WAYNE STATE UNIVERSITY

Interpretable Machine Learning (Explainable AI)

Dr. Ratna Babu Chinnam

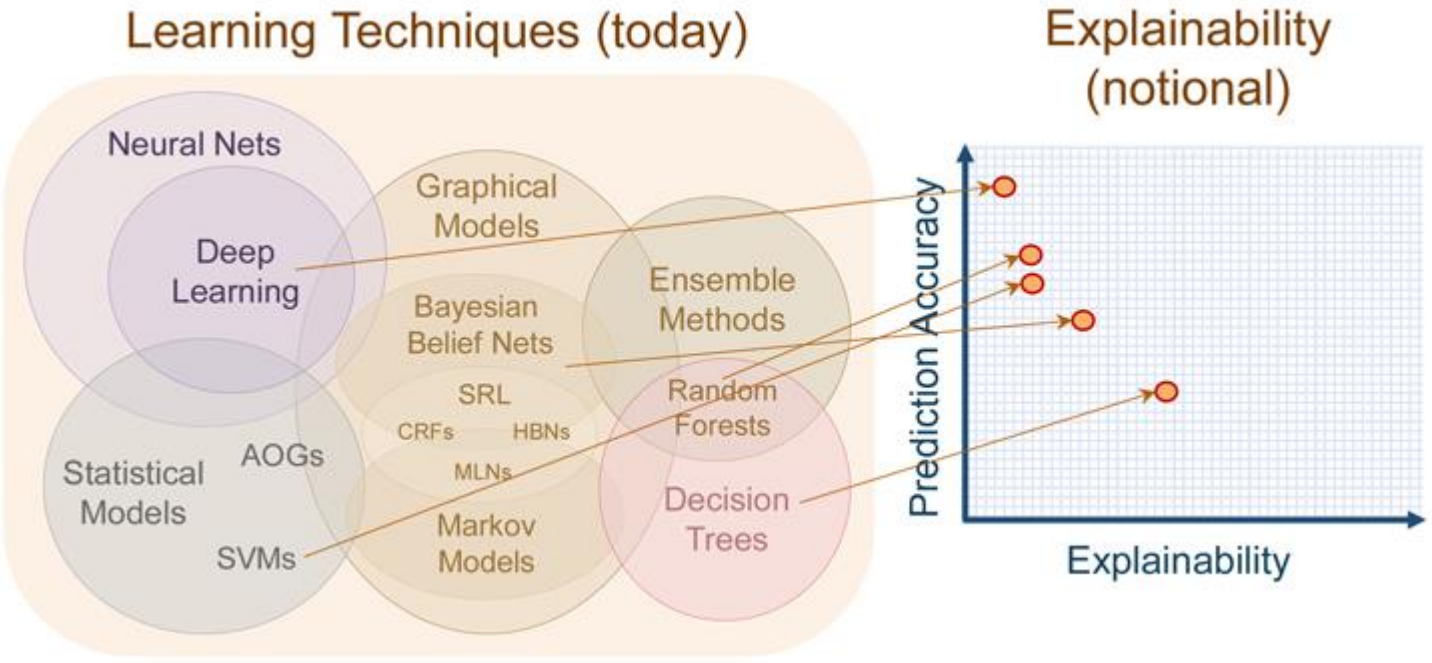
Industrial & Systems Engineering

Wayne State University

Primary Source: Manojit Nandi | Link | Contact

What is Interpretability?

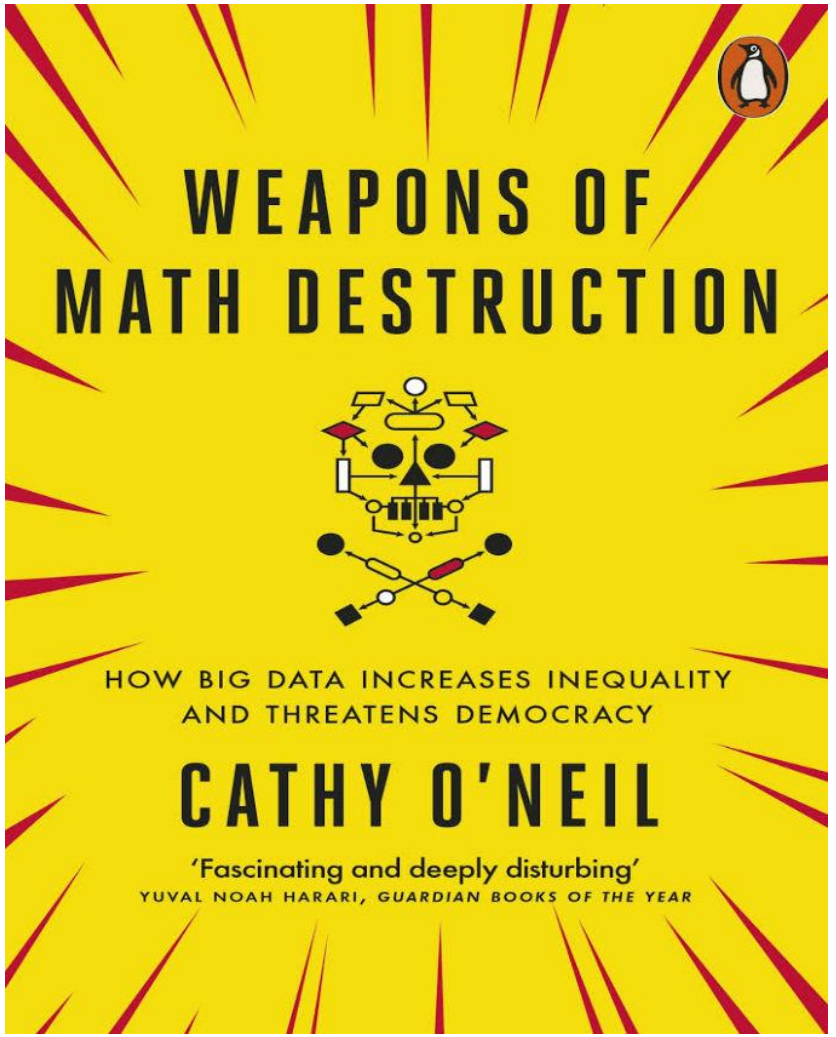
- **Interpretability** is defined as the "*ability to explain how a statistical or machine learning model came to a particular outcome based on the inputs*".
- Generally, techniques with higher predictive accuracy tend to be harder to interpret.



Source: Bornstein - Is A.I. Permanently Inscrutable | Link

Fairness, Accountability, and Transparency in ML

- Here are some things currently determined by statistical models:
  1. Whether a prisoner will be released on parole based on likelihood to reoffend?
  2. Who gets a credit loan from a bank or other lending institution?
  3. Whether a teacher will be fired based on a teaching evaluation "score"?



Benefits of Explainable Models

- If output of machine learning model can be explained, then we can:
  1. **Validate Fairness**: Check if vulnerable groups are disparately impacted by the algorithm.
  2. **"De-bug" Models**: Identify reasons for systematic errors in your model.
  3. **Contestability**: We can only dispute the results of a model if we understand how it came to a decision.

(a) Husky classified as wolf (b) Explanation

Source: Ribeiro et. al (2016) | Paper

GDPR Article 22 and the "Right to Explanation"

- EU General Data Protection Regulations (GDPR), Article 22 states any EU **citizen can opt out of automated decision-making**.
  - Automated individual decision-making is a decision made by automated means without any human involvement. Examples:
    - an online decision to award a loan; and
    - a recruitment aptitude test using per-programmed algorithms and criteria.
  - Automated individual decision-making does not have to involve profiling, although it often will.
  - Profiling is "any form of automated processing of personal data consisting of the use of personal data to evaluate certain personal aspects relating to a natural person, in particular to analyze or predict aspects concerning that natural person's performance at work, economic situation, health, personal preferences, interests, reliability, behavior, location or movements." (see Article 4(4) of GDPR).
- Article 12 **allows individuals to inquire as to why a particular algorithmic decision was made for them**.

Article 22 (GDPR): "Automated individual decision-making, including profiling"

The data subject shall have the right not to be subject to a decision based solely on automated processing, including profiling, which produces legal effects concerning him or her or similarly significantly affects him or her.

WAYNE STATE UNIVERSITY

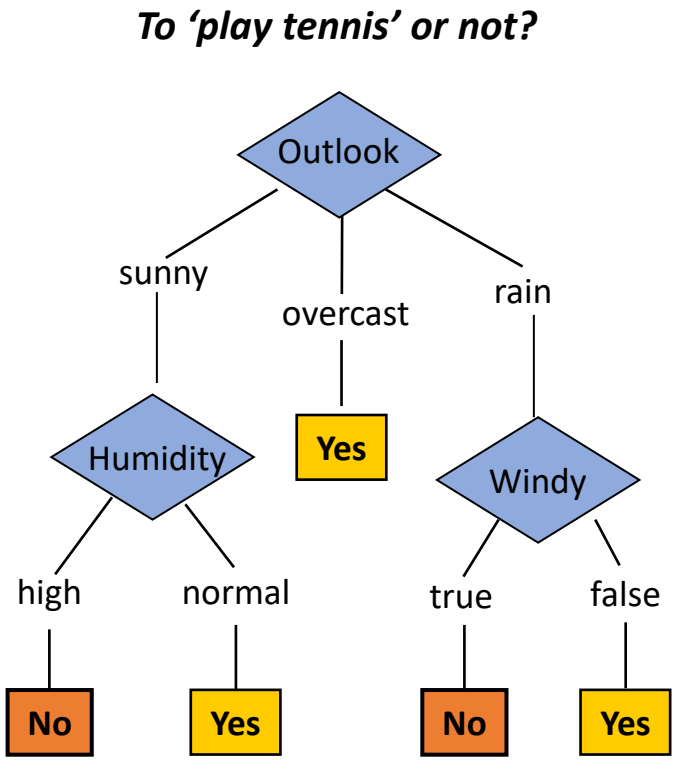
LIME: Locally Interpretable Model-Agnostic Explanations

Marco Tulio Ribeiro, Sameer Singh, Carlos Guestr, University of Washington

Paper | GitHub

Interpretable Models

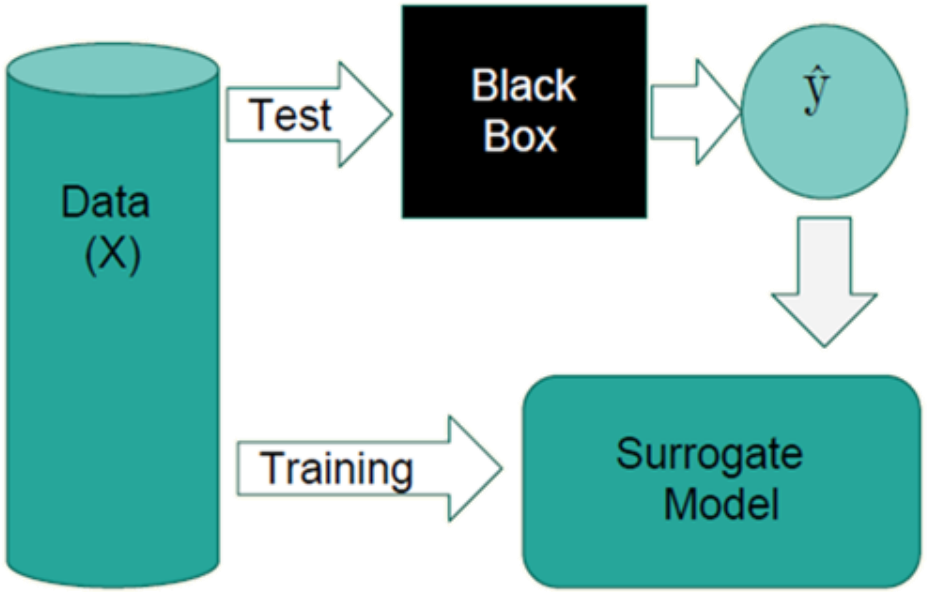
- **Linear models are more interpretable by humans.**
- **Linear Regression, Logistic Regression**: Coefficients learned by the model tell us the expected change in Y, given a change in the input X.
- **Decision Trees**: Branches of the trees tell us the order the features were evaluated and what the threshold was.



Decision Tree Example

Surrogate Models

- Start with some “black-box” predictive model.
- On some dataset X, get the predictions $\hat{y}$ from the model.
- Train an “interpretable surrogate model” on the dataset X and the model predictions $\hat{y}$.
- Rely on surrogate model for explanations.



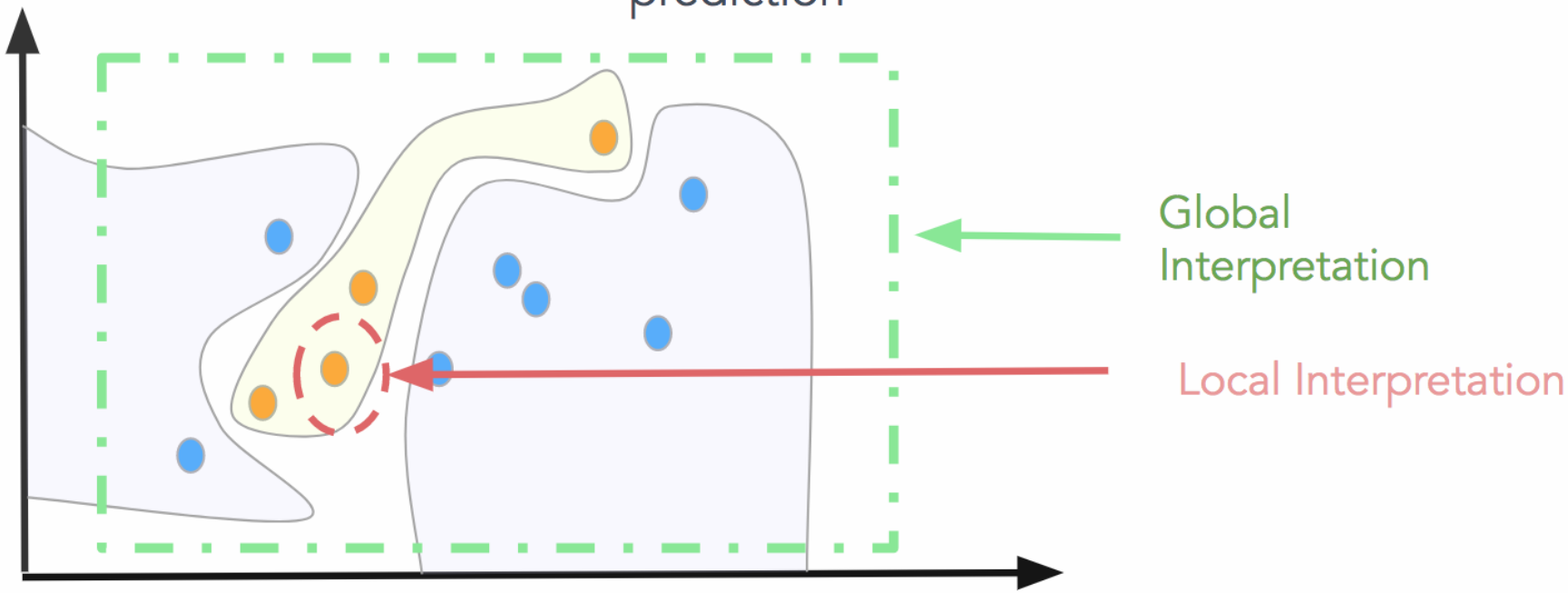
Global vs Local Interpretations

Global Interpretation

Being able to explain the conditional interaction between dependent(*response*) variables and independent(*predictor, or explanatory*) variables based on the complete dataset

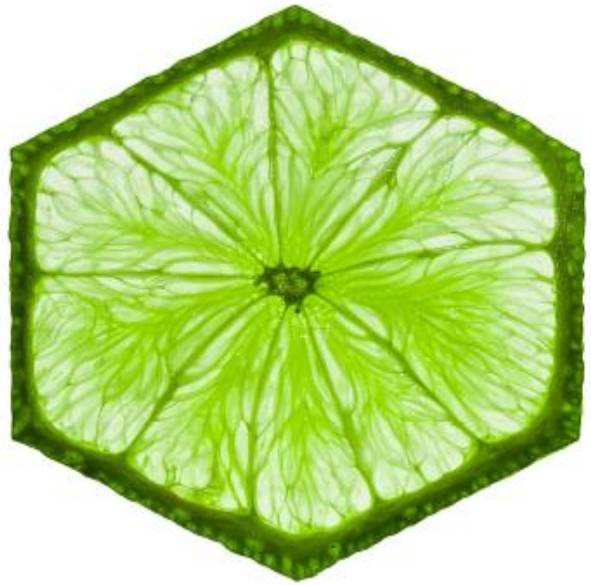
Local Interpretation

Being able to explain the conditional interaction between dependent(*response*) variables and independent(*predictor, or explanatory*) variables wrt to a single prediction



LIME

- With surrogate models, we use interpretable linear models to estimate the “average effects” of a black-box model (Global Interpretability).
- Rather than caring about general effects of different features, we aim to understand why the model made a particular decision (Local Interpretability)?
- **LIME library allows us to create local explanations for a test point**: https://github.com/marcotcr/lime



LIME: Methodology

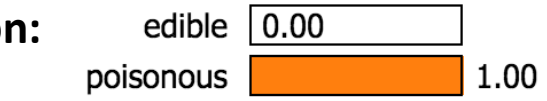
- Choose some point x whose output you wish to explain and get the model's prediction $\hat{y}$.
- Sample new points by perturbing the point x. Let's call these points X'.
- Evaluate these points with your black-box model. Call these predictions Y'.
- Now fit some interpretable local model on the sampled points X' and their associated predictions Y'.

Is "mushroom" poisonous?

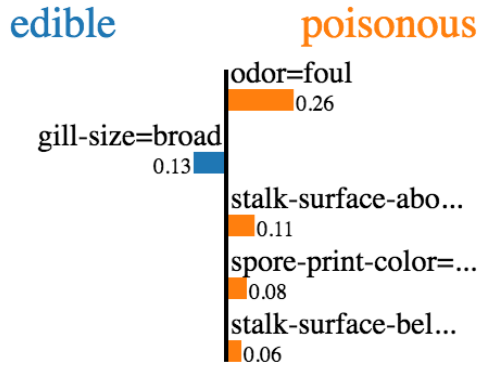
Input:

Feature	Value
odor=foul	True
gill-size=broad	True
stalk-surface-above-ring=silky	True
spore-print-color=chocolate	True
stalk-surface-below-ring=silky	True

Prediction:



Explanation:



Source: LIME | Link

WAYNE STATE UNIVERSITY

SHAP: SHapley Additive exPlanations

Scott Lundberg, Gabriel G. Erion, Su-In Lee, University of Washington

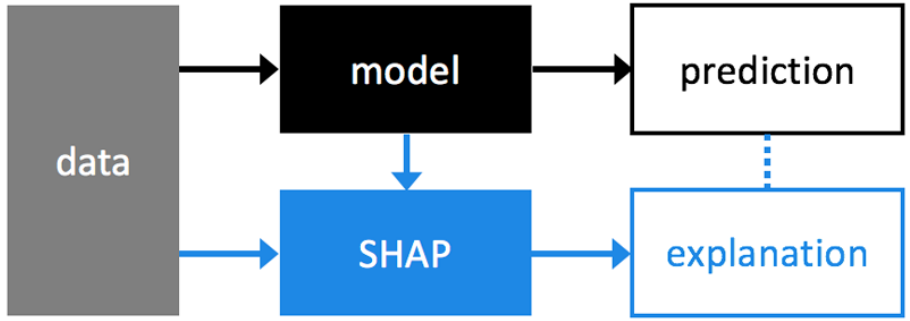
Paper | GitHub

Co-operative Game Theory

- **Game Theory**: Branch of micro-economics dealing with interactions between decision-making agents.
- **Cooperative Game Theory**: Sub-field of game theory where players are “working together” to achieve a common goal.
- **Shapely Value**: Helps determine a “payoff for each player” when each player might have “contributed more or less” than others.
- **Insight for ML**: We can view the features of the model as the “players”, and the outcome of the model (prediction) as the “game’s result”.

Shapley Values

- **Definition**: Is the expected marginal contribution of a player after all possible combinations have been considered.
- **Question**: How to measure each player's contribution to the team's outcome?
- **Heuristic**: If we remove a player from the team and the outcome doesn't change, then the player wasn't "useful".



- To compute Shapley value for each player, we compute each outcome where the player was present and compare it to the outcome where the player was not present.
  - Player i present vs. Player i not present

"Deep Learning Model Development Team" Example

- **Target**: Deliver a DL model that needs 100 lines of code
- **Development Team**: Has three data scientists (L, M, N)
  - Must work together to deliver the project

V(X)	Line of codes
L	10
M	30
N	5
L, M	50
L, N	40
M, N	35
L, M, N	100

Given: Contribution Among Different Coalitions

Order	L Contribution	M Contribution	N Contribution
L, M, N	V(L) = 10	V(L,M) – V(L) = 50 – 10 = 40	V(L,M,N) – V(L,M) = 100 – 50 = 50
L, N, M	V(L) = 10	V(L,M,N) – V(L, N) = 100 – 40 = 60	V(L,N) – V(L) = 40 – 10 = 30
M, L, N	V(L,M) – V(M) = 50 – 30 = 20	V(M) = 30	V(L,M,N) – V(L,M) = 100 – 50 = 50
M, N, L	V(L,M,N) – V(M,N) = 100 – 35 = 65	V(M) = 30	V(M,N) – V(M) = 35 – 30 = 5
N, L, M	V(L,N) – V(L) = 40 – 5 = 35	V(L,M,N) – V(L,N) = 100 – 40 = 60	V(N) = 5
N, M, L	V(L,M,N) – V(M,N) = 100 – 35 = 65	V(M,N) – V(N) = 35 – 5 = 30	V(N) = 5

Marginal Contribution by different orders (3 players, hence 3! orders)

Contributor	Shapley Calculation	Shapley Value
L	1/6(10+10+20+65+35+65)	34.17
M	1/6(40+60+30+30+60+30)	41.7
N	1/6(50+30+50+5+5+5)	24.17

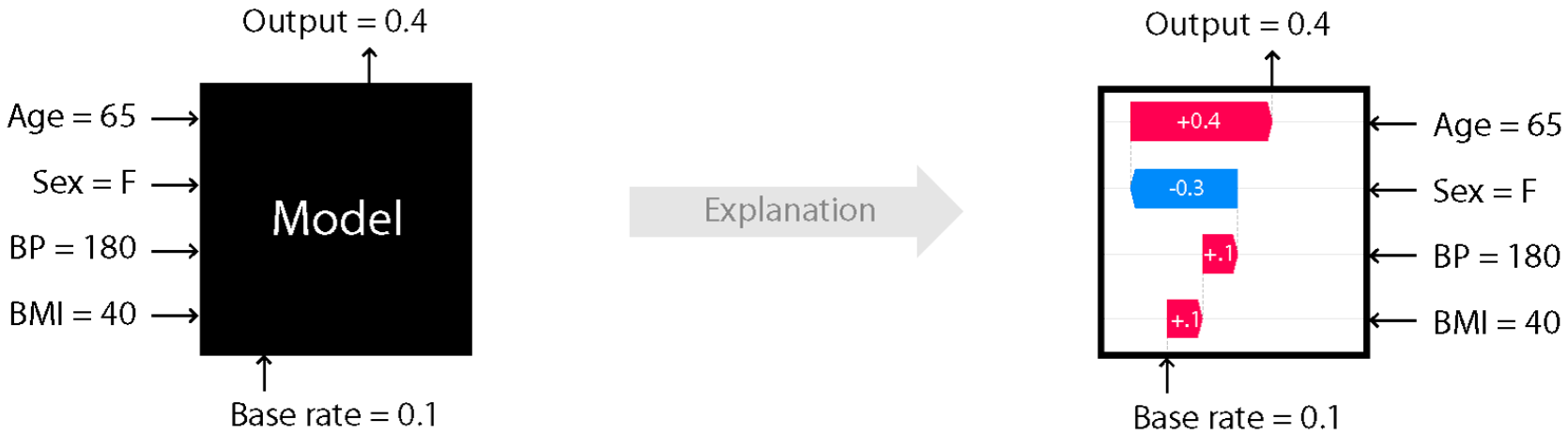
Shapley Values

Although M is 6 times more capable than N (30 vs 5), M should get 41.7% of reward while N should get 24.17% reward!

Source: Edward Ma | Link

TreeSHAP

- TreeShap is an implementation of SHAP integrated with XGBoost, a popular gradient boosting tree algorithm.
- Brute-forcing all 2^N feature combinations is inefficient, so TreeSHAP uses the structure of decision trees to quickly approximate the Shapley Values.
- Python Library: https://github.com/slundberg/shap



Source: SHAP | Link

SHAP Explainers & Example Notebooks

WAYNE STATE UNIVERSITY

TreeExplainer: Tree SHAP, a fast and exact algorithm to compute SHAP values for trees and ensembles of trees.

- **Census income classification with LightGBM** - Using the standard adult census income dataset, notebook trains a gradient boosting tree model with LightGBM and then explains predictions using SHAP.

DeepExplainer: An implementation of Deep SHAP, a faster (but only approximate) algorithm to compute SHAP values based on connections between SHAP and the DeepLIFT algorithm.

- **MNIST Digit classification with Keras** - Explains predictions using SHAP
- **Keras LSTM for IMDB Sentiment Classification** - Explains predictions using SHAP

GradientExplainer: An implementation of expected gradients to approximate SHAP values for deep learning models. Slower than DeepExplainer and makes different approximation assumptions.

- **Explain an Intermediate Layer of VGG16 on ImageNet** - Explain output of a pre-trained VGG16 ImageNet model using an internal convolutional layer.

LinearExplainer: For a linear model with independent features we can analytically compute the exact SHAP values. We can also account for feature correlation if we are willing to estimate the feature covaraince matrix. LinearExplainer supports both of these options.

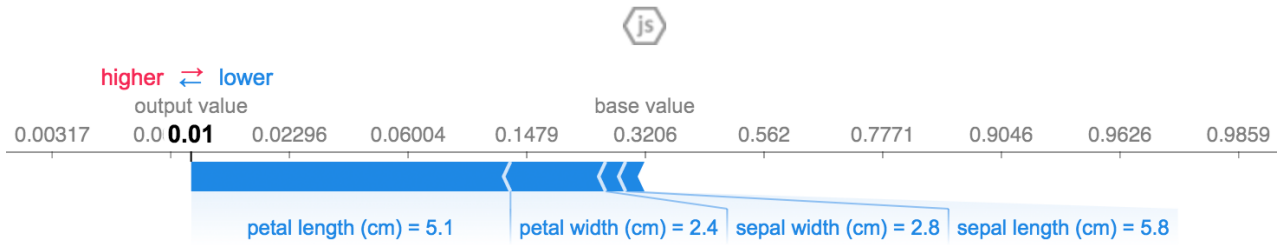
- **Sentiment Analysis with Logistic Regression** - Explain a linear logistic regression sentiment analysis model.

KernelExplainer: Kernel SHAP, a model agnostic method to estimate SHAP values for any model. Because it makes no assumptions about the model type, it is slower than the other model type specific algorithms.

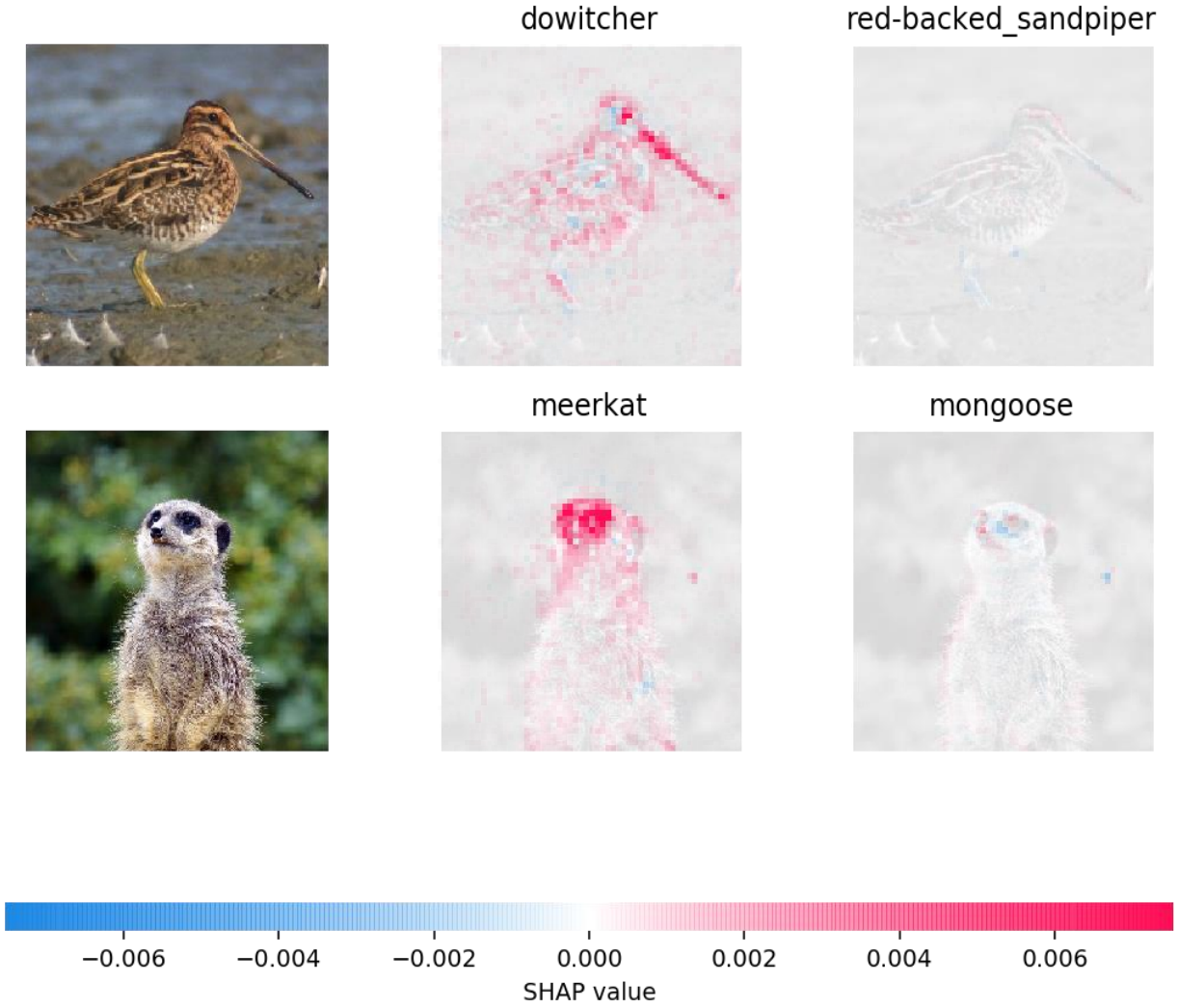
- **Census income classification with scikit-learn** - Notebook trains a k-nearest neighbors classifier using scikit-learn and then explains predictions using SHAP

Source: SHAP | Link

SHAP Examples:



Model agnostic example for IRIS Dataset with KernelExplainer (explains any function)



Deep learning example with GradientExplainer (TensorFlow/Keras/PyTorch models)

Source: SHAP | Link

WAYNE STATE UNIVERSITY

Recourse Analysis

Berk Ustun (Harvard U.), Alexander Spangher (U. of Southern California), Yang Liu (U. of California, Santa Cruz)

Paper | GitHub

Interpretable & Actionable

- We covered tools for understanding why an algorithm made a particular decision, but this **explanation may not necessarily be "actionable"**.
- For individuals who receive a negative outcome by some automated decision-making process, we would ideally want to be able to *recommend actions they could take to improve their outcome for next time*.

Recourse Analysis

- Given a linear model, our goal is to produce a flipset for an individual: A set of actions the individual can undertake to change her outcome.
- Formally, for an individual with features x and outcome $f(x) = -1$, does some action a exist such that $f(x + a) = 1$?
- With a linear model, we can use integer optimization to find the easiest action for getting recourse.
- Python Library: https://github.com/ustunb/actionable-recourse

Using LIME & SHAP: Python Case Study

Human-Intepretable Machine Learning - The Road to Explainable AI using Census Dataset

Primary Source: *Tutorial by* *Dipanjan (DJ) Sarkar*

Interpreting Machine Learning models is no longer a luxury but a necessity. To understand and need and importance of human-interpretable machine learning, feel free to check out the article, *Explainable Artificial Intelligence (Part 1)— The Importance of Human Interpretable Machine Learning*.

In this tutorial we will take a look at various ways to explain potential black-box machine learning models in a model-agnostic way. We will be working on a real-world dataset on Census income, also known as the Adult dataset available in the UCI ML Repository where we will be predicting if the potential income of people is more than $50K/yr or not.

The purpose of this tutorial is manifold. The first main objective is to familiarize ourselves with the major state-of-the-art model interpretation frameworks out there (a lot of them being extensions of LIME - the original framework and approach proposed for model interpretation).

LIME (short for local interpretable model-agnostic explanations) is based on the work presented in the paper, "Why Should I Trust You?": Explaining the Predictions of Any Classifier" which talks about a novel explanation technique that explains the predictions of any classifier in an interpretable and faithful manner, by learning an interpretable model locally around the prediction. It also covers a method to explain models by presenting representative individual predictions and their explanations in a non-redundant way.

We cover usage of the following model interpretation frameworks in our tutorial.

- **LIME**
- **SHAP**

Canvas Course Website:

- Python Jupyter Notebook Code
- model_evaluation_utils.py
- HTML Output