Explainable AI Introduction

Anthony Harrington
PyData Meetup
Warsaw May 7th 2019

Explainable AI

- Higher accuracy typically comes at the expense of
- interpretability.
- Explainable AI aims to create a suite of machine learning techniques that [1]:
  - Produce more explainable models, while maintaining a high level of learning performance (prediction accuracy); and
  - Enable human users to understand, appropriately trust, and effectively manage the emerging generation of artificially intelligent partners

Why is it important?

- Interpretable fair and transparent models are a series legal mandate in regulated sectors such as banking, insurance, healthcare, etc. [1]
- EU GDPR Article 22 legislates a right to explanation for EU citizens impacted by algorithmic decisions.*
- Regulatory requirements also change and are a key driver of what constitutes interpretability in machine learning. (Risk Basel III A-IRB)
- Without interpretability there is no certainty that algorithms are not relearning and applying human biases and there are no assurances that humans have not designed a machine to make intentionally erroneous decisions
- Basic emotional need to understand and trust decisions made by ML algorithms.
- Hacking and adversarial attacks are difficult to detect unless we understand more about decision making process.

Adversarial Attacks Stop Sign to 45 mph speed-limit

Robust Physical-World Attacks on Deep Learning Visual Classification
CVPR2018 - https://arxiv.org/pdf/1707.08945.pdf

Interpretability Impact on Adoption and Oversight

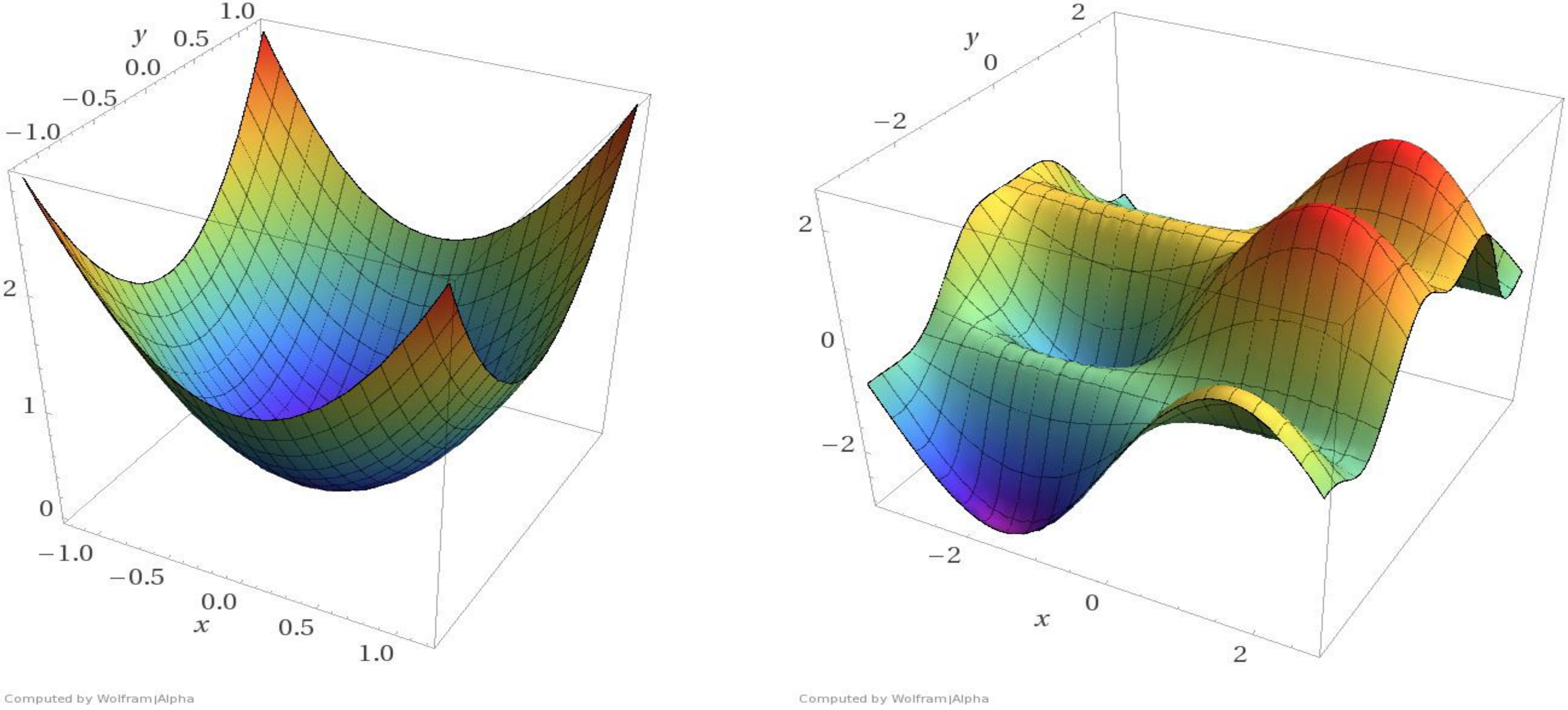
- 82 percent of all enterprises are now considering or moving ahead with AI adoption, attracted by the technology's ability to drive revenues, improve customer service, lower costs, and manage risk. However, … 60 percent of those companies fear liability issues (IBM Institute for Business Value)

- AI4PEOPLE EU consortium founded to examine the need for ML explainability and transparency, possibly supported by **auditing mechanisms**; formulating redress or compensation processes; the need for **appropriate metrics for AI trustworthiness**; developing a **new EU oversight agency responsible for** the protection of public welfare through the **evaluation and supervision of AI**;

(https://jia.sipa.columbia.edu/building-trust-artificial-intelligence)

Why is it difficult?

- Fundamentally difficult and very new field
- Multiplicity of good models [6]
  - given the same set of input variables and prediction targets, complex ML algorithms can produce multiple accurate models with similar but different architectures
  - if you have a convex error surface and fit a linear model there is basically 1 best model
  - if there is no obvious global minimum and a concave surface then there will be multiple models with different weightings for making decisions.



Left: a convex function. Right: a non-convex function.
It is much easier to find the bottom of the surface in the convex function than the non-convex surface. (Source: Reza Zadeh)
https://www.oreilly.com/ideas/the-hard-thing-about-deep-learning

Linear Modeling vs Machine Learning

- In general linear models are focused on understanding and predicting average behaviour whereas ML can provide more accurate but more difficult to explain predictions for subtler aspects of modeled phenomenon. [1]

- Linear model evaluation includes hypotheses testing, confidence intervals, distributions of the residual sum-of-squares, goodness of fit, e.g. R squared. $$R^2 \equiv 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

- ML models tend to be evaluated in predictive accuracy

- *Key idea - In-sample versus out-of-sample accuracy*

Explainable AI Community

- FATML – Fairness Accountability and Transparency – academic driven with a broad social and commercial focus
  http://www.fatml.org/
- *Algorithms and the data that drive them are designed and created by people -- There is always a human ultimately responsible for decisions made or informed by an algorithm. "The algorithm did it" is not an acceptable excuse if algorithmic systems make mistakes or have undesired consequences, including from machine-learning processes.*
- DARPA funded – XAI or Explainable AI
  https://www.darpa.mil/program/explainable-artificial-intelligence
- The XAI program is focused on two areas: (1) machine learning problems to classify events of interest in heterogeneous, multimedia data; and (2) machine learning problems to construct decision policies for an autonomous system to perform a variety of simulated missions.
- DARPA focus is around security applications

Taxonomy of Model Interpretability 1

- High Interpretability → Linear Monotonic Functions.
  - Eg. traditional regression algorithm such as Ordinary Least Squares.
  - Any change in a given input variable will result in a change in the response variable in one direction and at a magnitude shown by the coefficient.
- Medium Interpretability → Nonlinear monotonic functions.
  - Some ML algorithms can be constrained to be monotonic with a given independent variable/s. There may not be a coefficient that represents that change but the change will be in one direction.
  - Can obtain relative variable importance measures.
- Low Interpretability → Non-linear, non-monotonic functions.
  - Most ML models fall in this category.
  - Can change in a positive or negative rate for any change in an input variable.
  - Typically these models only provide relative variable importance measures

https://www.oreilly.com/library/view/an-introduction-to/9781492033158/

High Interpretability Linear Models

- Traditional linear regression algorithm with OLS. (unbiased estimator)

$$y_i = \beta_0 1 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^\mathsf{T} \boldsymbol\beta + \varepsilon_i, \qquad i = 1, \ldots, n,$$

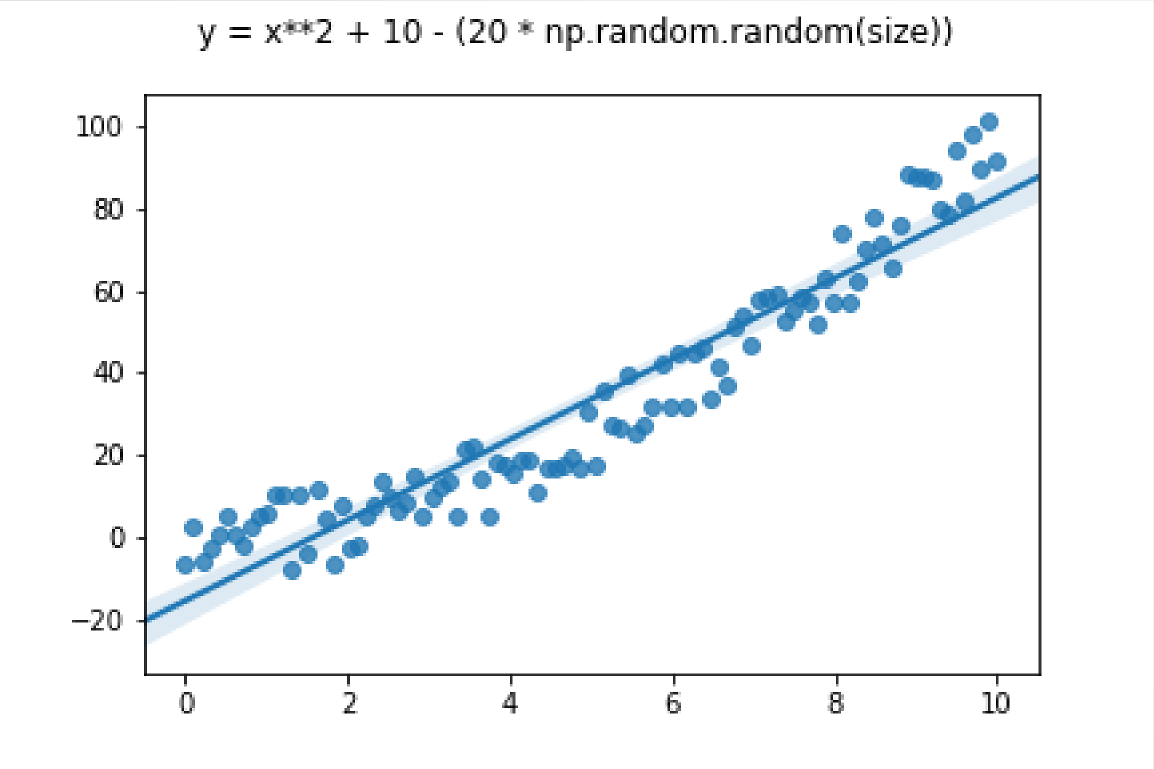
- If we assume that errors are *iid* N(0, Var) then *B_0* and *B_xi* are also normally distributed. (The CLT shows that OLS estimators are asymptotically normal even when the error terms are not normally distributed). Note that regressors(independent variables) should not be linear functions of other regressors(multi-collinearity)

- If the assumptions about the residuals are met then the distribution of the regression coefficients are normal and so we can calculate Confidence Intervals using CLT results. (Jerzy Spława-Neyman(1894-1981) (1924 PhD University of Warsaw))

- Enables inferential statistics to make inferences about the ***population*** from the sample.

- In-sample versus Out-of-sample accuracy

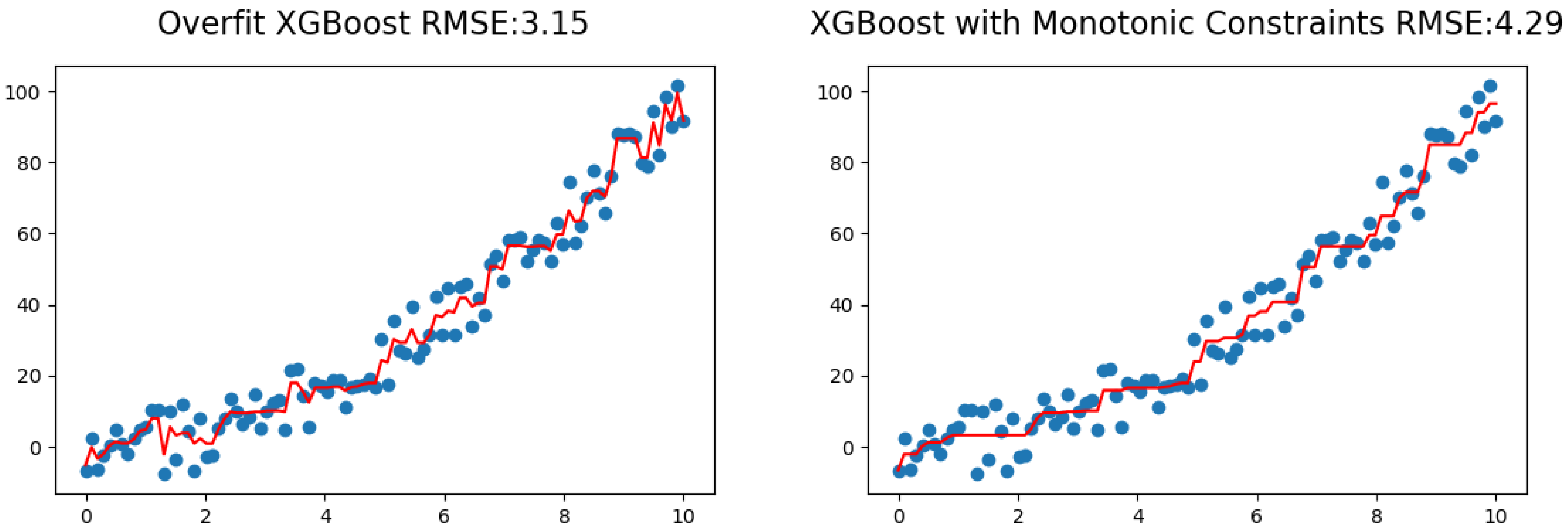
Medium Interpretability Example
Non-Linear Monotonic Functions



Monotonic change in independent
Variable is in one direction
No scikit-learn support
XGBoost has support.
LightGBM has support.

https://blog.datadive.net/monotonicity-constraints-in-machine-learning/

Medium Interpretability Example
Non-Linear Monotonic Functions



In-sample versus out-of-sample accuracy

https://blog.datadive.net/monotonicity-constraints-in-machine-learning/

Low Interpretability → Non-linear, non-monotonic functions Credit Card Default Dataset

Feature	Description
ID	Customer Identifier
LIMIT_BAL	Amount of the given credit. It includes both the individual consumer credit and his/her family (supplementary) credit.
SEX	Gender (1 = male
EDUCATION	Education (1 = graduate school
MARRIAGE	Marital status (1 = married
AGE	Age (year).
PAY_0	September 2005 Payment Status. -1 = pay duly 1 = payment delay for one month, 2 = payment delay for two months,…
PAY_2	August 2005 Payment Status
PAY_3	July 2005 Payment Status
PAY_4	June 2005 Payment Status
PAY_5	May 2005 Payment Status
PAY_6	April 2005 Payment Status
BILL_AMT1	September 2005 Bill Amount
BILL_AMT2	August 2005 Bill Amount
BILL_AMT3	July 2005 Bill Amount
BILL_AMT4	June 2005 Bill Amount
BILL_AMT5	May 2005 Bill Amount
BILL_AMT6	April 2005 Bill Amount
PAY_AMT1	Previous Payment September 2005
PAY_AMT2	Previous Payment August 2005
PAY_AMT3	Previous Payment July 2005
PAY_AMT4	Previous Payment June 2005
PAY_AMT5	Previous Payment May 2005
PAY_AMT6	Previous Payment April 2005
default payment	

Lasso and Random Forest Feature Importance Example

	Default Lasso Regression Sklearn
PAY_0	0.626778
BILL_AMT1	-0.39271
PAY_AMT2	-0.28479
PAY_AMT1	-0.20089
BILL_AMT3	0.169526
LIMIT_BAL	-0.11896
PAY_2	0.104651
PAY_3	0.091173
EDUCATION	-0.08618
BILL_AMT2	0.083418
MARRIAGE	-0.07728
AGE	0.074526
PAY_AMT5	-0.05364
SEX	-0.05325
BILL_AMT5	0.046767
PAY_5	0.045712
PAY_AMT4	-0.04256
PAY_AMT3	-0.03882
PAY_4	0.026819
PAY_AMT6	-0.02406
BILL_AMT4	-0.02017
BILL_AMT6	0.018744
Random	**-0.01696**
PAY_6	0.001152

	Default Random Forest Sklearn
PAY_0	0.086489
Random	**0.071762**
AGE	0.057357
BILL_AMT1	0.056032
LIMIT_BAL	0.055345
BILL_AMT2	0.052133
BILL_AMT4	0.048894
BILL_AMT3	0.046784
BILL_AMT6	0.046684
PAY_AMT1	0.046309
PAY_AMT3	0.043985
PAY_AMT6	0.043471
BILL_AMT5	0.043468
PAY_AMT2	0.043273
PAY_AMT4	0.040225
PAY_AMT5	0.038063
PAY_2	0.032155
PAY_3	0.031326
PAY_4	0.030506
PAY_6	0.025432
PAY_5	0.017893
EDUCATION	0.017761
MARRIAGE	0.01285
SEX	0.011805

In scikit-learn, we implement the importance as described in [1]. It is sometimes called "gini importance" or "mean decrease impurity" and is defined as: the total decrease in node impurity (weighted by the probability of reaching that node (which is approximated by the proportion of samples reaching that node) averaged over all trees of the ensemble.

https://stackoverflow.com/questions/15810339/how-are-feature-importances-in-randomforestclassifier-determined

GB Feature Importance Example

	Default GB Sklearn
PAY_0	0.149105
BILL_AMT1	0.110814
LIMIT_BAL	0.064468
BILL_AMT3	0.058056
PAY_AMT1	0.055928
BILL_AMT4	0.050508
Random	**0.048404**
BILL_AMT2	0.042515
PAY_2	0.038439
PAY_AMT3	0.035066
PAY_6	0.03416
AGE	0.032995
PAY_AMT2	0.030408
BILL_AMT6	0.029722
MARRIAGE	0.029597
PAY_3	0.027875
PAY_AMT6	0.027586
BILL_AMT5	0.024986
PAY_AMT5	0.024721
PAY_4	0.021623
EDUCATION	0.021109
SEX	0.015233
PAY_AMT4	0.014284
PAY_5	0.0124

	GridSearchCV GB
PAY_0	0.199625
BILL_AMT1	0.087447
Random	**0.066727**
LIMIT_BAL	0.051992
BILL_AMT2	0.046552
PAY_AMT1	0.045622
BILL_AMT3	0.039142
BILL_AMT6	0.038931
PAY_AMT3	0.037041
AGE	0.036882
PAY_AMT6	0.036171
PAY_AMT5	0.03514
BILL_AMT4	0.03479
PAY_AMT2	0.033824
BILL_AMT5	0.032348
PAY_AMT4	0.028242
PAY_2	0.028024
PAY_3	0.025673
PAY_5	0.021518
MARRIAGE	0.018973
PAY_6	0.017634
PAY_4	0.016749
EDUCATION	0.01484
SEX	0.006115

Scikit-learn GB uses Friedman-MSE as a purity function to sum up how much splitting on each feature reduced the impurity across all the splits in the tree.

The features are always randomly permuted at each split. Therefore, the best found split may vary, even with the same training data.

https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html

XGB Feature Importance Example

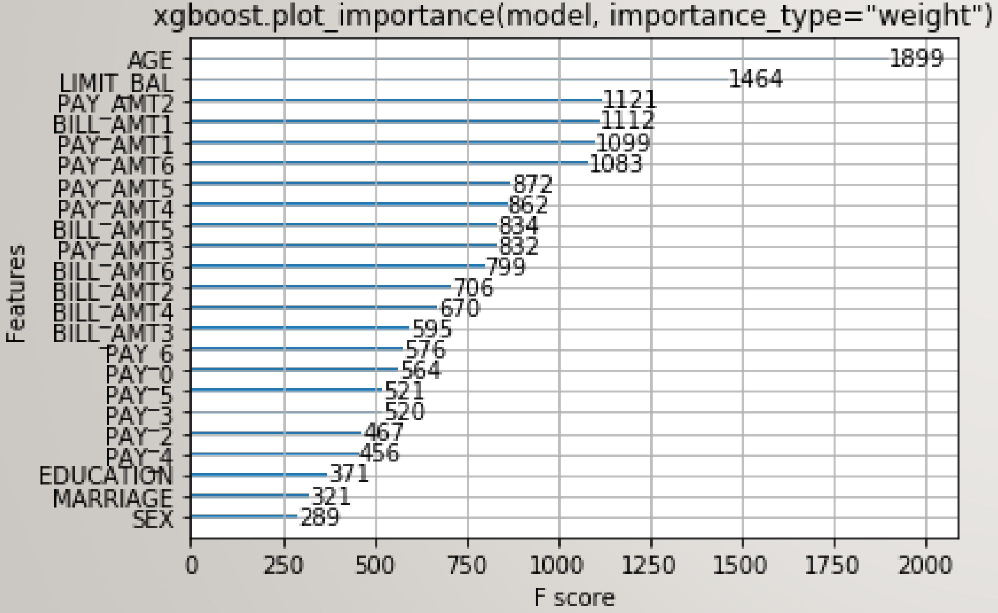
	Default XGBoost
BILL_AMT1	0.117820323
LIMIT_BAL	0.097201765
PAY_0	0.085419737
PAY_AMT2	0.060382918
Random	0.057437409
PAY_AMT1	0.050073639
BILL_AMT4	0.03976436
BILL_AMT2	0.038291607
PAY_AMT3	0.036818851
BILL_AMT3	0.033873342
PAY_3	0.033873342
PAY_AMT6	0.033873342
EDUCATION	0.032400589
PAY_6	0.030927835
MARRIAGE	0.030927835
PAY_AMT4	0.030927835
PAY_AMT5	0.030927835
AGE	0.030927835
BILL_AMT6	0.027982326
PAY_5	0.025036819
BILL_AMT5	0.023564065
PAY_2	0.02209131
PAY_4	0.019145804
SEX	0.010309278

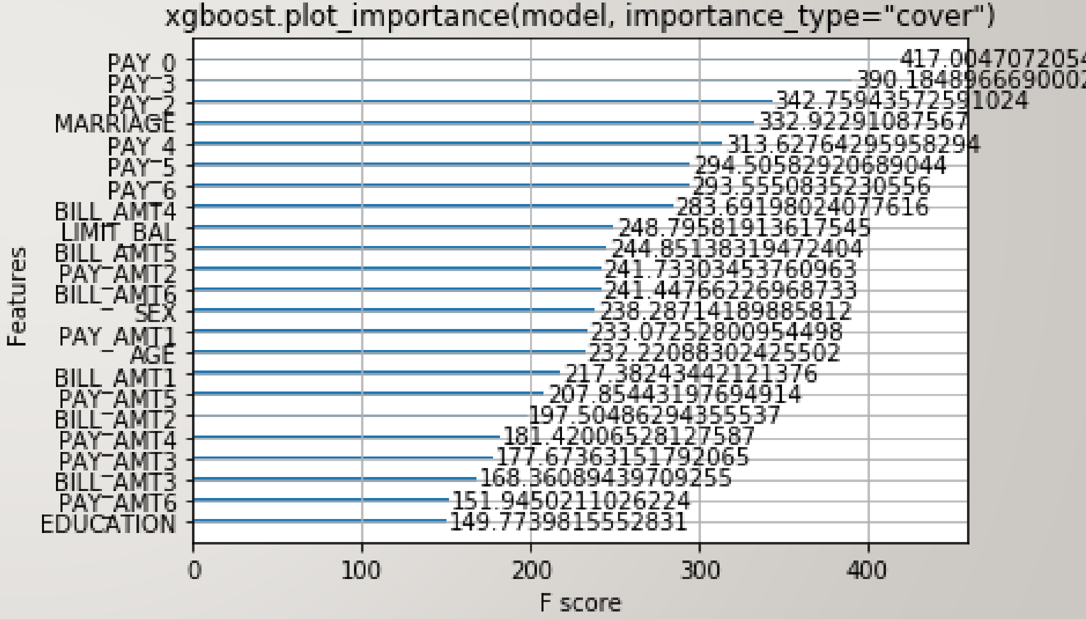
	GridSearchCV XGBoost
BILL_AMT1	0.103659
PAY_0	0.091463
LIMIT_BAL	0.089939
PAY_AMT2	0.064024
Random	**0.04878**
PAY_AMT1	0.047256
BILL_AMT2	0.045732
BILL_AMT4	0.044207
PAY_AMT3	0.041159
PAY_AMT6	0.041159
BILL_AMT3	0.03811
PAY_3	0.035061
PAY_6	0.033537
EDUCATION	0.033537
AGE	0.030488
BILL_AMT5	0.028963
PAY_AMT5	0.028963
PAY_AMT4	0.027439
BILL_AMT6	0.025915
PAY_4	0.02439
MARRIAGE	0.022866
PAY_2	0.021341
PAY_5	0.019817
SEX	0.012195

Feature importance is only defined when the decision tree model is chosen as base learner (booster=gbtree).
It is not defined for other base learner types, such as linear learners (booster=gblinear).

‘weight’: the number of times a feature is used to split the data across all trees.
‘gain’: the average gain across all splits the feature is used in.
‘cover’: the number of samples affected by a split averaged over all splits the feature is used in.
‘total_gain’: the total gain across all splits the feature is used in.
‘total_cover’: the total coverage across all splits the feature is used in.

XGBoost - Weight and Cover Importance Type



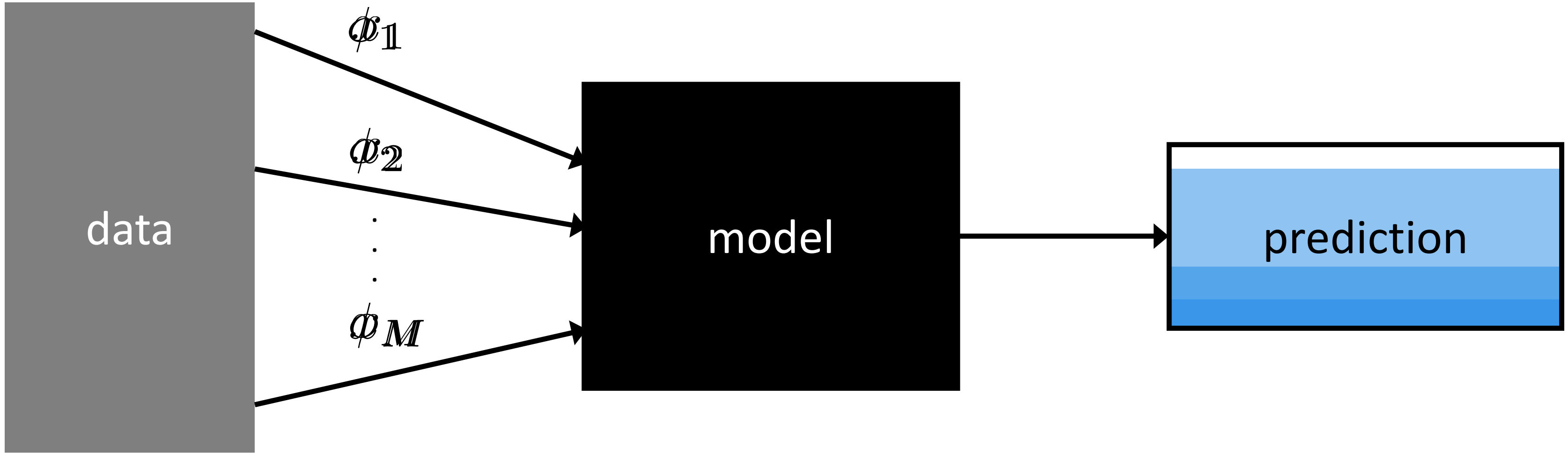


Taxonomy of Model Interpretability 2

- Global Interpretability
  - Global explanations of machine-learned relationship between the prediction target and the input variables
- Local Interpretability
  - in small regions – clusters of input records or subsets if data rows – you can get a typically more accurate local explanation.
- Model agnostic versus model specific interpretability -
  - E.g. LIME is model agnostic (Local Interpretable Model-Agnostic Explanations)
  - E.g, decision tree interpreter is model specific

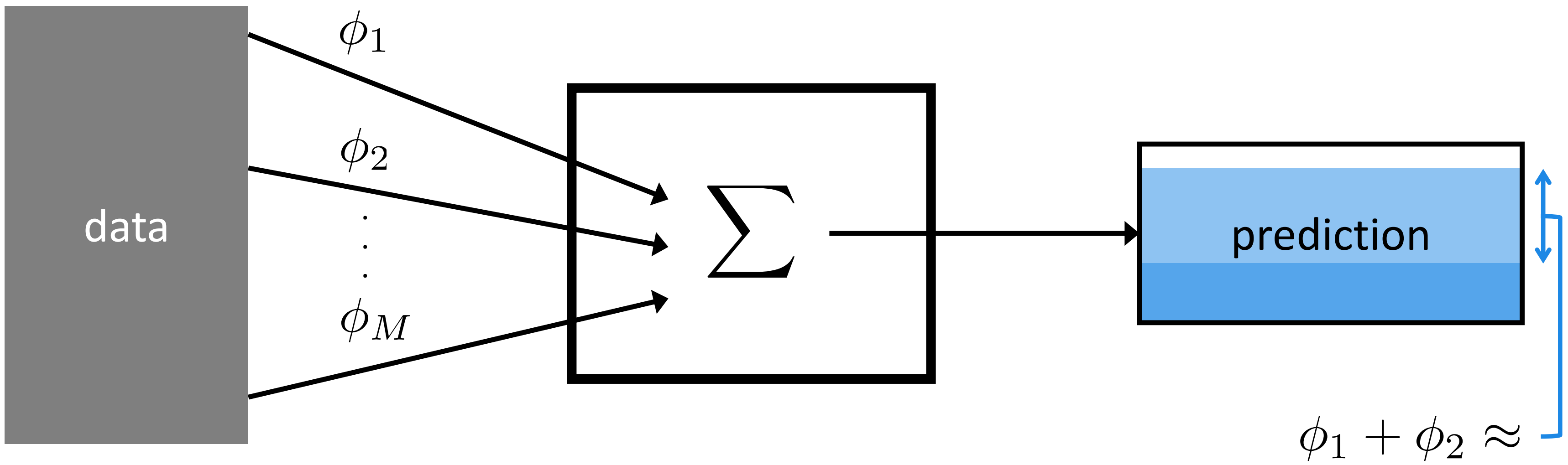
https://www.oreilly.com/library/view/an-introduction-to/9781492033158/

Explaining a complex model through additive feature effects



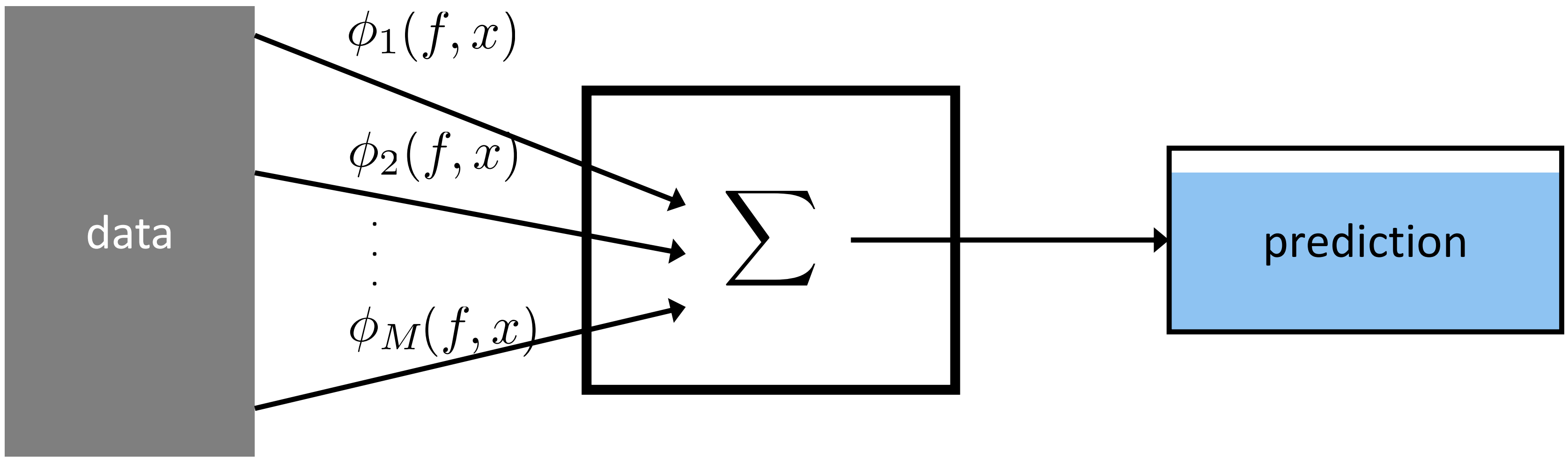
https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx

Model as a sum of feature attributions



https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx

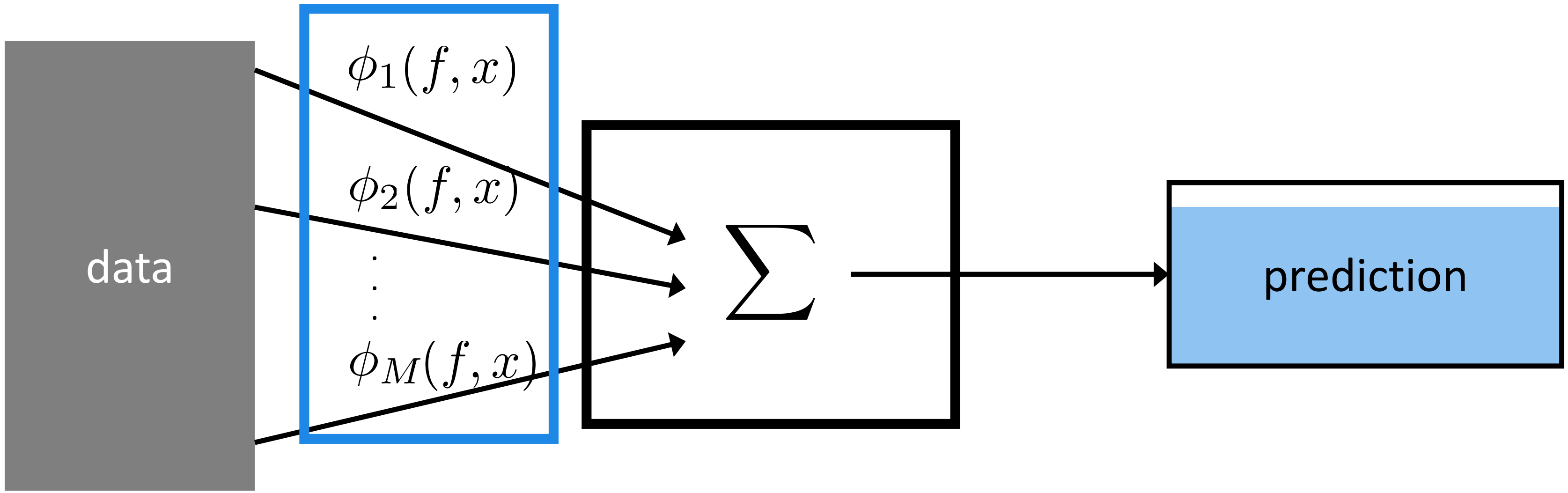
Model and data-set specific



In-sample vs Out-of-sample accuracy

https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx

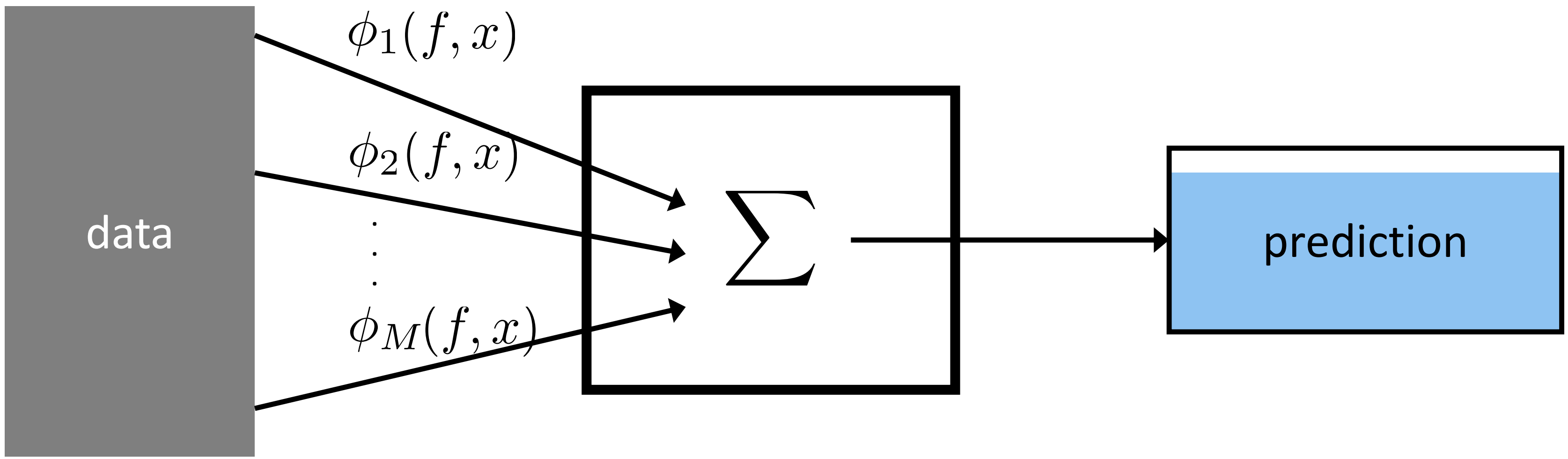
Additive feature attribution methods



Only one way to assign feature attributions given two properties!

https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx

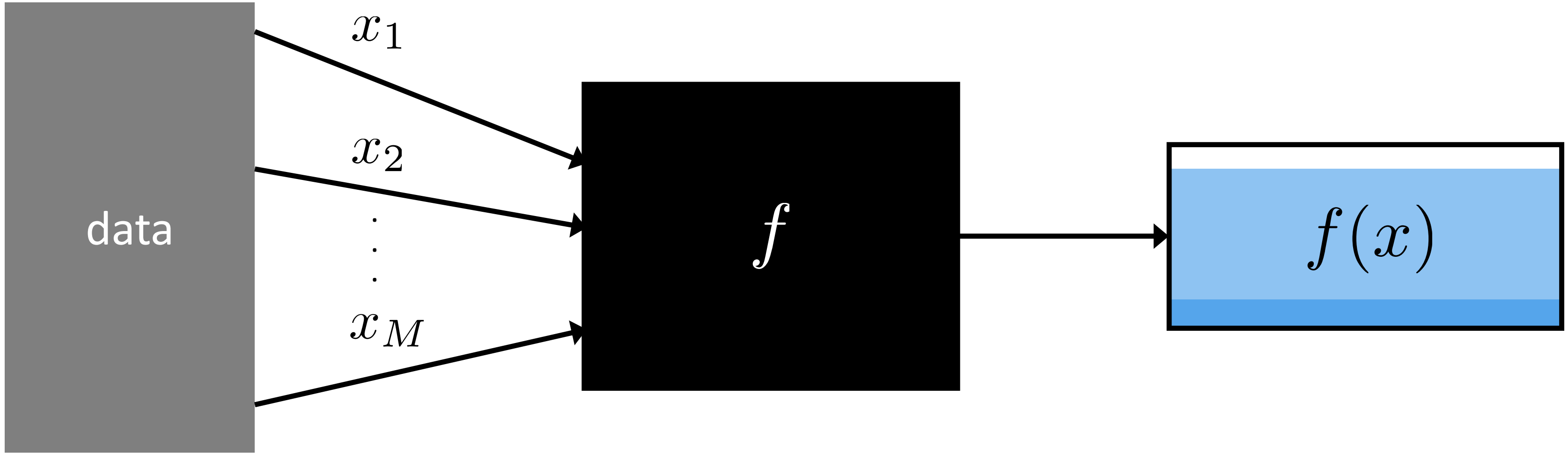
Additive feature attribution methods



1 Local accuracy

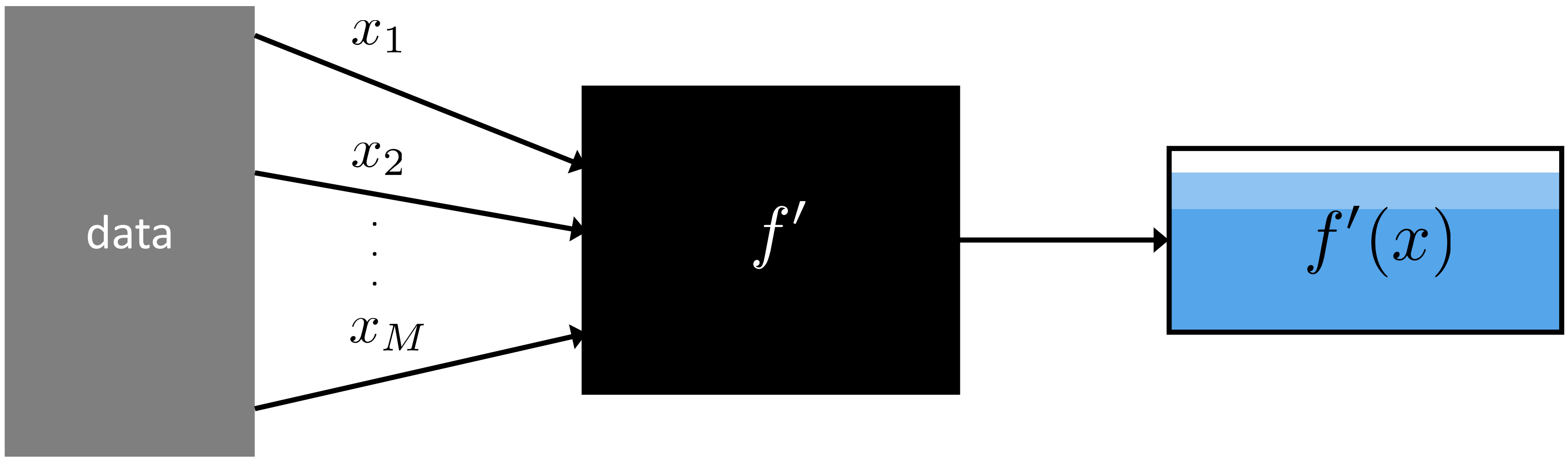
$$\sum_{i=0}^{M} \phi_i = f(x), \quad \phi_0 = f(\emptyset)$$

https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx



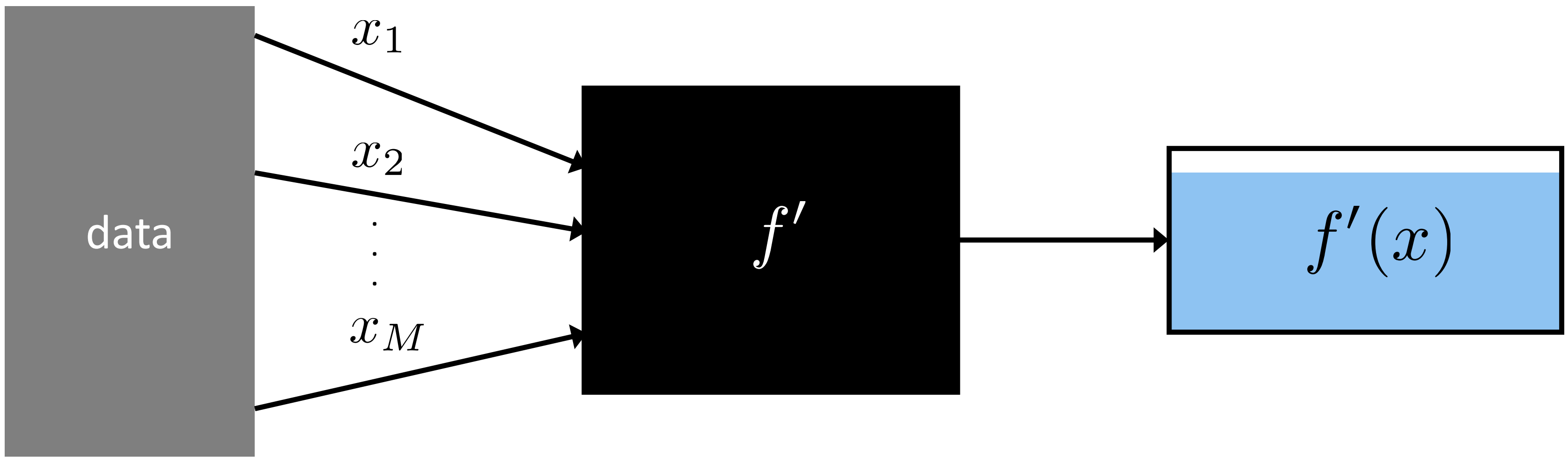
2 Consistency

https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx



2 Consistency

https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx



2 Consistency

$$\phi_1(f, x) \geq \phi_1(f', x)$$

https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx

Shap Values

- If consistency fails to hold, then we can't compare the attributed feature importances between any two models, because *then having a higher assigned attribution doesn't mean the model actually relies more on that feature*.
- If accuracy fails to hold then we don't know how the attributions of each feature combine to represent the output of the whole model.
- A proof from game theory (Shapley Values 1954) on the fair allocation of profits leads to a uniqueness result for feature attribution methods in machine learning.
- Tree SHAP is a fast algorithm that can exactly compute SHAP values for trees in polynomial time instead of the classical exponential runtime.

SHapley Additive exPlanation (SHAP) values

Base rate

20%

$E[f(x)]$

Prediction for John

55%

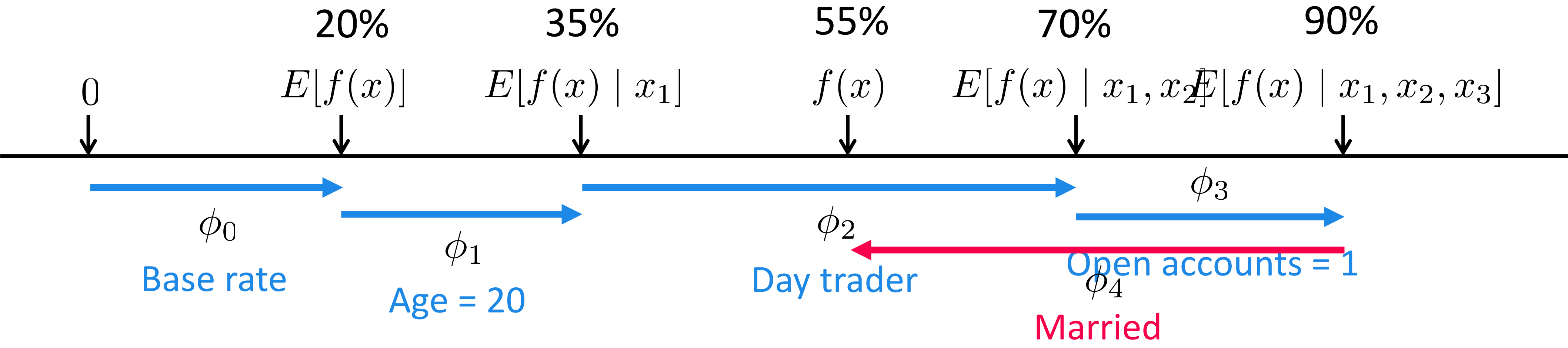
$f(x)$

0

How did we get here?

https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx

SHapley Additive exPlanation (SHAP) values

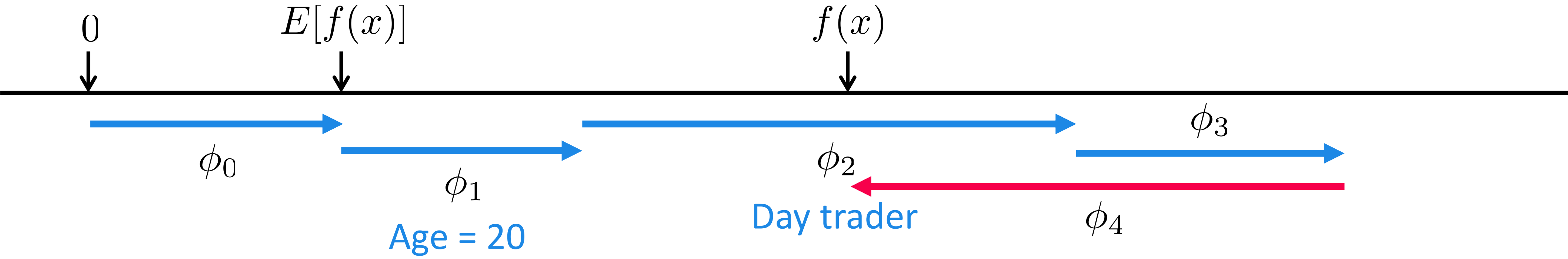


https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx

SHapley Additive exPlanation (SHAP) values

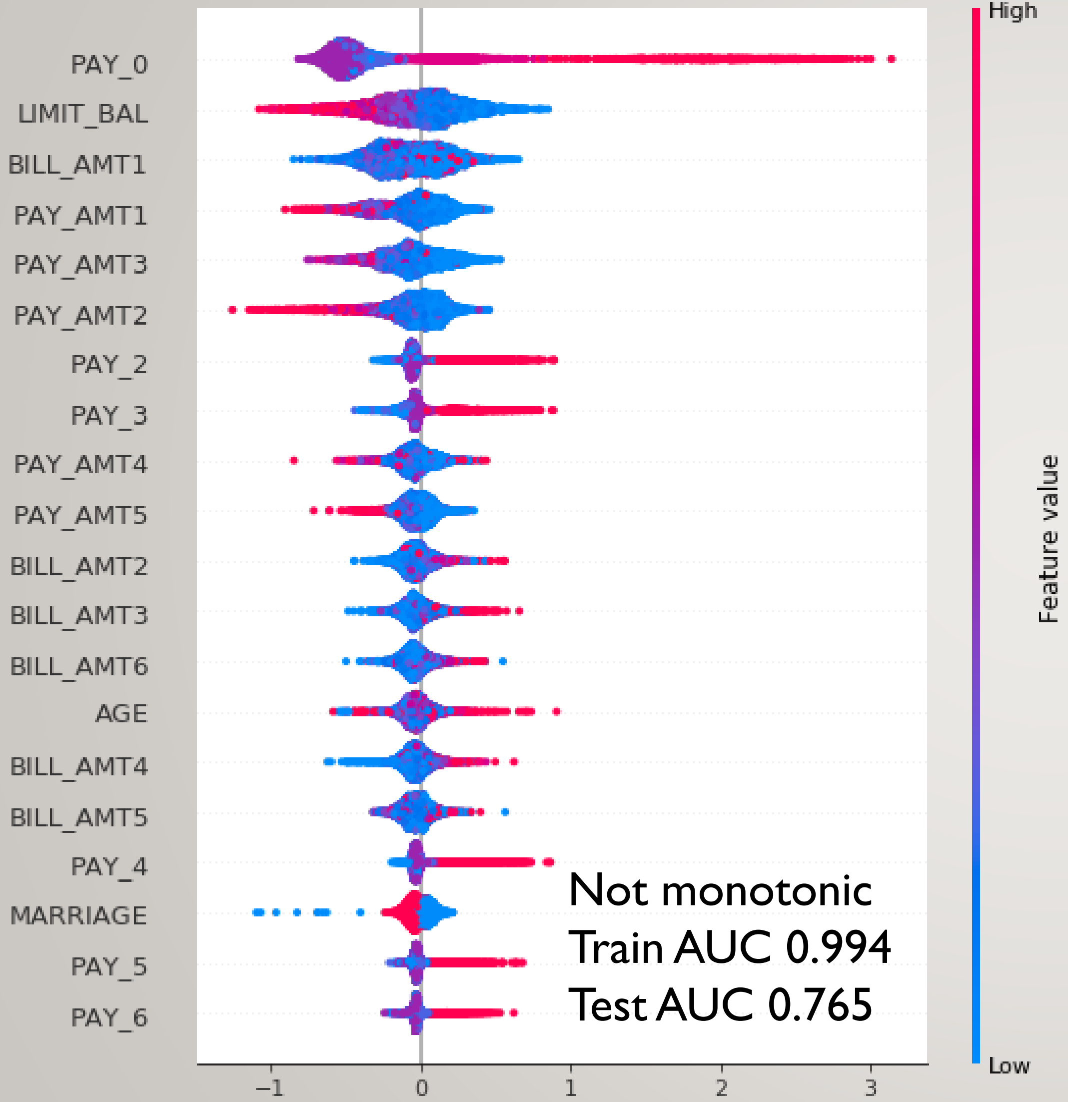
The order matters!

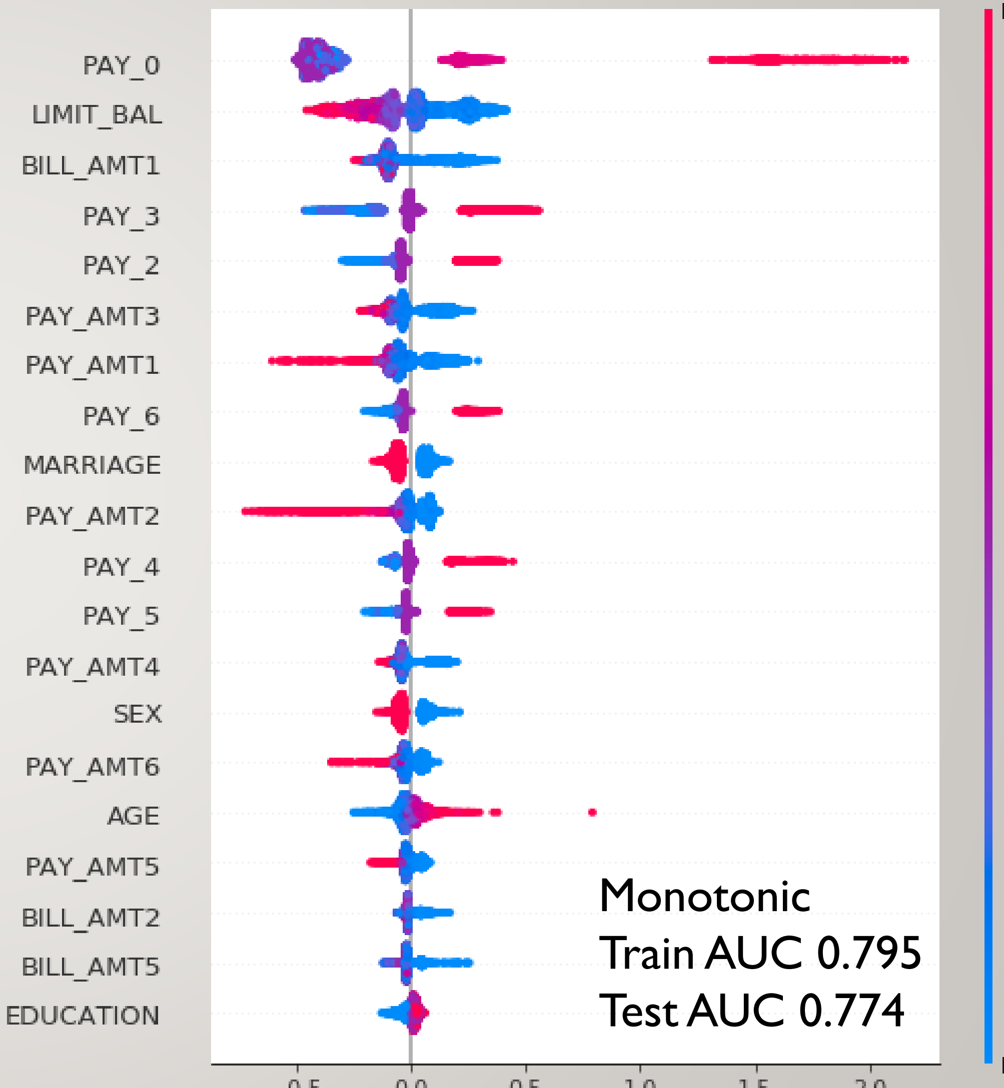
SHAP values result from averaging over all N! possible orderings.



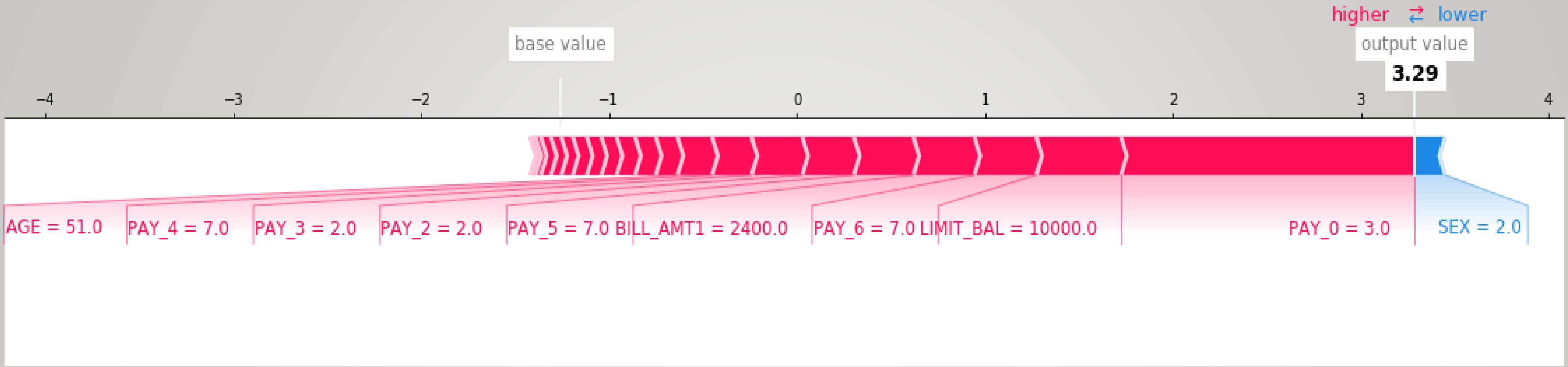
https://github.com/slundberg/shap/blob/master/docs/presentations/February%202018%20Talk.pptx

SHAP GLOBAL VALUES





SHAP Local Explanations for riskiest customer in dataset



Base Value – avg model output over the training dataset

Values are log-odds for XGBoost by default

Shap values will sum to model output

Reason Code	Value
PAY_0	3 months delayed
Limit Balance	Limit balance is too low at 10,000
PAY_6	7 months delayed

SHAP Variants

- Tree Shap - a fast and exact algorithm to compute shap values for trees and ensembles of trees. (xgboost/lightgbm/catboost/scikit-learn models)

- Deep Shap- fast, approximate algorithm to compute shap values for deep learning models that is based on connections between shap and the deeplift algorithm. (tensorflow/keras models)

- Gradient Shap - an implementation of expected gradients to approximate shap values for deep learning models. it is based on connections between shap and the integrated gradients algorithm.

- Kernel Shap - a model agnostic method to estimate shap values for any model. it makes no assumptions about the model type and is slower than the other model type specific algorithms.

- Scott Lundberg University of Washington https://github.com/slundberg/shap

Discussion Points

- What does a good interpretability workflow look like?
- Dataset management and e.g. classification prediction accuracy are well understood and usually very mature in organizations. What would a mature ML interpretability workflow look like?
- What tests would we do/results we could show?
- Is anyone working with regulators/in a regulated industry and could share some of their best practices?
- How are you presenting ML Interpretability results?

References

1. O'Reilly An Introduction to machine learning Interpretability
2. https://github.com/jphall663 - Patrick Hall H20.ai
3. https://christophm.github.io/interpretable-ml-book/ Christopher Molnar
4. https://github.com/slundberg/shap Scott Lundberg
5. https://github.com/pbiecek/xai_resources Przemysław Biecek
6. Statistical Modelling, The Two Cultures, Statistical Science 2001, Vol. 16, No. 3, 199-231 http://www2.math.uu.se/~thulin/mm/breiman.pdf