APPLIED ROOTS

XAI : SHAP → (2017)

Applied AI Course

APPLIED ROOTS

Shapely values

- Co-operative Game Theory
- Lloyd Shapely 1951; Nobel Prize (2012)
- Study in ML context
- Lots of mathematical properties & research (~70yrs)

APPLIED ROOTS

Game - Theory

Players: $f_1, f_2, f_3, \ldots f_d$ (features)

Game → co-operatively work together

→ generate a model output

[GAIN/profit]

APPLIED ROOTS

Shapely value

- measures how much has each player (feature) contributed the profit (output)

- Mathematically rigorous
- Intuition ↓

https://towardsdatascience.com/shap-explained-the-way-i-wish-someone-explained-it-to-me-ab81cc69ef30

APPLIED ROOTS

Intuition : Regression Problem

Model Predict Salary

features Age, gender, Job

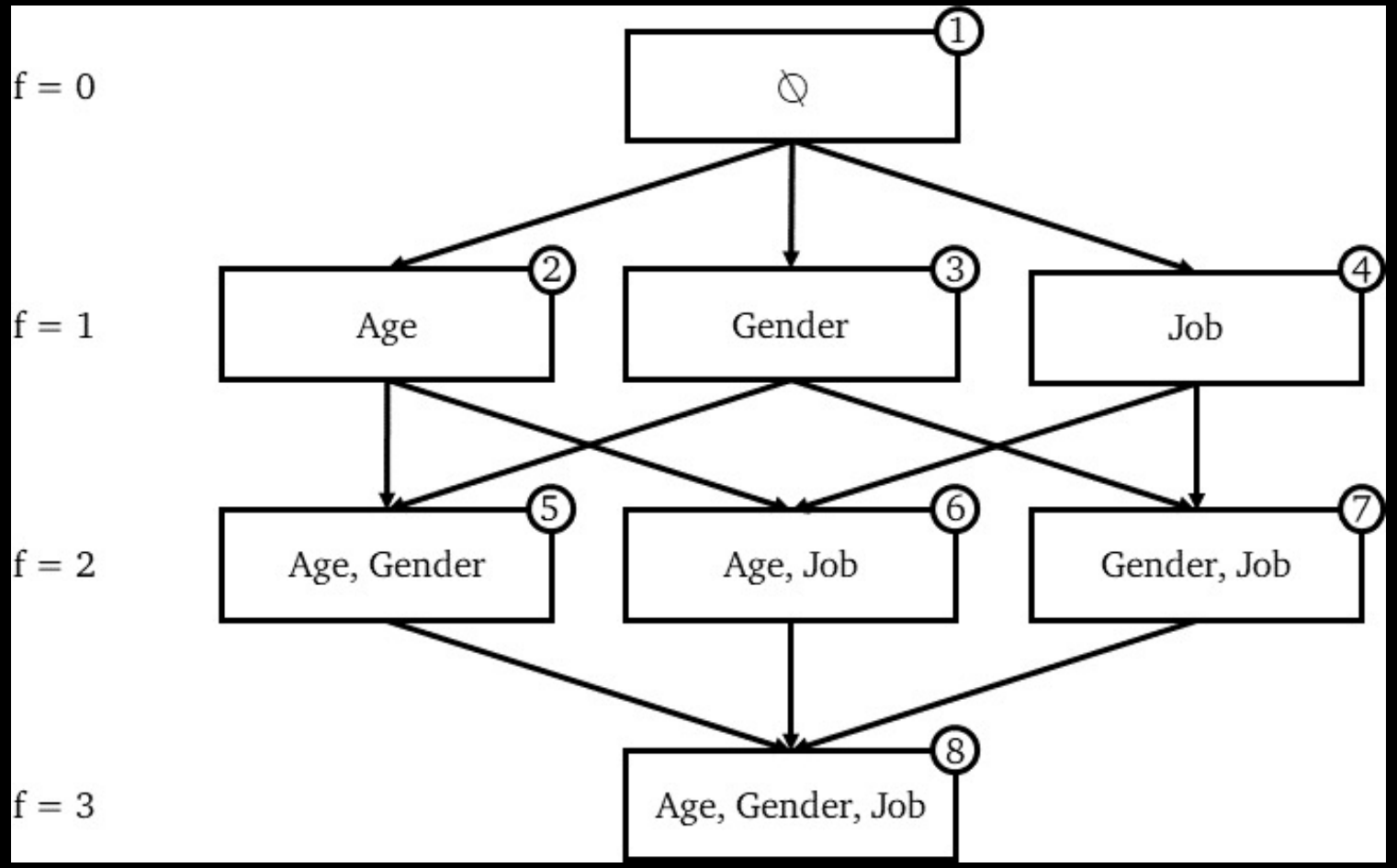
output y_i

$\bar{y}$ = avg of y_i's

x_0 : Query point

$\hat{y}_0$: predicted value for x_0

APPLIED ROOTS

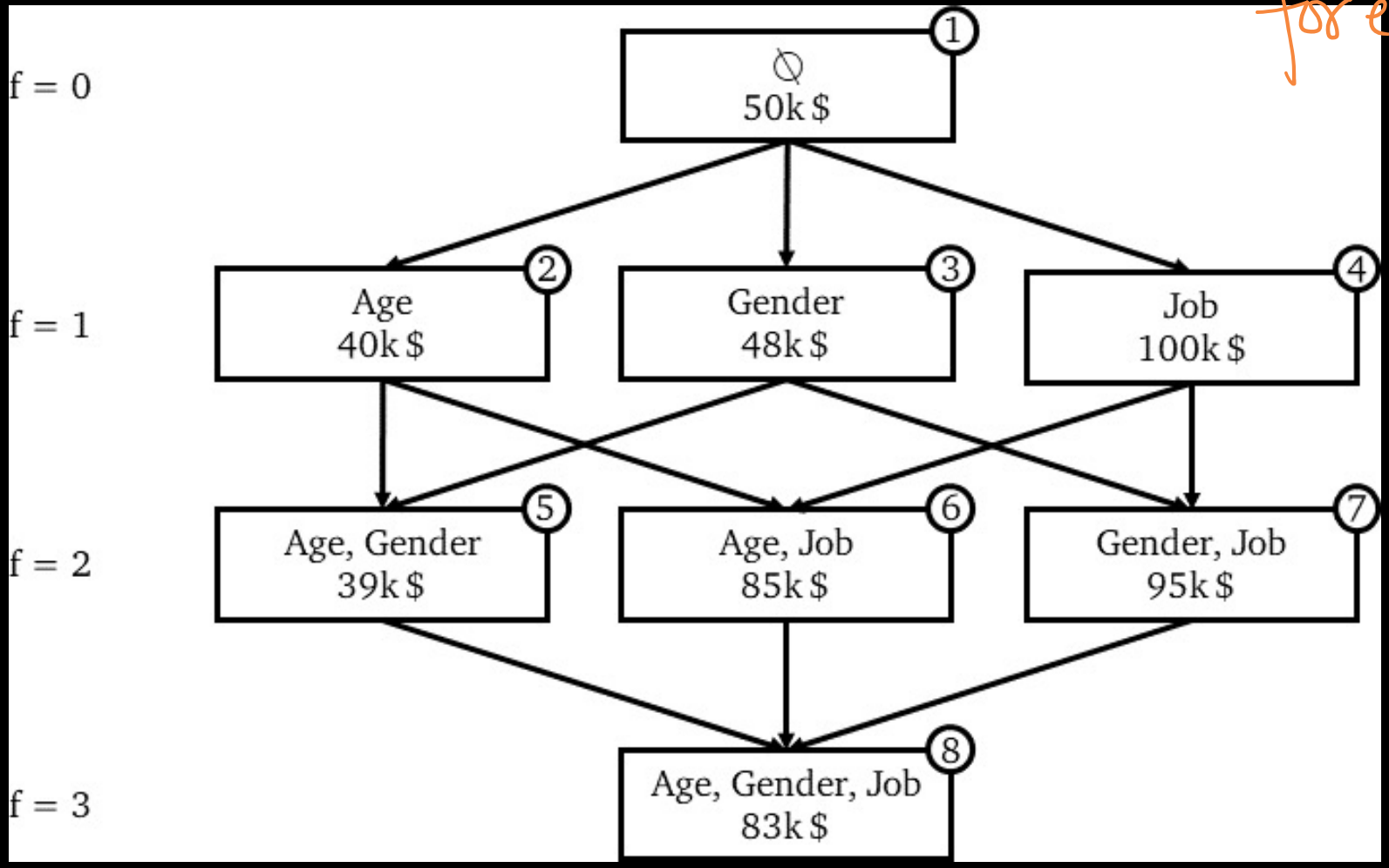


d- features

2^d - models

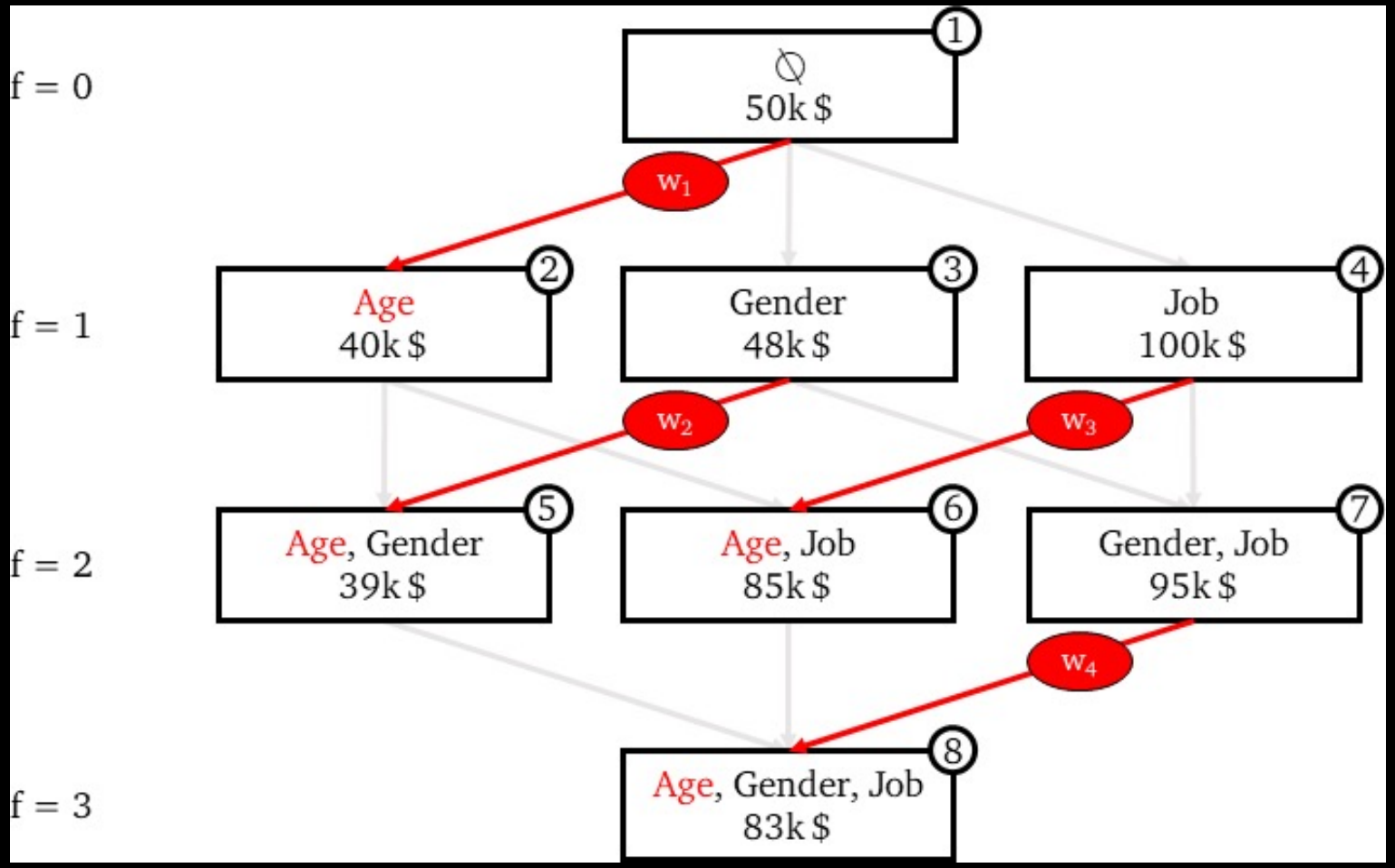
APPLIED ROOTS

$\hat{y}_q$ given x_q for each of the 2^d models



APPLIED ROOTS

Marginal Contribution of a feature (say Age)

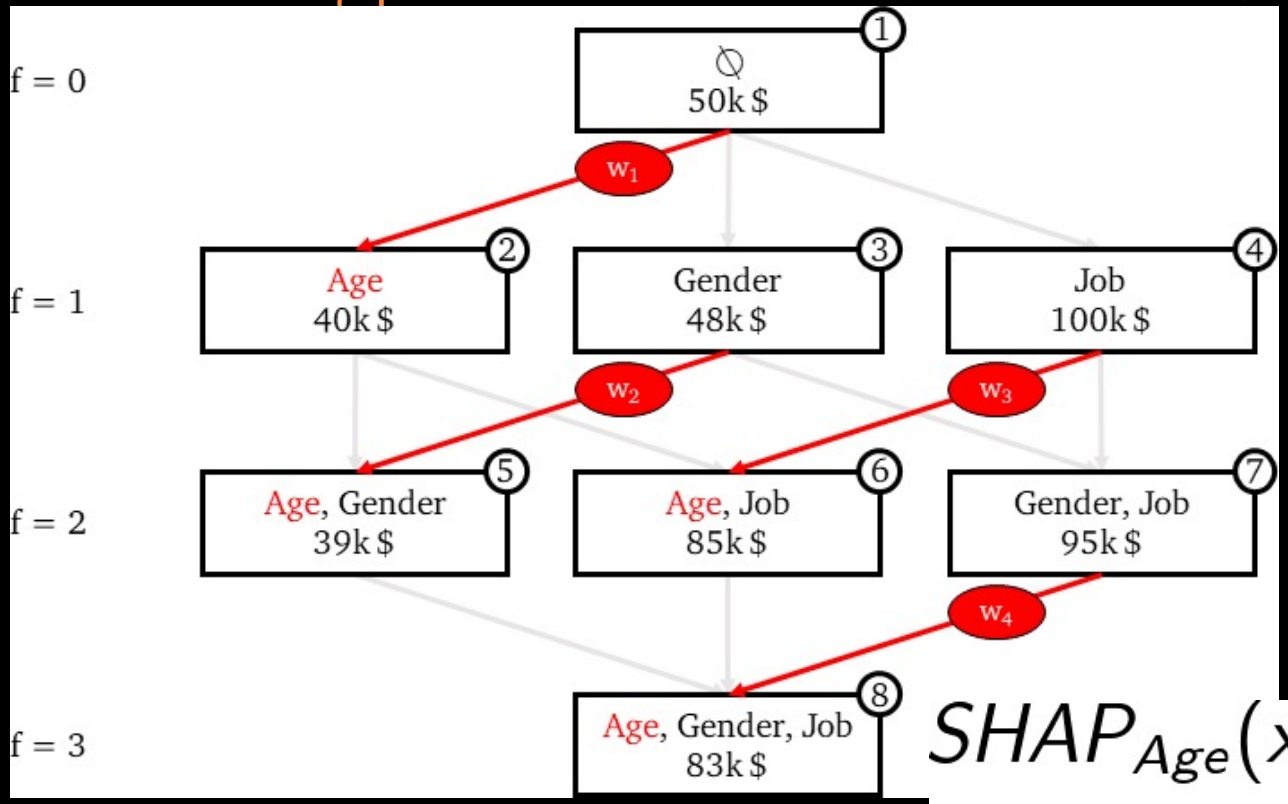


$MC_{Age, \{Age\}}(x_0) = -10k$

$MC_{Age, \{Age, Gender\}}(x_0) = -9k$

APPLIED ROOTS

Marginal contribution of a feature (say Age)



$$SHAP_{Age}(x_0) = w_1 \times MC_{Age,\{Age\}}(x_0) + w_2 \times MC_{Age,\{Age,Gender\}}(x_0) + w_3 \times MC_{Age,\{Age,Job\}}(x_0) + w_4 \times MC_{Age,\{Age,Gender,Job\}}(x_0)$$

APPLIED ROOTS

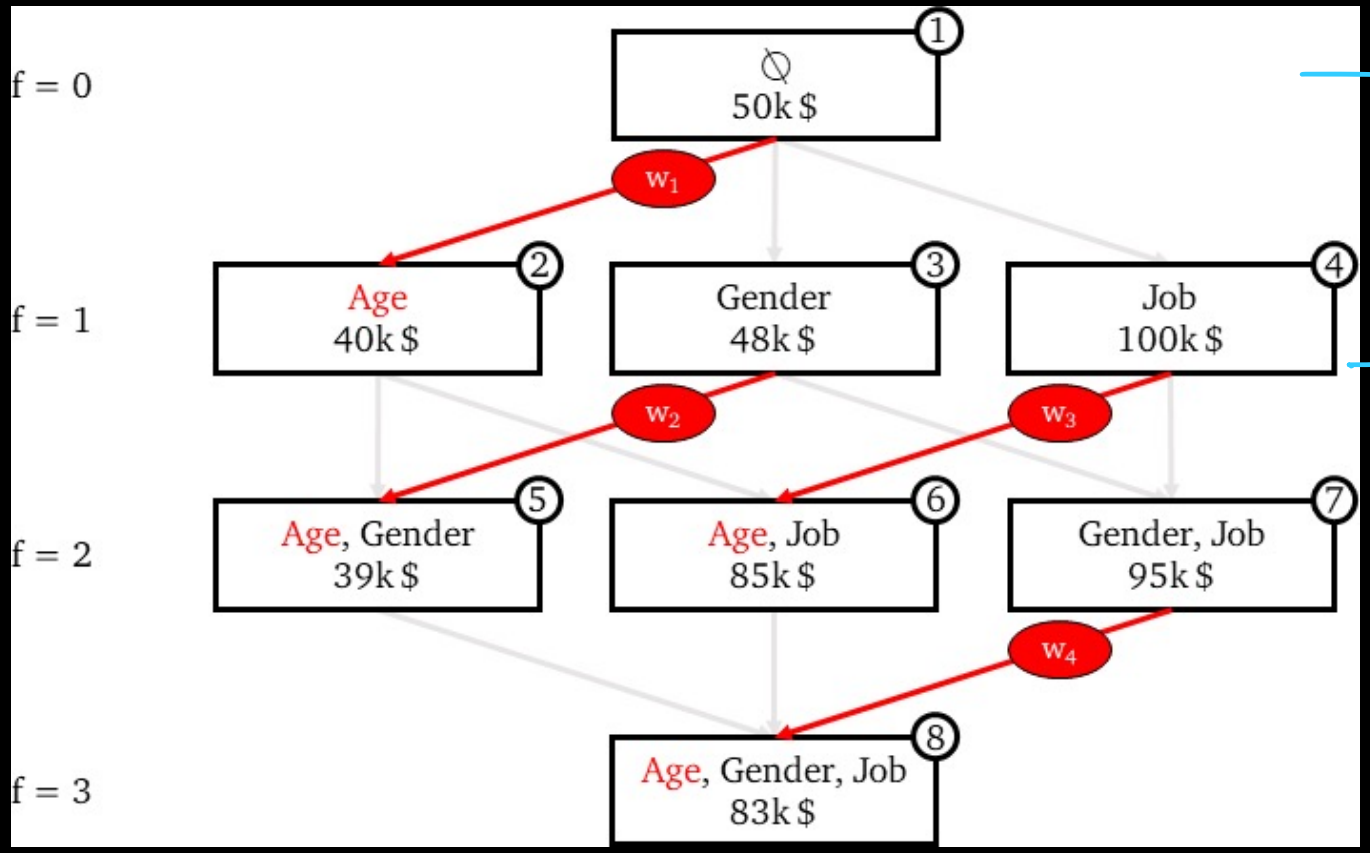
$$
\begin{aligned}
SHAP_{Age}(x_0) = \; & w_1 \times MC_{Age,\{Age\}}(x_0) + \\
& w_2 \times MC_{Age,\{Age,Gender\}}(x_0) + \\
& w_3 \times MC_{Age,\{Age,Job\}}(x_0) + \\
& w_4 \times MC_{Age,\{Age,Gender,Job\}}(x_0)
\end{aligned}
$$

$w_1 + w_2 + w_3 + w_4 = 1$

How to compute the weights?

APPLIED ROOTS

$w_1 + w_2 + w_3 + w_4 = 1$



w_1

||

$w_2 + w_3$

||

w_4

& $w_2 = w_3$

$$\Rightarrow w_1 = 1/3$$

$$w_2 = w_3 = 1/6$$

$$w_4 = 1/3$$

APPLIED ROOTS

Fcf

	N. of Nodes $\binom{F}{f}$	N. of Edges $f \times \binom{F}{f}$
f = 0	1	
f = 1		3
	3	
f = 2		6
	3	
f = 3		3
	1	
Sum	$2^F=8$	$F\times2^{F-1}=12$

APPLIED ROOTS

$$
\begin{aligned}
SHAP_{Age}(x_0) = & \left[\left(1 \times \binom{3}{1}\right]^{-1} \times MC_{Age,\{Age\}}(x_0) \; + \right. \\
& \left[\left(2 \times \binom{3}{2}\right]^{-1} \times MC_{Age,\{Age,Gender\}}(x_0) \; + \right. \\
& \left[\left(2 \times \binom{3}{2}\right]^{-1} \times MC_{Age,\{Age,Job\}}(x_0) \; + \right. \\
& \left[\left(3 \times \binom{3}{3}\right]^{-1} \times MC_{Age,\{Age,Gender,Job\}}(x_0) \; + \right. \\
= & \frac{1}{3} \times (-10k\$) + \frac{1}{6} \times (-9k\$) + \frac{1}{6} \times (-15k\$) + \frac{1}{3} \times (-12k\$) \\
= & -11.33k\$
\end{aligned}
$$

APPLIED ROOTS

$$SHAP_{feature}(x) = \sum_{set:feature \in set} [|set| \times \binom{F}{|set|}]^{-1} [Predict_{set}(x) - Predict_{set \setminus feature}(x)]$$

https://towardsdatascience.com/shap-explained-the-way-i-wish-someone-explained-it-to-me-ab81cc69ef30

APPLIED ROOTS

SHapely Additive exPlanations → SHAP

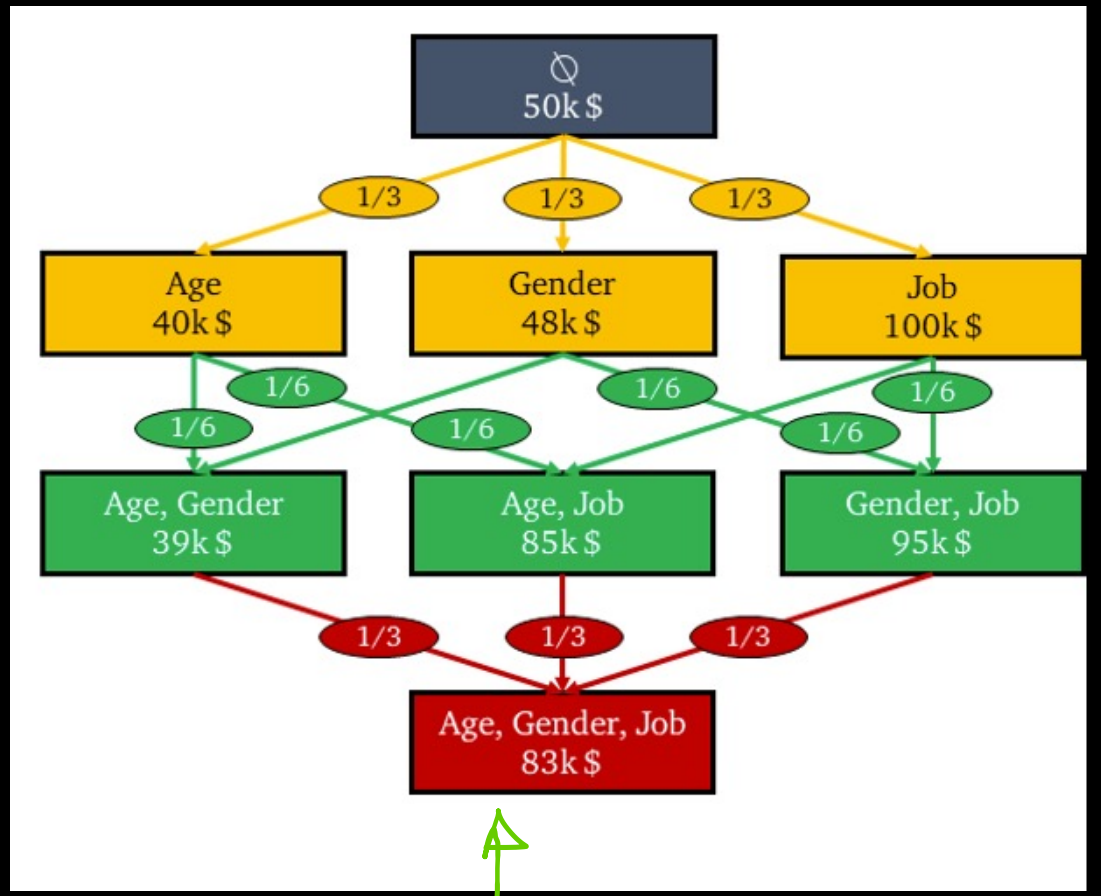
$SHAP_{Age}(x_0) = -11.33\,k$

$SHAP_{Gender}(x_0) = -2.33\,k$

$SHAP_{Job}(x_0) = +46.66\,k$

$SUM = 33\,k$

83 - 50



Sum of shapely values

$$= \hat{y}_F - \hat{y}_\phi$$

ϕ → no-features

F → all features

APPLIED ROOTS

Setup of explainability → same as LIME

$x \longrightarrow x' \in \{0,1\}^M$: interpretable representation

$x = h_x(x')$

$g(x')$: Surrogate model

$g(x') \approx f(h_x(x'))$

APPLIED ROOTS

Additive Feature Attribution:

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

→ #dim in interpretable representation

ϕ_i ← effect of feature i ← Shapely values

Linearity

APPLIED ROOTS

Expectations in Probability

$$E(X) = \sum_x x \cdot P(X=x) \quad \text{(or)} \quad \int_x x \cdot P(x)\, dx$$

$P(X=x)$ → PMF

$P(x)$ → PDF

↳ expected or average value of X

APPLIED ROOTS

$$E(f(x)) = \sum_{x} f(x) \cdot P(X=x) \quad \text{(or)} \quad \int_{x} f(x)\, P(x)\, dx$$

Say. X^2

APPLIED ROOTS

$$E(X|Y=y_1) = \sum_x x \cdot P(X|Y=y_1)$$

Example:

https://en.wikipedia.org/wiki/Conditional_expectation#Example_1:_Dice_rolling

APPLIED ROOTS

Expectations & Shapely values.

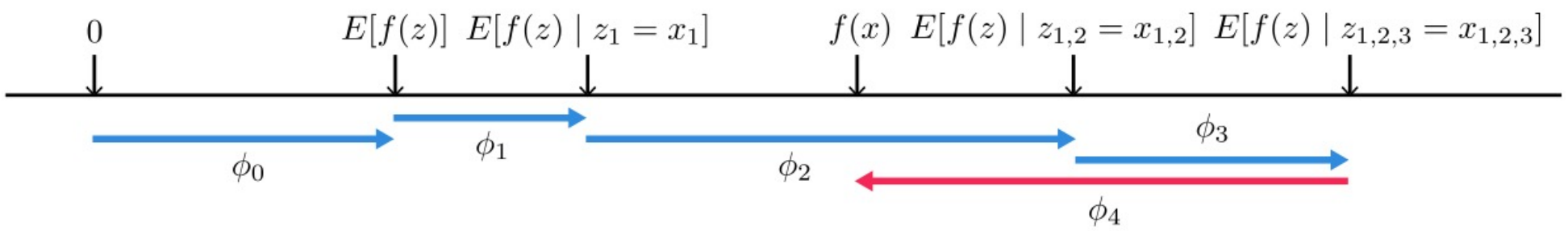


Figure 1: SHAP (SHapley Additive exPlanation) values attribute to each feature the change in the expected model prediction when conditioning on that feature. They explain how to get from the base value $E[f(z)]$ that would be predicted if we did not know any features to the current output $f(x)$. This 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.

https://arxiv.org/pdf/1705.07874.pdf

APPLIED ROOTS

Computing shapely values = computing conditional expectations

Very time consuming exp-time-complexity [NP-Hard]

Lots of approximate Algorithms

APPLIED ROOTS

Kernel SHAP ⟷ LIME + shapley values.

↳ as LIME uses a $\Pi_{x'}(\ )$: local kernel

→ Model agnostic

APPLIED ROOTS

Kernel SHAP

$$\pi_x(z') = \frac{M - 1}{\binom{M}{|z'|} |z'| \left(M - |z'|\right)}$$

Theorem (2)

M → # dim in z'

$|z'|$ → # features present in z'

VS LIME

APPLIED ROOTS

$$\pi_x(z') = \frac{M-1}{\binom{M}{|z'|}\,|z'|\,(M-|z'|)}$$

constant

intuition

- $|z'|$: v.small or v.large $\Rightarrow$ $\pi_x(z')$: larger
- $|z'|$: in between $\Rightarrow$ $\pi_x(z')$: smaller

APPLIED ROOTS

Model specific approximate Shapley computation

↳ TreeSHAP (2018)

↳ fast & model specific (RF, GBDT..)

APPLIED ROOTS

Pros

(+) Solid Theory

(+) LIME + SHAP

(+) Tree-SHAP is fast

Cons

(−) Kernel SHAP is slow

(−) Tree SHAP's unintuitive results

(−) Approximations (ignores feature dependence)

APPLIED ROOTS

Code: https://shap.readthedocs.io/en/latest/index.html

Q&A