A Unified Approach to Interpreting Model Predictions

Scott M. Lundberg
Paul G. Allen School of Computer Science
University of Washington
Seattle, WA 98105
slund1@cs.washington.edu

Su-In Lee
Paul G. Allen School of Computer Science
Department of Genome Sciences
University of Washington
Seattle, WA 98105
suinlee@cs.washington.edu

NIPS 2017

Presenter: Jack Lanchantin

Explanation Models

- *f*: original prediction model to be explained
- *g*: the explanation model.
- Explanation models often use simplified inputs *x'* that map to the original inputs through a mapping function $x = h_x(x')$.

- Focus on **local methods** designed to explain a prediction *f*(*x*) based on a single input *x*.
- Local methods try to ensure $g(z') \approx f(h_x(z'))$ whenever $z' \approx x'$

Additive Feature Attribution Methods

Methods with an explanation model that is a linear function of binary variables:

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i, \tag{1}$$

where $z' \in \{0,1\}^M$, M *is the number of simplified input features, and* $\phi_i \in \mathbb{R}$.

LIME

The loss function to force g to well approximate f

Optional regularization of g

$$\xi = \arg\min_{g \in \mathcal{G}} L(f, g, \pi_{x'}) + \Omega(g)$$

Kernel specifies what 'local' means

A class of interpretable models (linear models)

DeepLIFT

Minimizes the following function:

$$\sum_{i=1}^{n} C_{\Delta x_i \Delta o} = \Delta o,$$

where $o = f(x)$ is the model output, $\Delta o = f(x) - f(r)$, $\Delta x_i = x_i - r_i$, and r is the reference input. If we let $\phi_i = C_{\Delta x_i \Delta o}$ and $\phi_0 = f(r)$, then DeepLIFT's explanation model matches Equation 1 and is thus another additive feature attribution method.

Shapley Value Estimation

- Assigns an importance value to each feature that represents the effect on the model prediction of including that feature.
- Given feature subsets $S \subseteq F$, where F is the set of all features:
  - One model $f_{S \cup \{i\}}$ is trained with feature i present (i not in S)
  - Another model f_S is trained with the feature withheld.

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} \left[f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S) \right]$$

Properties of Additive Feature Attribution

A surprising attribute of the class of additive feature attribution methods is the presence of a single unique solution in this class with three desirable properties.

Property 1: Local Accuracy

$$f(x) = g(x') = \phi_0 + \sum_{i=1}^{M} \phi_i x'_i \tag{5}$$

The explanation model $g(x')$ matches the original model $f(x)$ when $x = h_x(x')$, where $\phi_0 = f(h_x(\mathbf{0}))$ represents the model output with all simplified inputs toggled off (i.e. missing).

Property 2: Missingness

$$x_i' = 0 \implies \phi_i = 0 \tag{6}$$

Missingness constrains features where $x_i' = 0$ *to have no attributed impact.*

Property 3: Consistency

If a model changes so that some z_i's contribution increases or stays the same regardless of the other inputs, that z_i's attribution should not decrease.

Property 3 (Consistency) *Let* $f_x(z') = f(h_x(z'))$ *and* $z' \setminus i$ *denote setting* $z'_i = 0$. *For any two models* f *and* f', *if*

$$f'_x(z') - f'_x(z' \setminus i) \geq f_x(z') - f_x(z' \setminus i) \tag{7}$$

for all inputs $z' \in \{0, 1\}^M$, *then* $\phi_i(f', x) \geq \phi_i(f, x)$.

Unifying Theorem for Properties 1-3

Theorem 1 *Only one possible explanation model g follows Definition 1 and satisfies Properties 1, 2, and 3:*

$$\phi_i(f, x) = \sum_{z' \subseteq x'} \frac{|z'|!(M - |z'| - 1)!}{M!} [f_x(z') - f_x(z' \setminus i)] \tag{8}$$

where $|z'|$ is the number of non-zero entries in z', and $z' \subseteq x'$ represents all z' vectors where the non-zero entries are a subset of the non-zero entries in x'.

SHAP (SHapley Additive exPlanation) Values

$$\phi_i(f, x) = \sum_{z' \subseteq x'} \frac{|z'|!(M - |z'| - 1)!}{M!} [f_x(z') - f_x(z' \setminus i)] \quad (8)$$

- This paper proposes SHAP values as a unified measure of feature importance.
- These are the Shapley values of a conditional expectation function of the original model
- I.e. they are the solution to Equation 8, where $f_x(z') = f(h_x(z')) = E[f(z) \mid z_S]$, and S is the set of non-zero indexes in z'
- SHAP values provide the unique additive feature importance measure that adheres to properties 1-3 and uses conditional expectations to define simplified inputs.

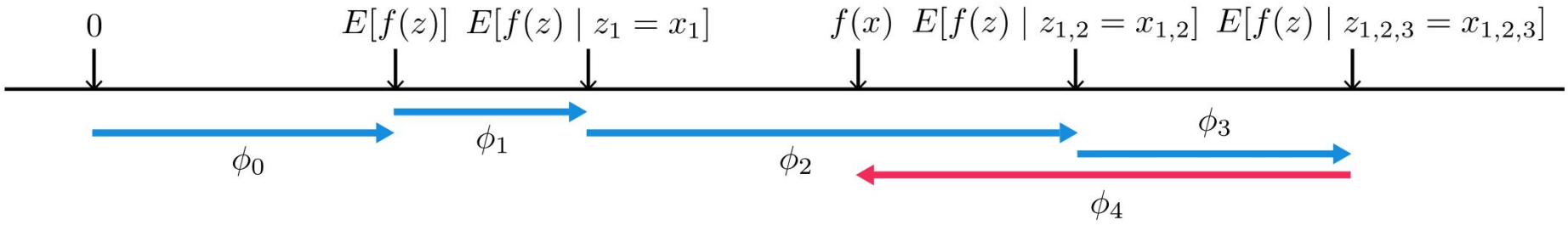
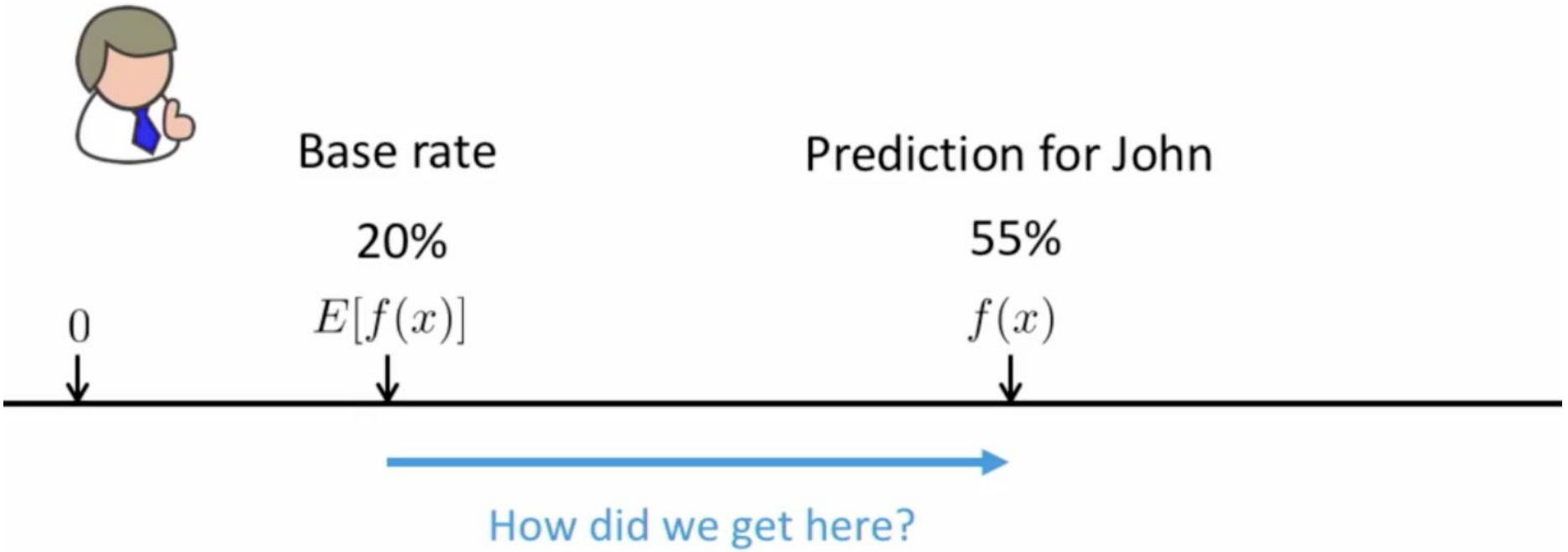
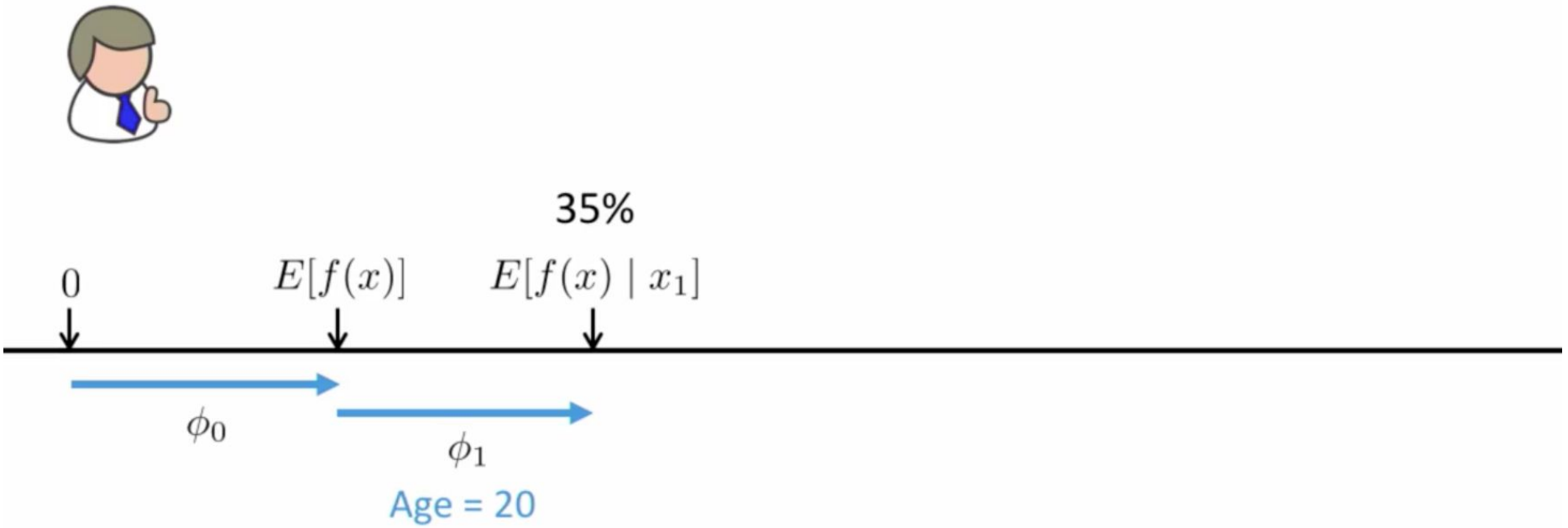


Figure 1: SHAP (SHapley Additive exPlanation) values attribute to each feature the change in the expected model prediction when conditioning on that feature. They explain how to get from the base value $E[f(z)]$ that would be predicted if we did not know any features to the current output $f(x)$. This diagram shows a single ordering. When the model is non-linear or the input features are not independent, however, the order in which features are added to the expectation matters, and the SHAP values arise from averaging the ϕ_i values across all possible orderings.


Base rate
20%
$E[f(x)]$
Prediction for John
55%
$f(x)$
0
How did we get here?

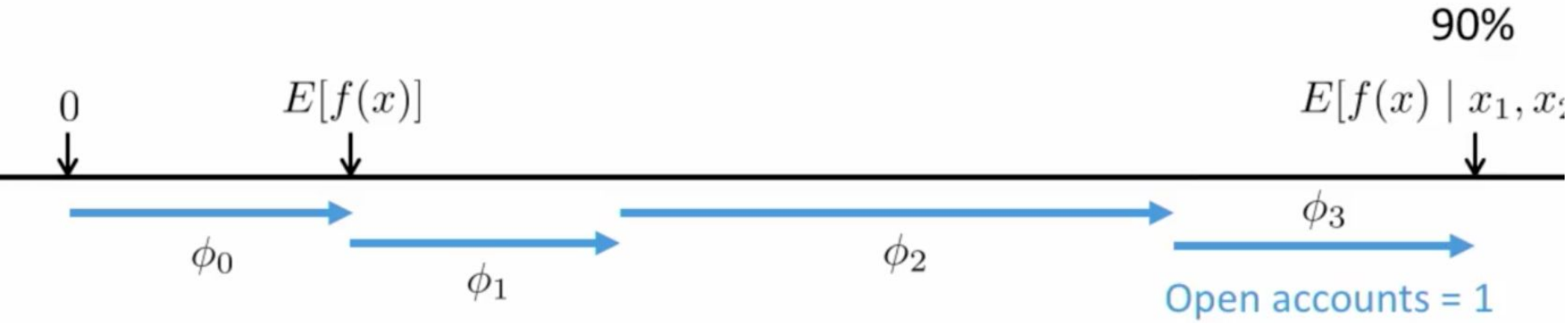

35%
0
$E[f(x)]$
$E[f(x) \mid x_1]$
ϕ_0
ϕ_1
Age = 20

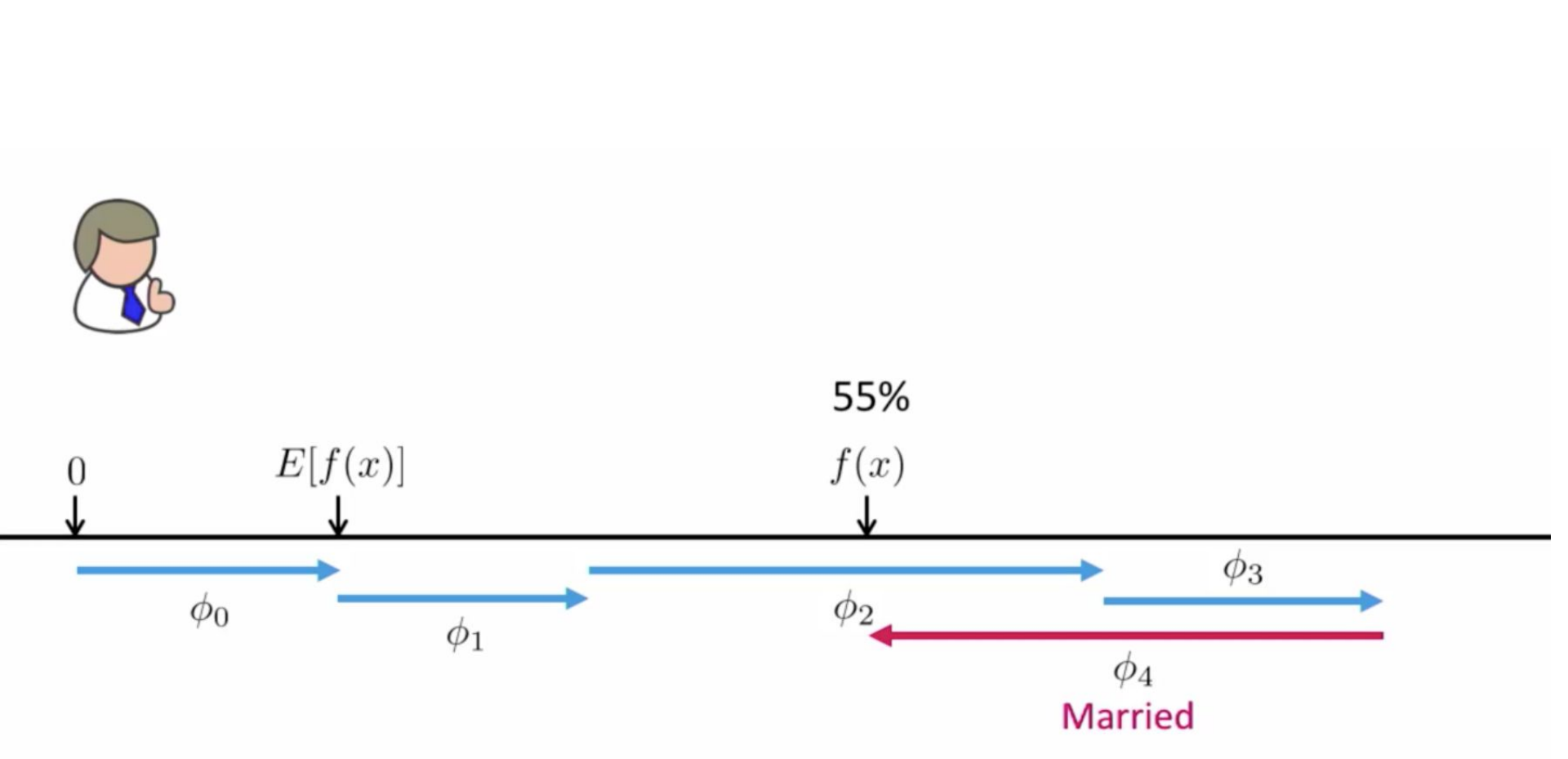
70%

0 $E[f(x)]$ $E[f(x) \mid x_1, x_2]$

ϕ_0 ϕ_1 ϕ_2

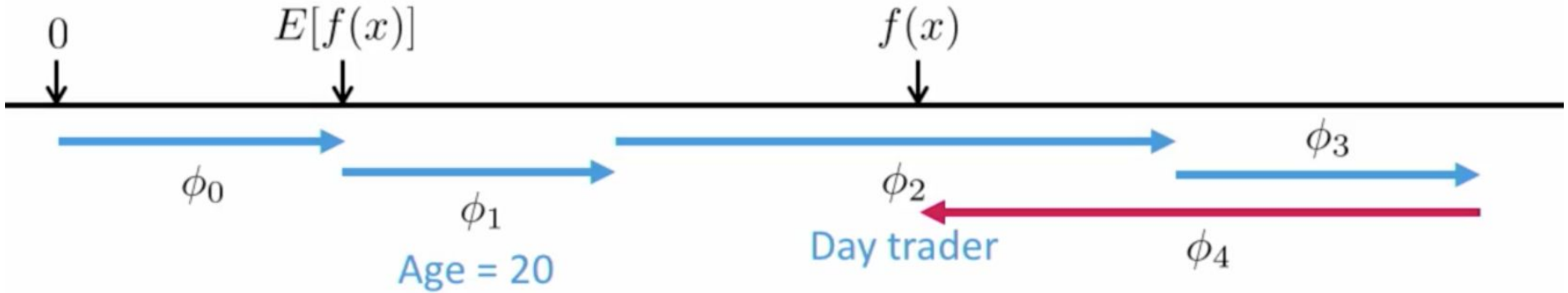
Day trader


90%
0
$E[f(x)]$
$E[f(x) \mid x_1, x_2$
ϕ_0
ϕ_1
ϕ_2
ϕ_3
Open accounts = 1


55%
0
$E[f(x)]$
$f(x)$
ϕ_0
ϕ_1
ϕ_2
ϕ_3
ϕ_4
Married

The order matters!

SHAP values result from averaging over all N! possible orderings.



SHAP (SHapley Additive exPlanation) Values

- Implicit in this definition of SHAP values is a simplified input mapping, $h_x(z') = z_S$ where z_S has missing values for features not in the set S.
- Since most models cannot handle arbitrary patterns of missing input values, we approximate $f(z_S)$ with $E[f(z) \mid z_S]$.

Model-Agnostic SHAP Approximations

1. Shapley sampling values method (previous work)
2. Kernel SHAP

Kernel SHAP (Linear LIME + Shapley values)

$$\xi = \underset{g \in \mathcal{G}}{\arg\min}\ L(f, g, \pi_{x'}) + \Omega(g). \tag{2}$$

Theorem 2 (Shapley kernel) *Under Definition 1, the specific forms of $\pi_{x'}$, L, and Ω that make solutions of Equation 2 consistent with Properties 1 through 3 are:*

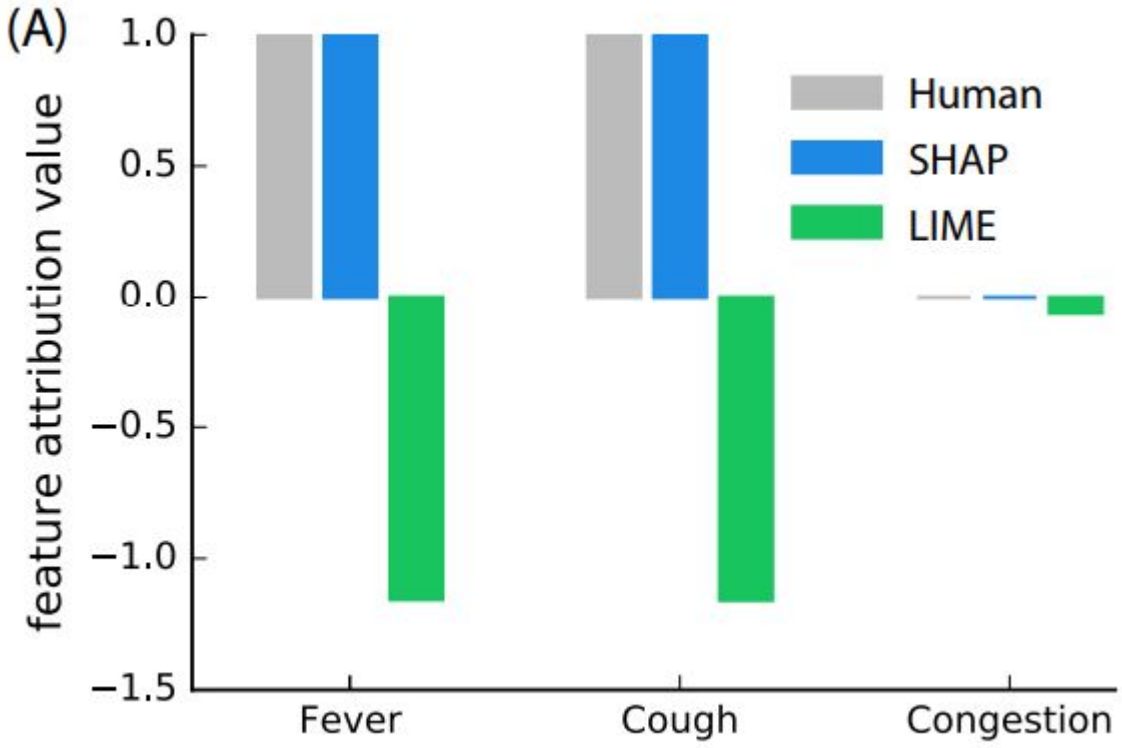
$$\Omega(g) = 0,$$

$$\pi_{x'}(z') = \frac{(M-1)}{(M \text{ choose } |z'|)|z'|(M-|z'|)},$$

$$L(f, g, \pi_{x'}) = \sum_{z' \in Z} \left[f(h_x^{-1}(z')) - g(z')\right]^2 \pi_{x'}(z'),$$

where $|z'|$ is the number of non-zero elements in z'.

Kernel SHAP (Linear LIME + Shapley values)



Model-Specific SHAP Approximations

1. Linear SHAP
2. Deep SHAP

Linear SHAP

For linear models, if we assume input feature independence (Equation 11), SHAP values can be approximated directly from the model's weight coefficients.

Corollary 1 (Linear SHAP) *Given a linear model* $f(x) = \sum_{j=1}^{M} w_j x_j + b$*:* $\phi_0(f, x) = b$ *and*

$$\phi_i(f, x) = w_j(x_j - E[x_j])$$

Deep SHAP (DeepLIFT + Shapley values)

Deep SHAP combines SHAP values computed for smaller components of the network into SHAP values for the whole network. It does so by recursively passing DeepLIFT's multipliers, now defined in terms of SHAP values, backwards through the network

$$m_{x_j f_3} = \frac{\phi_i(f_3, x)}{x_j - E[x_j]} \tag{13}$$

$$\forall_{j \in \{1,2\}} \quad m_{y_i f_j} = \frac{\phi_i(f_j, y)}{y_i - E[y_i]} \tag{14}$$

$$m_{y_i f_3} = \sum_{j=1}^{2} m_{y_i f_j} m_{x_j f_3} \qquad \text{chain rule} \tag{15}$$

$$\phi_i(f_3, y) \approx m_{y_i f_3}(y_i - E[y_i]) \qquad \text{linear approximation} \tag{16}$$

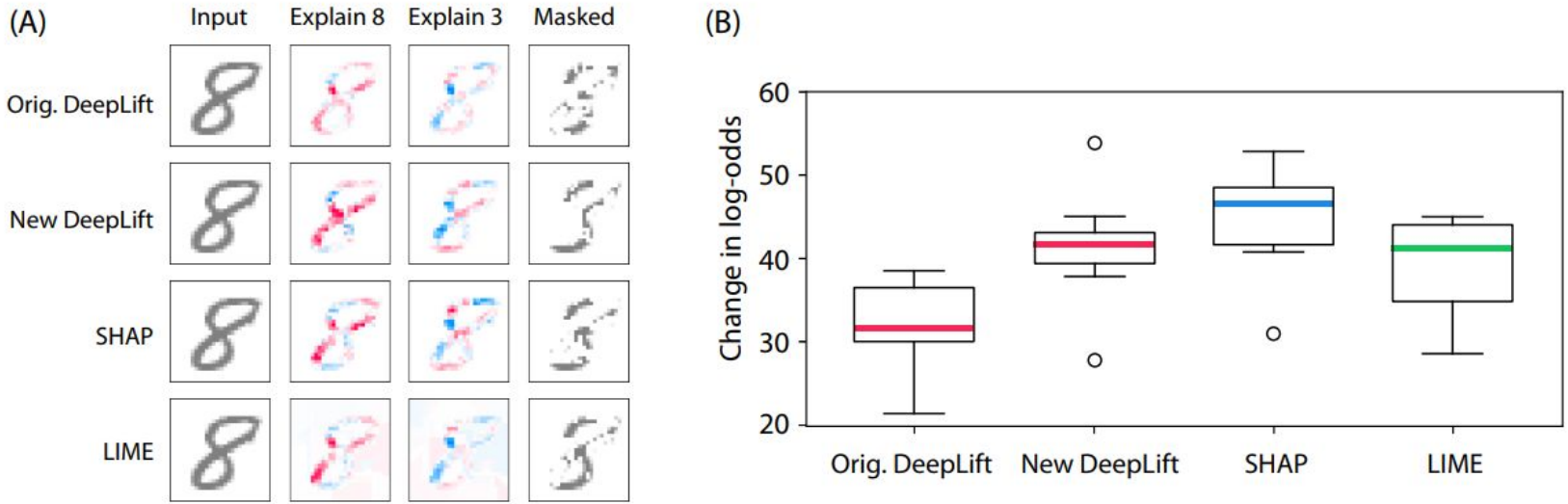


Figure 5: Explaining the output of a convolutional network trained on the MNIST digit dataset. Orig. DeepLIFT has no explicit Shapley approximations, while New DeepLIFT seeks to better approximate Shapley values. (A) Red areas increase the probability of that class, and blue areas decrease the probability. Masked removes pixels in order to go from 8 to 3. (B) The change in log odds when masking over 20 random images supports the use of better estimates of SHAP values.