Mahidol University
Faculty of Medicine Ramathibodi Hospital
Department of Clinical Epidemiology and Biostatistics

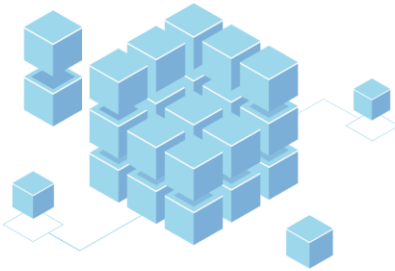
Machine Learning Explainability in Nasopharyngeal Cancer Survival using LIME and SHAP

Rasheed Omobolaji Alabi, Mohammed Elmusrati, Ilmo Leivo, Alhadi Almangush & Antti A. Mäkitie

Commentator: Habib ur Rehman Owasi
Msc Data Science for Healthcare and Clinical Informatics

Journal Club 21 March 2024

C&B



Introduction

Why ML in healthcare?

- Improves diagnostic accuracy
- Helps in personalized treatment planning
- Predicts survival outcomes

Why is explainability critical?

- Black-box models limit clinical adoption
- Regulatory and ethical requirements
- Enhances trust and transparency for clinicians and patients

High Accuracy ≠ High Interpretability

A model can be highly accurate but still be a black box.

Example: XGBoost, Random Forest, Deep Neural Networks can achieve very high accuracy in medical imaging or survival prediction — but they are not inherently interpretable.

These models learn complex, non-linear relationships.

Their decision-making process involves thousands of internal parameters (e.g., trees, weights), which are not human-readable.

Explainability Methods: LIME and SHAP

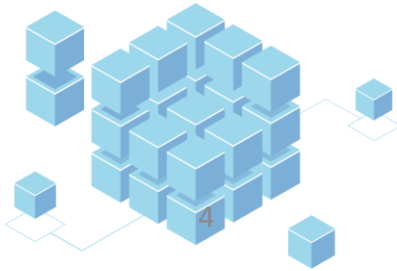
LIME (Local Interpretable Model-Agnostic Explanations)

- Perturbs input features to analyze their impact on model predictions
- Provides a local explanation for each individual prediction

SHAP (Shapley Additive Explanations)

- Based on cooperative game theory
- Considers all feature combinations to determine their contribution
- Provides both local and global explanations

C&B

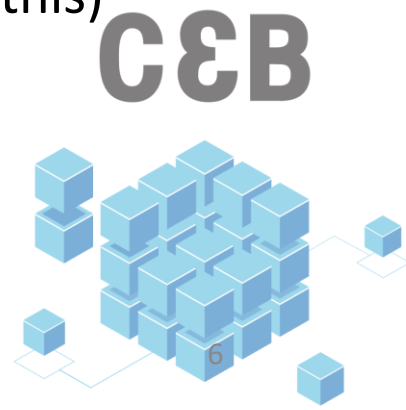


Comparison: LIME vs. SHAP

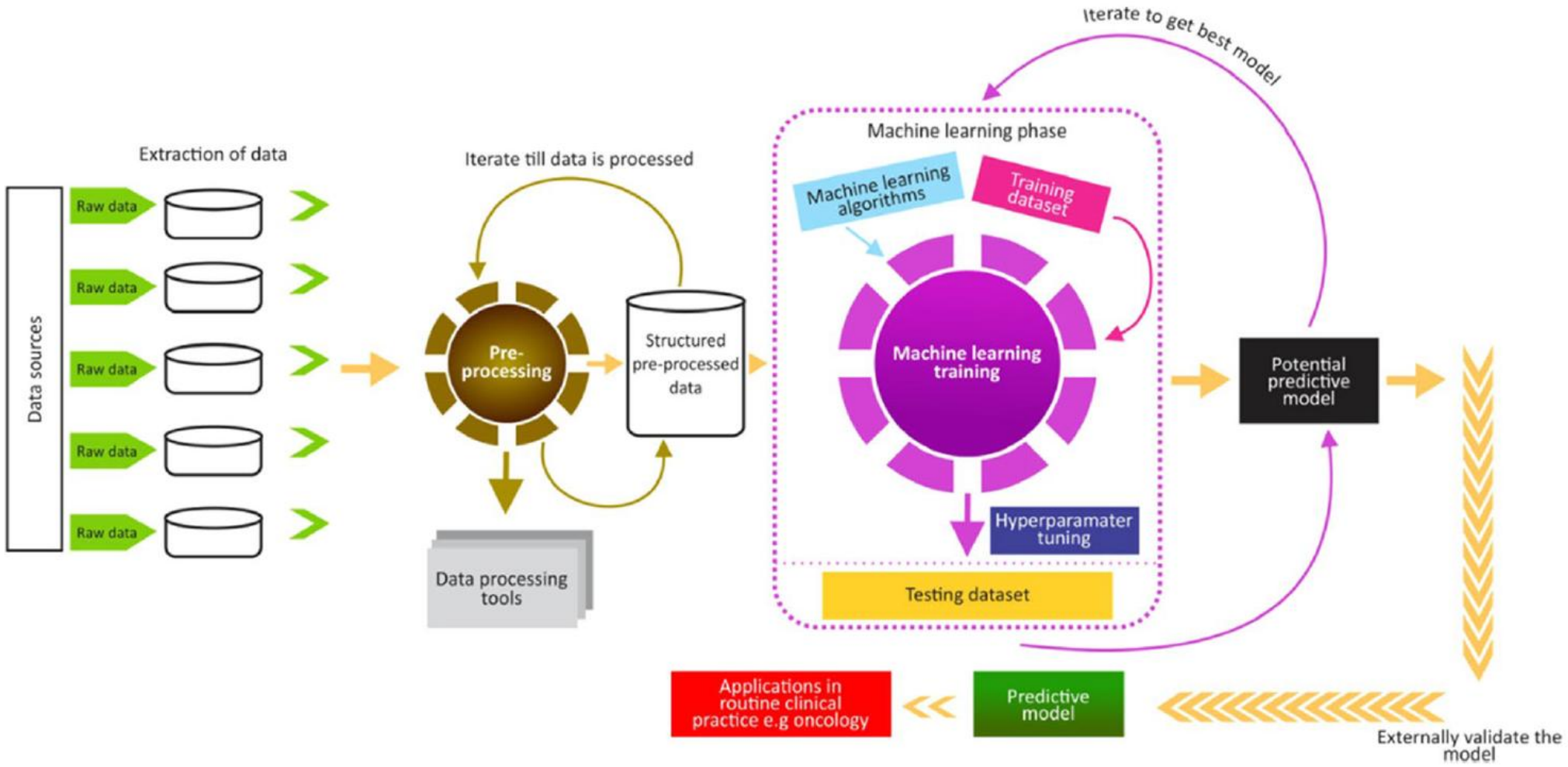
Feature	LIME	SHAP
Interpretability	Local only	Global + Local
Stability of Explanations	Moderate (Perturbation-based)	High (based on Shapley values)
Computational Efficiency	Faster	Slower
Clinical Usefulness	Quick insight for individual patients	Robust insight for patient groups and individuals

MS-CPFI vs. LIME & SHAP: What's Different?

- **MS-CPFI** (Model-Agnostic Counterfactual Perturbation Feature Importance)
  - **Perturbs feature values** to analyze their importance
  - **Counterfactual approach** (i.e., what happens if a feature had a different value?)
  - Designed for **multi-state survival models**
- **Key differences:**
  - MS-CPFI works on multi-state models (LIME and SHAP are not designed for this)
  - MS-CPFI does not rely on perturbation of real samples but counterfactual variations



A typical ML training process



ML in Nasopharyngeal Cancer Prognosis

Objective: To predict nasopharyngeal cancer (NPC) survival using machine learning models.

Dataset: Utilized 1,094 NPC patient records from the SEER database for training and validation.

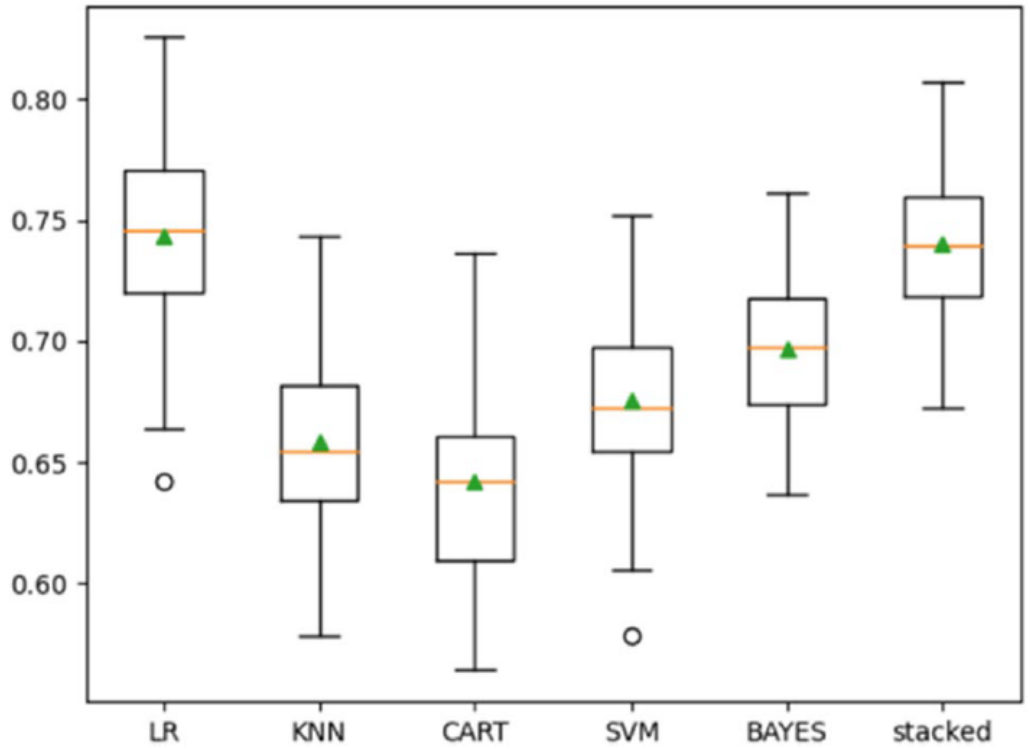
Models Compared: Stacked ML model (ensemble of 5 algorithms) vs. XGBoost (state-of-the-art approach).

Validation: Performance tested through internal, geographic external validation.

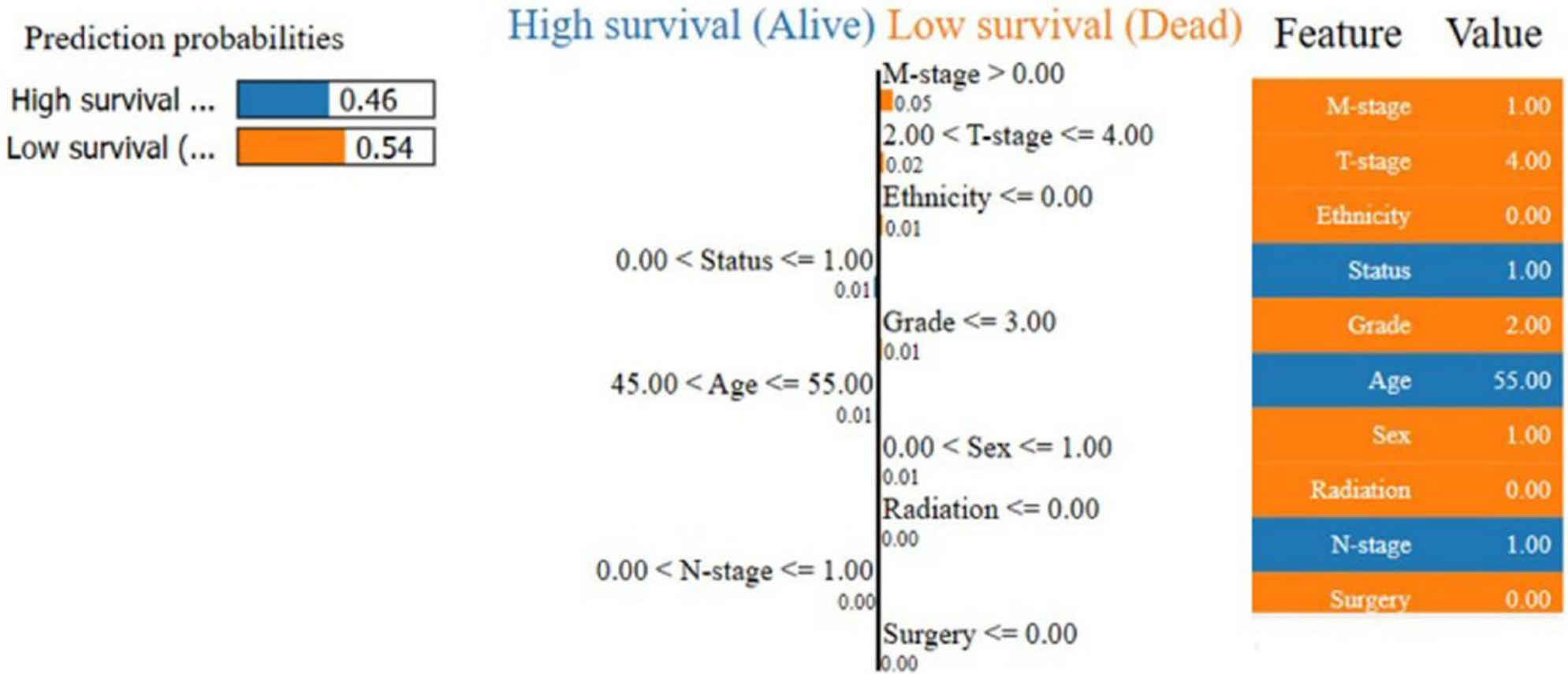
Training performance of the individual algorithm and the stacked algorithm.

Stacked Model
Combines 5 algorithms, achieving 85.9% accuracy in NPC survival prediction.

XGBoost Model
State-of-the-art boosting technique with 84.5% accuracy, comparable to stacking.



LIME explainability of a single instance



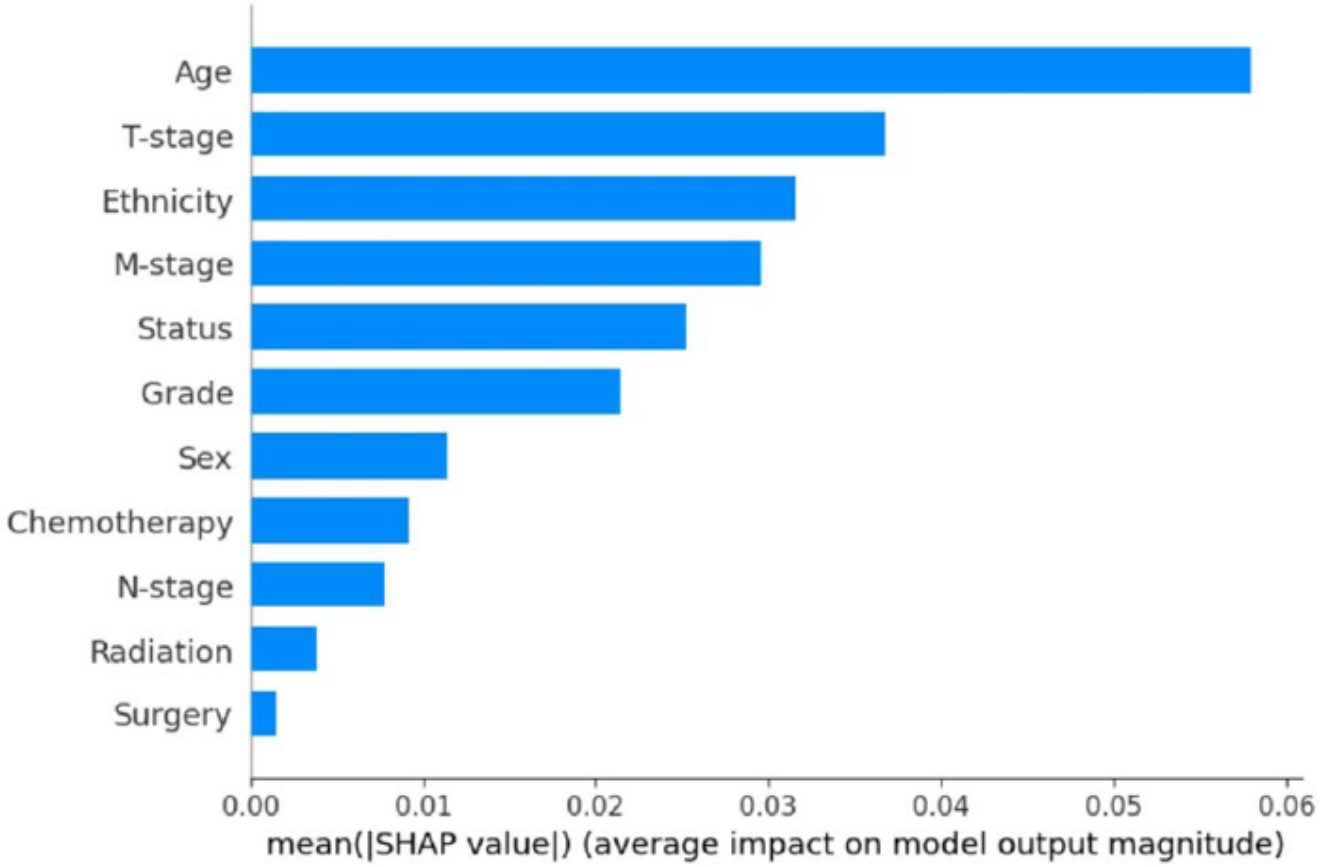
Mahidol University
Faculty of Medicine Ramathibodi Hospital
Department of Clinical Epidemiology and Biostatistics

SHAP force plot

(a) high chance of survival (b,c) low risk of survival

Contribution of each feature to the prediction

y-axis *is the features in the dataset*



They are ordered from the highest to the lowest effect on the prediction.

It does not matter if the feature affects the prediction in a positive or negative way, it only shows how much a single feature affected the prediction.

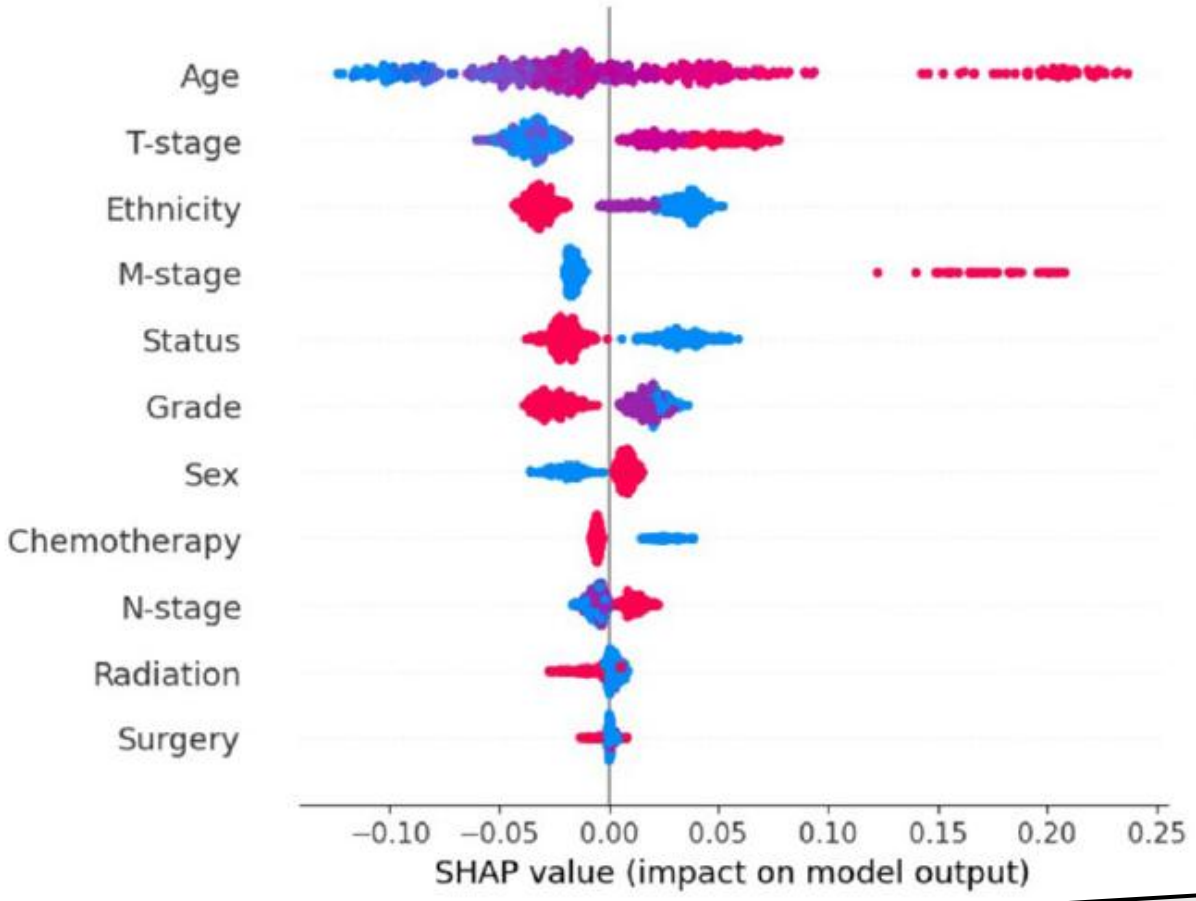
x-axis *is the mean absolute SHAP value*

C&B

SHAP Beeswarm plot

Its designed to display an information-dense summary of how the top features in a dataset impact the model's output.

y-axis *is the features in the dataset*



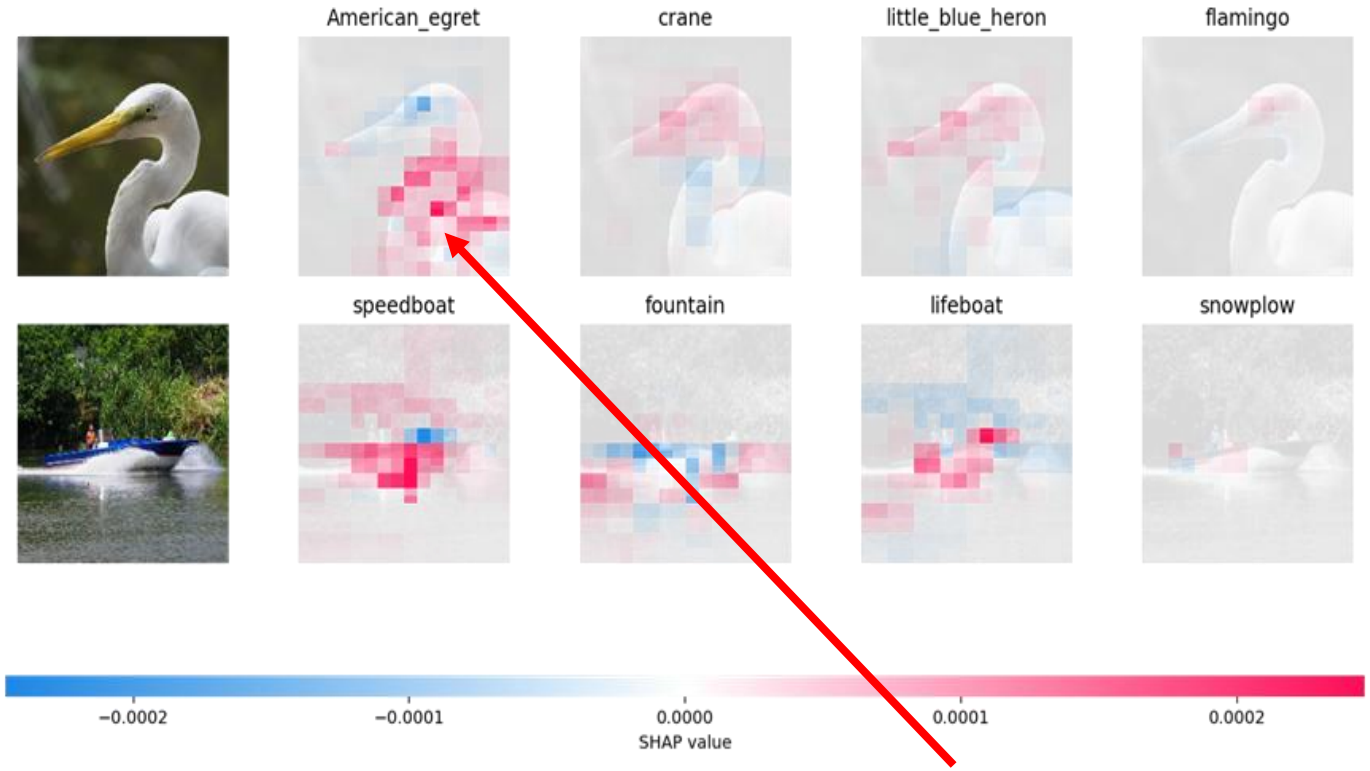
The color shows how higher and lower values of the feature will affect the result.

They are also ordered from the highest to the lowest effect on the prediction.

x-axis *represents the SHAP value*

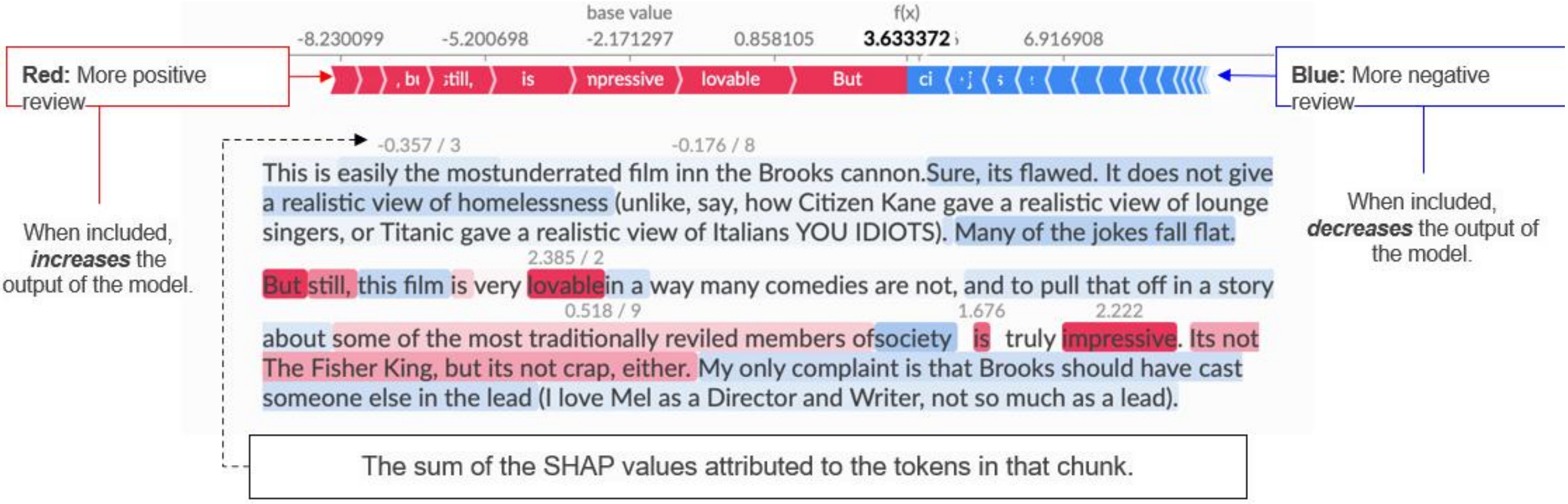
SHAP Plots: *Image Plot*

- Image plot is used to explain prediction based on images.

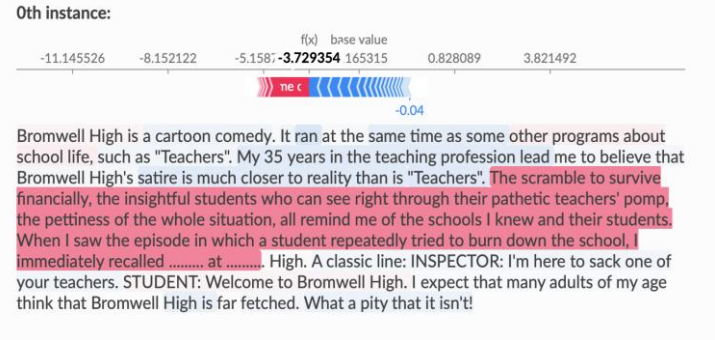


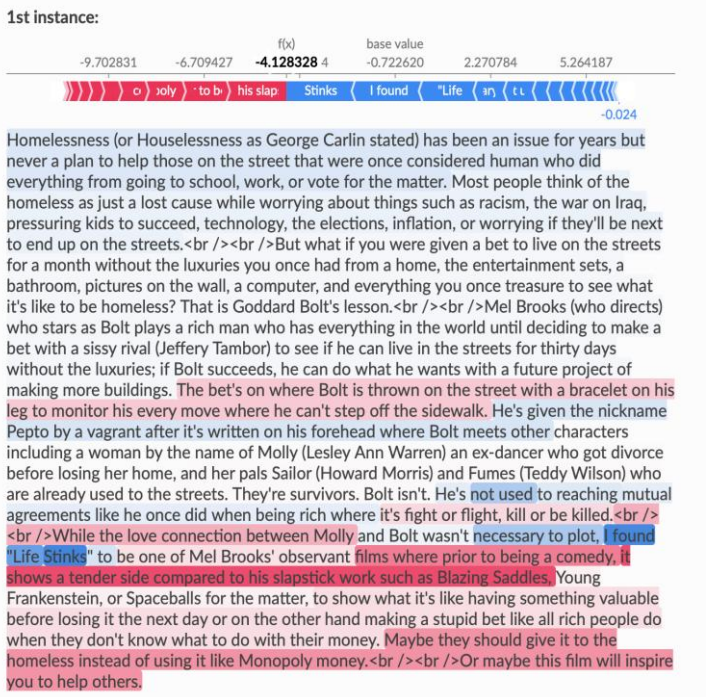
The bird is American egret. The bump at the neck is red, meaning it contribute more toward predicting the image as American_egret.

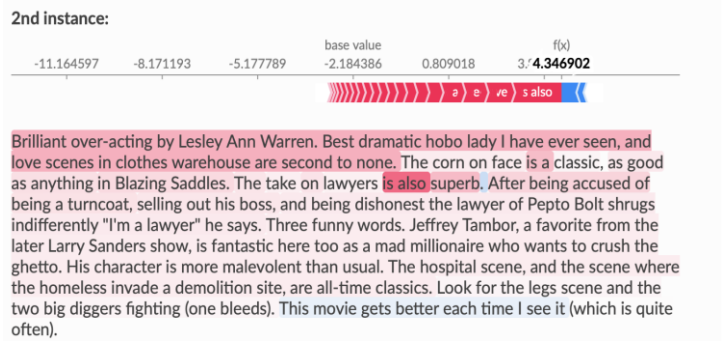
SHAP Plots: ***Text Plot - Single Instance Text Plot***



SHAP Plots: Text Plot - Multiple Instance Text Plot







C&B

When we pass a multi-row explanation object to the text plot, we get the single instance plots for each input instance scaled so they have consistent comparable x-axis and color ranges.

Limitations

1. **Feature Dependencies:**
   - When two or more model features are associated/correlated (value of one feature *depends* on the value of another).
   - Two ways Feature Dependencies impact SHAP:
     - The first comes from how SHAP values are approximated → ***can be misleading***
     - Feature dependencies can also lead to some confusion when interpreting SHAP plots.

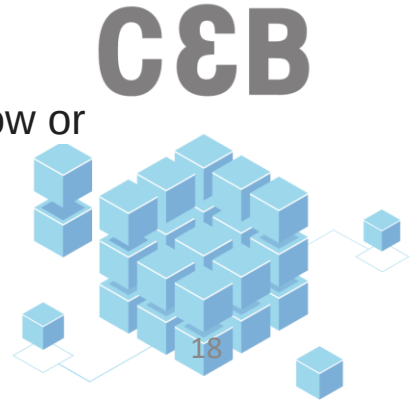
2. **Causal Inference:**
   - Cannot be used for causal inference (finding an event's true causes).
   - Cannot tell us how the features contributed to the target variable because a model is not necessarily a good representation of reality.

SHAP is not a measure of "how important a given feature is in the real world", it is simply "how important a feature is to the model". — Gianlucca Zuin

Limitations (Continued)

3. **Human Error/Feature Importance Consistence:**
   - False narratives can be created during SHAP values analysis through ***Confirmation Bias.***
   - It can also be done maliciously to support a conclusion that will benefit someone.
   - SHAP values is strongly related to the “objective” of the model.
   - SHAP output should always be analyzed considering the model objective in mind.

4. **Model Agnostic in Theory, Not Always in Practice**
   - SHAP's TreeSHAP works well for tree-based models like Random Forest.
   - KernelSHAP, which is fully model-agnostic, is computationally expensive.
   - SHAP for deep learning models (DeepSHAP) is hard to implement in frameworks like TensorFlow or PyTorch.



Conclusion

- The LIME and SHAP techniques are both model-agnostic techniques for providing explanations to the prediction made by an ML model.
- Both LIME and SHAP enhanced clinical interpretability:
  - SHAP identified global critical features (age, stage, ethnicity).
  - LIME provided intuitive local explanations for individual patient prognosis.
- SHAP generally preferred for robust, comprehensive clinical interpretations.

Thank You

C&B