Model explainability and Model Diagnosis

Srimugunthan
Data science-Lead

Brief about me..

- My name is **Srimugunthan**
- **Data Science & Analytics-Tech Lead** @ Wells Fargo
- **Area of interest/Expertise**: Prediction models, Recommender systems, Reinforcement learning Experimentation and causal inference

Outline

Part1: Model explainability techniques

Understand different explainability techniques and how they compare against each other

.

Part2: Demo, Model understanding using explainability

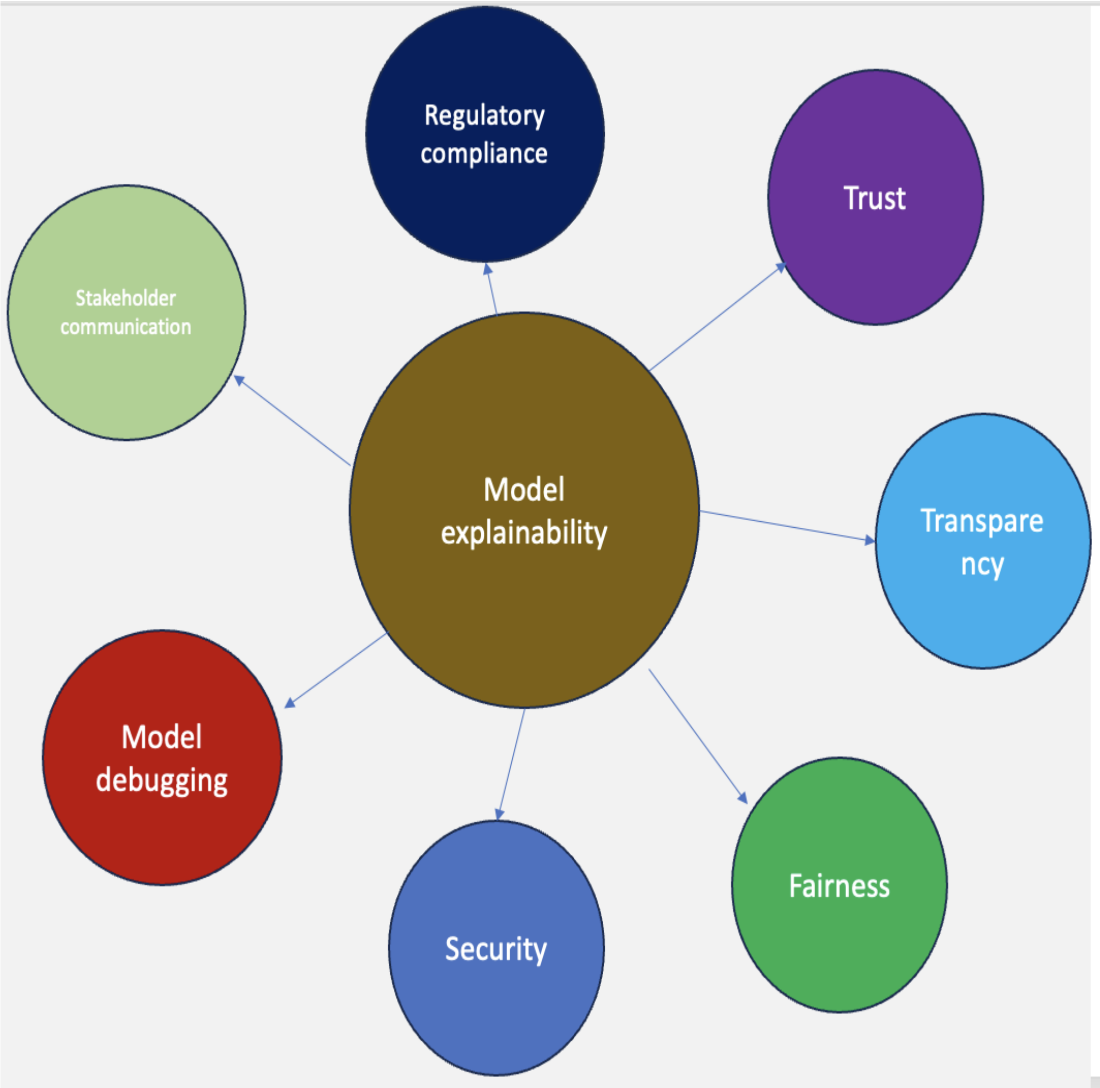
Using explainability to understand model behavior

Part3: Model diagnosis and debugging

Use explainability techniques to debug a model

Part-1: Model explainability techniques

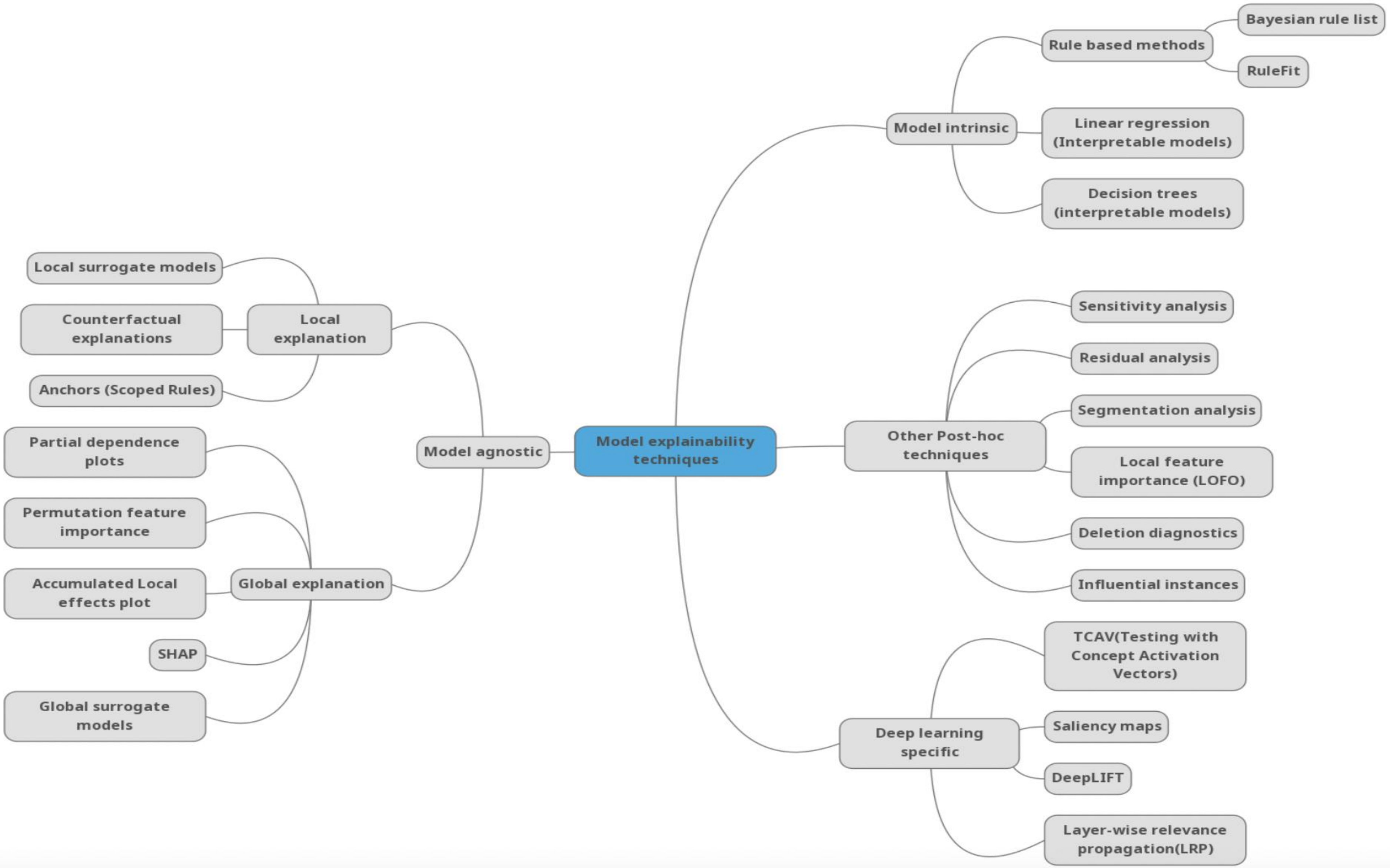
Why is Model explainability important



- **Trust and Transparency:** Understand black box models. Model based Decisions need to be transparent
- **Ethical considerations**

  Bias and Fairness: In applications where fairness and avoiding discrimination are critical, understanding the model can help identify and mitigate biases.
- **Regulatory Compliance:**
  - EU's General Data Protection Regulation (GDPR) requires a "right to explanation,"
  - Banking institutions require to follow SR 11-7 regulation
  - Mitigating Model Risks :
    - Manage AI incidents and Needs AI incident response
- **Stakeholder Communication**
- **Model Debugging and ensuring model quality**

Model explainability techniques mind map



Partial dependence plot

Partial dependence plot: plot the average model outcome in terms of different values of the predictor

Step 0

X1	X2	X3	Y
a1	b1	c1	Y1
a2	b2	c2	Y2
a3	b3	c3	Y3

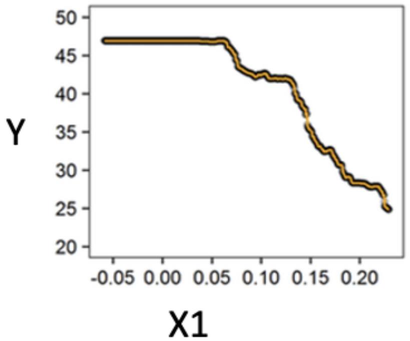
Step 1

X1	X2	X3	Y	Mean
a1	b1	c1	Y11	
a1	b2	c2	Y12	Y (a1)
a1	b3	c3	Y13	
a2	b1	c1	Y21	
a2	b2	c2	Y22	Y (a2)
a2	b3	c3	Y23	
a3	b1	c1	Y31	
a3	b2	c2	Y32	Y (a3)
a3	b3	c3	Y33	

Step 2

X1	a1	a2	a3
Y	Y (a1)	Y (a2)	Y (a3)

Step 3



ALGORITHM

1. **Select the Feature of Interest:** Choose the feature for which you want to estimate the partial dependence.
2. **Define the Range:** Determine the range of values for the selected feature that you want to explore.
3. **Generate Random Samples:** Instead of creating a fixed grid of values, generate random samples from the selected feature's range.
4. **Make Predictions:** For each randomly sampled value, set the selected feature to that value while keeping all other features constant. Then, make predictions using your model.
5. **Calculate Average Predictions:** Calculate the average prediction across all the random samples for each value of the selected feature.
6. **Plot the Results:** Plot the selected feature values on the x-axis and the corresponding average predictions on the y-axis to create the Monte Carlo estimate of the Partial Dependence Plot.

Code from scratch: https://github.com/h2oai/mli-resources/blob/master/notebooks/pdp_ice.ipynb
https://medium.com/dataman-in-ai/how-is-the-partial-dependent-plot-computed-8d2001a0e556

ICE plot

- Individual conditional expectation plot: plots model outcome for each random sample of predictor
- While a PDP visualizes the averaged relationship between features and predicted responses, a set of ICE plots disaggregates the averaged information and visualizes an individual dependence for each observation

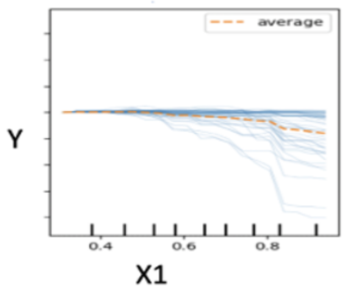
Step 0

X1	X2	X3	Y
a1	b1	c1	Y1
a2	b2	c2	Y2
a3	b3	c3	Y3

Step 1

X1	X2	X3	Y
a1	b1	c1	Y11
a1	b2	c2	Y12
a1	b3	c3	Y13
a2	b1	c1	Y21
a2	b2	c2	Y22
a2	b3	c3	Y23
a3	b1	c1	Y31
a3	b2	c2	Y32
a3	b3	c3	Y33

Step 2



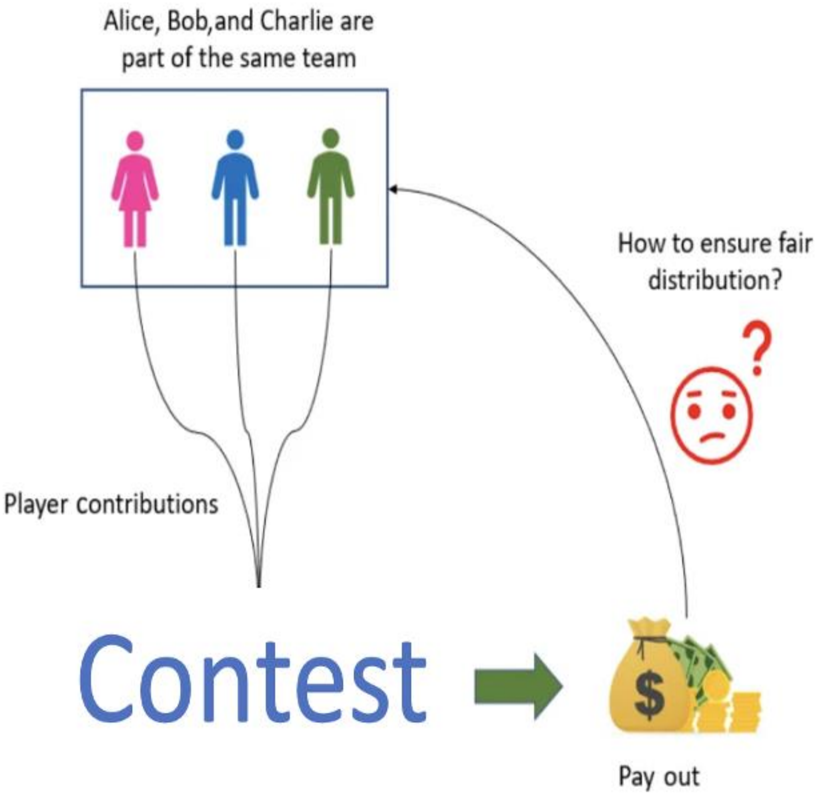
ALGORITHM

ICE Plot:

- **Individual Instances:** For each instance in your dataset, fix the values of all features except the one of interest.
- **Vary Feature of Interest:** Change the value of the feature of interest for that specific instance while keeping the other features constant.
- **Make Predictions:** For each value of the feature of interest, make predictions using the model for that specific instance.
- **Plot the Results:** Plot the predictions against the varying values of the feature of interest for that specific instance.

Code from scratch: https://github.com/h2oai/mli-resources/blob/master/notebooks/pdp_ice.ipynb

SHAP values



ALGORITHM

The algorithm: Approximate Shapley estimation for single feature value

Output: Shapley value for the value of the j-th feature

Required: Number of iterations M, instance of interest x, feature index j, data matrix X, and machine learning model f

For all $m = 1, \ldots, M$:

- Draw random instance z from the data matrix X
- Pick a random subset of feature column indices o (with j not in o).
- Construct two new instances
  - With j from x: x_{+j}, where all values in x with index in o are replaced by the respective values in z.
  - Without j from x: x_{-j}, where all values in x with index in o are replaced by the respective values in z and also the value for j is replaced by the value in z.
- Compute marginal contribution: $\phi_j^m = \hat{f}(x_{+j}) - \hat{f}(x_{-j})$

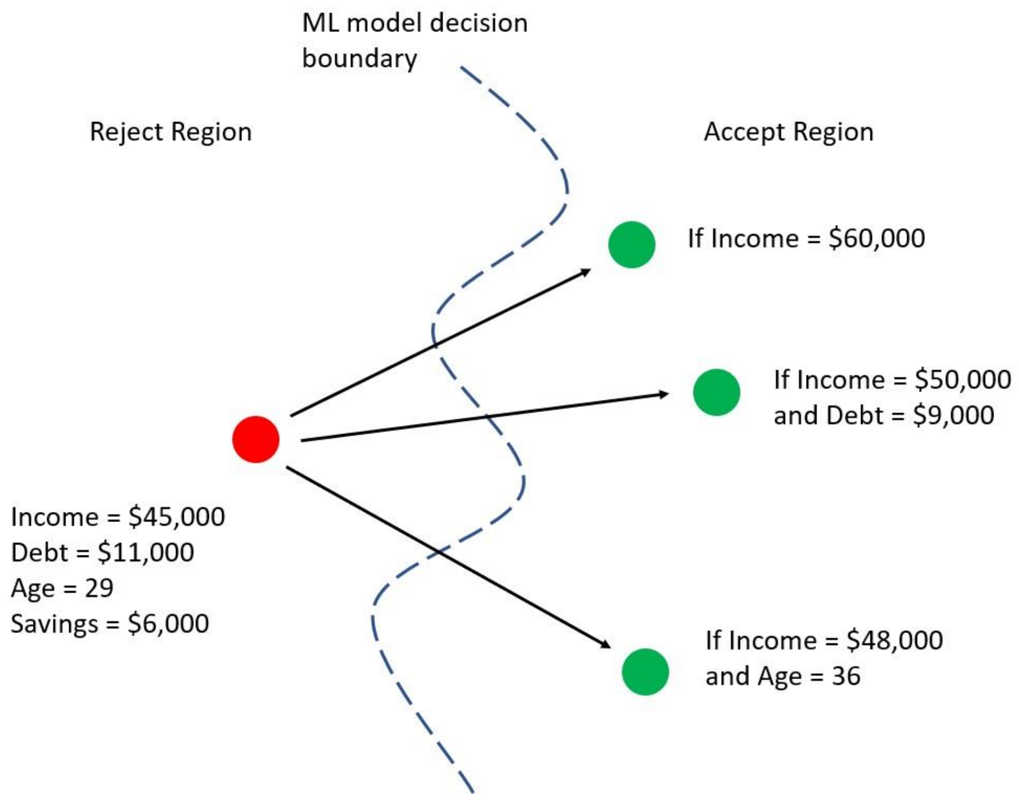
Compute Shapley value as the average: $\phi_j(x) = \frac{1}{M} \sum_{m=1}^{M} \phi_j^m$

Code-from-scratch : https://www.depends-on-the-definition.com/shapley-values-from-scratch/

https://towardsdatascience.com/understand-the-working-of-shap-based-on-shapley-values-used-in-xai-in-the-most-simple-way-d61e4947aa4e

Counterfactuals

Definition: A counterfactual is a datapoint close to a given data point, such that the model predicts it to be in a different class



ALGORITHM

1. INPUT:
- The data point or the instance.
- The features of interest that you want to perturbate.
- Additional Constraints

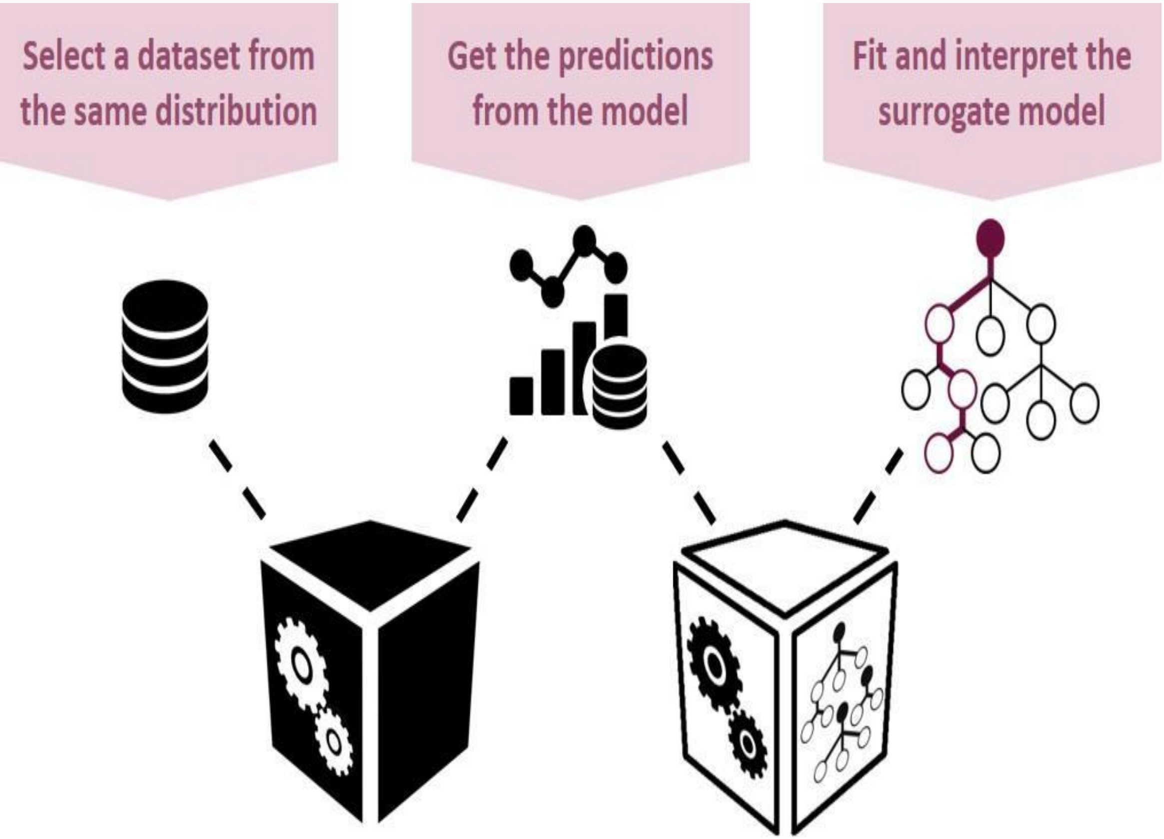
2. Define the Distance Metric:
- Choose a distance metric to quantify the dissimilarity between instances. (Eg: Euclidean distance, Manhattan distance, or a custom distance function).

3. Optimization Solver :
- Formulate an optimization problem to find a new instance that is close to the original but results in a different prediction.
- Define an objective function that balances the proximity of the counterfactual to the original instance and the change in the model's prediction, subject to the constraints
- Use an optimization solver (e.g., gradient-based or evolutionary algorithms) to find the values of the features that minimize the objective function while satisfying the defined constraints.

4. Output the Counterfactual Explanation:
- Present the counterfactual instance if it results in a different prediction and highlight the changed features as an explanation for the model's decision.

[Implementation from scratch: https://www.kaggle.com/code/kyosukemorita/re-focus-counterfactual-explanations-for-trees](https://www.kaggle.com/code/kyosukemorita/re-focus-counterfactual-explanations-for-trees)

Global Surrogate model explanation



ALGORITHM

STEP1: Choose a dataset This could be the same dataset that was used for training the black box model or a new dataset from the same distribution.

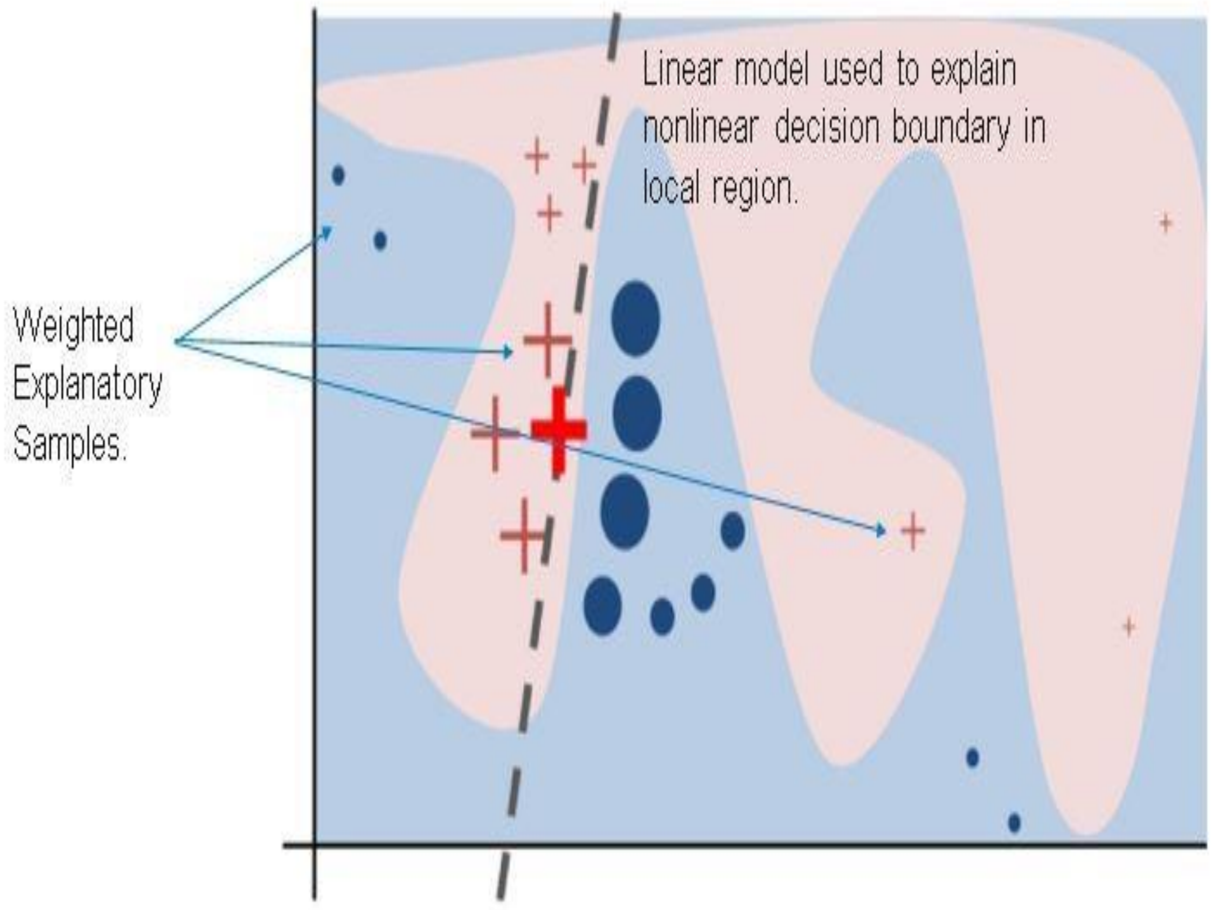
STEP2: For the chosen dataset, get the predictions of your base black box model.

STEP3: Choose an interpretable surrogate model (linear model, decision tree, …).

STEP4: Train the interpretable model on the dataset and its predictions. This is the surrogate model.

STEP5: Interpret / visualize the surrogate model.

Local surrogate model explanation



ALGORITHM

1. Choose your instance of interest for which you want to have an explanation of the predictions of your black box model.

2. Perturb your dataset and get the black box predictions for these new points.

3. Weight the new samples by their proximity to the instance of interest.

4. Fit a weighted, interpretable (surrogate) model on the dataset with the variations.

5. Explain prediction by interpreting the local model.

Algo toy-implementation: https://fat-forensics.org/how_to/transparency/tabular-surrogates.html
https://www.kdnuggets.com/2018/12/explainable-ai-model-interpretation-strategies.html/2

Techniques Compared

	PDP/ICE	SHAP	Counterfactual	Local surrogate	Surrogate
Type of explainability	PDP: Global ICE: Local	Global as well as local	Local	Local	Global
Cons	Correlated features create invalid points in plot	Sensitive to order of input Can be adversarially manipulated to hide bias Computationally complex	Rashomon effect: Multiple contradictory explanations	Explanations can be unstable. Can be adversarially manipulated to hide bias	fail for non-linear relationships and high-dimensional interactions among features.
When to use	Helps diagnose monotonicity	Lends to great visualizations	Useful in misprediction analysis	Lime works for tabular data, text and images	When you need a simpler understandable proxy model

Part-2: Demo, Model explainability for model understanding

Demo: Model explainability

- **Titanic Notebook(Local surrogate, Lime):** https://github.com/srimugunthan/Modelexplainability-session/blob/main/4_Lime/LIME-Titanic.ipynb
- **Titanic Notebook(Counterfactuals DiCe):** https://github.com/srimugunthan/Modelexplainability-session/blob/main/5_DiCe/DiceOnTitanic.ipynb
- **Titanic Notebook (PDP/ICE) :** https://github.com/srimugunthan/Modelexplainability-session/blob/main/1_PDP_ICE/PDP-ICE-titanic.ipynb
- **Titanic Notebook (SHAP) :** https://github.com/srimugunthan/Modelexplainability-session/blob/main/2_SHAP/SHAP-titanic.ipynb
- **Housing Prices (Surrogate tree model)** : https://github.com/h2oai/mli-resources/blob/master/notebooks/dt_surrogate.ipynb

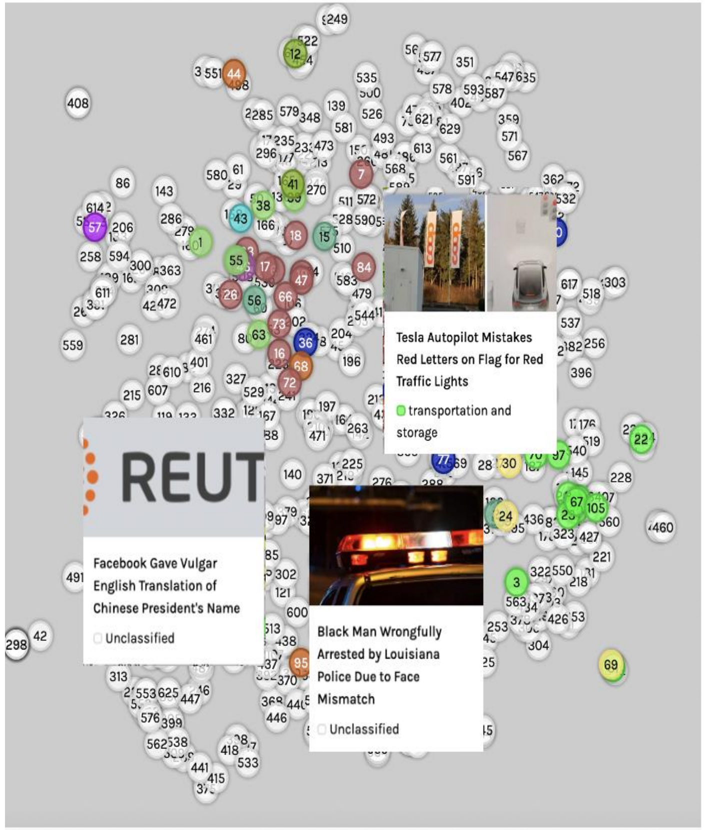
Part-3: Model diagnosis and debugging

Common AI/ML model Bugs

INDEPENDENT MODEL LEVEL → PRODUCTION DEPLOYMENT AND INTEGRATION LEVEL → PRODUCT LEVEL

SOME COMMON ISSUES IN ML MODEL	DESCRIPTION
Data leakage	Wrong methodology of cross validation, including features with target leakage
Biased training data	Training data preparation error , Data augmentation errors, Intentional data poisoning
Feature interactions	Correllated features, proxy features, Non-additive effect from multiple features
Source Data errors not handled	Missing values, outliers, sparse data Incomplete data due to non-availability Duplicate data rows
Wrong Feature engineering	Disparate feature scales. Information loss on certain features
Overfitting, Underfitting, bias-variance isssues	Errors in Hyper parameter and Model selection, Feature selection high cardinality of certain categorical features
Error in target variable distribution	Imbalance in target. Target distribution not representing reality
Model instability	Model results do not have reproducibility
Distribution shift	Dataset selection issues, using very old data etc
Bias , Fairness issues	Underperformance in segment, data sparsity in certain regions

- Data loss at source, missing data errors
- Broken upstream models
- Changes in Table update cadence
- Mismatch in feature engineering during training and production
- Drift issues
  - Concept drift
  - Data drift
  - Model degradation
- Adversarial attacks



https://incidentdatabase.ai/summaries/spatial/

“Machine learning for High risk applications” https://www.oreilly.com/library/view/machine-learning-for/9781098102425/

Example : Diagnosing and Debugging biased ML model

Credit card Example (from " Machine learning from High-risk applications" book) :

- https://github.com/ml-for-high-risk-apps-book/Machine-Learning-for-High-Risk-Applications-Book/blob/main/code/Chapter-10/Testing_and_Remediating_Bias_constrained.ipynb

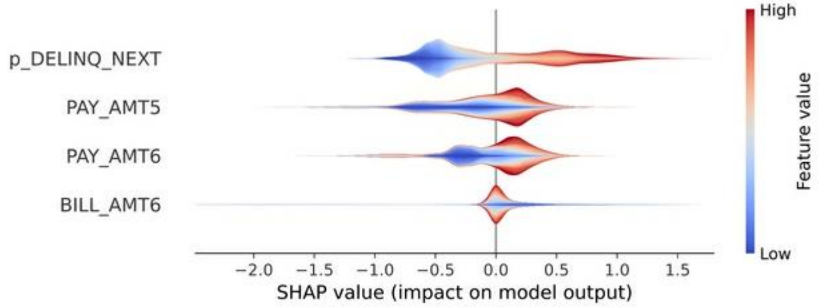
Diagnosing Bias

Metrics by segment

	Prevalence	Accuracy	True Positive Rate	Precision	Specificity	Negative Predicted Value	False Positive Rate	False Discovery Rate	False Negative Rate	False Omissions Rate
hispanic	0.399393	0.732053	0.564557	0.705696	0.843434	0.744428	0.156566	0.294304	0.435443	0.255572
black	0.386707	0.719033	0.544271	0.667732	0.829228	0.742647	0.170772	0.332268	0.455729	0.257353
white	0.107075	0.817718	0.565476	0.308442	0.847966	0.942109	0.152034	0.691558	0.434524	0.057891
asian	0.101010	0.832323	0.620000	0.326316	0.856180	0.952500	0.143820	0.673684	0.380000	0.047500

Debugging Bias

- Hunt for proxy features encoding race by building an **adversarial model that predicts the race**
- Identify problematic feature from SHAP plot

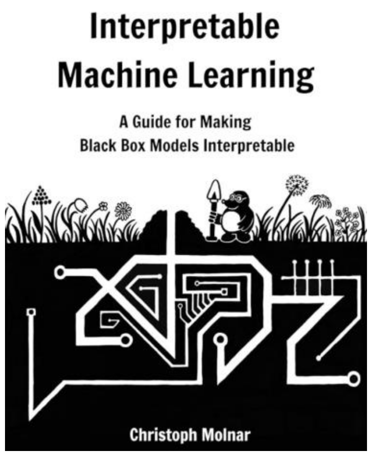


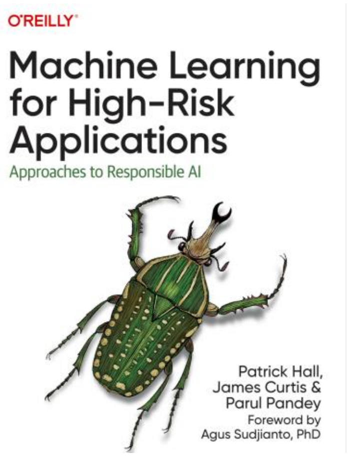
Fixing Bias, Ways of Remediation

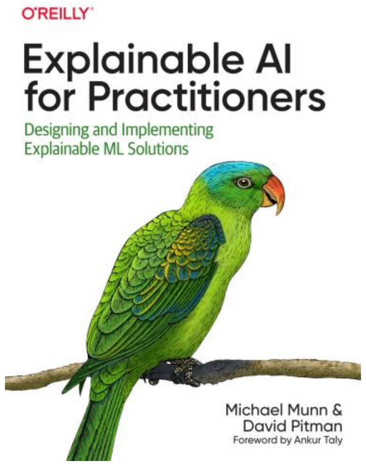
- Re-sampling training data(during pre-processing)
- Do regularisation, training a weighted model, a customized objective function or Select model for performance as well as fairness measure (during modelling phase)
- Modify prediction scores (during post-processing)

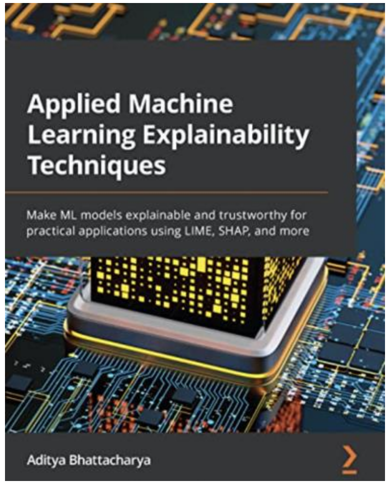
Thanks

Books









LinkedIn connect

https://www.linkedin.com/in/srimugunthan-dhandapani/