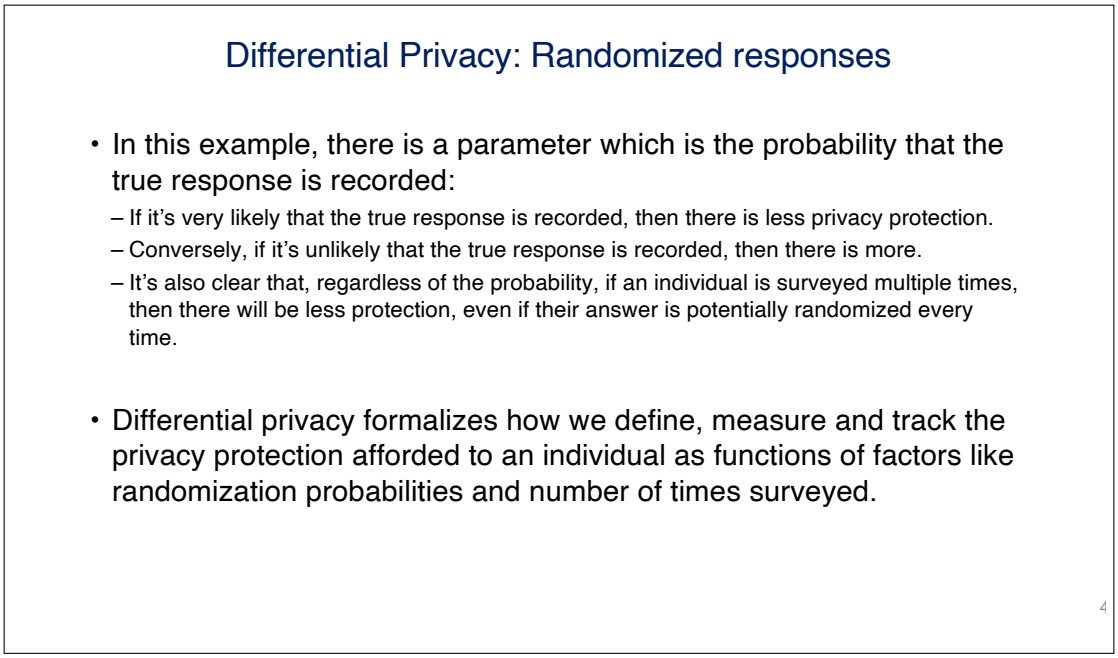



**Figure 41:** caption

## 1.16 Differential prviacy

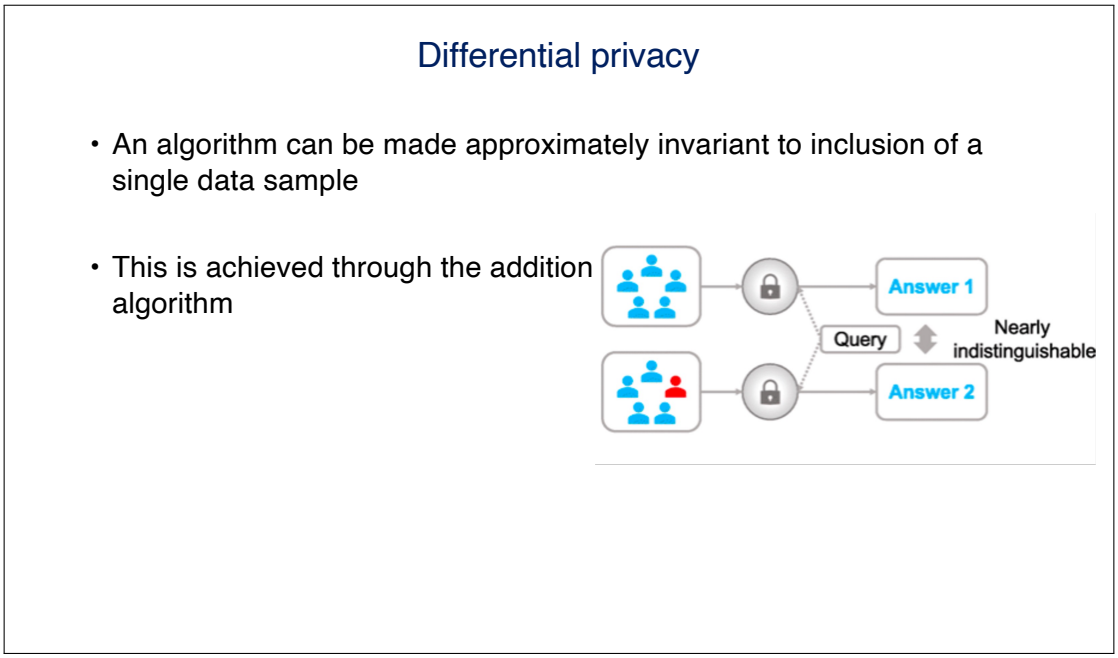



**Figure 42:** caption

## Differential privacy (continued)

- This can be mathematically defined using the following (relaxed) expression:

A process A is ε-differentially private if for all databases $D_1$ and $D_2$ which differ in only one individual:

$$\mathbb{P}[A(D_1) = O] \leq e^{\varepsilon} \cdot \mathbb{P}[A(D_2) = O]$$

**Figure 43:** caption

## Differential privacy (continued)

- This is done through the addition of noise to the output of the query so then the results in two datasets look approximately like this:

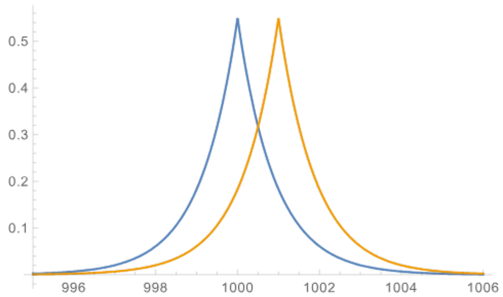



**Figure 44:** caption

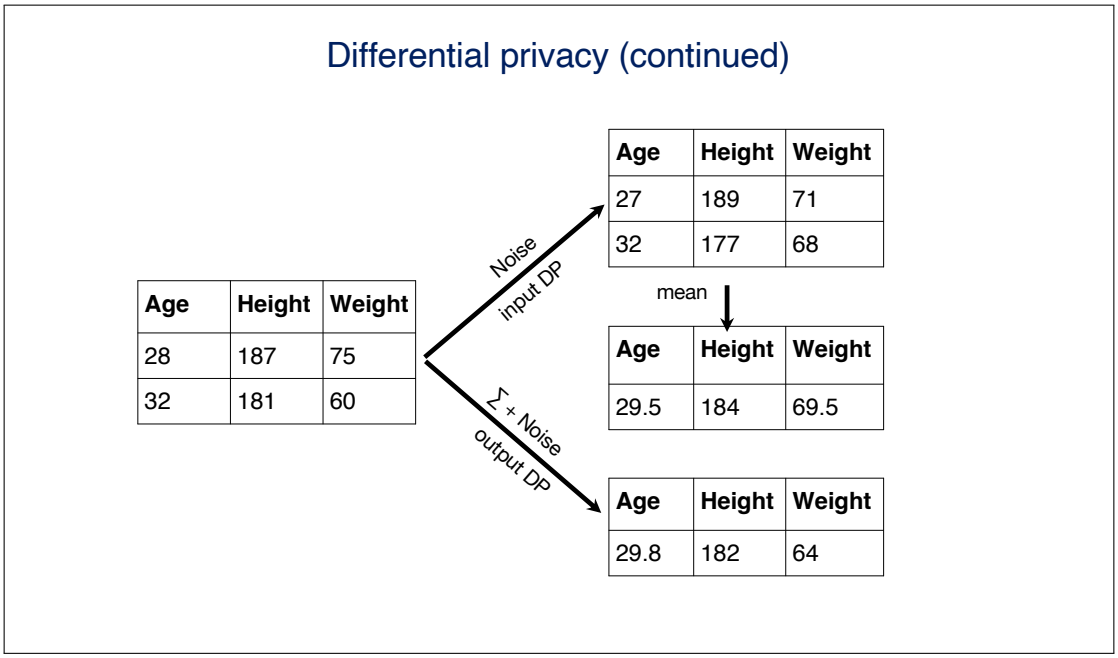



**Figure 45:** caption

## 1.17 Concrete mitigation

### 1.17.1 Differentially private stochastic gradient descent (DP-SGD)

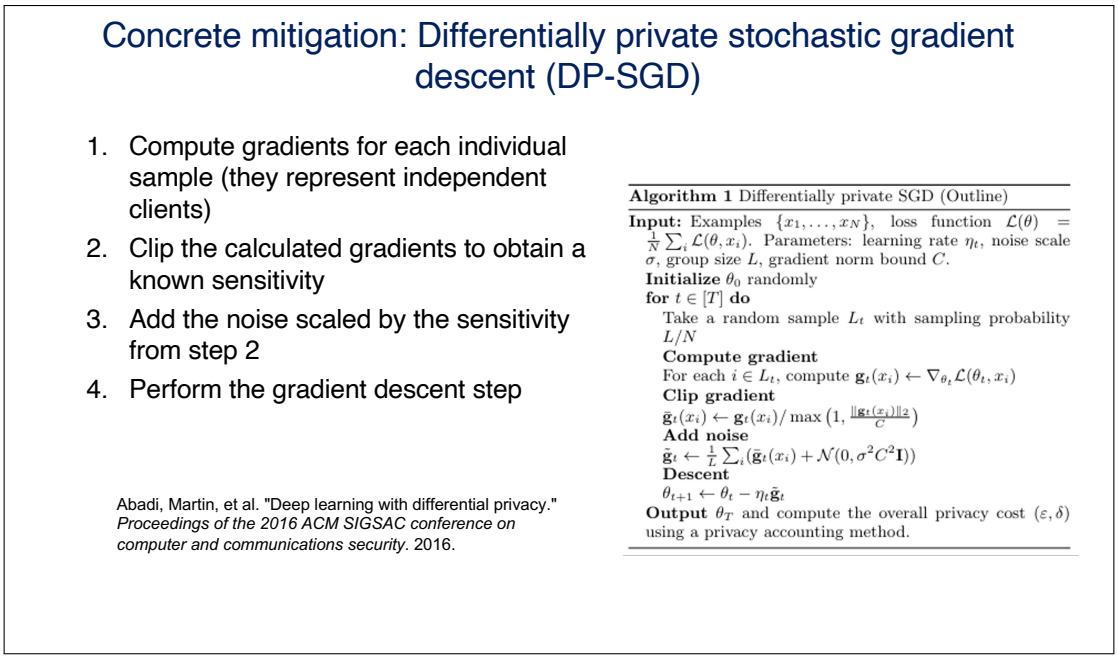



**Figure 46:** caption

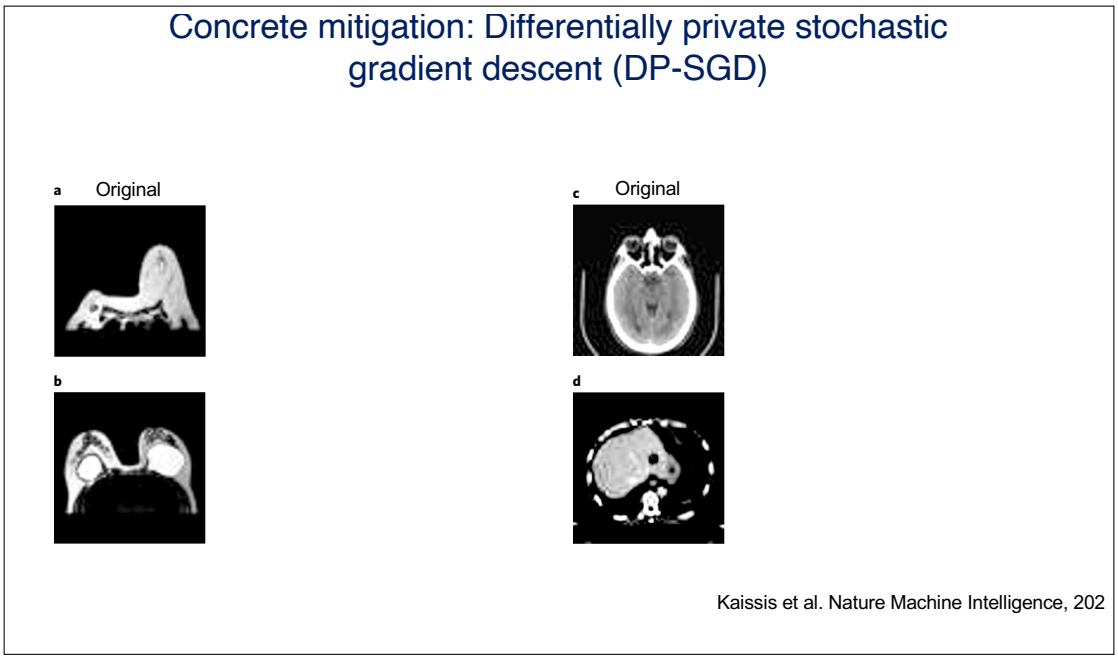



**Figure 47:** caption

# 2 Interpretability and Explainability

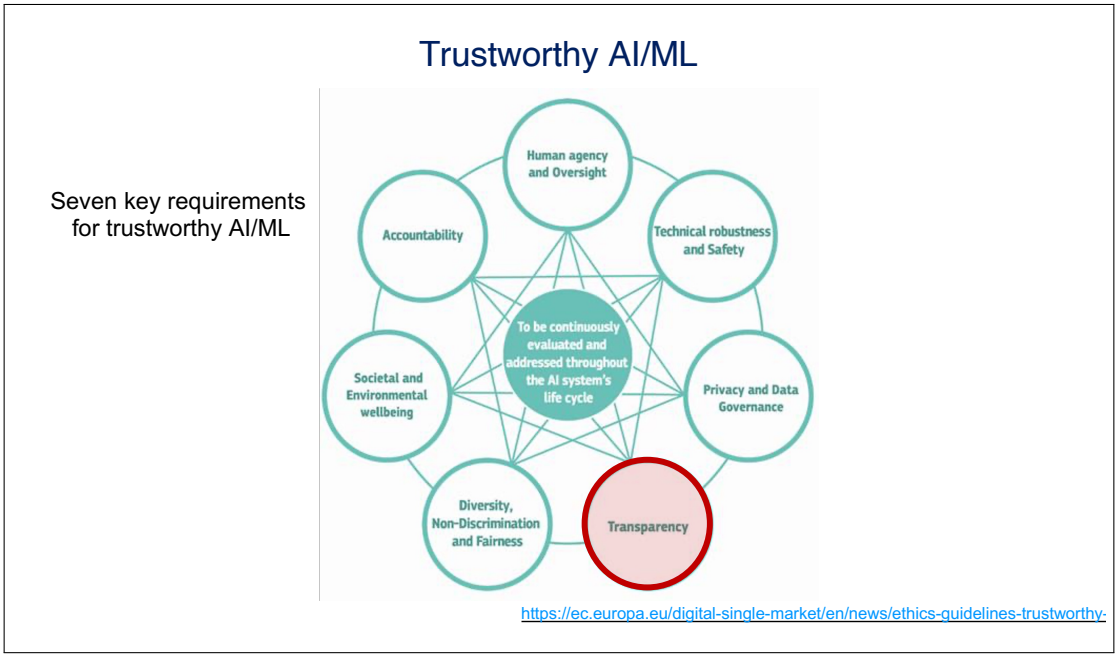



**Figure 48:** caption

## 2.1 Why is it important?

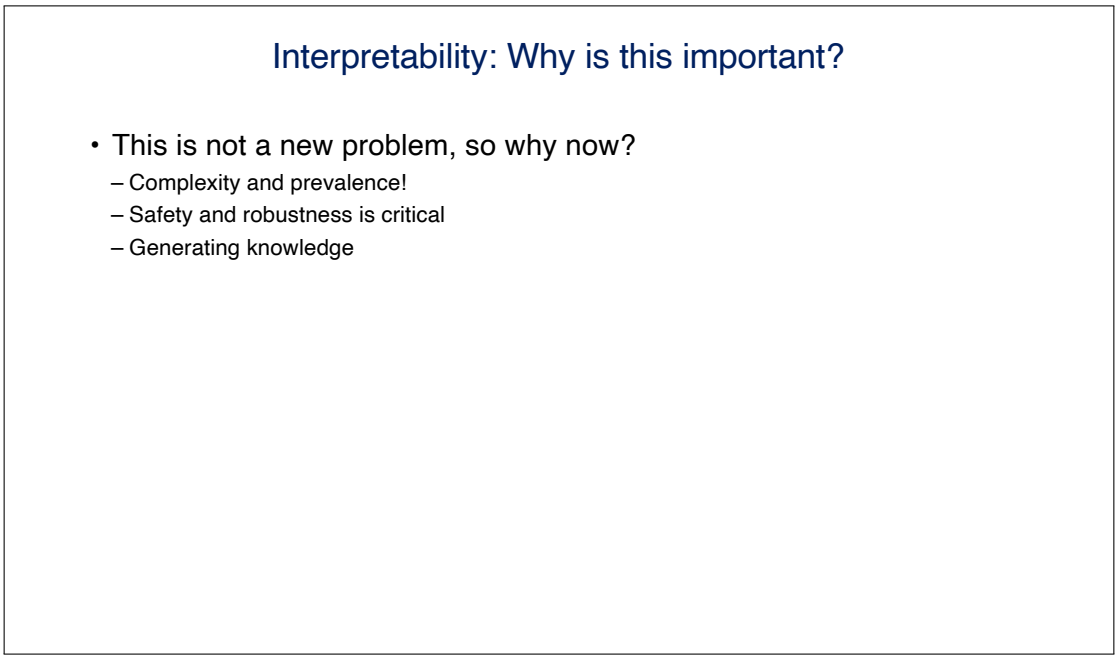



**Figure 49:** caption

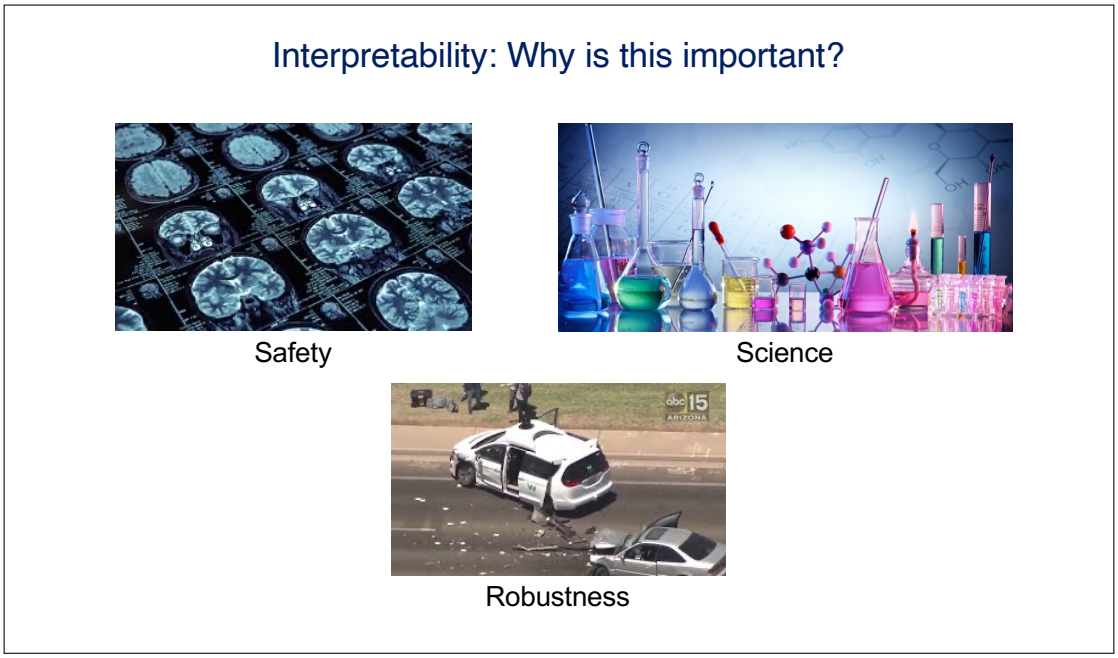



**Figure 50:** caption

## Interpretability: Why is this important?

- This is not a new problem, so why now?
  - Complexity and prevalence!
  - Safety is critical
  - Generating knowledge
- Debugging machine learning
  - Why does my model not train?
  - Why does my model exhibit unexpected/unintuitive behaviour?

**Figure 51:** caption

## Interpretability: Why is this important?

- Debugging machine learning models

Data during training

Data during deployment

**Figure 52:** caption

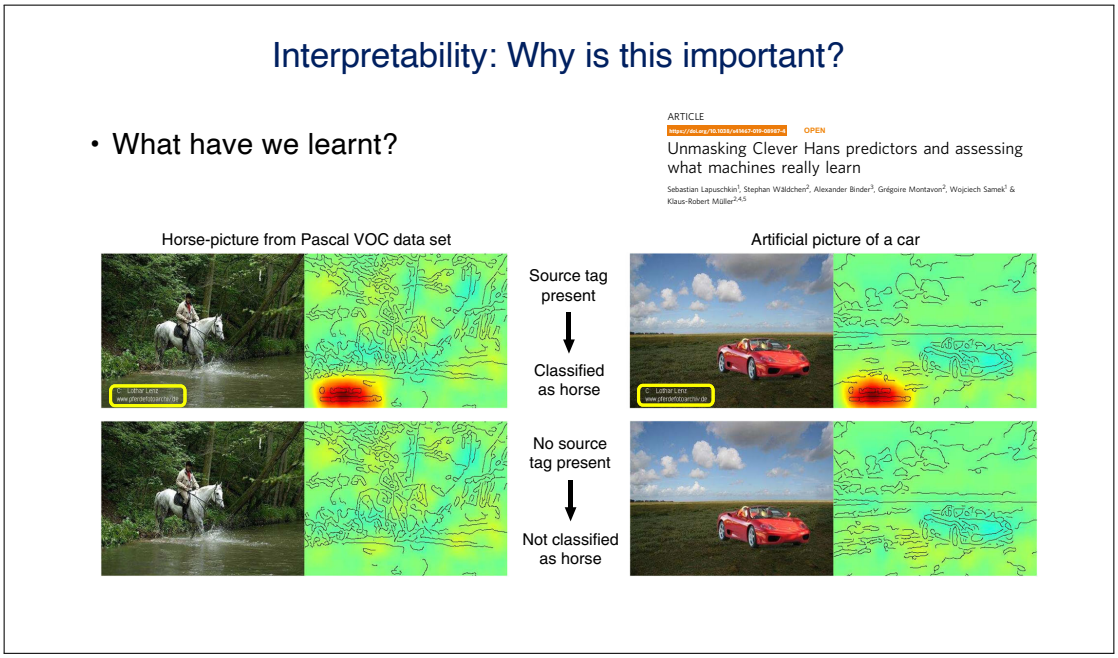



**Figure 53:** caption

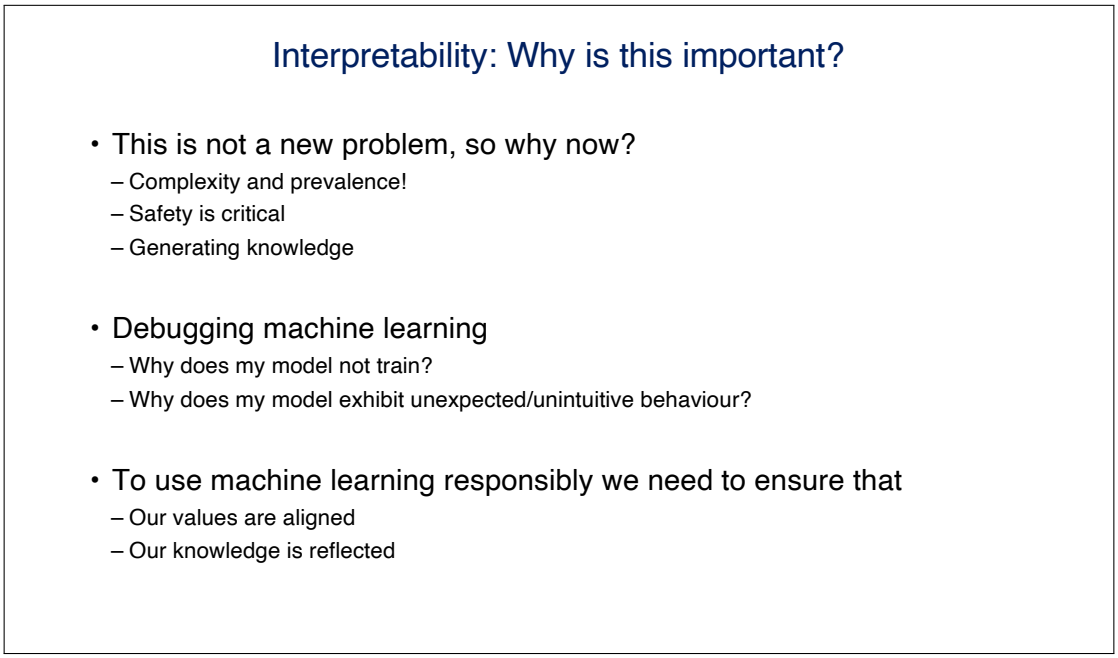



**Figure 54:** caption

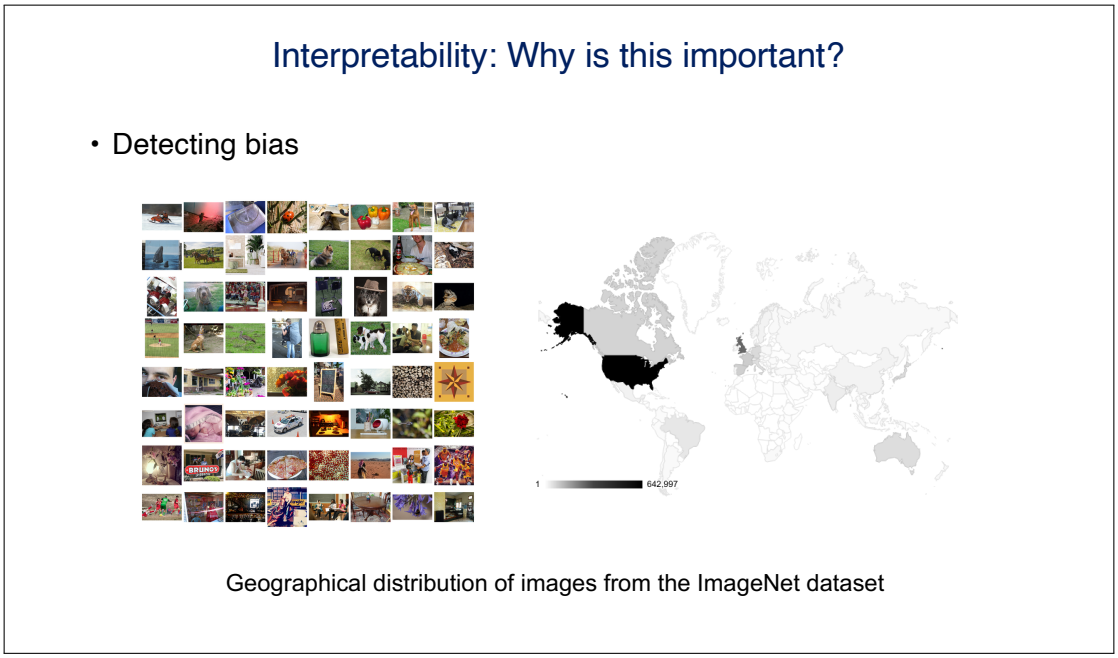



**Figure 55:** caption

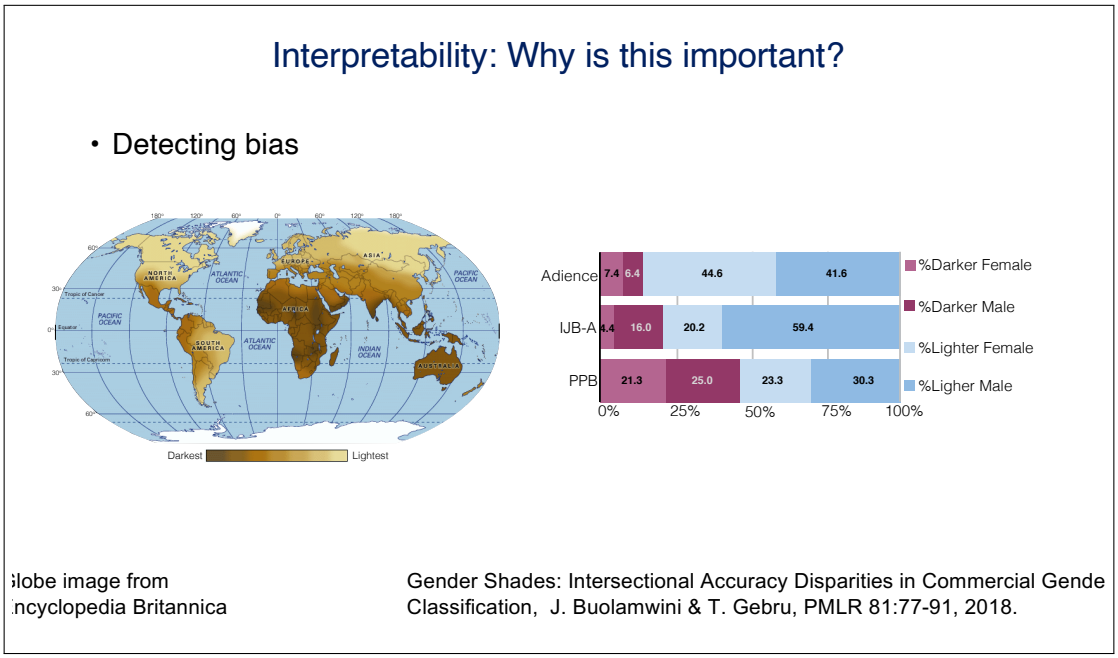



**Figure 56:** caption

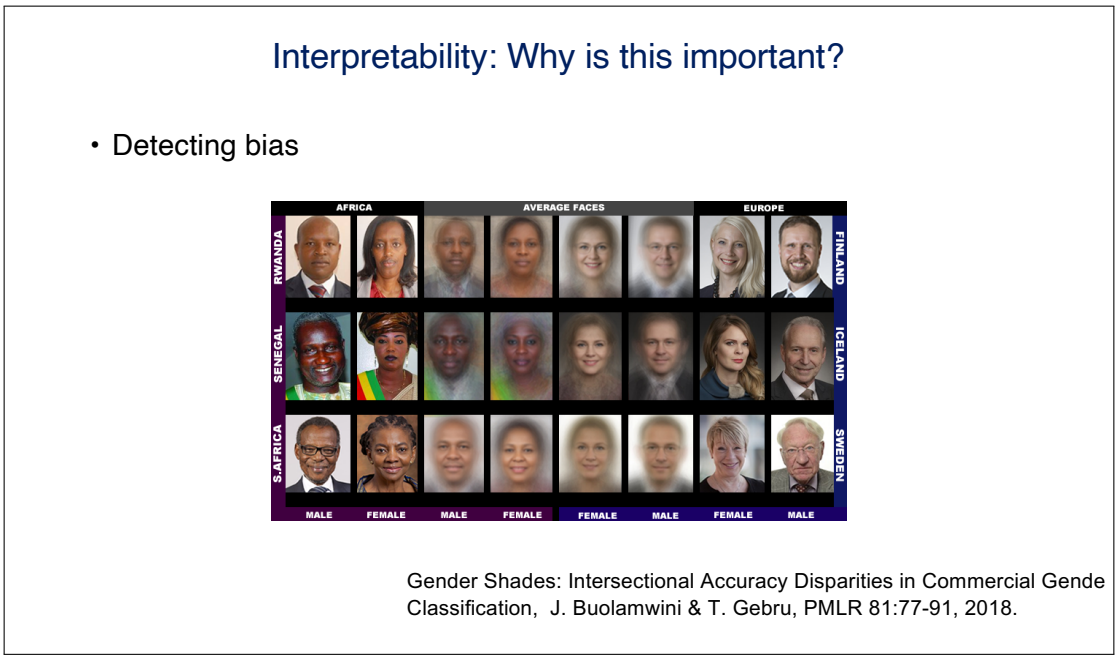



**Figure 57:** caption

## 2.2 Common misunderstandings

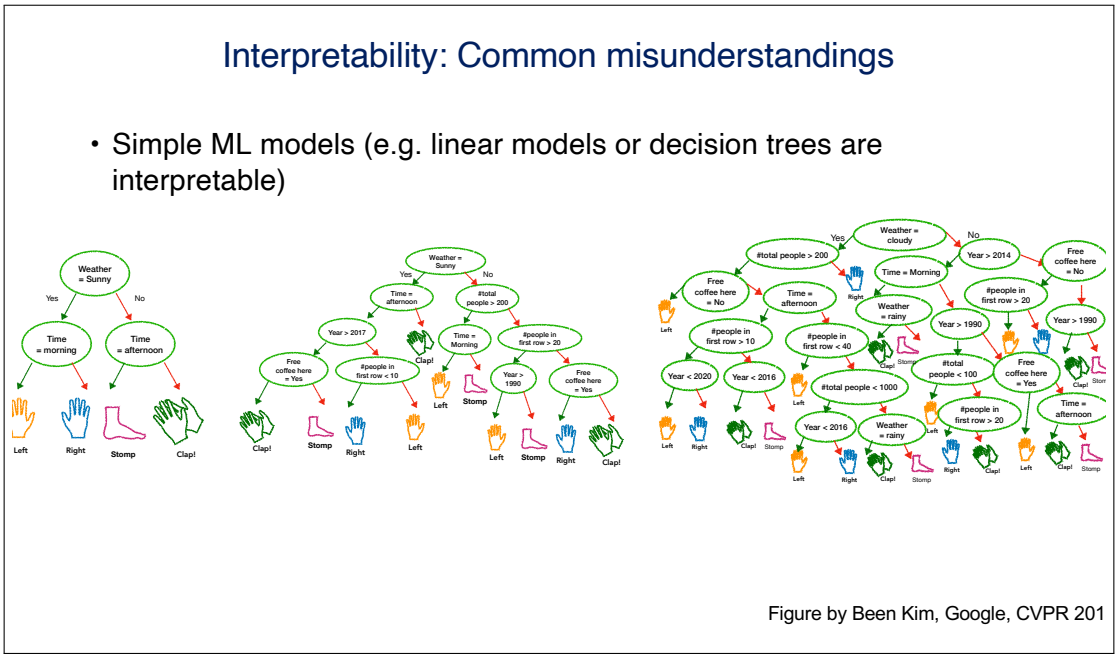



**Figure 58:** caption

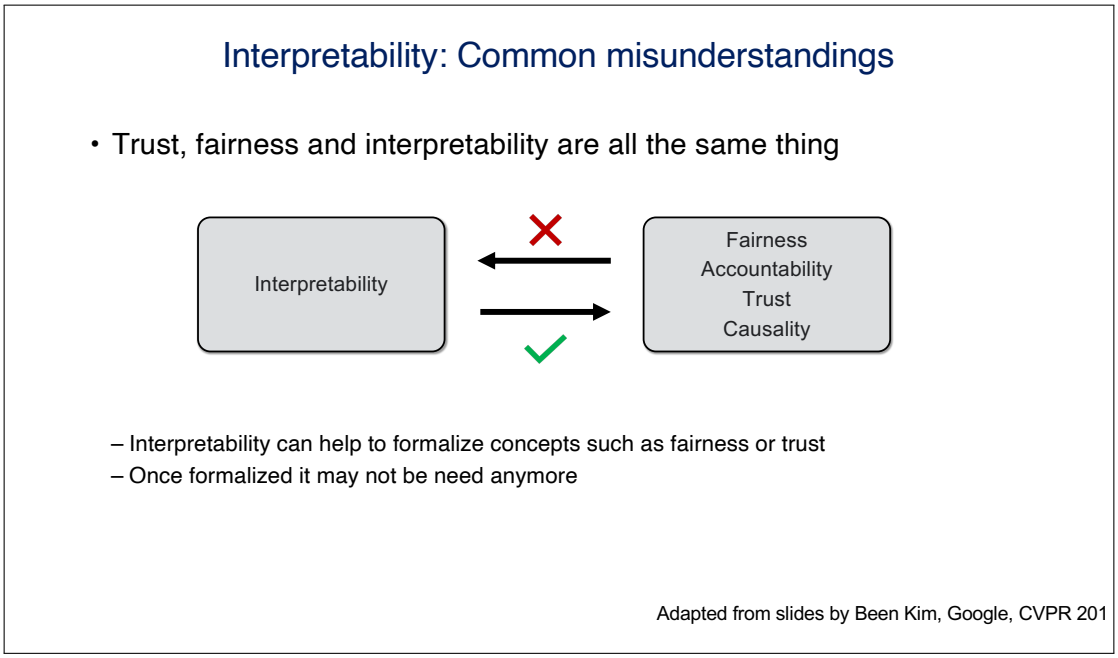



**Figure 59:** caption


**Figure 60:** caption

## 2.3 Types of methods

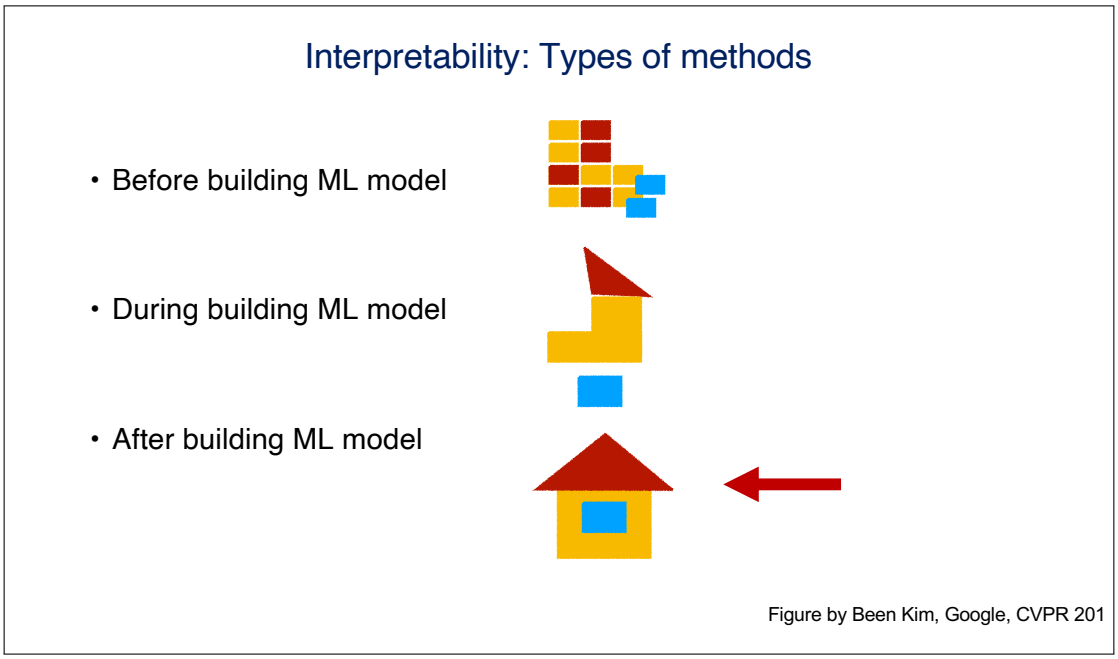



**Figure 61:** caption

## 2.4 How can we interpret an existing ML model?

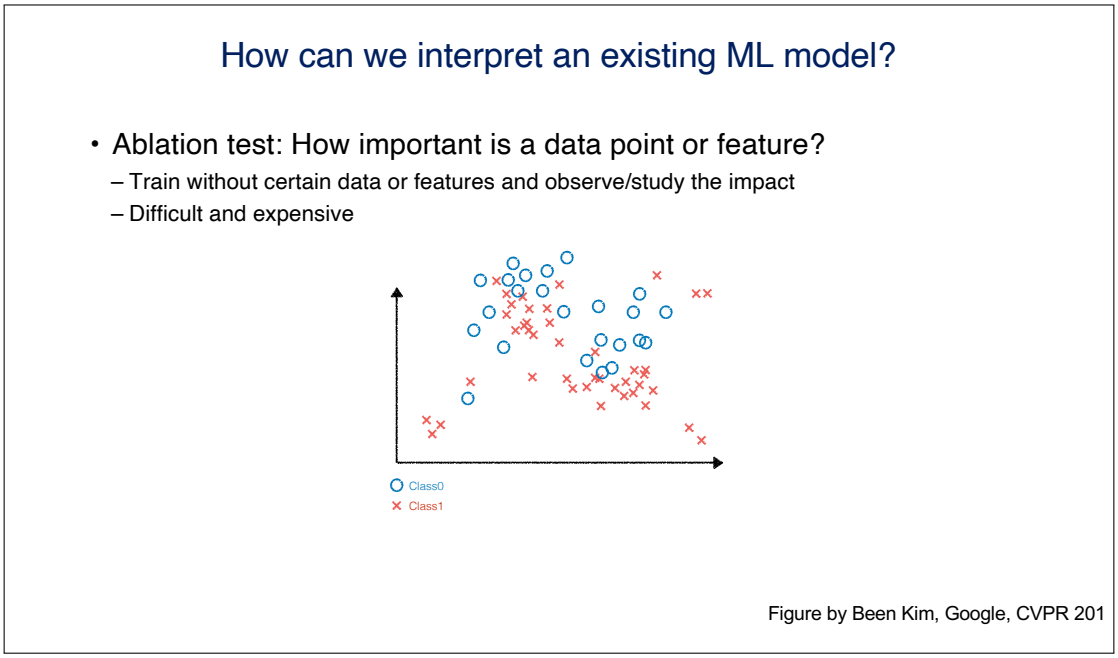



**Figure 62:** caption

## How can we interpret an existing ML model?

- Ablation test: How important is a data point or feature?
  - Train without certain data or features and observe/study the impact
  - Difficult and expensive

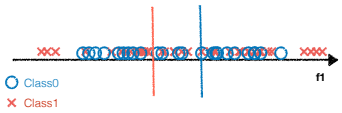



Figure by Been Kim, Google, CVPR 201

**Figure 63:** caption

## How can we interpret an existing ML model?

- Fit functions (use first derivatives)
  - Sensitivity analysis
  - Saliency maps

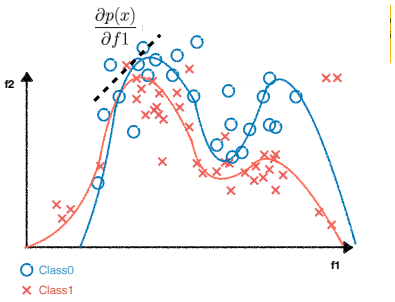



Figure by Been Kim, Google, CVPR 201

**Figure 64:** caption

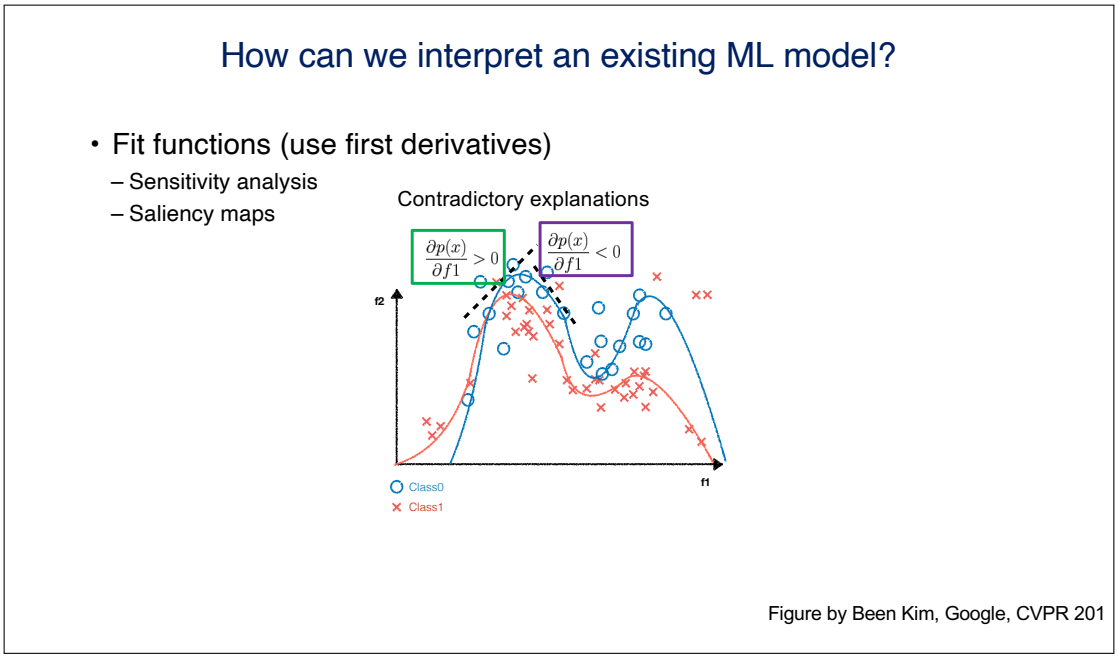



**Figure 65:** caption

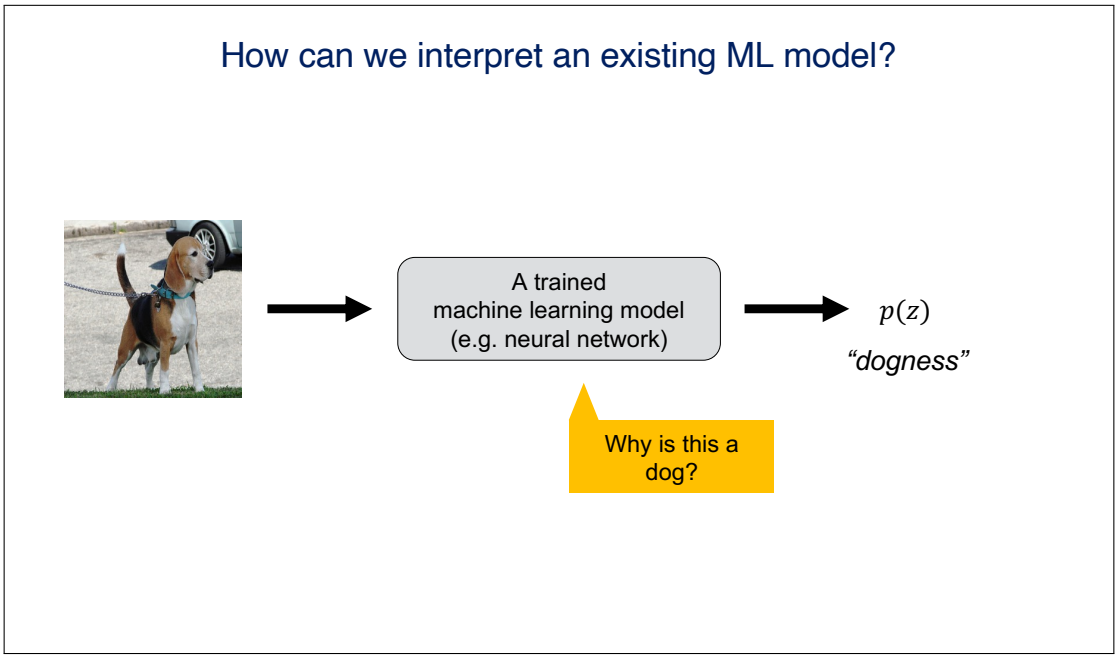



**Figure 66:** caption

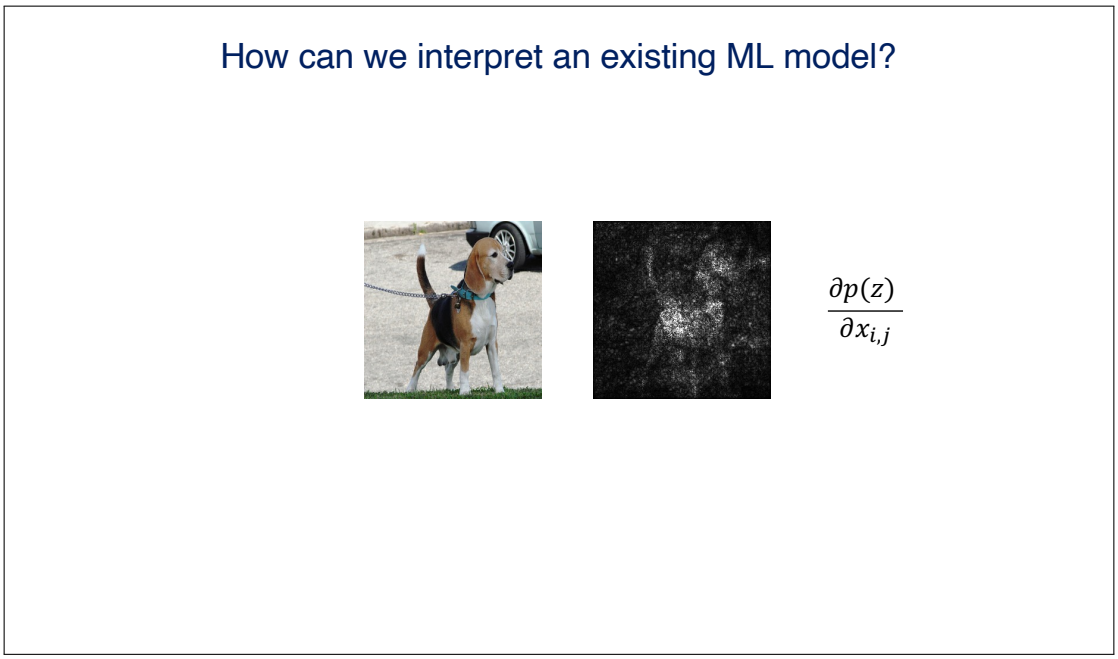



**Figure 67:** caption

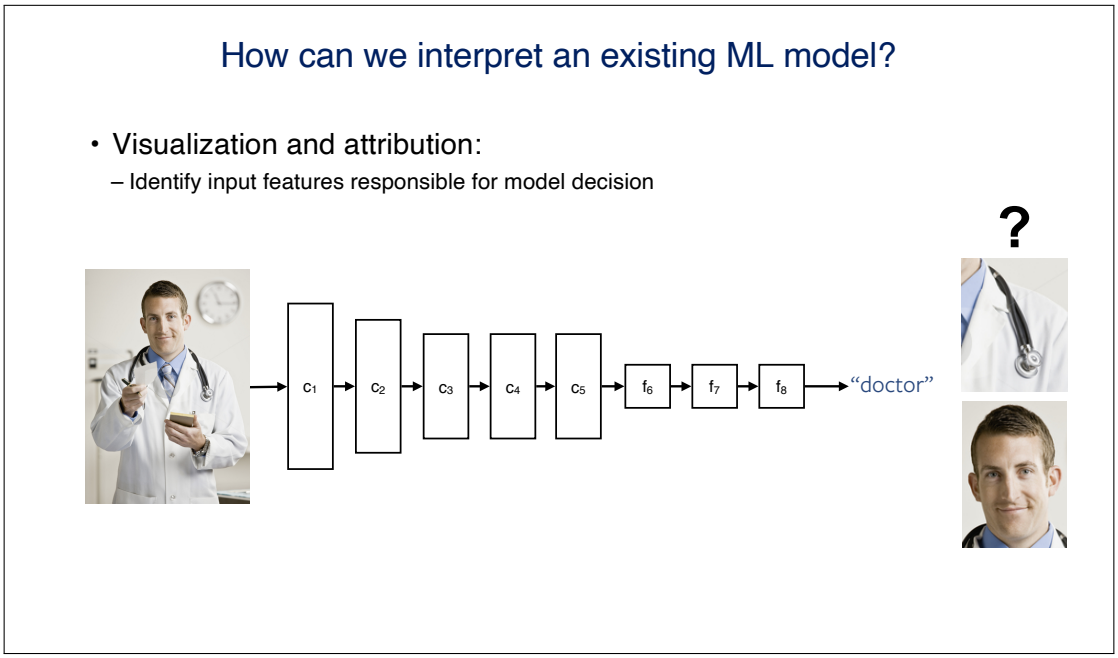
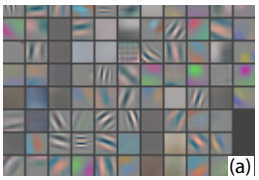
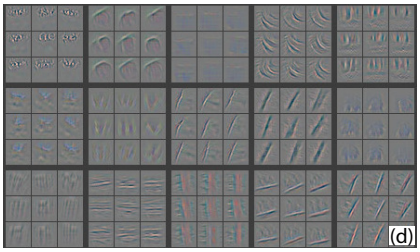



**Figure 68:** caption

## How can we interpret an existing ML model?

- Direct visualization of filters
- Easy to implement
- Limited practical value
  - First layers are easy to interpret (mostly low-level features)
  - Higher layers are more difficult to interpret (non-interpretable features)




https://arxiv.org/abs/1311.290

**Figure 69:** caption

## How can we interpret an existing ML model?

- Problem: Visualization of filters has limited value
- Solution: Instead visualize activations generated by kernels
  - Strong response: Feature is present
  - Weak response: Feature is not present
  - Easy to implement
  - Easy to interpret for early layers

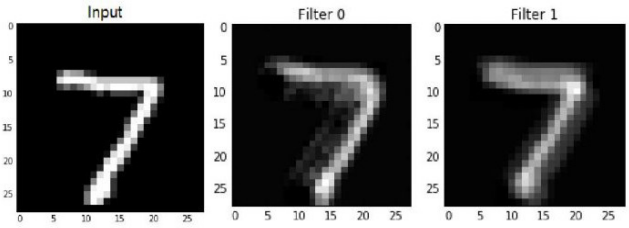



**Figure 70:** caption

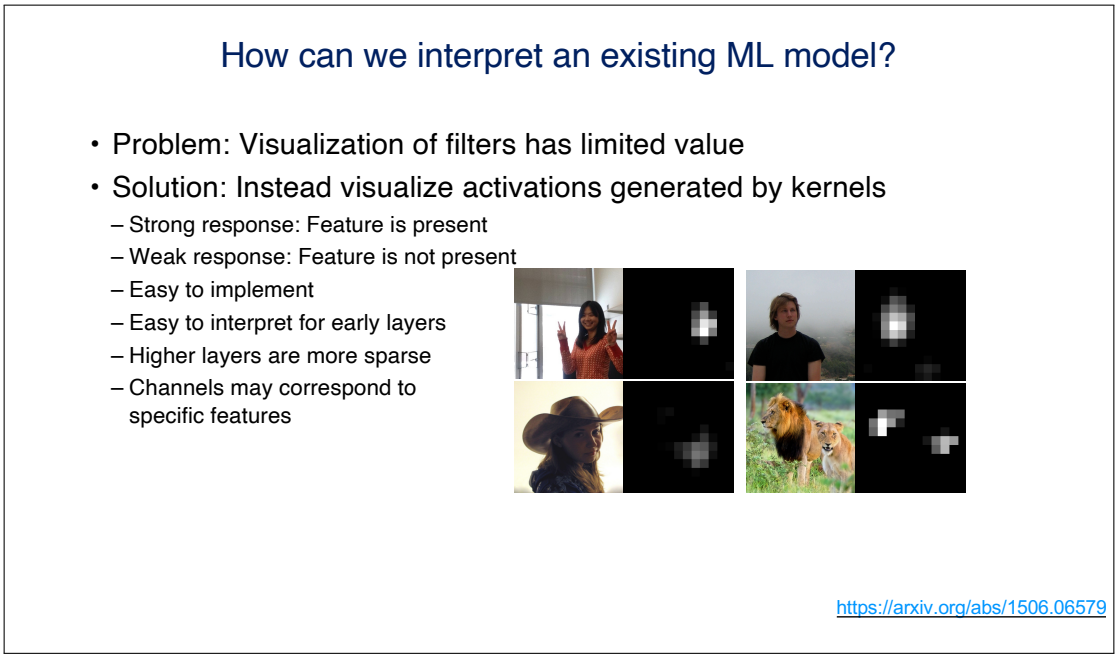



**Figure 71:** caption

### 2.4.1 Occlusions

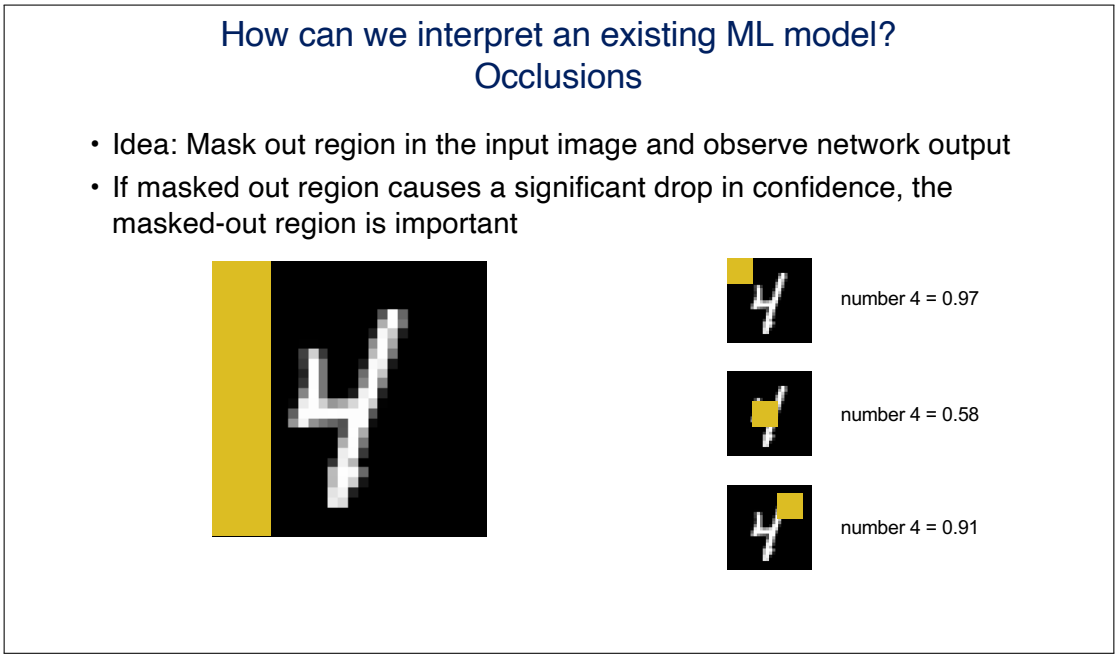
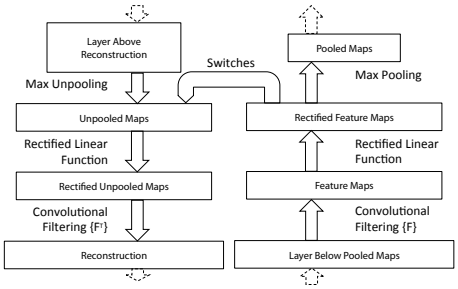



**Figure 72:** caption

### 2.4.2 Saliency maps

## How can we interpret an existing ML model? Saliency maps

- **DeconvNet**
  - Given a trained network and an image
  - Chose activation at one layer (set all others to zero)
  - Invert network



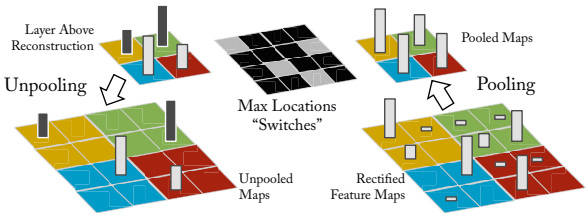

  - No training involved
  - Backward pass in network is almost identical to backpropagation (apart from ReLUs)

Visualizing and Understanding Convolutional Networks, M. D. Zeiler and R. Fergus, ECCV 201

**Figure 73:** caption

## How can we interpret an existing ML model? Saliency maps

- **DeconvNet**
  - Given a trained network and an image
  - Chose activation at one layer (set all others to zero)
  - Invert network




  - No training involved
  - Backward pass in network is almost identical to backpropagation (apart from ReLUs)

Visualizing and Understanding Convolutional Networks, M. D. Zeiler and R. Fergus, ECCV 201

**Figure 74:** caption

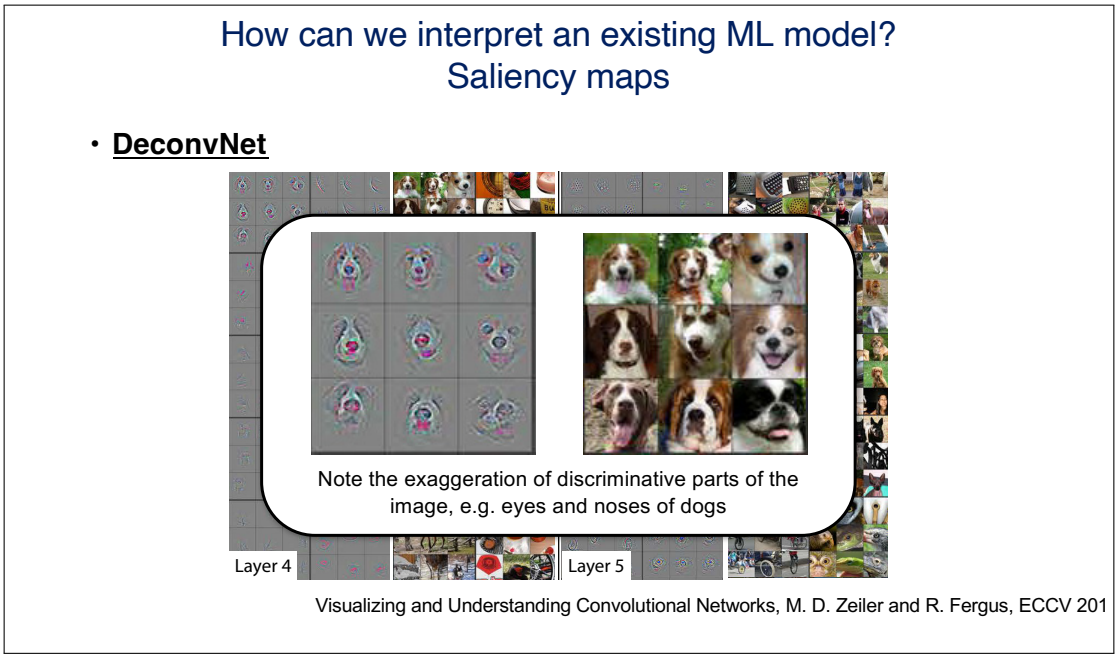



**Figure 75:** caption

### 2.4.3 Occlusions

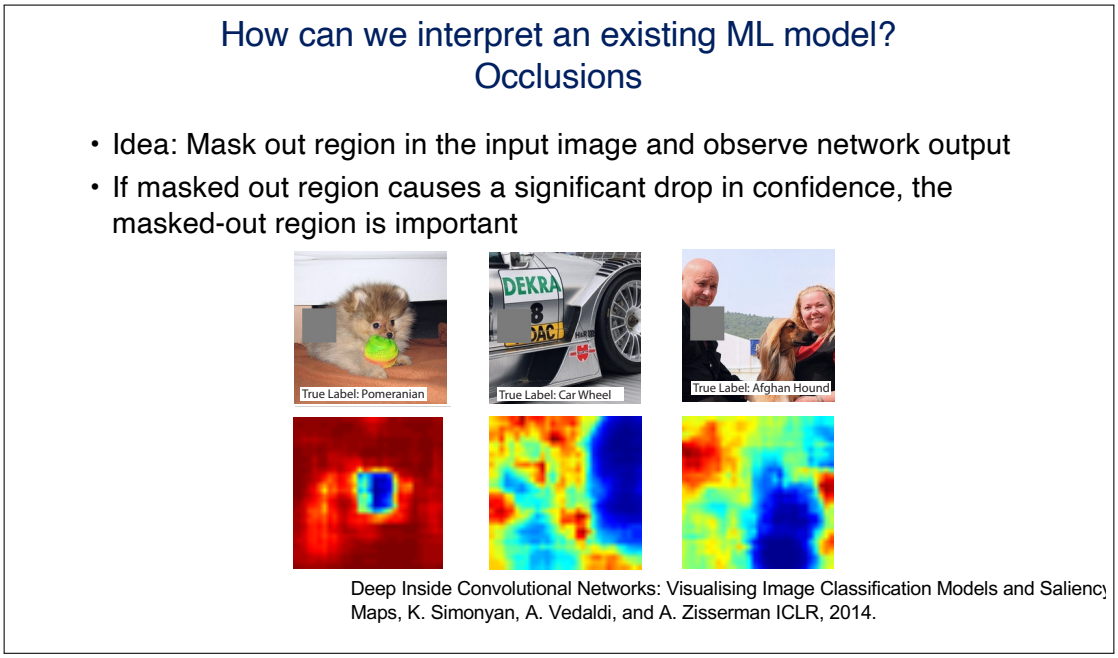



**Figure 76:** caption

### 2.4.4 Saliency maps

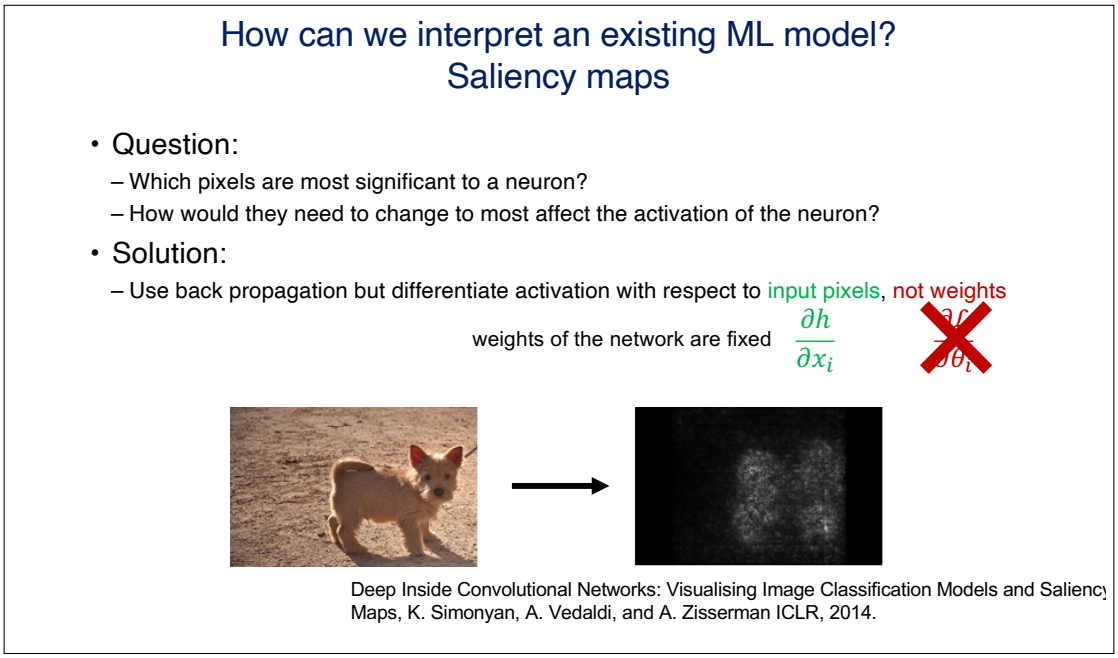



**Figure 77:** caption

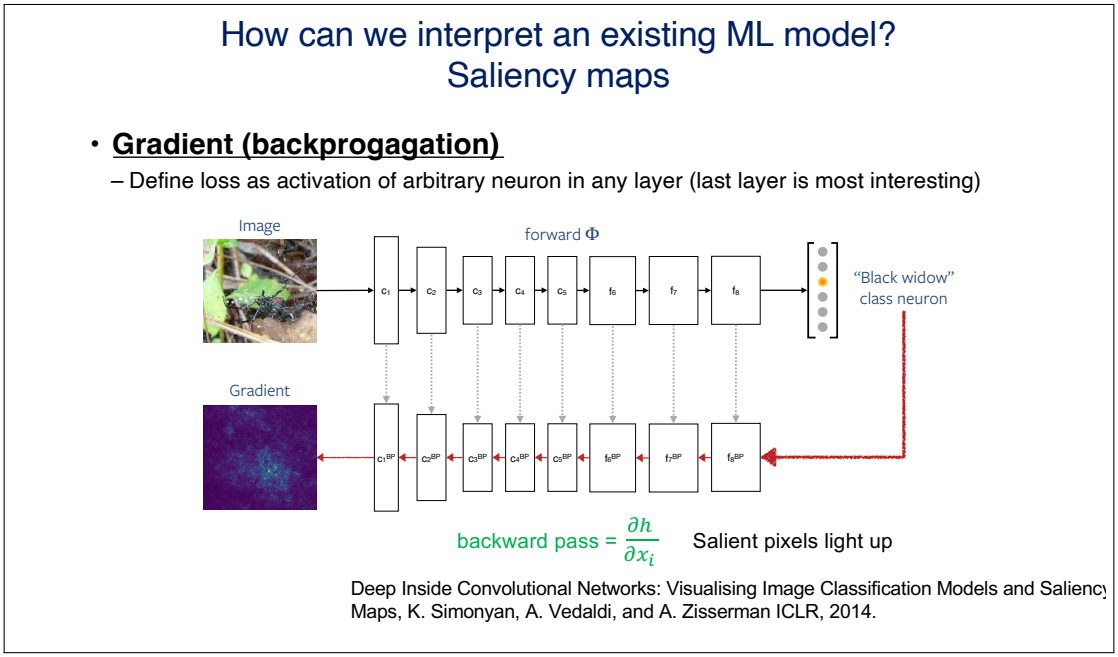



**Figure 78:** caption

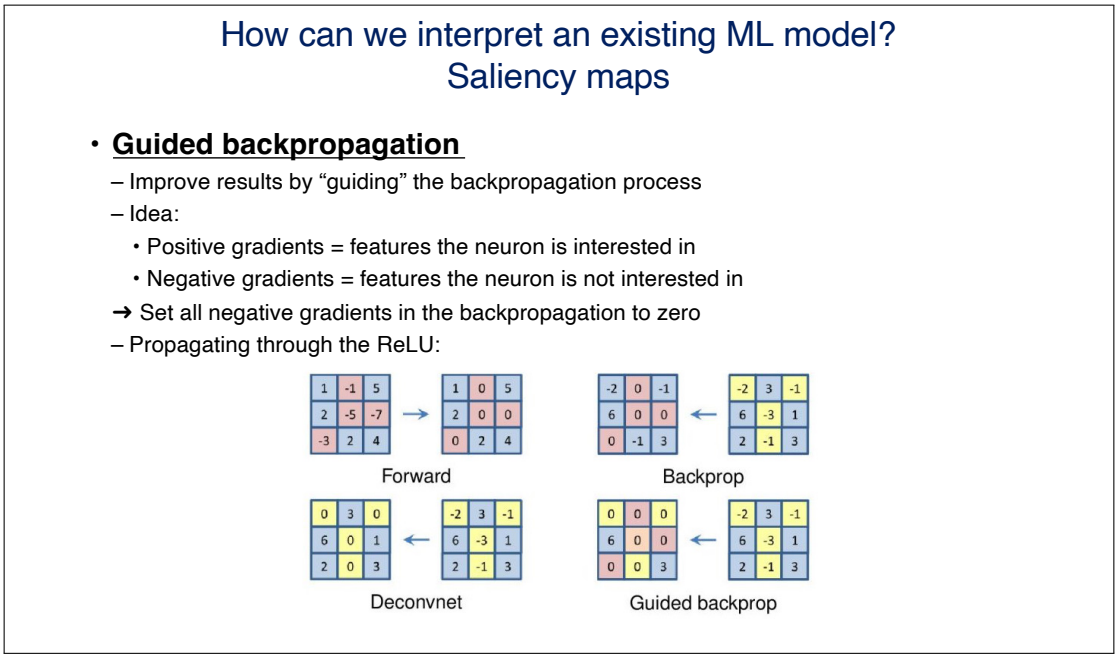



**Figure 79:** caption

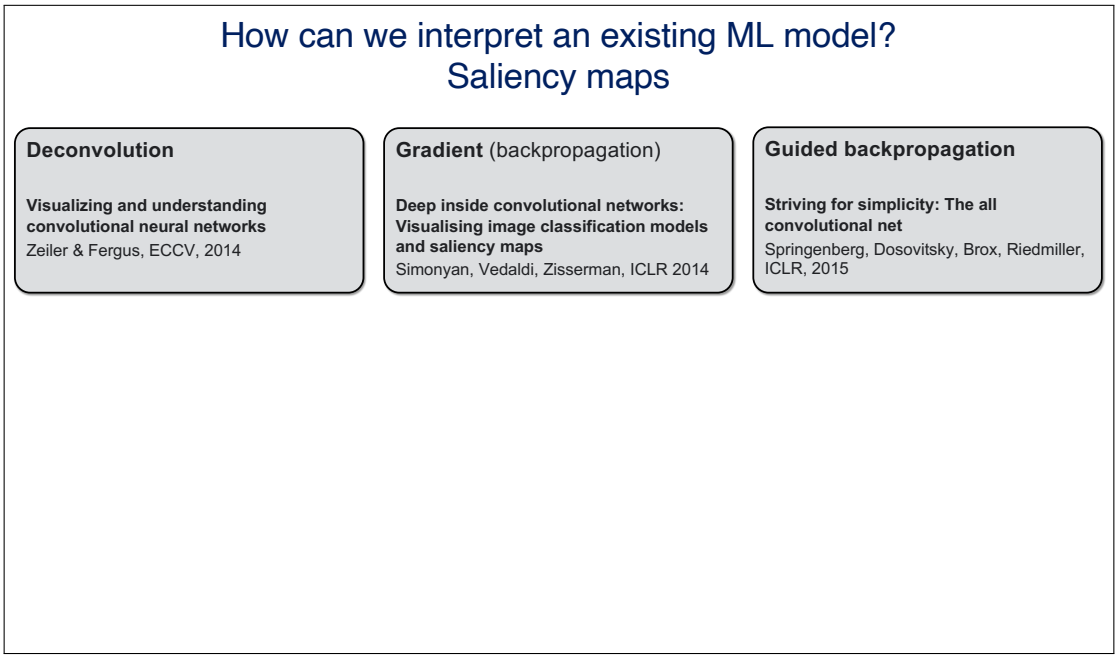



**Figure 80:** caption

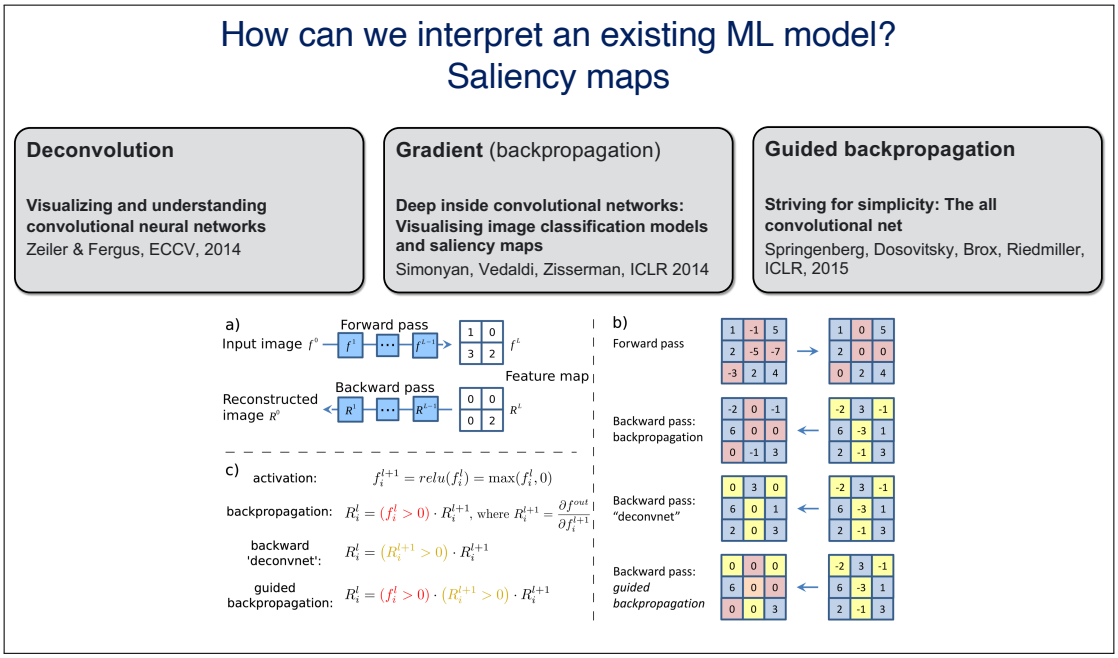



**Figure 81:** caption

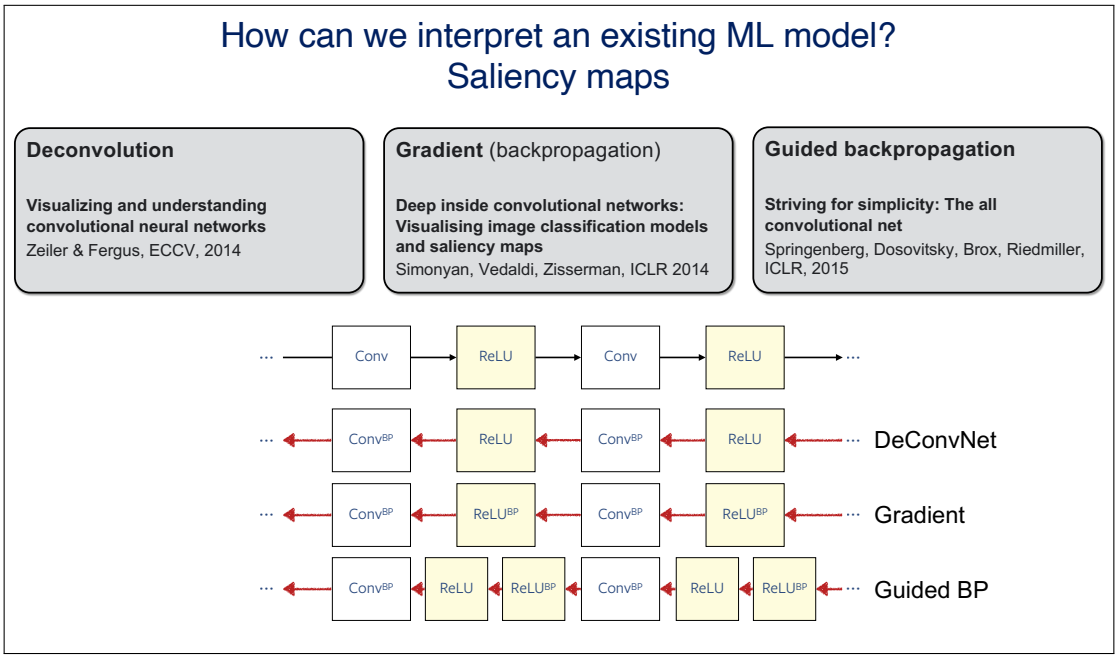



**Figure 82:** caption

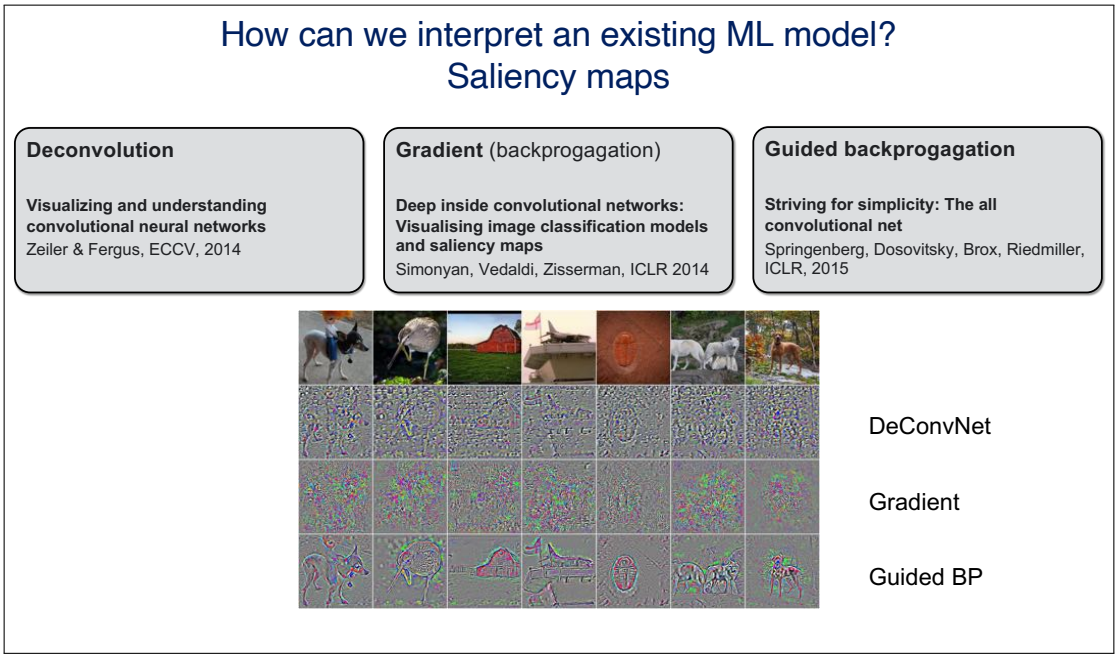



**Figure 83:** caption

## 2.5 Lack of channel specificity

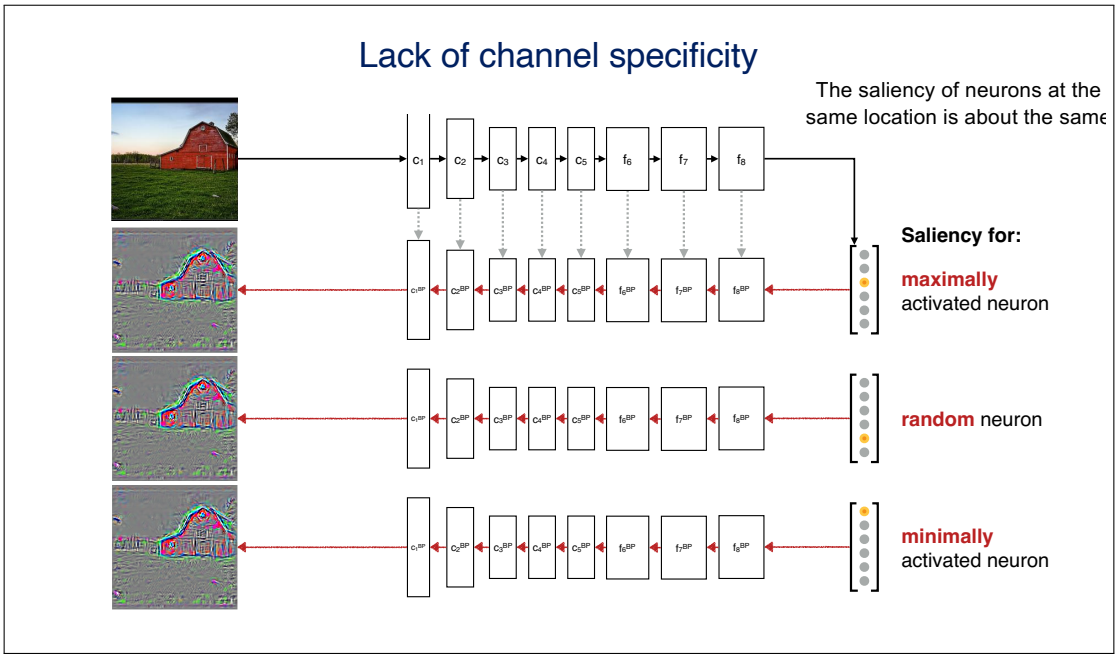



**Figure 84:** caption

## 2.6 Cam and Grad-Cam

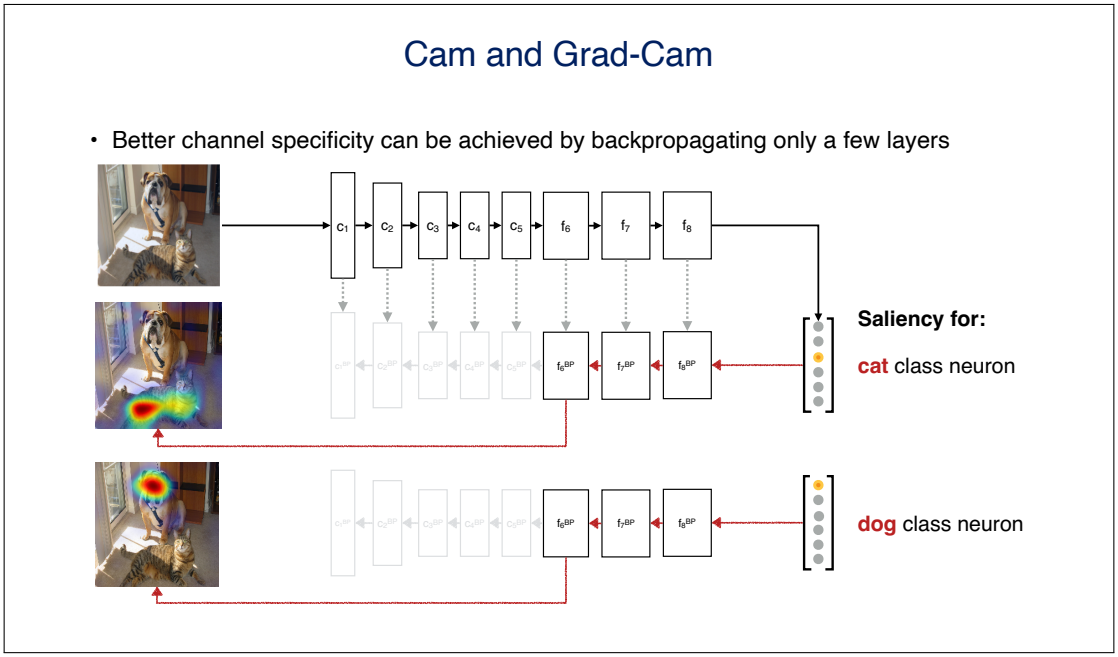



**Figure 85:** caption

## 2.7 DeepDream / Inceptionism

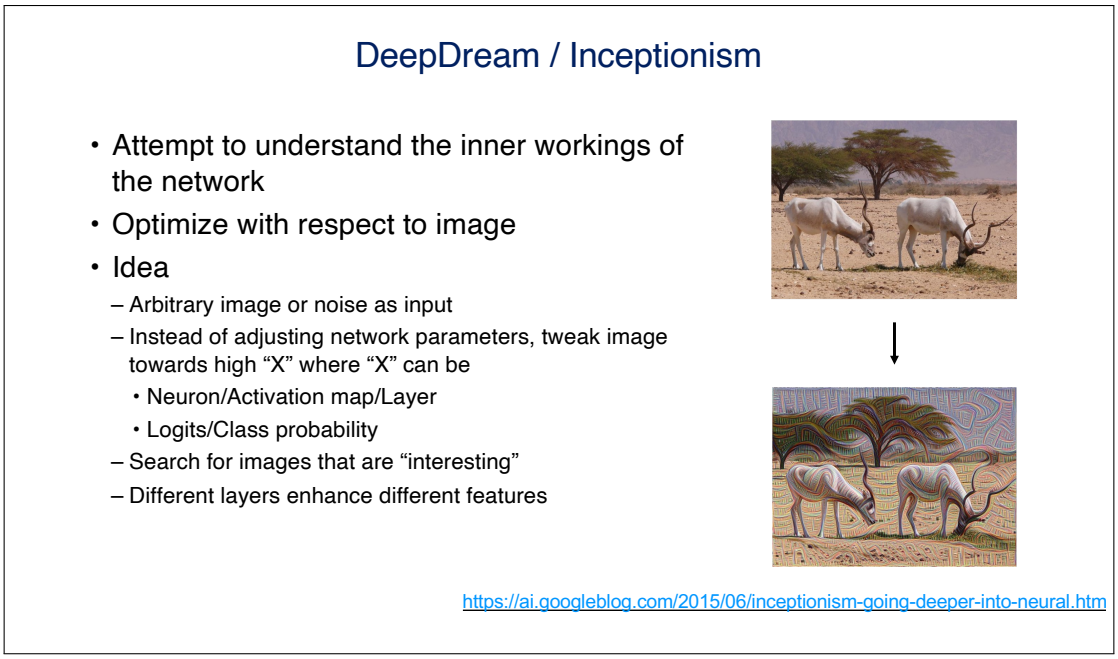



**Figure 86:** caption

## DeepDream / Inceptionism

- Find an image $x$ such that the activation $\phi_n(x)$ at layer $n$ is high

$$\max_x \phi_n(x) - \boxed{\lambda \mathcal{R}(x)}$$

Regularizer: e.g. L1 or L2 norm of image

- Algorithm
  - Forward propagate to layer $n$
  - No minimization of loss. Instead maximize L2 norm of activations of a particular NN layer
  - Backpropagate to input layer
- Resulting image will show learned features

https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.htm

**Figure 87:** caption

## 2.8 Inversion

## Inversion

- Inversion attempts to construct an image from a layer activation $\hat{y}$:

$$\hat{x} = \min_x (\|\phi(x) - \hat{y}\|_2^2 + \lambda \mathcal{R}(x))$$

- $\hat{x}$ is the reconstructed image
- $\phi(x)$ is the network output for input image $x$
- $\hat{y}$ is the desired activation
- $\mathcal{R}$ is the regularizer

**Figure 88:** caption

# 3 Robustness: Adversarial Methods

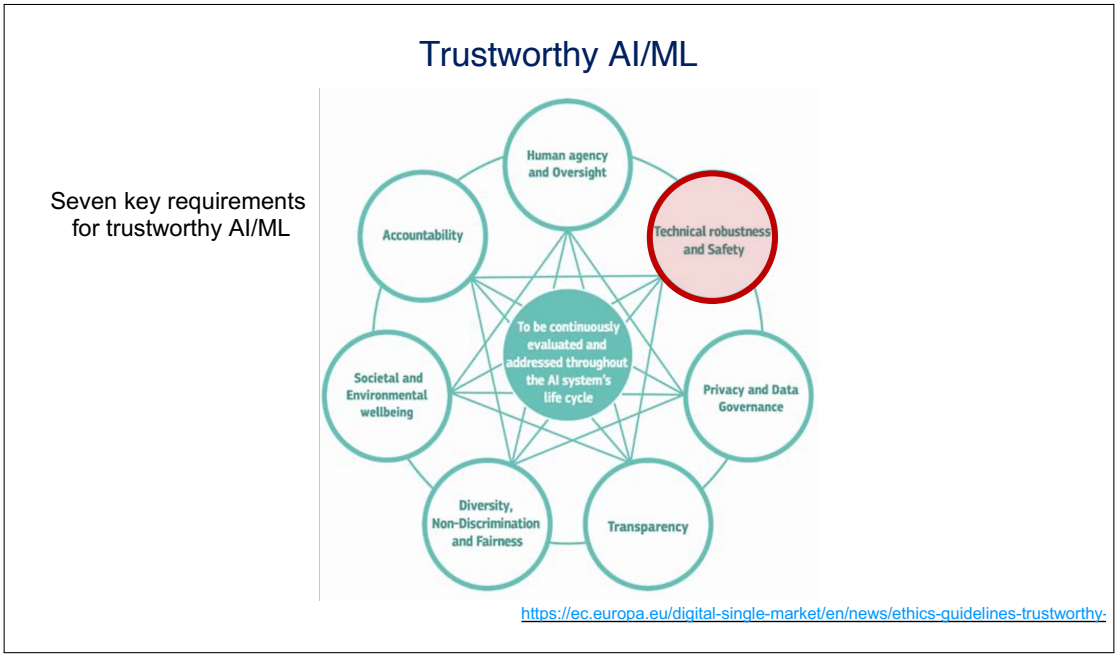



**Figure 89:** caption

## 3.1 Adversarial Attacks

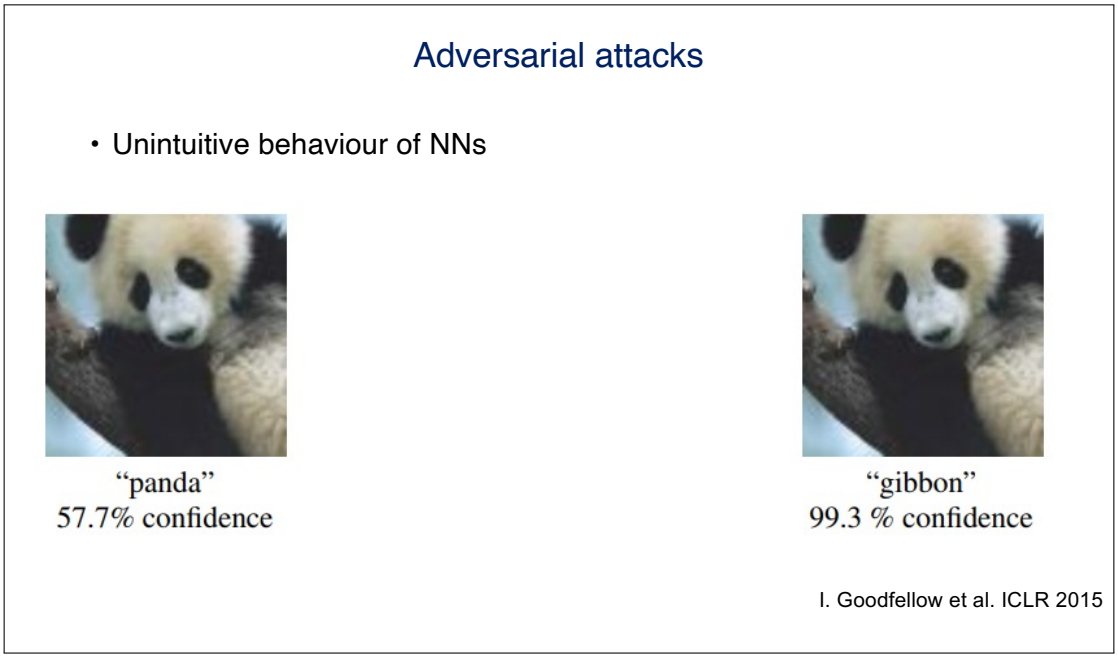
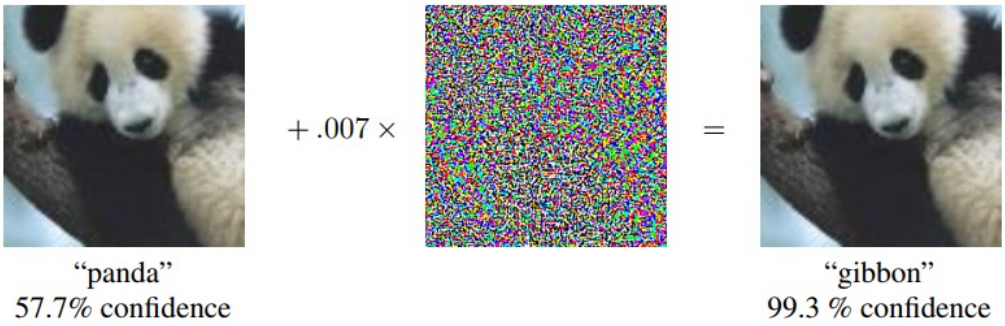



**Figure 90:** caption

## Adversarial attacks

- Unintuitive behaviour of NNs




I. Goodfellow et al. ICLR 2015

**Figure 91:** caption

### 3.1.1 Pertubation

## Adversarial attacks - Perturbation

- Assume a linear classifier:

$$\theta^T x$$

- We can think of an adversarial example that contains a small, non-perceivable perturbation to the input. Let's denote the perturbation as $\eta$:

$$\tilde{x} = x + \eta$$

- Then, the logits of the classifier would be

$$\theta^T \tilde{x} = \theta^T (x + \eta)$$
$$= \theta^T x + \theta^T \eta$$

**Figure 92:** caption

## Adversarial attacks - Perturbation

- Given a small perturbation $\eta$, the effect of the perturbation on the logits of the classifier is given by $\theta^T \eta$.
- Idea:
  - Find a $\eta$ that causes a change that is non-perceivable and ostensibly innocuous to the human eye, yet destructive and adverse enough for the classifier to the extent that its predictions are no longer accurate.
  - An adversarial example is one that which maximizes the value of $\theta^T \eta$ to sway the model into making a wrong prediction

**Figure 93:** caption

## Adversarial attacks - Perturbation

- Problem:
  - Need a constraint on $\eta$; otherwise, one could just apply a large perturbation to the input
- Solution:
  - Apply a constraint such that

$$\|\eta\|_\infty \leq \epsilon \qquad \|\boldsymbol{x}\|_\infty = \max_i \{|x_i|\}$$

- Assume a perturbation:

$$\eta = \epsilon \cdot \text{sign}(\theta)$$

- What are the bounds of this perturbation?

I. Goodfellow et al. ICLR 2015

**Figure 94:** caption

## Adversarial attacks - Perturbation

- What are the bounds of this perturbation?

$$\eta = \epsilon \cdot \text{sign}(\theta)$$

$$\theta^T \eta = \epsilon \cdot \theta^T \text{sign}(\theta)$$

$$= \epsilon \|\theta\|_1$$

$$= \epsilon m n$$

Here the average magnitude of an element of $\theta$ is given by $m$

I. Goodfellow et al. ICLR 2015

**Figure 95:** caption

## Adversarial attacks - Perturbation

- This means that the change in activation given by the perturbation increases linearly with respect to $n$ (or the dimensionality).
- If $n$ is large, one can expect even a small perturbation capped at $\epsilon$ to produce a perturbation big enough to render the model susceptible to an adversarial attack.
- Remember that for images $n = \text{no. of pixels}$
- Such perturbed examples as referred to as ***adversarial examples***

I. Goodfellow et al. ICLR 2015

**Figure 96:** caption

### 3.1.2 Fast Gradient Sign Method

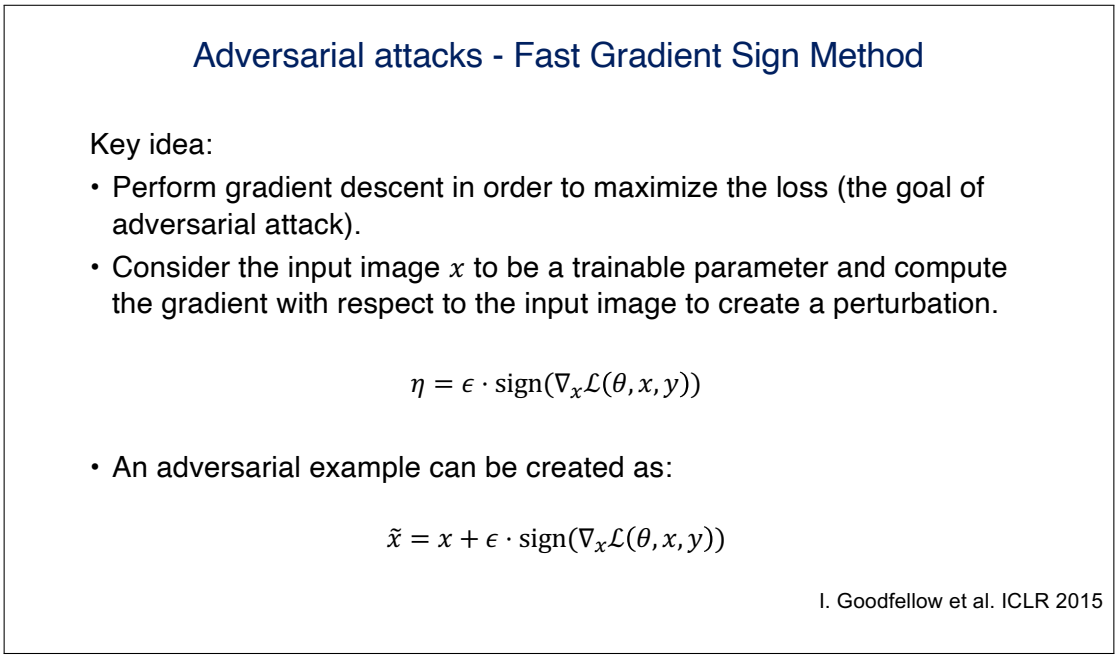



**Figure 97:** caption

## 3.2 Adversarial attacks

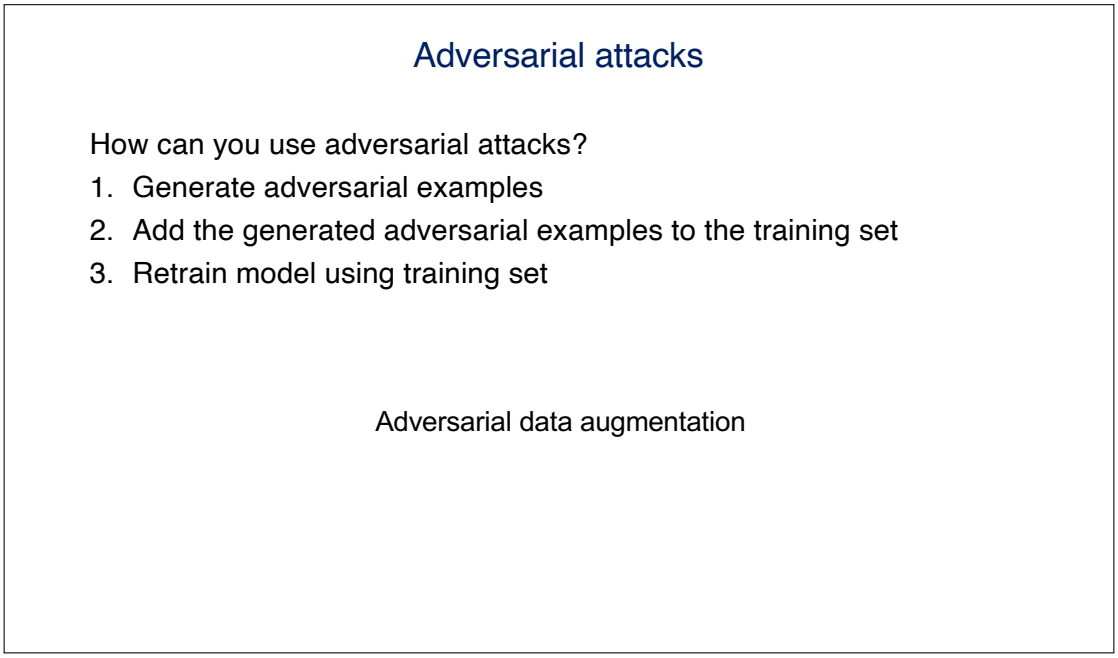



**Figure 98:** caption

# A References

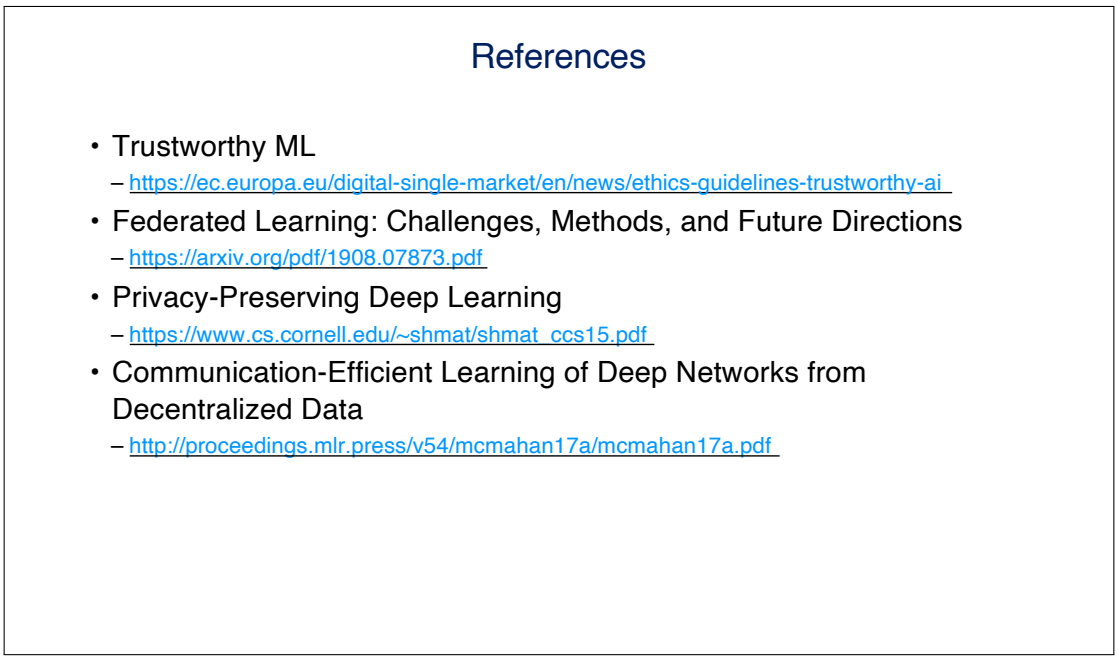



**Figure 99:** caption

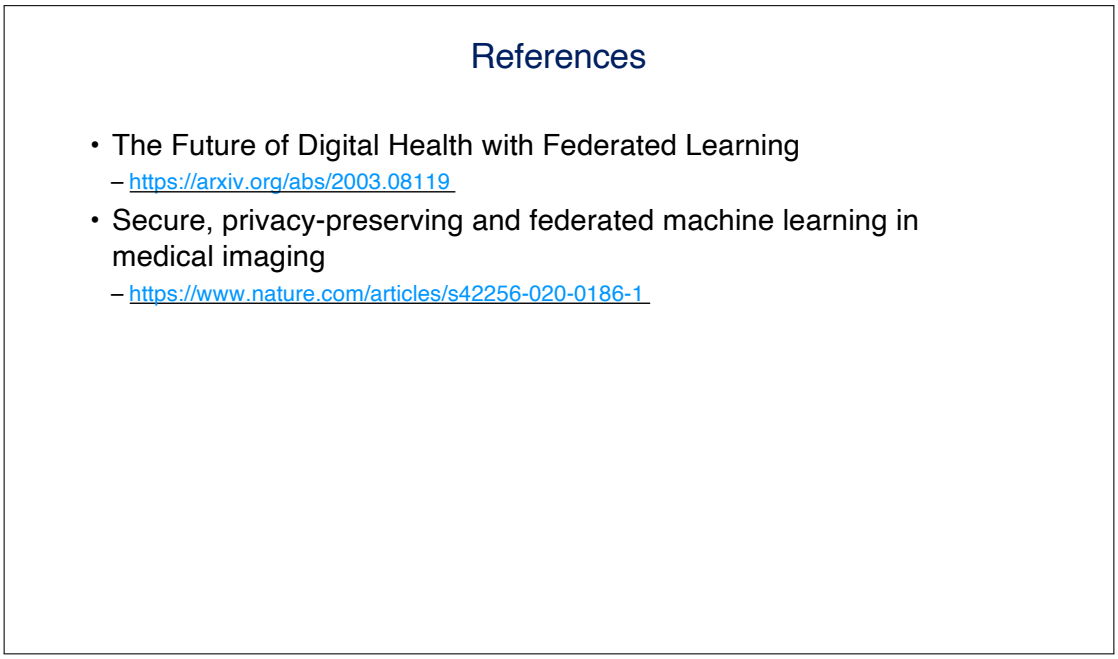



**Figure 100:** caption

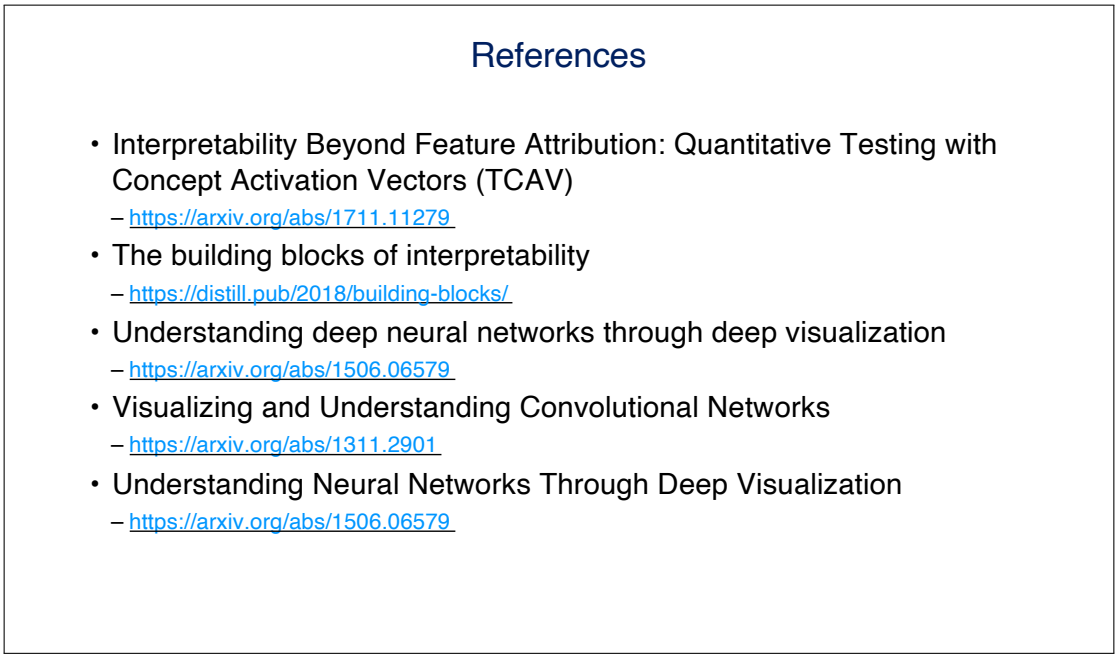



**Figure 101:** caption