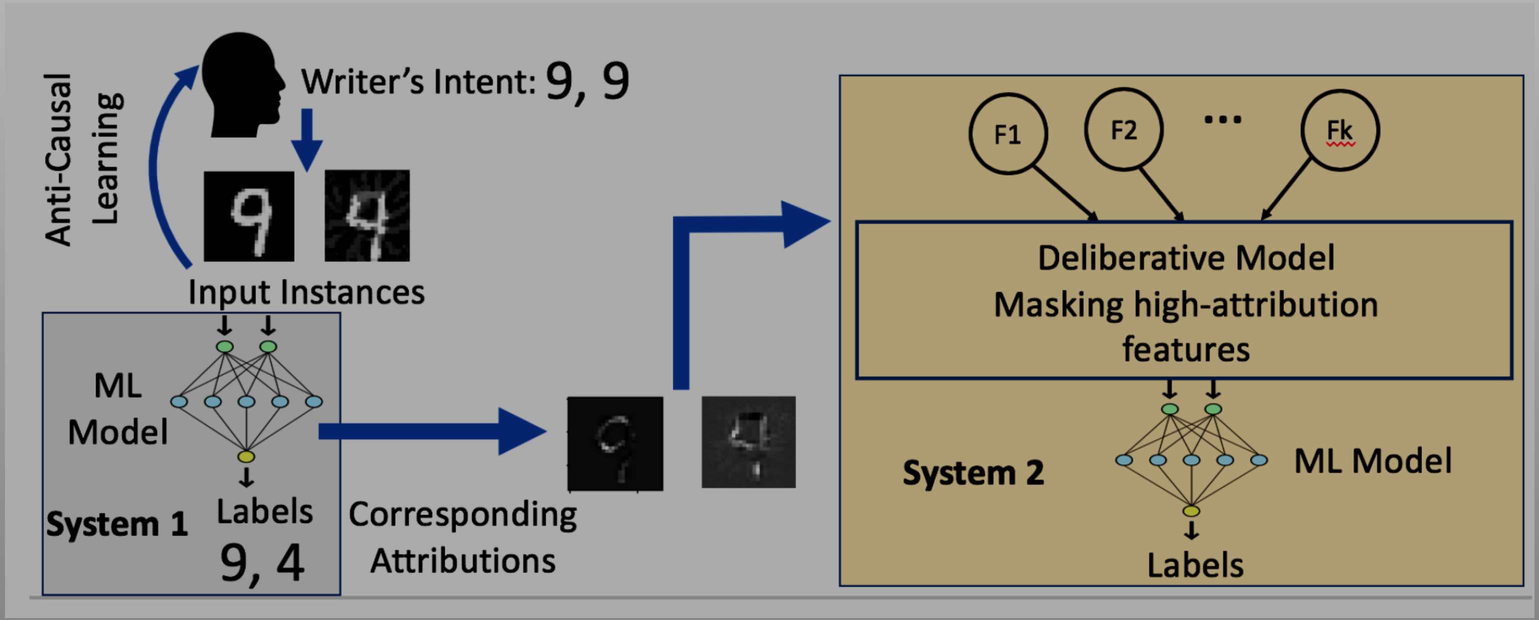
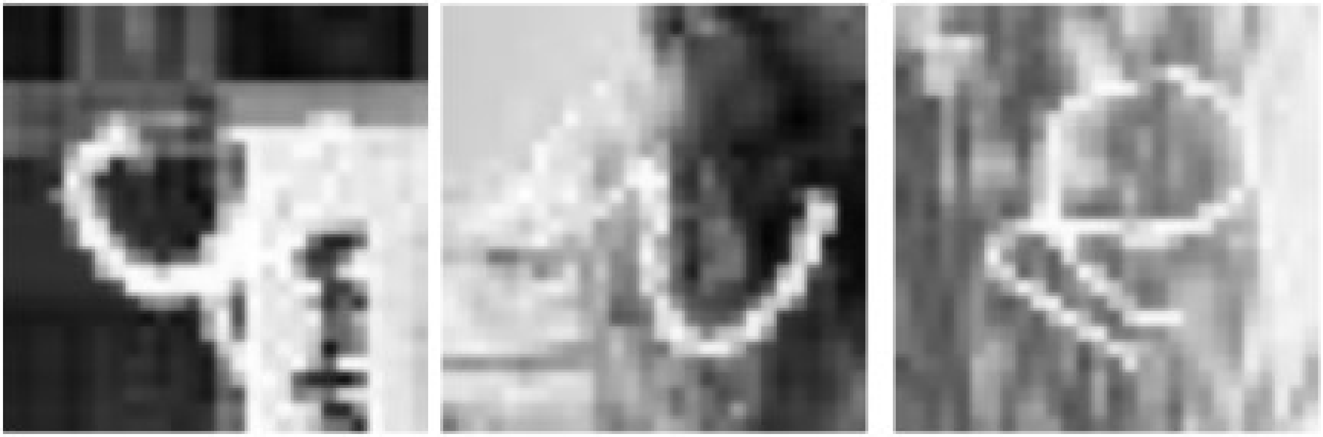
Attribution-Based Confidence Metric For Deep Neural Networks



Paper Authors: Susmit Jha, Sunny Raj, Steven Lawrence Fernandes, Sumit Kumar Jha, Somesh Jha, Brian Jalaian, Gunjan Verma, Ananthram Swami

Motivation

There are things we know that we know. There are known unknowns. That is to say there are things that we now know we don't know. But there are also unknown unknowns. There are things we do not know we don't know.



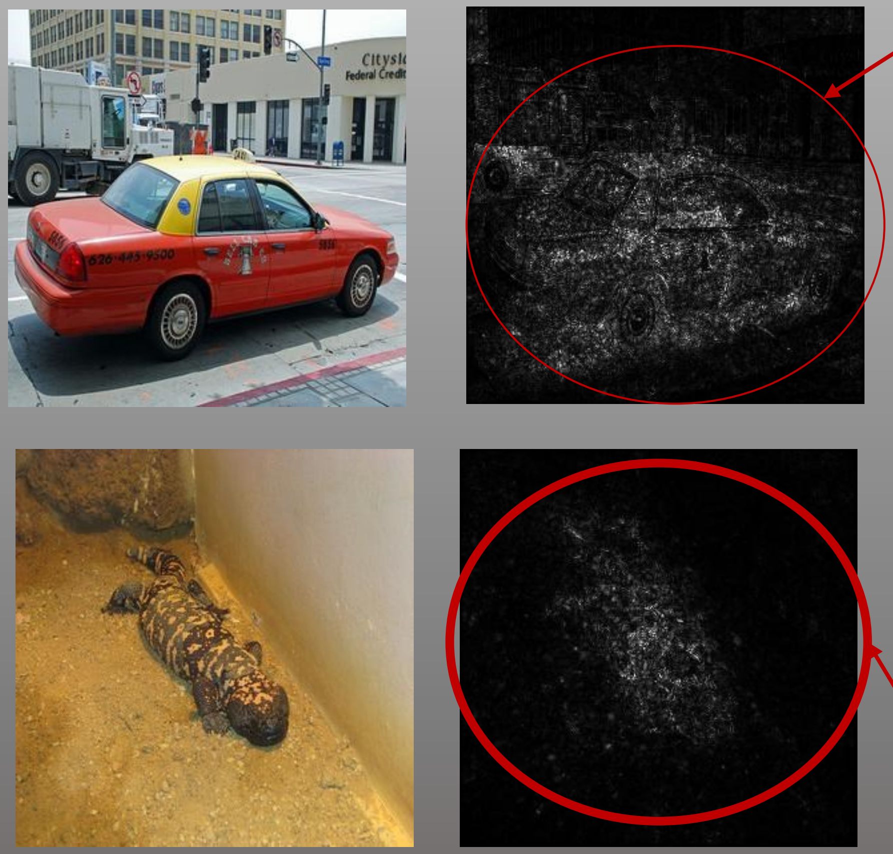
2 misclassified as 9
AttributeConf=0.28

3 misclassified as 2
AttributeConf=0.41

8 misclassified as 2
AttributeConf=0.89

AI does not know that it does now know!

Explainable AI and Attributions



Images

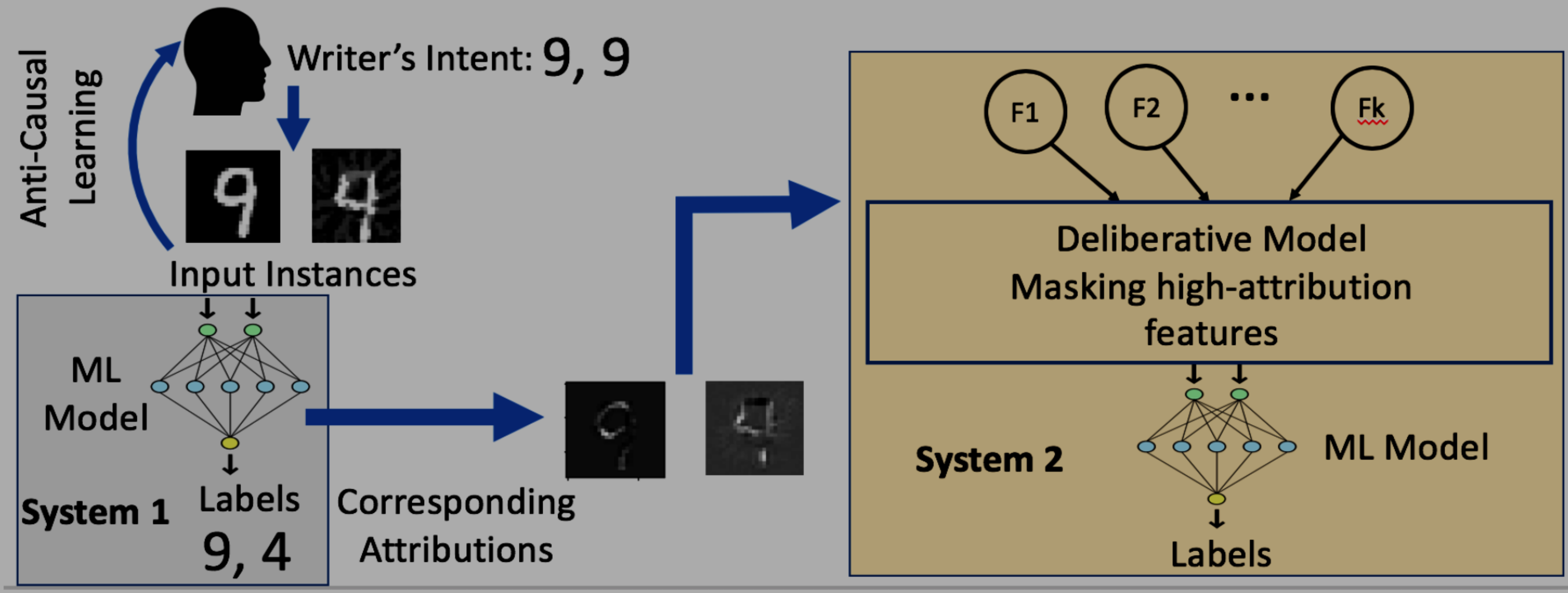
Explanations

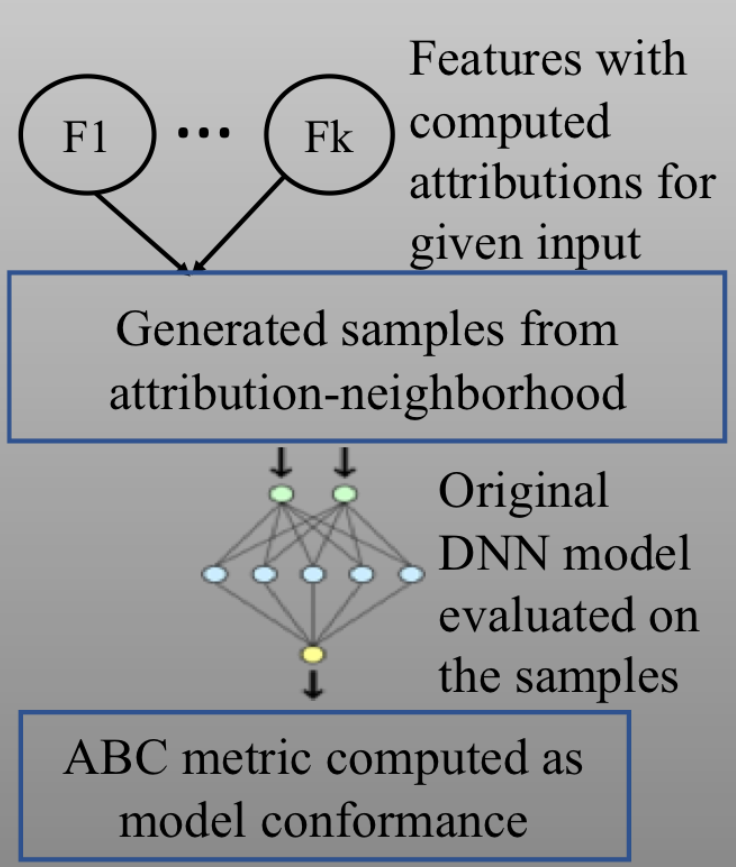
Explanation showing outline of a car

Explanation showing outline of a Gila Monster

- The values of the features in the explanation are called attributions
- Attribution-based confidence (ABC) metric

Attribution-based confidence (ABC) metric



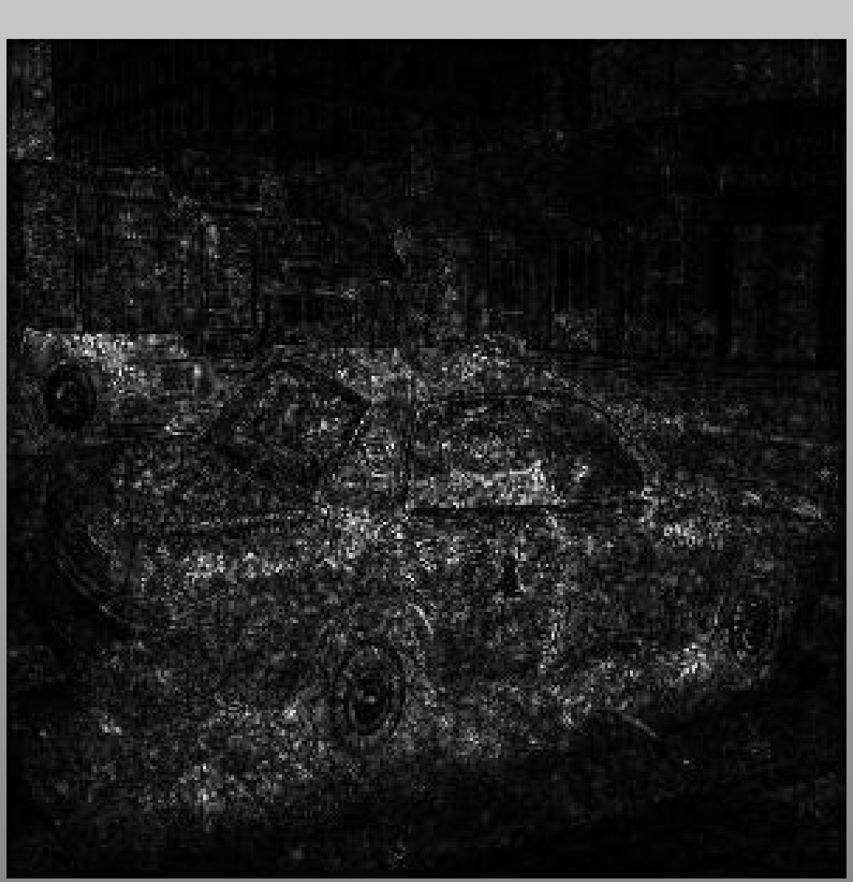


Features with computed attributions for given input

Generated samples from attribution-neighborhood

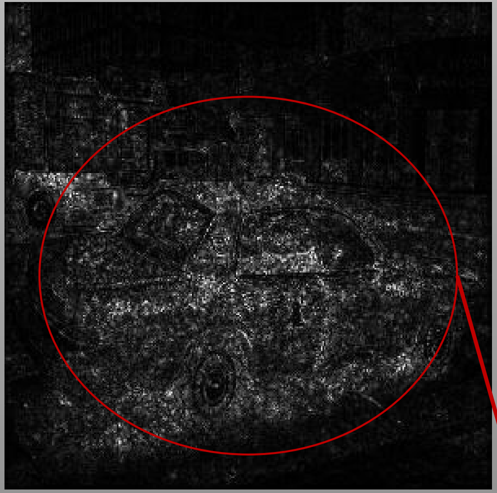
Original DNN model evaluated on the samples

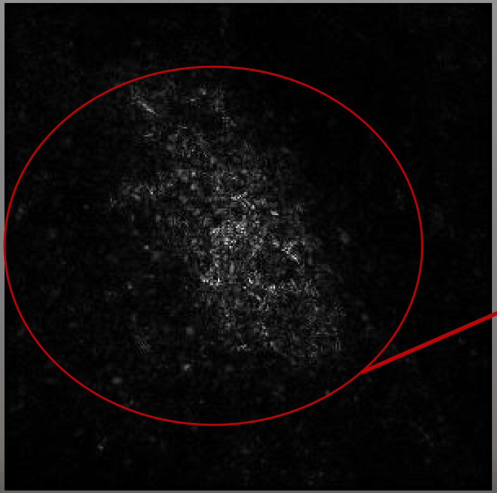
ABC metric computed as model conformance

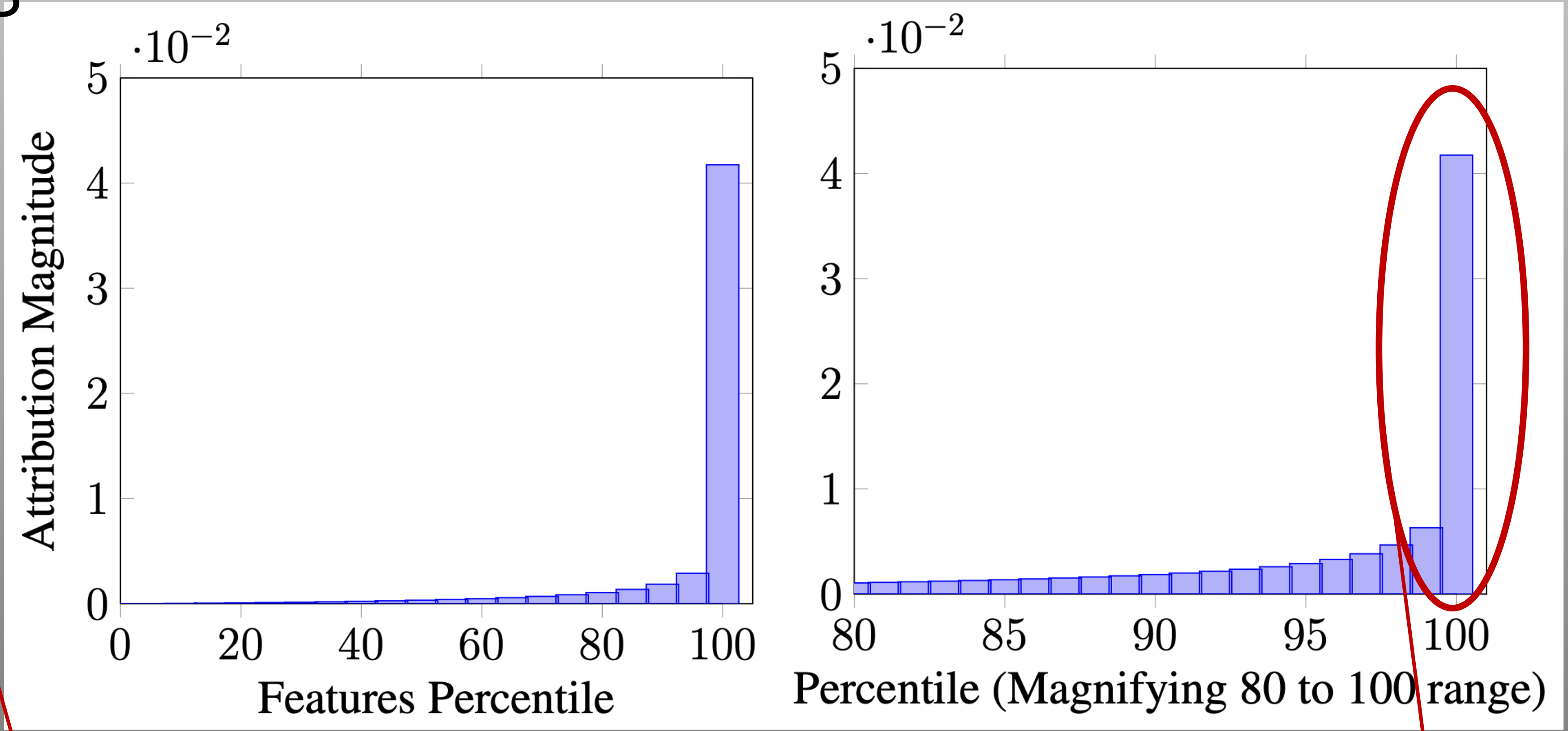


Attribution-based confidence (ABC) metric

Observations







Only top 1 percent of features have significant attributions

Number of high attributions features is low, making sampling over important features a more viable strategy.

How to generate such attribution-based samples?

- $\mathcal{F}$ is the DNN function, $\mathcal{A}$ is the attribution, $\mathbf{x}$ is the input, $\mathbf{x}^b$ is the baseline.

- **Assumption 1**: Attribution is dominated by the first order derivative. Most explanation generation methods assume this.

$$\mathcal{A}_j^i(\mathbf{x}) = (\mathbf{x}_j - \mathbf{x}_j^b) \times \int_{\alpha=0}^{1} \partial_j \mathcal{F}^i(\mathbf{x}^b + \alpha(\mathbf{x} - \mathbf{x}^b))d\alpha$$

- **Assumption 2**: Attributions are complete.

$$\mathcal{F}(\mathbf{x}) - \mathcal{F}(\mathbf{x}^b) = \textstyle\sum_{k=1}^{n} \mathcal{A}_k(\mathbf{x}) \text{ where } \mathbf{x} \text{ has } n \text{ features.}$$

- **Theorem**: The sensitivity of the output with respect to an input feature can be approximated as the ratio of the attribution and the input feature $\frac{\mathcal{A}_j(\mathbf{x})}{\mathbf{x}_j}$.

- **Proof:** Given an input $\mathbf{x}$ and its neighbor $\mathbf{x}' = \mathbf{x} + \delta\mathbf{x}$, we can use Taylor series expansion to express $\mathcal{F}(\mathbf{x}')$ as:

$$\mathcal{F}(\mathbf{x}') = \mathcal{F}(\mathbf{x}) + \boxed{\sum_{k=1}^{n} \left(\frac{\partial \mathcal{F}(\mathbf{x})}{\partial \mathbf{x}_k} \delta\mathbf{x}_k\right)} + \max_{k=1,\ldots,n} O(\delta\mathbf{x}_k^2) \ .$$

- Using completeness assumption on $\mathcal{F}(\mathbf{x}')$: $\mathcal{F}(\mathbf{x}') - \mathcal{F}(\mathbf{x}^b) = \sum_{k=1}^{n} \mathcal{A}_k(\mathbf{x}')$.
- Eliminating $\mathbf{x}^b$ term and doing Taylor expansion we get:

$$\begin{aligned} \mathcal{F}(\mathbf{x}') - \mathcal{F}(\mathbf{x}) &= \sum_{k=1}^{n} \left(\mathcal{A}_k(\mathbf{x}') - \mathcal{A}_k(\mathbf{x})\right) \\ &\boxed{= \sum_{k=1}^{n} \left(\frac{\partial \mathcal{A}_k(\mathbf{x})}{\partial \mathbf{x}_k} \delta\mathbf{x}_k\right)} + \max_{k=1,\ldots,n} O(\delta\mathbf{x}_k^2) \ . \end{aligned}$$

Sensitivity of the model with respect to the input feature $\mathbf{x}_j$ is $\frac{\partial \mathcal{A}_j(\mathbf{x})}{\partial \mathbf{x}_j}$.

- Sensitivity of the model with respect to the input feature $\mathbf{x}_j$ is $\frac{\partial \mathcal{A}_j(\mathbf{x})}{\partial \mathbf{x}_j}$.

$$\boxed{\mathcal{A}_j^i(\mathbf{x}) = (\mathbf{x}_j - \mathbf{x}_j^b) \times \int_{\alpha=0}^{1} \partial_j \mathcal{F}^i(\mathbf{x}^b + \alpha(\mathbf{x} - \mathbf{x}^b))d\alpha}$$

- Differentiating attributions and then removing non-linear term:

$$\begin{aligned}
\frac{\partial \mathcal{A}_j(\mathbf{x})}{\partial \mathbf{x}_j} &= \int_{\alpha=0}^{1} \frac{\partial \mathcal{F}(\mathbf{x}^{\mathbf{b}} + \alpha(\mathbf{x} - \mathbf{x}^b))}{\partial \mathbf{x}_j} d\alpha + \mathbf{x}_j \frac{\partial}{\partial \mathbf{x}_j} \Big(\int_{\alpha=0}^{1} \frac{\partial \mathcal{F}(\mathbf{x}^b + \alpha(\mathbf{x} - \mathbf{x}^b))}{\partial \mathbf{x}_j} d\alpha \\
&= \int_{\alpha=0}^{1} \frac{\partial \mathcal{F}(\mathbf{x}^{\mathbf{b}} + \alpha(\mathbf{x} - \mathbf{x}^b))}{\partial \mathbf{x}_j} d\alpha + \mathbf{x}_j \left(\int_{\alpha=0}^{1} \frac{\partial^2 \mathcal{F}(\mathbf{x}^b + \alpha(\mathbf{x} - \mathbf{x}^b))}{\partial \mathbf{x}_j^2} d\alpha \right) \\
&\approx \int_{\alpha=0}^{1} \frac{\partial \mathcal{F}(\mathbf{x}^{\mathbf{b}} + \alpha(\mathbf{x} - \mathbf{x}^b))}{\partial \mathbf{x}_j} d\alpha \boxed{=} \frac{\mathcal{A}_j(\mathbf{x})}{\mathbf{x}_j} \text{ with baseline feature } \mathbf{x}_j^b = 0.
\end{aligned}$$

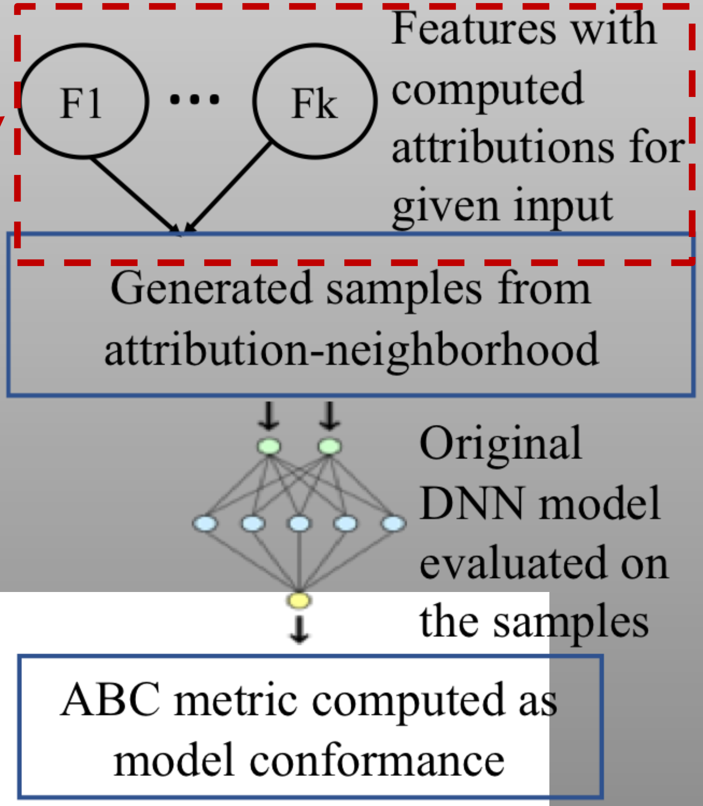
How to generate such attribution-based samples?

- **Theorem**: The sensitivity of the output with respect to an input feature can be approximated to the ratio of the attribution and the input feature $\frac{\mathcal{A}_j(\mathbf{x})}{\mathbf{x}_j}$.
- Probability of mutating input feature:

$$P(\mathbf{x}_j) = \frac{|\mathcal{A}_j/\mathbf{x}_j|}{\sum_{k=1}^{n} |\mathcal{A}_k/\mathbf{x}_k|}$$

- Generate samples by mutating feature $\mathbf{x}_j$ of input $\mathbf{x}$ to baseline $\mathbf{x}_j^b$ with probability $P(\mathbf{x}_j)$.

ABC algorithm (Attribution-Based Confidence)



Features with computed attributions for given input

Generated samples from attribution-neighborhood

Original DNN model evaluated on the samples

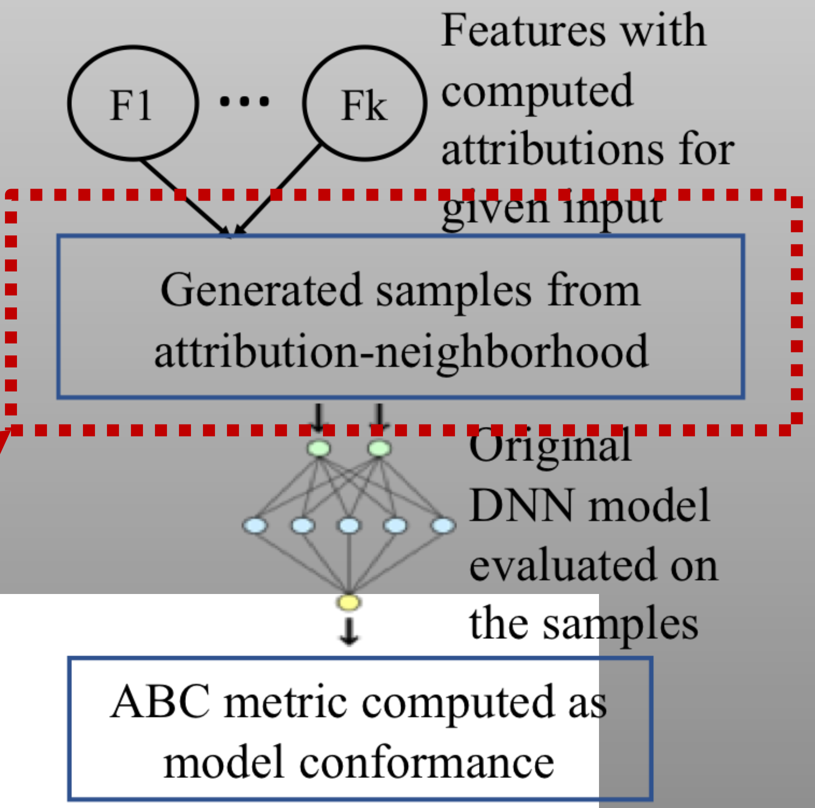
ABC metric computed as model conformance

Input: Model $\mathcal{F}$, Input $\mathbf{x}$ with features $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$, Sample size S

Output: ABC metric $c(\mathcal{F}, \mathbf{x})$

```
$\mathcal{A}_1, \ldots \mathcal{A}_n \leftarrow$ Attributions of features $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$ from input $\mathbf{x}$
$i \leftarrow \mathcal{F}(\mathbf{x})$ {Obtain model prediction}
for $j = 1$ to $n$ do
    $P(\mathbf{x}_j) \leftarrow \frac{|\mathcal{A}_j / \mathbf{x}_j|}{\sum_{k=1}^{n} |\mathcal{A}_k / \mathbf{x}_k|}$
end for
Generate $S$ samples by mutating feature $\mathbf{x}_j$ of input $\mathbf{x}$ to baseline $\mathbf{x}_j^b$ with probability $P(\mathbf{x}_j)$
Obtain the output of the model on the $S$ samples.
$c(\mathcal{F}, \mathbf{x}) \leftarrow S_{conform}/S$ where model's output on $S_{conform}$ samples is $i$
return $c(\mathcal{F}, \mathbf{x})$ as confidence metric (ABC) of prediction by the model $\mathcal{F}$ on the input $\mathbf{x}$
```

ABC algorithm (Attribution-Based Confidence)



Features with computed attributions for given input

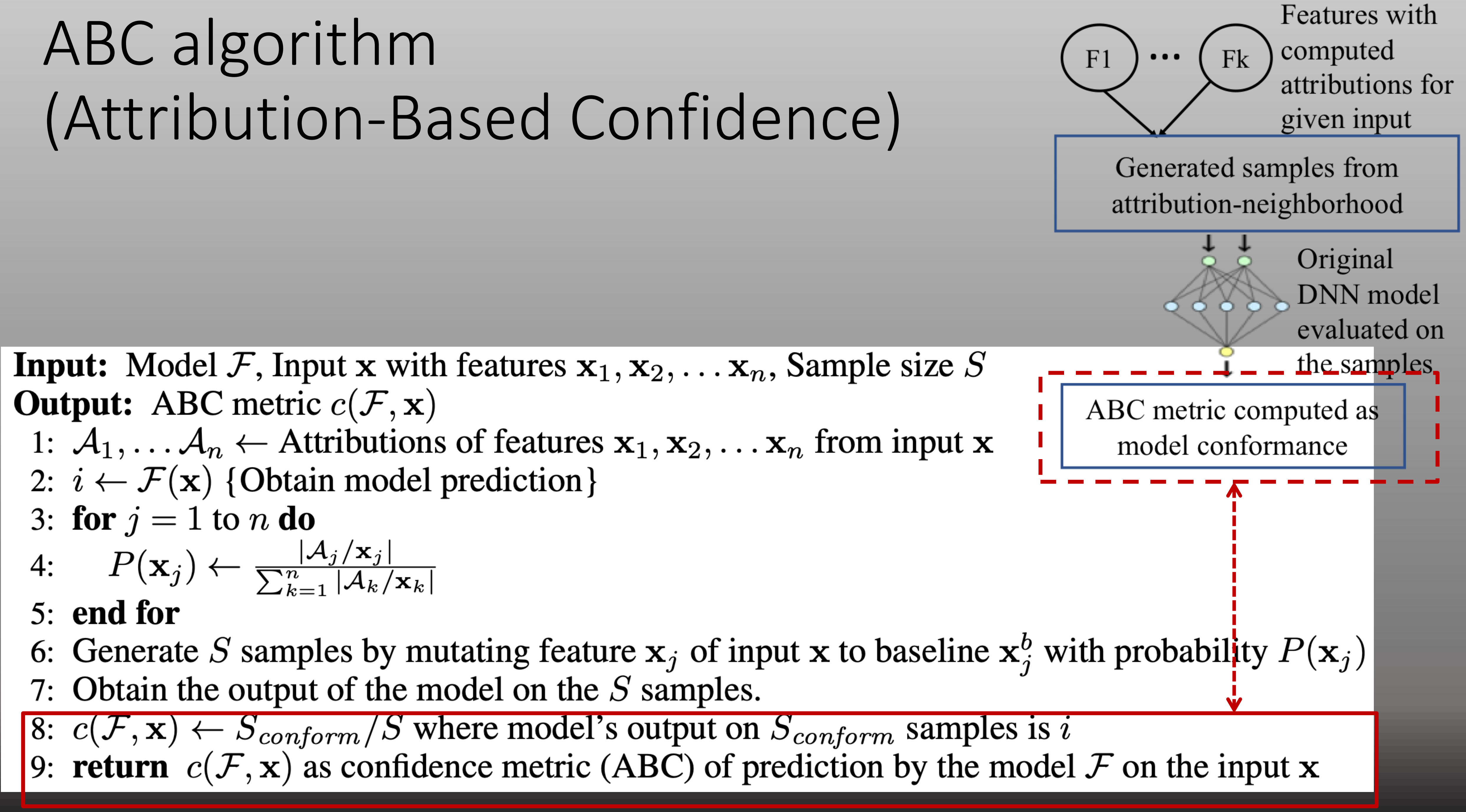
Generated samples from attribution-neighborhood

Original DNN model evaluated on the samples

ABC metric computed as model conformance

```
Input: Model $\mathcal{F}$, Input $\mathbf{x}$ with features $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$, Sample size $S$
Output: ABC metric $c(\mathcal{F}, \mathbf{x})$
$\mathcal{A}_1, \ldots \mathcal{A}_n \leftarrow$ Attributions of features $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$ from input $\mathbf{x}$
$i \leftarrow \mathcal{F}(\mathbf{x})$ {Obtain model prediction}
for $j = 1$ to $n$ do
    $P(\mathbf{x}_j) \leftarrow \frac{|\mathcal{A}_j / \mathbf{x}_j|}{\sum_{k=1}^{n} |\mathcal{A}_k / \mathbf{x}_k|}$
end for
Generate $S$ samples by mutating feature $\mathbf{x}_j$ of input $\mathbf{x}$ to baseline $\mathbf{x}_j^b$ with probability $P(\mathbf{x}_j)$
Obtain the output of the model on the $S$ samples.
$c(\mathcal{F}, \mathbf{x}) \leftarrow S_{conform}/S$ where model's output on $S_{conform}$ samples is $i$
return $c(\mathcal{F}, \mathbf{x})$ as confidence metric (ABC) of prediction by the model $\mathcal{F}$ on the input $\mathbf{x}$
```

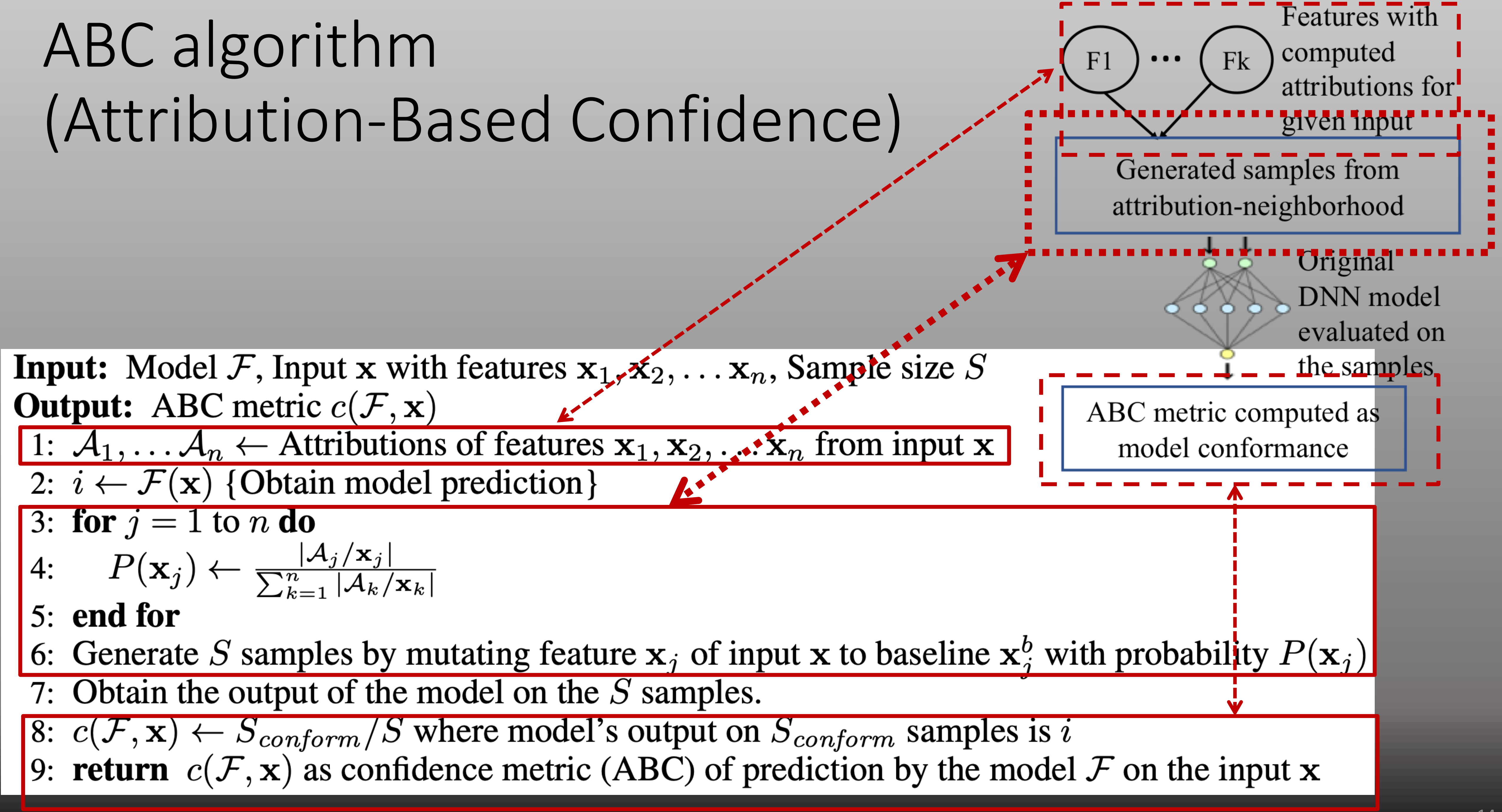
ABC algorithm (Attribution-Based Confidence)



F1
...
Fk
Features with computed attributions for given input
Generated samples from attribution-neighborhood
Original DNN model evaluated on the samples
ABC metric computed as model conformance

```
Input: Model $\mathcal{F}$, Input $\mathbf{x}$ with features $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$, Sample size $S$
Output: ABC metric $c(\mathcal{F}, \mathbf{x})$
$\mathcal{A}_1, \ldots \mathcal{A}_n \leftarrow$ Attributions of features $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$ from input $\mathbf{x}$
$i \leftarrow \mathcal{F}(\mathbf{x})$ {Obtain model prediction}
for $j = 1$ to $n$ do
    $P(\mathbf{x}_j) \leftarrow \frac{|\mathcal{A}_j / \mathbf{x}_j|}{\sum_{k=1}^{n} |\mathcal{A}_k / \mathbf{x}_k|}$
end for
Generate $S$ samples by mutating feature $\mathbf{x}_j$ of input $\mathbf{x}$ to baseline $\mathbf{x}_j^b$ with probability $P(\mathbf{x}_j)$
Obtain the output of the model on the $S$ samples.
$c(\mathcal{F}, \mathbf{x}) \leftarrow S_{conform}/S$ where model's output on $S_{conform}$ samples is $i$
return $c(\mathcal{F}, \mathbf{x})$ as confidence metric (ABC) of prediction by the model $\mathcal{F}$ on the input $\mathbf{x}$
```

ABC algorithm (Attribution-Based Confidence)



```
Input: Model $\mathcal{F}$, Input $\mathbf{x}$ with features $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$, Sample size $S$
Output: ABC metric $c(\mathcal{F}, \mathbf{x})$
$\mathcal{A}_1, \ldots \mathcal{A}_n \leftarrow$ Attributions of features $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$ from input $\mathbf{x}$
$i \leftarrow \mathcal{F}(\mathbf{x})$ {Obtain model prediction}
for $j = 1$ to $n$ do
    $P(\mathbf{x}_j) \leftarrow \frac{|\mathcal{A}_j / \mathbf{x}_j|}{\sum_{k=1}^{n} |\mathcal{A}_k / \mathbf{x}_k|}$
end for
Generate $S$ samples by mutating feature $\mathbf{x}_j$ of input $\mathbf{x}$ to baseline $\mathbf{x}_j^b$ with probability $P(\mathbf{x}_j)$
Obtain the output of the model on the $S$ samples.
$c(\mathcal{F}, \mathbf{x}) \leftarrow S_{conform} / S$ where model's output on $S_{conform}$ samples is $i$
return $c(\mathcal{F}, \mathbf{x})$ as confidence metric (ABC) of prediction by the model $\mathcal{F}$ on the input $\mathbf{x}$
```

F1
...
Fk
Features with computed attributions for given input
Generated samples from attribution-neighborhood
Original DNN model evaluated on the samples
ABC metric computed as model conformance

Finding conformance and deciding when to stop sampling

- SPRT to determine optimal stopping criteria (Wald, 1945)

C	**N**	**C**	**C**	**C**	**C**	**C**	**N**	**C**	**N**

C = 7, N = 3

- Checking just first four samples gives us an idea about the final outcome

C	**N**	**C**	**C**	**C**	**C**	**C**	**N**	**C**	**N**

C = 3, N = 1

- If we don’t want to generate and test all 10 samples and if we allow some errors, then we can stop early

Sequential Probability Ratio Test (SPRT)

- Type I/II errors ϵ
- Indifference region $[p_0, p_1]$
- Stopping criteria: $S_{min} = log(\frac{\epsilon}{1-\epsilon})$, $S_{max} = log(\frac{1-\epsilon}{\epsilon})$
- Number of conforming samples seen till now is c
- Number of non-conforming samples seen till now is n
- Sequential probability ratio at each iteration is:

$$S = log(\frac{p_1^c(1-p_1)^n}{p_0^c(1-p_0)^n})$$

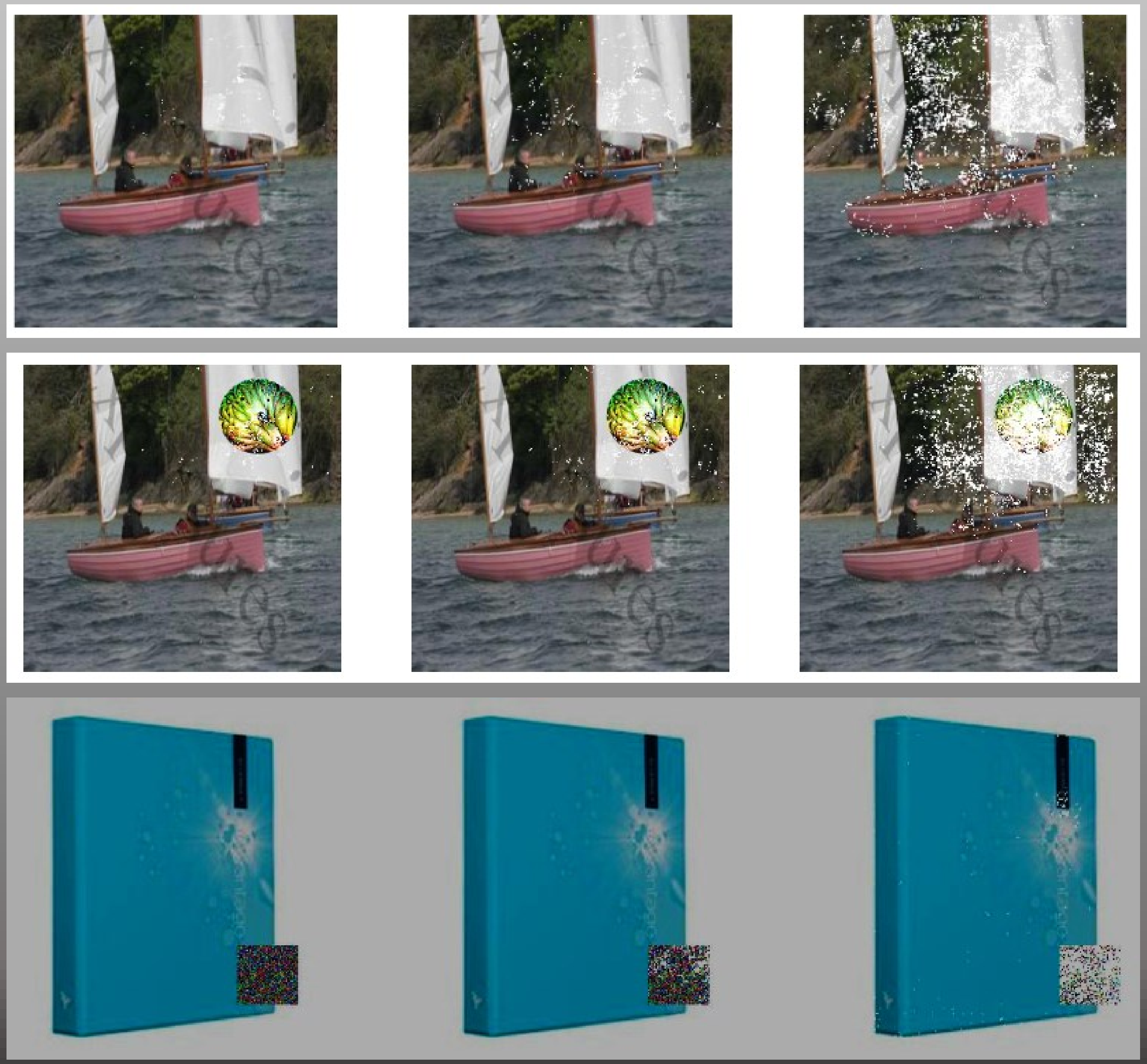
- If sequential probability ratio crosses stopping criteria, then stop sampling and count the sample output

If **you can't measure** it, you can’t manage it!

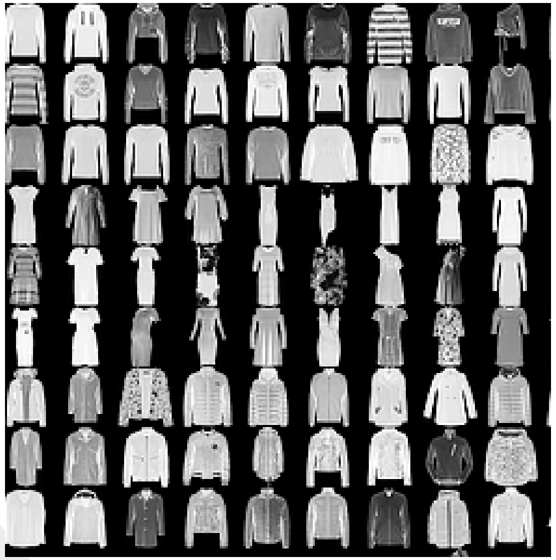
Results

Defense against patch attacks

- (top) Original images do not change label.
- (middle) Removing banana patch generated using adversarial patch attack.
- (bottom) Removing baseball patch generated using LaVAN method.



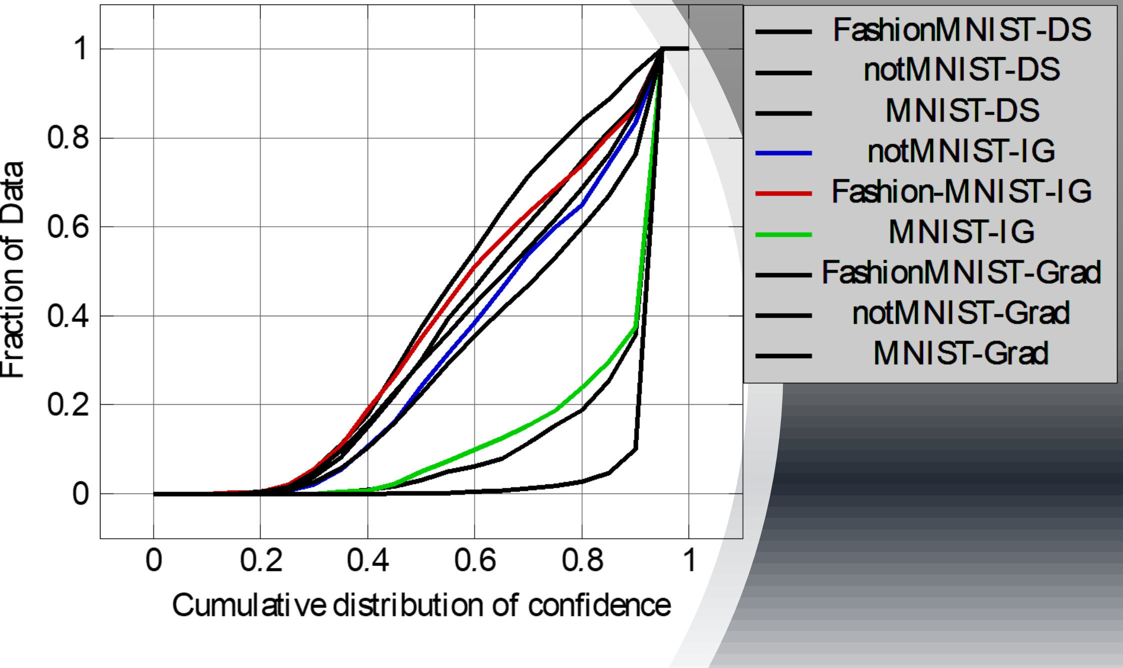
notMNIST



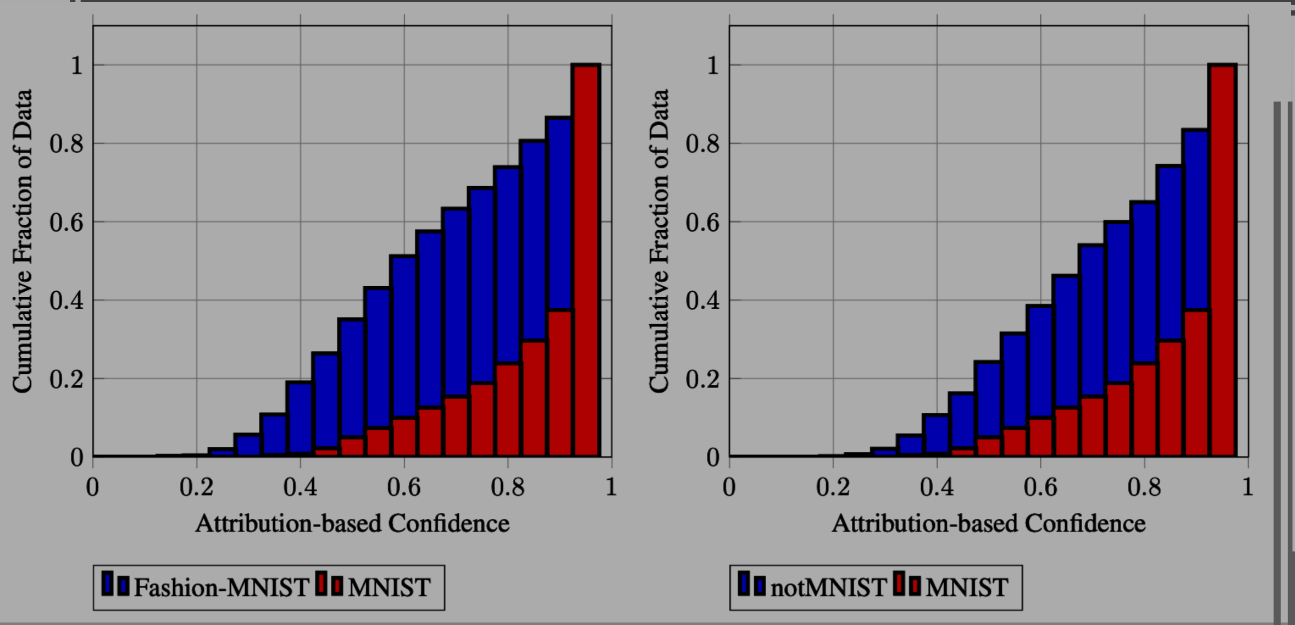
FashionMNIST

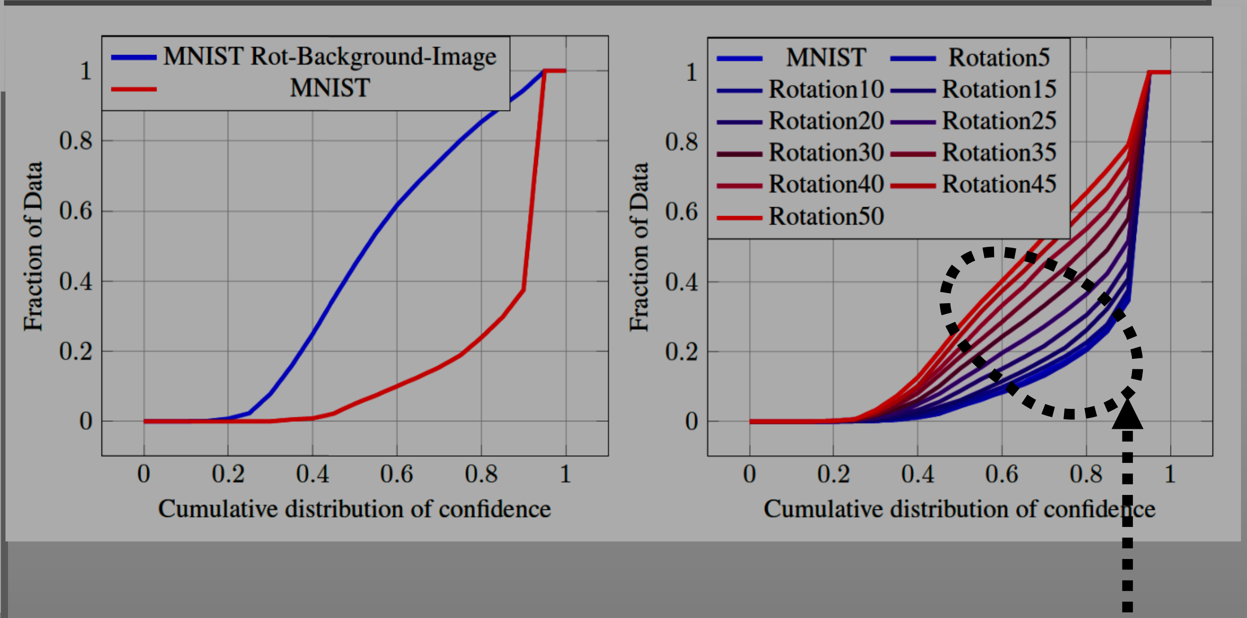
ABC metric for out-of-distribution data

- DNN trained on MNIST makes high confidence predictions for FashionMNIST and notMNIST input.
- Just the DNN prediction does not convey the whole picture.
- ABC metric helps.



ABC metric for out-of-distribution data

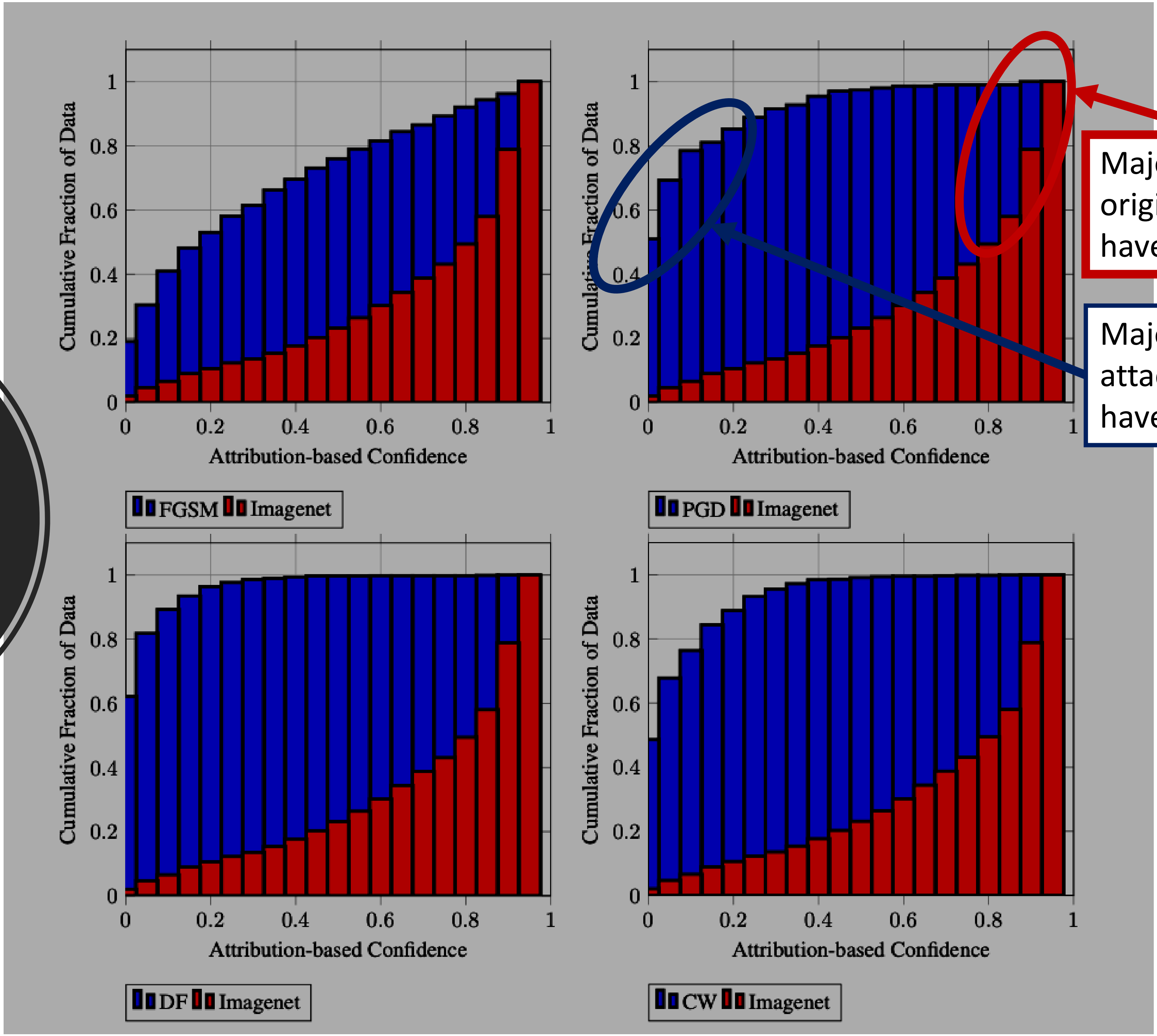




- ABC of in-distribution images is higher than ABC of out-of-distribution images.
- ABC decreases as the rotation angle of MNIST digit increases.

Lines do not cross each other.
ABC metric captures rotation effectively.

ABC metric for FGSM, PGD, DeepFool and CW



Majority of original images have high ABC

Majority of attacked images have low ABC

Future Research

- Different modalities
- Different adversarial attacks
- Sensitivity to attribution errors
- Deeper theoretical connections
    - with Bayesian approaches