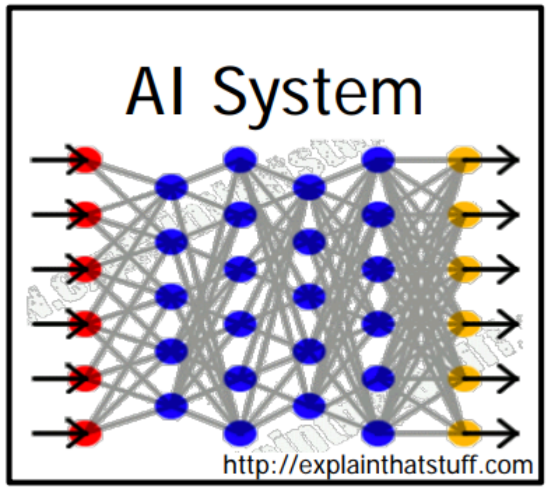
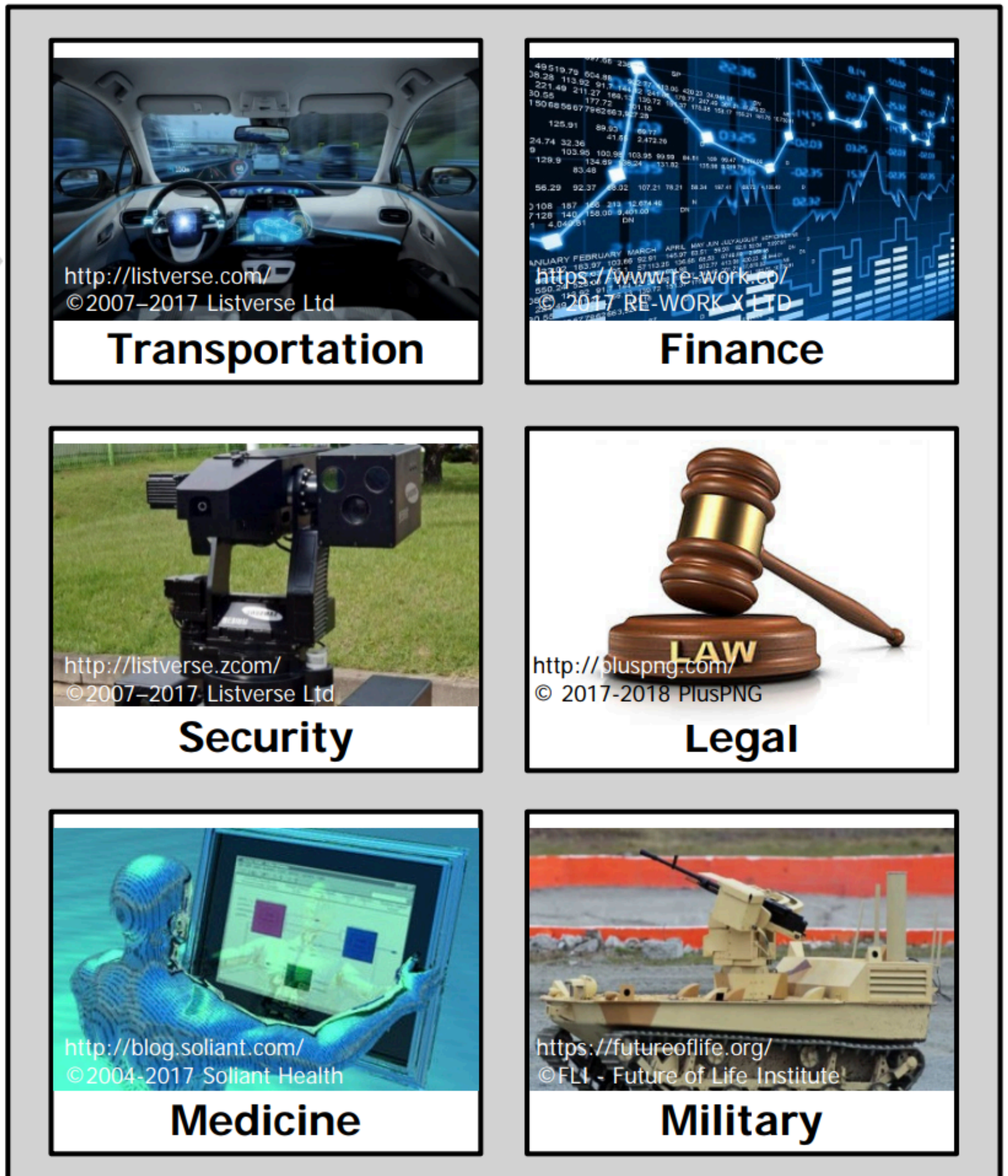
eXplainable Artificial Intelligence

The Need for Explainable AI



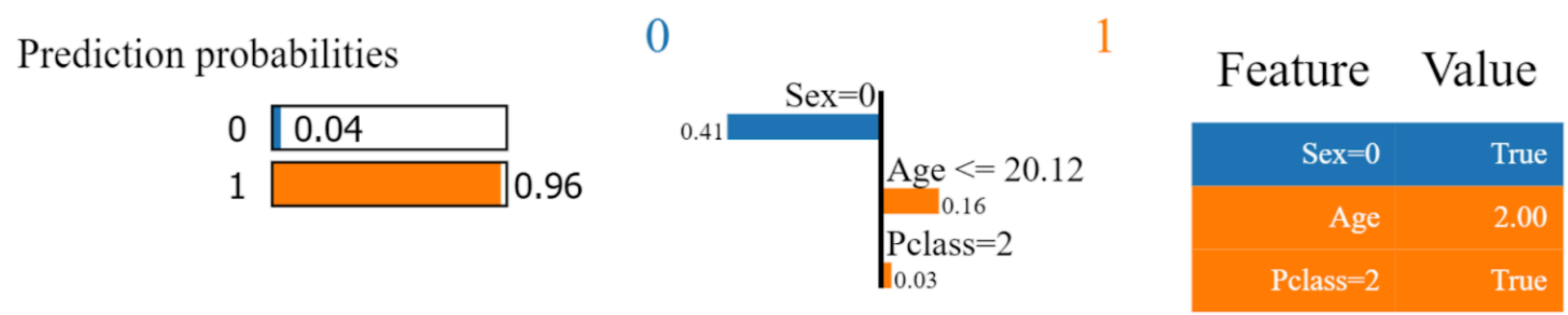
- We are entering a new age of AI applications
- Machine learning is the core technology
- Machine learning models are opaque, non-intuitive, and difficult for people to understand



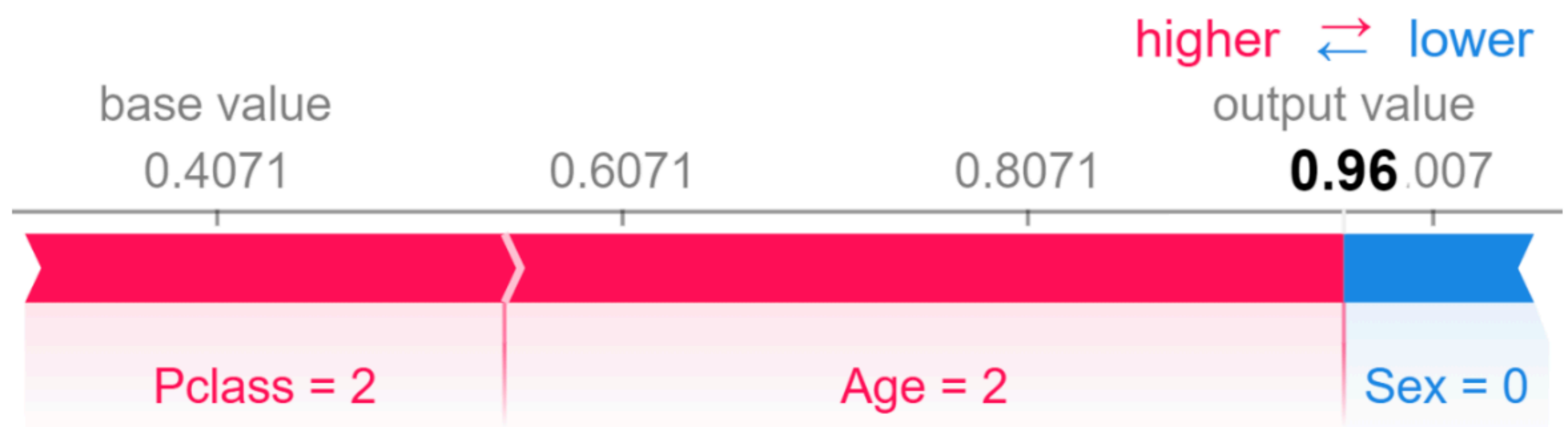
- Why did you do that?
- Why not something else?
- When do you succeed?
- When do you fail?
- When can I trust you?
- How do I correct an error?

- The current generation of AI systems offer tremendous benefits, but their effectiveness will be limited by the machine's inability to explain its decisions and actions to users
- Explainable AI will be essential if users are to understand, appropriately trust, and effectively manage this incoming generation of artificially intelligent partners

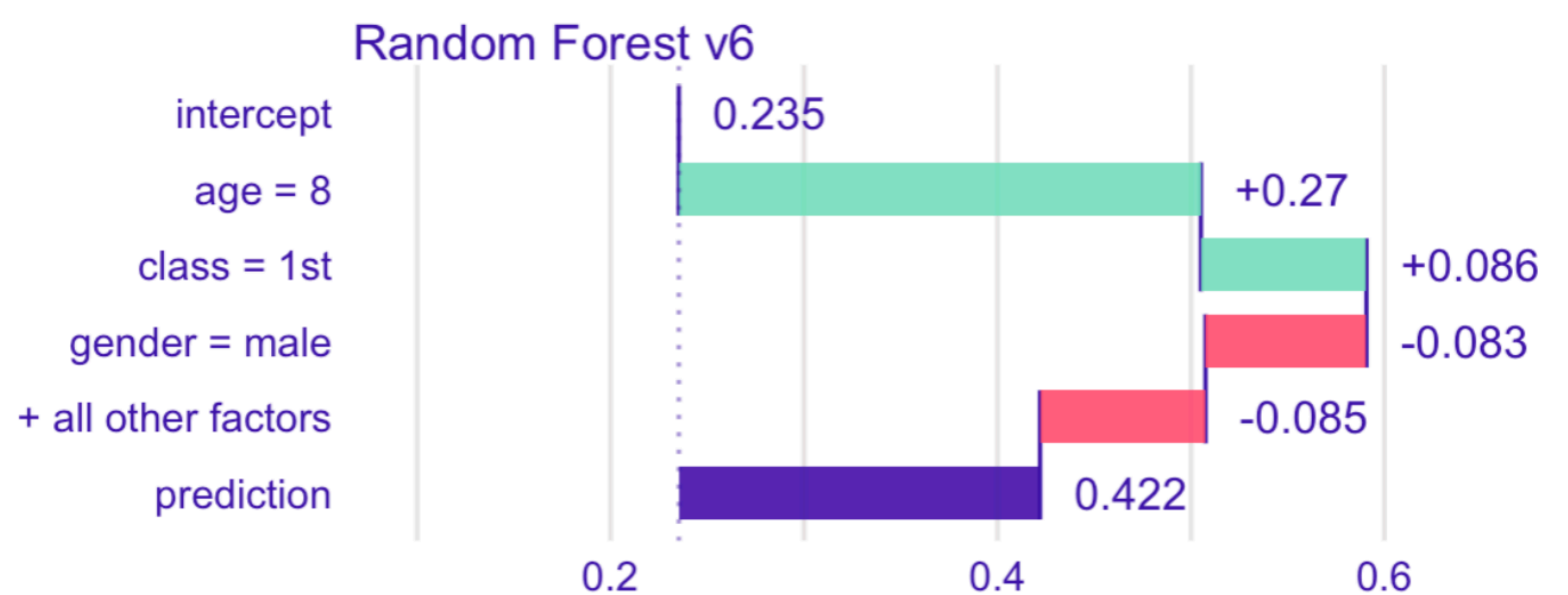
LIME



SHAP

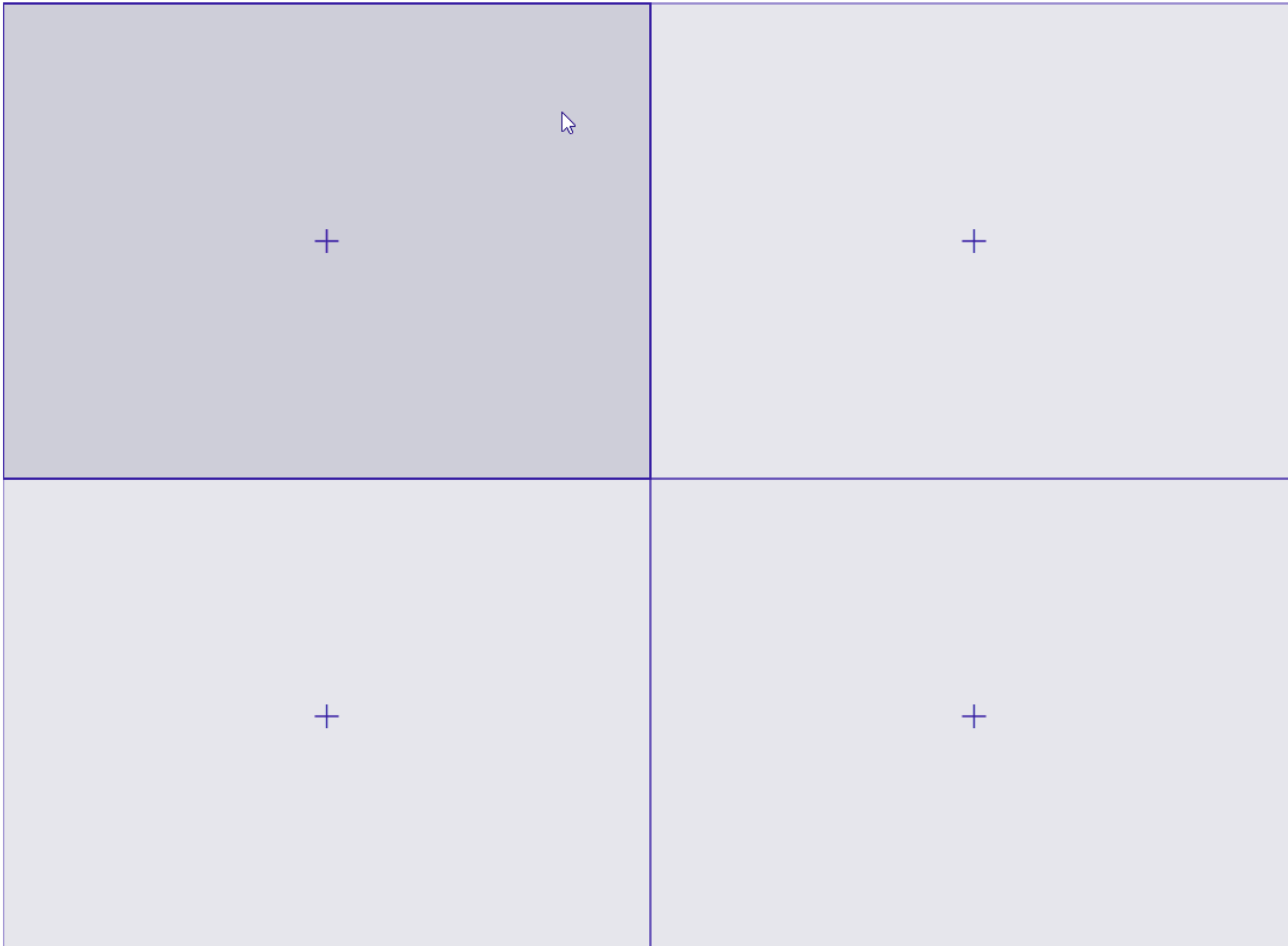


iBreakDown



Interactive Model Studio

Nancy ▾



Do Not Trust Additive Explanations

A Preprint

Alicja Gosiewska
Faculty of Mathematics and Information Science
Warsaw University of Technology
alicjagosiewska@gmail.com
https://orcid.org/0000-0001-6563-5742

Przemyslaw Biecek
Faculty of Mathematics, Informatics and Mechanics
University of Warsaw
Faculty of Mathematics and Information Science
Warsaw University of Technology
przemyslaw.biecek@gmail.com
https://orcid.org/0000-0001-8423-1823

December 2, 2019

Abstract

Explainable Artificial Intelligence (XAI) brings a lot of attention recently. Explainability is being presented as a remedy for a lack of trust in model predictions. Model agnostic tools such as LIME, SHAP, or Break Down promise instance level interpretability for any complex machine learning model. But how certain are these additive explanations? Can we rely on additive explanations for non-additive models?

In this paper, we (1) examine the behavior of the most popular instance-level explanations under the presence of interactions, (2) introduce a new method that can handle interactions for instance-level explanations, (3) perform a large scale benchmark to see how frequently additive explanations may be misleading.

1 INTRODUCTION

Predictive models are used in almost every aspect of our life, in school, at work, in hospitals, police stations, or dating services. They are useful, yet, at the same time can be a serious threat. Models that make unexplainable predictions may be harmful (O'Neil, 2016). Need for higher transparency and explainability of models is a hot topic of the recent year both in the Machine Learning community (Gill and Hall, 2018) as well as in the legal community that coined the phrase

SAFE ML: Surrogate Assisted Feature Extraction for Model Learning

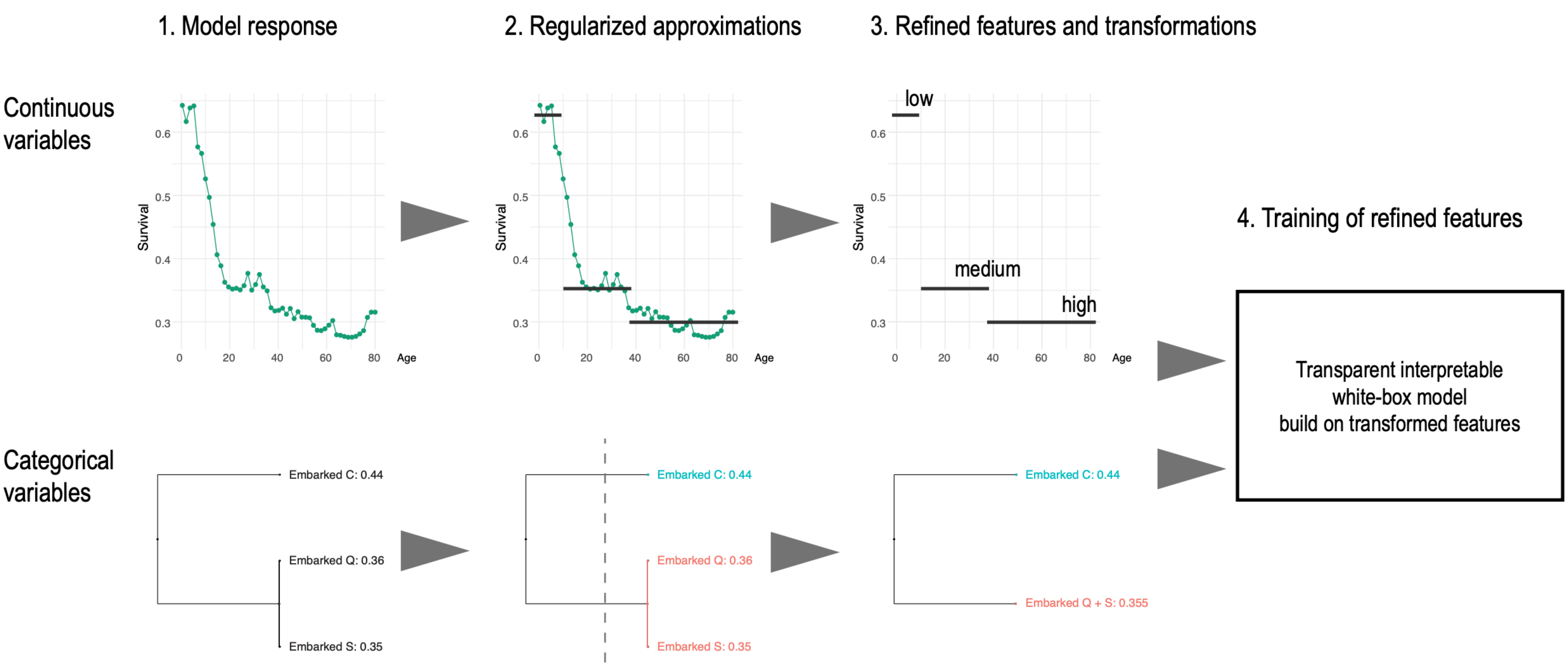
A Preprint

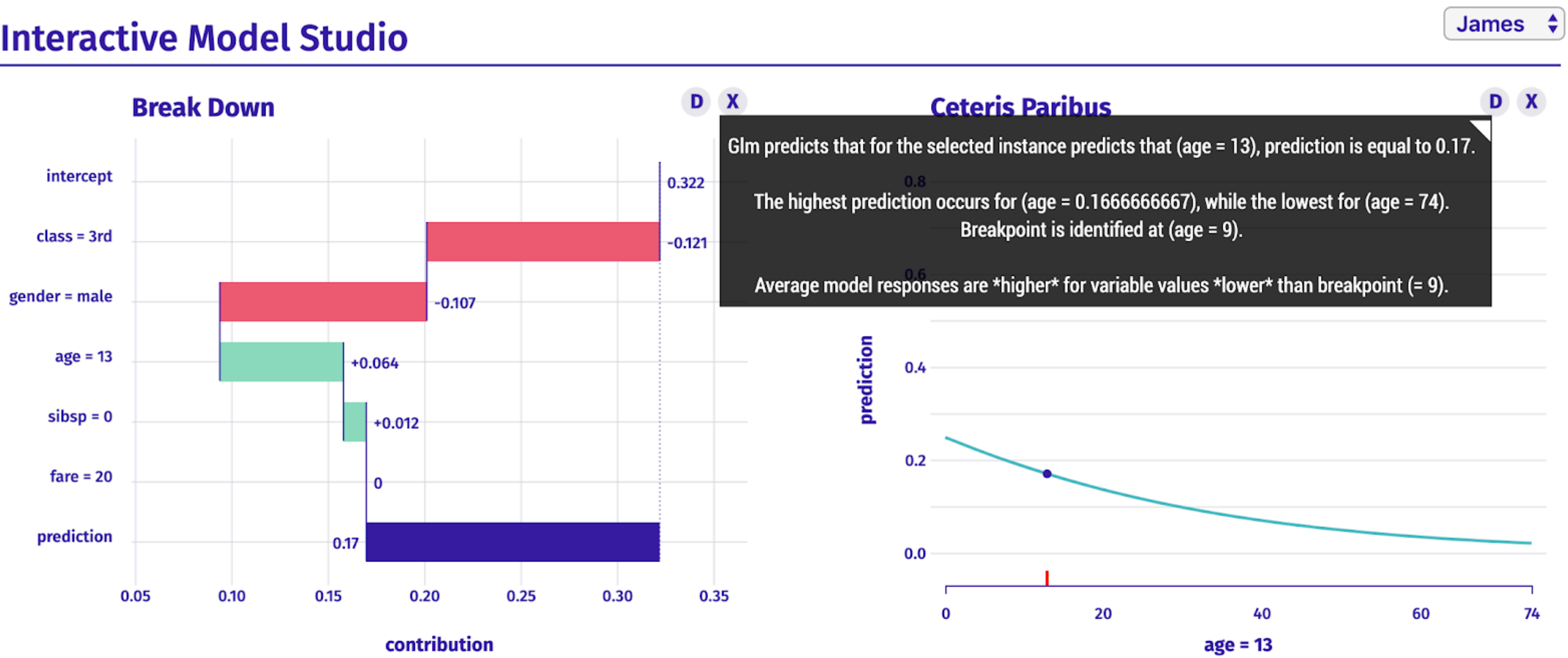
Alicja Gosiewska
Faculty of Mathematics and Information Science
Warsaw University of Technology
a.gosiewska@mini.pw.edu.pl

Aleksandra Gacek
Faculty of Mathematics and Information Science
Warsaw University of Technology
a.gacek@student.mini.pw.edu.pl

Piotr Lubon
Faculty of Mathematics and Information Science
Warsaw University of Technology
lubonp@student.mini.pw.edu.pl

Przemyslaw Biecek
Faculty of Mathematics, Informatics and Mechanics
University of Warsaw
Faculty of Mathematics and Information Science
Warsaw University of Technology
przemyslaw.biecek@gmail.com




Interactive Model Studio
James
Break Down
D X
intercept
class = 3rd
gender = male
age = 13
sibsp = 0
fare = 20
prediction
0.322
-0.121
-0.107
+0.064
+0.012
0
0.17
0.05
0.10
0.15
0.20
0.25
0.30
0.35
contribution
Ceteris Paribus
D X
Glm predicts that for the selected instance predicts that (age = 13), prediction is equal to 0.17.
The highest prediction occurs for (age = 0.1666666667), while the lowest for (age = 74).
Breakpoint is identified at (age = 9).
Average model responses are *higher* for variable values *lower* than breakpoint (= 9).
prediction
0.8
0.6
0.4
0.2
0.0
0
20
40
60
74
age = 13

XAI-BOT:

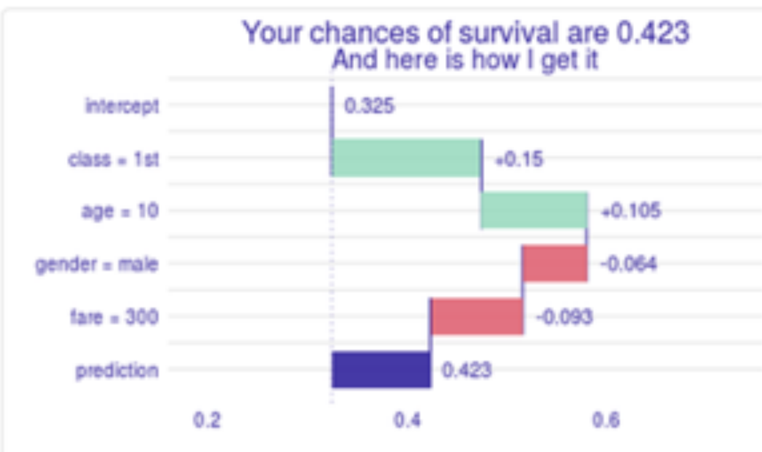
Talk to the model

Dialogflow Bot APP 12:38 PM
Your chance of survival equals 0.423. It's close to a toss of a coin!

Przemyslaw Biecek 12:38 PM
Why ?

Dialogflow Bot APP 12:38 PM
Creating a plot. It may take a few seconds...

Break down plot
This chart illustrates the contribution of variables to the final prediction (16 kB)



Przemyslaw Biecek 12:39 PM
what if I paid more?

Dialogflow Bot APP 12:39 PM
Creating a plot. It may take a few seconds...

Ceteris Paribus plot
This plot illustrates how the prediction changes when fare is changed and everything else is fixed (17 kB)

https://kmichael08.github.io

kmichael08.github.io

Titanic-explainer
GitHub Mail

40 female

Good news! You would've survived the disaster. Your chance of survival equals 0.8797

Bye

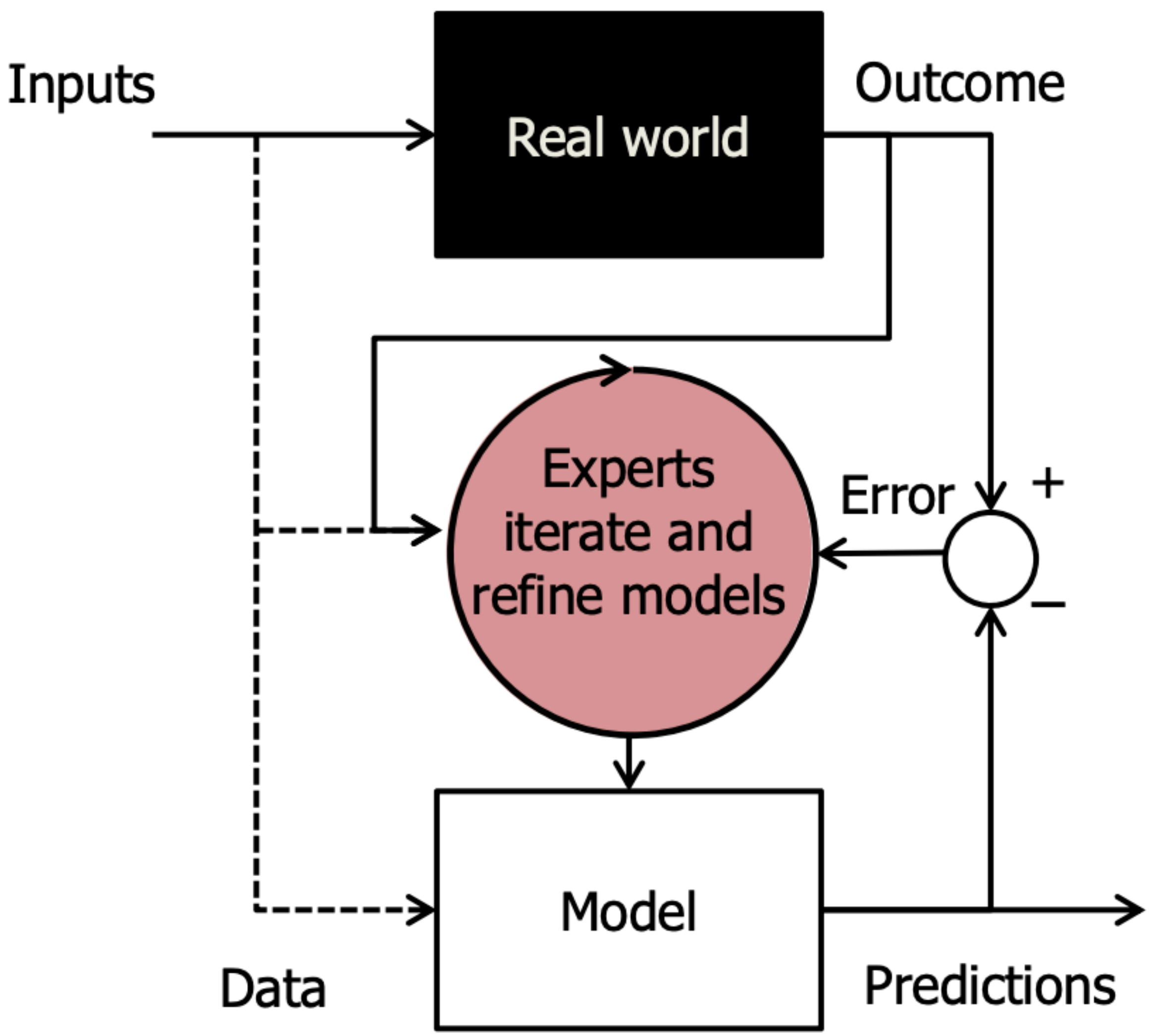
Bye :(Great talking to you! Come back later, as I will improve!

Type your message

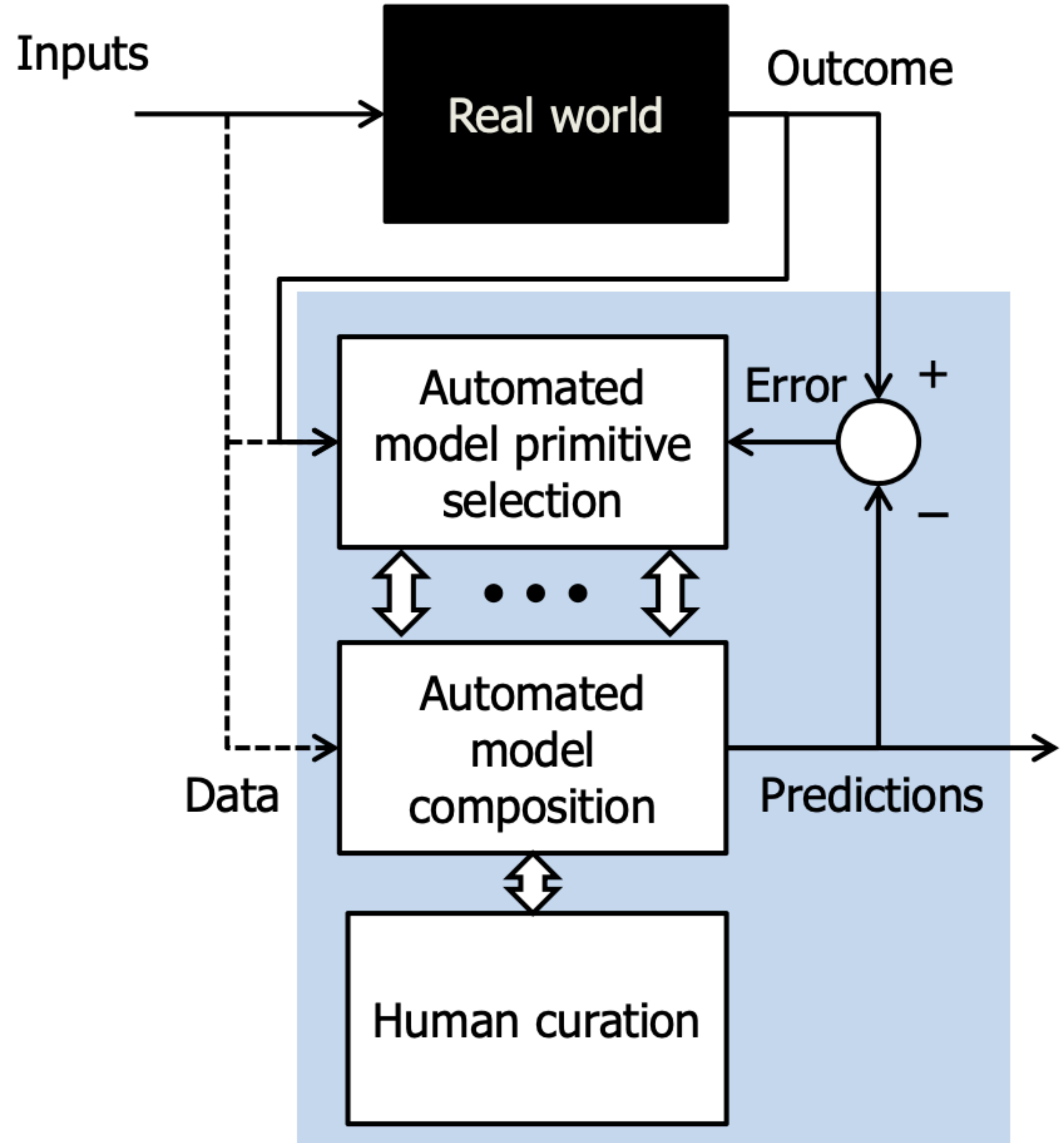
AutoML

D^3M: Data-driven discovery of models

Today: Manual

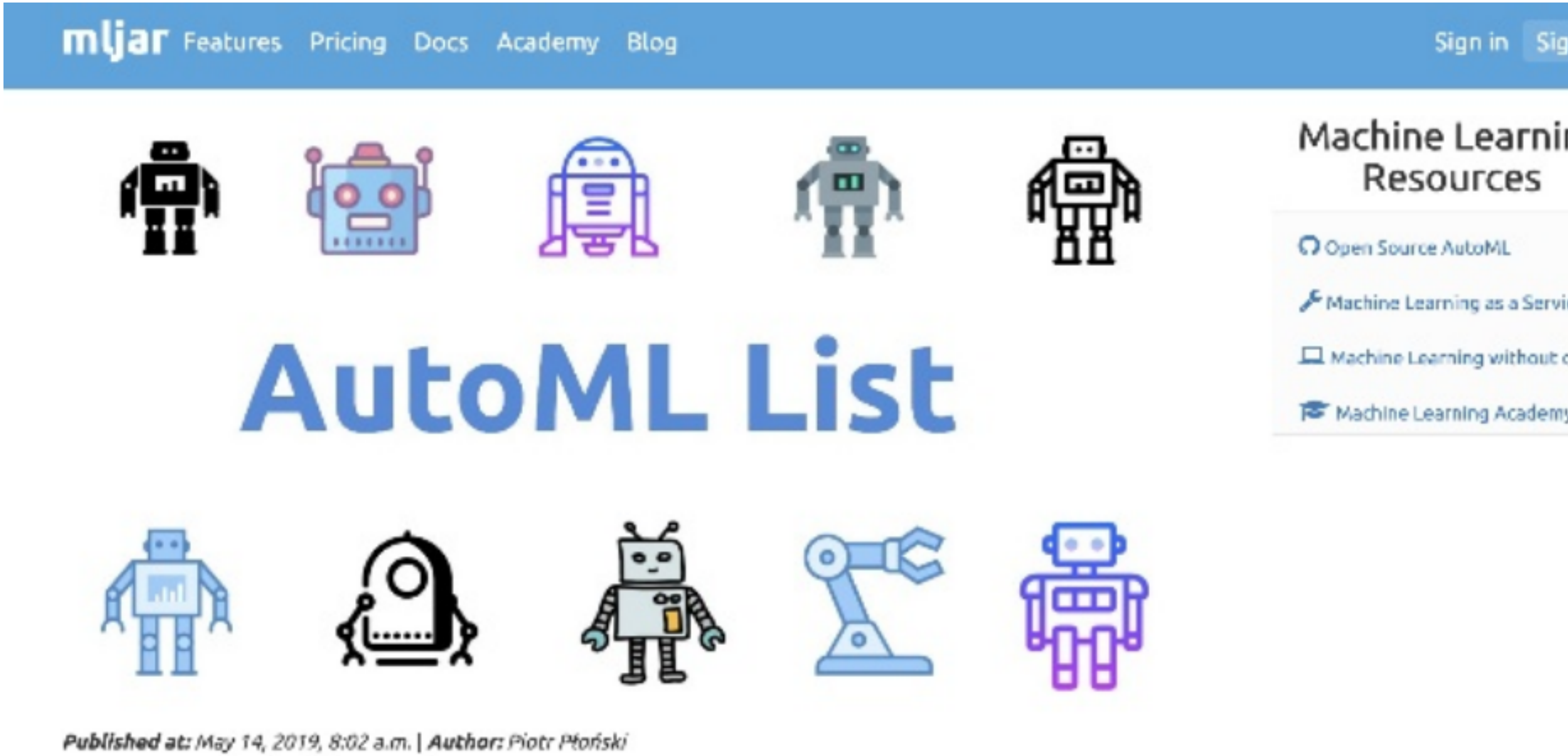


Tomorrow: Automated



- Model: representation of a real-world system
  - Examples
    - Inferring locations of images
    - Prediction of election outcomes
    - Estimation model for disease outbreaks
- Manual process: 10-1000s of person-years
- Teams of experts required to develop the model

- Automatically select problem-specific model primitives
  - Extend the library of modeling primitives
- Automatically compose complex models from primitives
- Facilitate user interaction with composed models



Auto-Keras	FeatureTools	Neural Network Intelligence (NNI)
auto-sklearn	h2o automl	tpot
automl-gs	Ludwig	TransmografAI
Auto-Weka	mljar-supervised	Auto_ml

Journal of Machine Learning Research 20 (2019) 1-32 Submitted 7/18; Revised 2/19; Published 3/19

Tunability: Importance of Hyperparameters of Machine Learning Algorithms

Philipp Probst PROBST@IBE.MED.UNI-MUENCHEN.DE
Institute for Medical Information Processing, Biometry and Epidemiology, LMU Munich
Marchioninistr. 15, 81377 München, Germany

Anne-Laure Boulesteix BOULESTEIX@IBE.MED.UNI-MUENCHEN.DE
Institute for Medical Information Processing, Biometry and Epidemiology, LMU Munich
Marchioninistr. 15, 81377 München, Germany

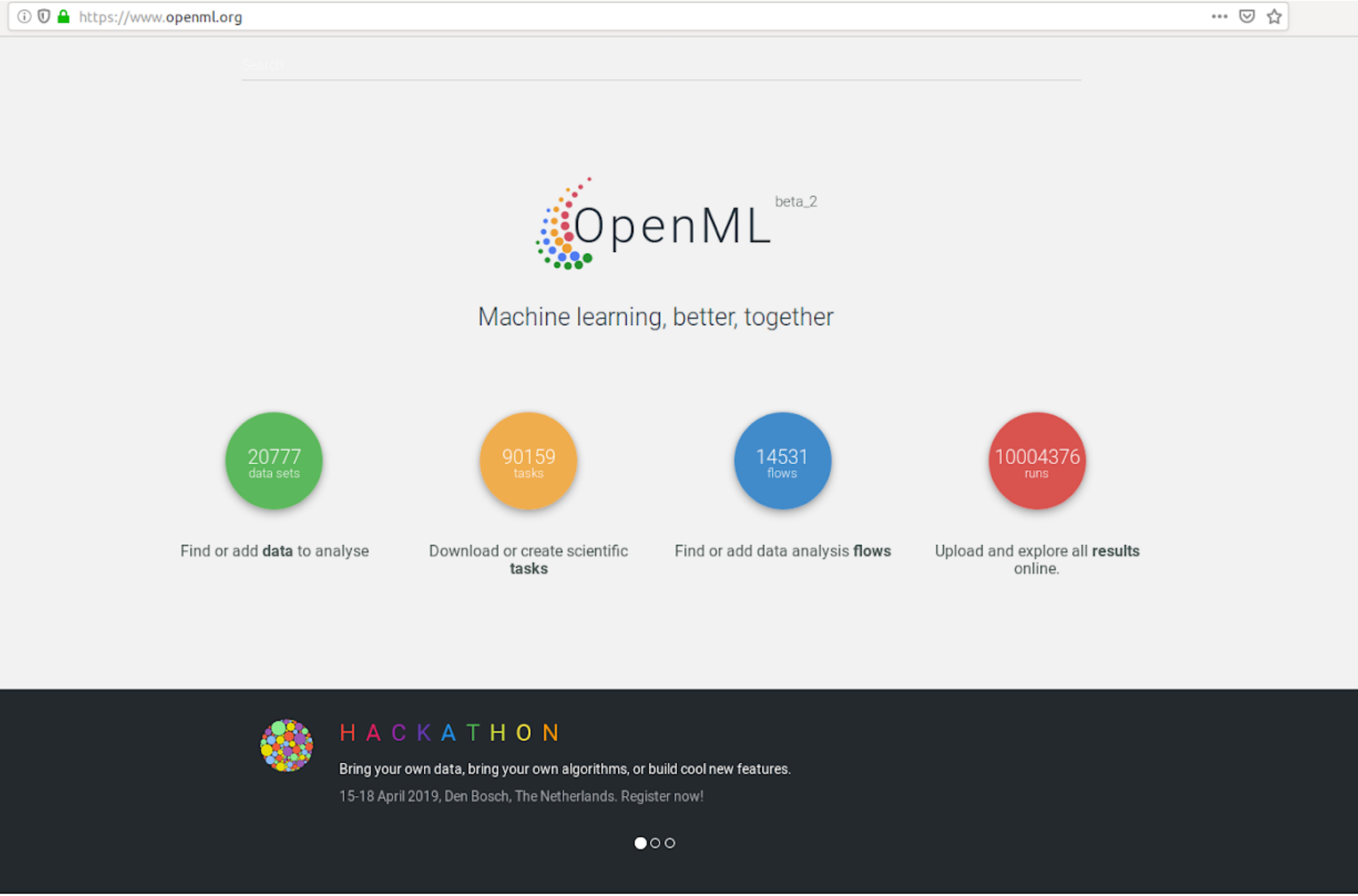
Bernd Bischl BERND.BISCHL@STAT.UNI-MUENCHEN.DE
Department of Statistics, LMU Munich
Ludwigstraße 33, 80539 München, Germany

Editor: Ryan Adams

Abstract

Modern supervised machine learning algorithms involve hyperparameters that have to be set before running them. Options for setting hyperparameters are default values from the software package, manual configuration by the user or configuring them for optimal predictive performance by a tuning procedure. The goal of this paper is two-fold. Firstly, we formalize the problem of tuning from a statistical point of view, define data-based defaults and suggest general measures quantifying the tunability of hyperparameters of algorithms. Secondly, we conduct a large-scale benchmarking study based on 38 datasets from the OpenML platform and six common machine learning algorithms. We apply our measures to assess the tunability of their parameters. Our results yield default values for hyperparameters and enable users to decide whether it is worth conducting a possibly time consuming tuning strategy, to focus on the most important hyperparameters and to choose adequate hyperparameter spaces for tuning.

Keywords: machine learning, supervised learning, classification, hyperparameters, tuning, meta learning


https://www.openml.org
OpenML
beta_2
Machine learning, better, together
20777
data sets
90159
tasks
14531
flows
10004376
runs
Find or add data to analyse
Download or create scientific tasks
Find or add data analysis flows
Upload and explore all results online.
HACKATHON
Bring your own data, bring your own algorithms, or build cool new features.
15-18 April 2019, Den Bosch, The Netherlands. Register now!

model_param_index	1036_1	1036_10	1049_9	1050_1	1050_
gbm_10	0.9991615	0.9989143	0.9360491	0.7967995	0.835
gbm_11	0.9992034	0.9988122	0.9302455	0.8016304	0.839
gbm_2	0.9991845	0.9988492	0.9306920	0.8230676	0.877
gbm_3	0.9992535	0.9986928	0.9328125	0.8302134	0.872
gbm_7	0.9992494	0.9986342	0.9335938	0.8420894	0.881
gbm_9	0.9991469	0.9989881	0.9301339	0.7863325	0.831
kknn_1	0.9534735	0.9704949	0.9048549	0.8125000	0.819
kknn_10	0.9534735	0.9704949	0.9048549	0.8125000	0.819
randomForest_1	0.9990141	0.9984963	0.9516183	0.8634259	0.893
randomForest_10	0.9989963	0.9983714	0.9477679	0.8469203	0.883
randomForest_11	0.9989503	0.9984312	0.9515067	0.8600543	0.893
randomForest_2	0.9990037	0.9985202	0.9520647	0.8614130	0.884
randomForest_3	0.9989807	0.9984800	0.9523996	0.8581421	0.886
randomForest_4	0.9990005	0.9984768	0.9536272	0.8640801	0.894
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Journal of Machine Learning Research 20 (2019) 1-32 Submitted 7/18; Revised 2/19; Published 3/19

Tunability: Importance of Hyperparameters of Machine Learning Algorithms

Philipp Probst PROBST@IBE.MED.UNI-MUENCHEN.DE
Institute for Medical Information Processing, Biometry and Epidemiology, LMU Munich
Marchioninistr. 15, 81377 München, Germany

Anne-Laure Boulesteix BOULESTEIX@IBE.MED.UNI-MUENCHEN.DE
Institute for Medical Information Processing, Biometry and Epidemiology, LMU Munich
Marchioninistr. 15, 81377 München, Germany

Bernd Bischl BERND.BISCHL@STAT.UNI-MUENCHEN.DE
Department of Statistics, LMU Munich
Ludwigstraße 33, 80539 München, Germany

Editor: Ryan Adams

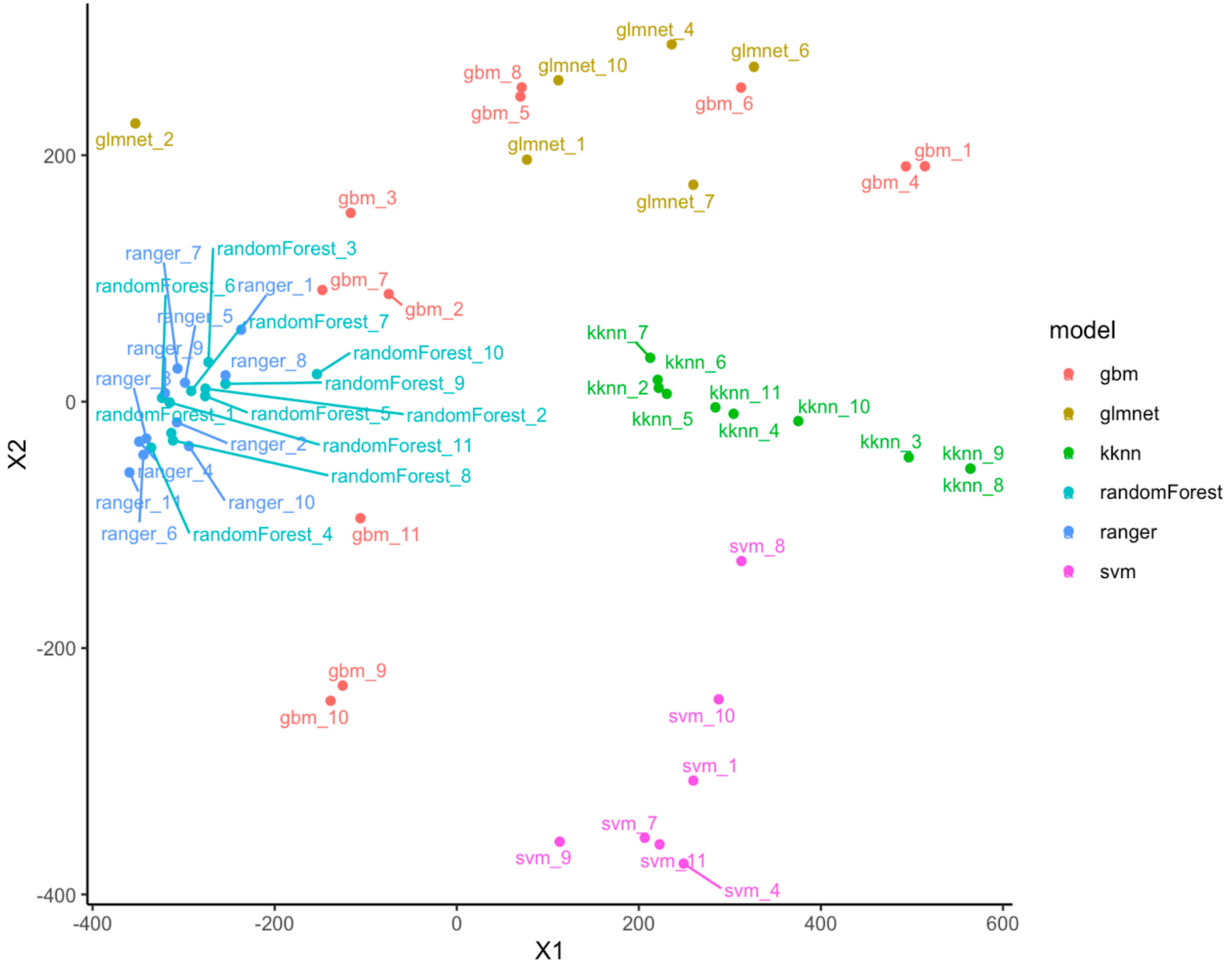
Abstract

Modern supervised machine learning algorithms involve hyperparameters that have to be set before running them. Options for setting hyperparameters are default values from the software package, manual configuration by the user or configuring them for optimal predictive performance by a tuning procedure. The goal of this paper is two-fold. Firstly, we formalize the problem of tuning from a statistical point of view, define data-based defaults and suggest general measures quantifying the tunability of hyperparameters of algorithms. Secondly, we conduct a large-scale benchmarking study based on 38 datasets from the OpenML platform and six common machine learning algorithms. We apply our measures to assess the tunability of their parameters. Our results yield default values for hyperparameters and enable users to decide whether it is worth conducting a possibly time consuming tuning strategy, to focus on the most important hyperparameters and to choose adequate hyperparameter spaces for tuning.

Keywords: machine learning, supervised learning, classification, hyperparameters, tuning, meta-learning

Parameter	Def.P	Def.O	Tun.P	Tun.O	$q_{0.05}$	$q_{0.95}$
glmnet			0.069	0.024		
alpha	1	0.403	0.038	0.006	0.009	0.981
lambda	0	0.004	0.034	0.021	0.001	0.147
rpart			0.038	0.012		
cp	0.01	0	0.025	0.002	0	0.008
maxdepth	30	21	0.004	0.002	12.1	27
minbucket	7	12	0.005	0.006	3.85	41.6
minsplit	20	24	0.004	0.004	5	49.15
kknn			0.031	0.006		
k	7	30	0.031	0.006	9.95	30
svm			0.056	0.042		
kernel	radial	radial	0.030	0.024		
cost	1	682.478	0.016	0.006	0.002	920.582
gamma	$1/p$	0.005	0.030	0.022	0.003	18.195
degree	3	3	0.008	0.014	2	4
ranger			0.010	0.006		
num.trees	500	983	0.001	0.001	206.35	1740.15
replace	TRUE	FALSE	0.002	0.001		
mple.fraction	1	0.703	0.004	0.002	0.323	0.974
mtry	$\sqrt{p}$	$p \cdot 0.257$	0.006	0.003	0.035	0.692
dered.factors	TRUE	FALSE	0.000	0.000		
min.node.size	1	1	0.001	0.001	0.007	0.513
xgboost			0.043	0.014		
nrounds	500	4168	0.004	0.002	920.7	4550.95
eta	0.3	0.018	0.006	0.005	0.002	0.355
subsample	1	0.839	0.004	0.002	0.545	0.958
booster	gbtree	gbtree	0.015	0.008		
max_depth	6	13	0.001	0.001	5.6	14
child_weight	1	2.06	0.008	0.002	1.295	6.984
mple_bytree	1	0.752	0.006	0.001	0.419	0.864
mple_bylevel	1	0.585	0.008	0.001	0.335	0.886
lambda	1	0.982	0.003	0.002	0.008	29.755
alpha	1	1.113	0.003	0.002	0.002	6.105

s (package defaults (Def.P) and optimal defaults (Def.O)), tunability
erparameters with the package defaults (Tun.P) and our optimal defaul
) as reference and tuning space quantiles ($q_{0.05}$ and $q_{0.95}$) for different p
rameters of the algorithms



X2
X1
model
gbm
glmnet
kknn
randomForest
ranger
svm
200
0
-200
-400
-400
-200
0
200
400
600
glmnet_4
glmnet_6
glmnet_10
glmnet_2
glmnet_1
glmnet_7
gbm_8
gbm_5
gbm_6
gbm_1
gbm_4
gbm_3
gbm_7
gbm_2
gbm_11
gbm_9
gbm_10
ranger_7
ranger_1
ranger_5
ranger_9
ranger_8
ranger_3
ranger_2
ranger_4
ranger_11
ranger_10
ranger_6
randomForest_3
randomForest_6
randomForest_7
randomForest_10
randomForest_9
randomForest_1
randomForest_5
randomForest_2
randomForest_11
randomForest_8
randomForest_4
kknn_7
kknn_6
kknn_2
kknn_11
kknn_5
kknn_10
kknn_4
kknn_3
kknn_9
kknn_8
svm_8
svm_10
svm_1
svm_7
svm_9
svm_11
svm_4

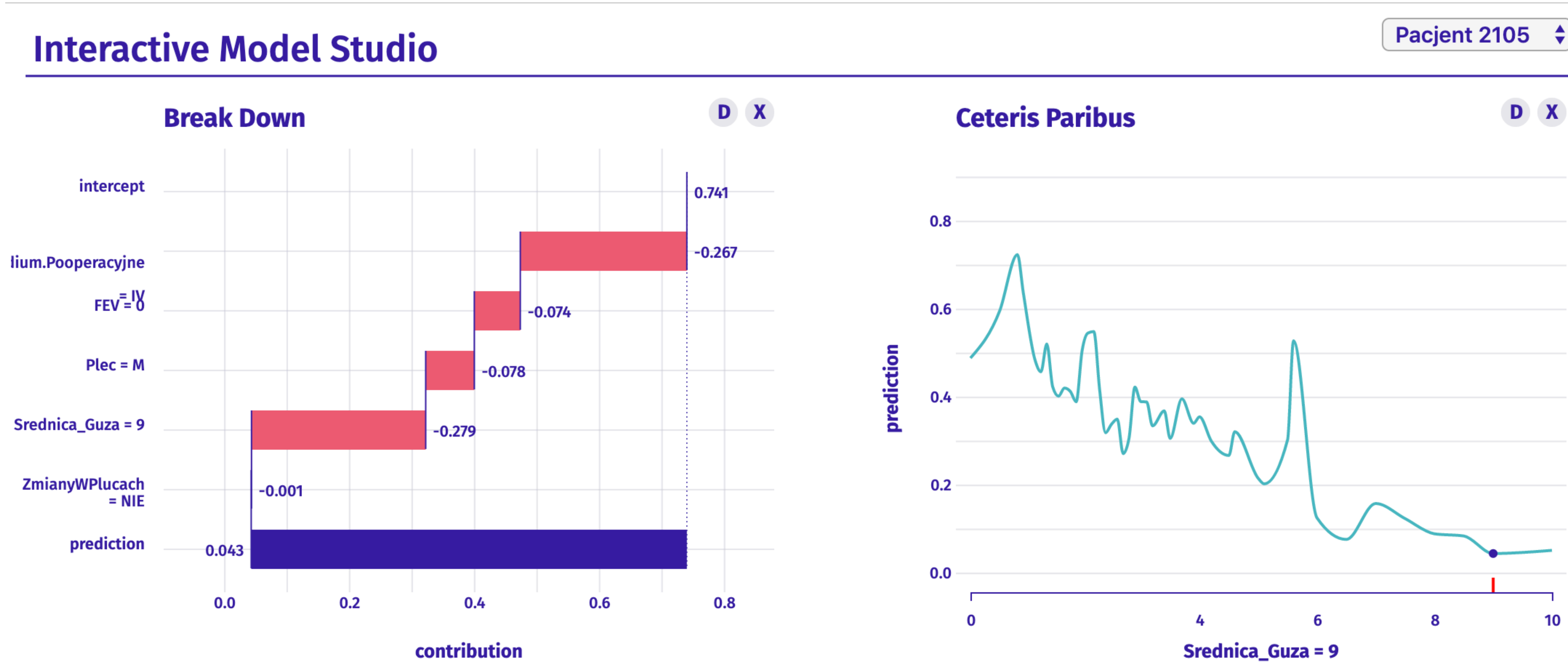
ML in computational oncology

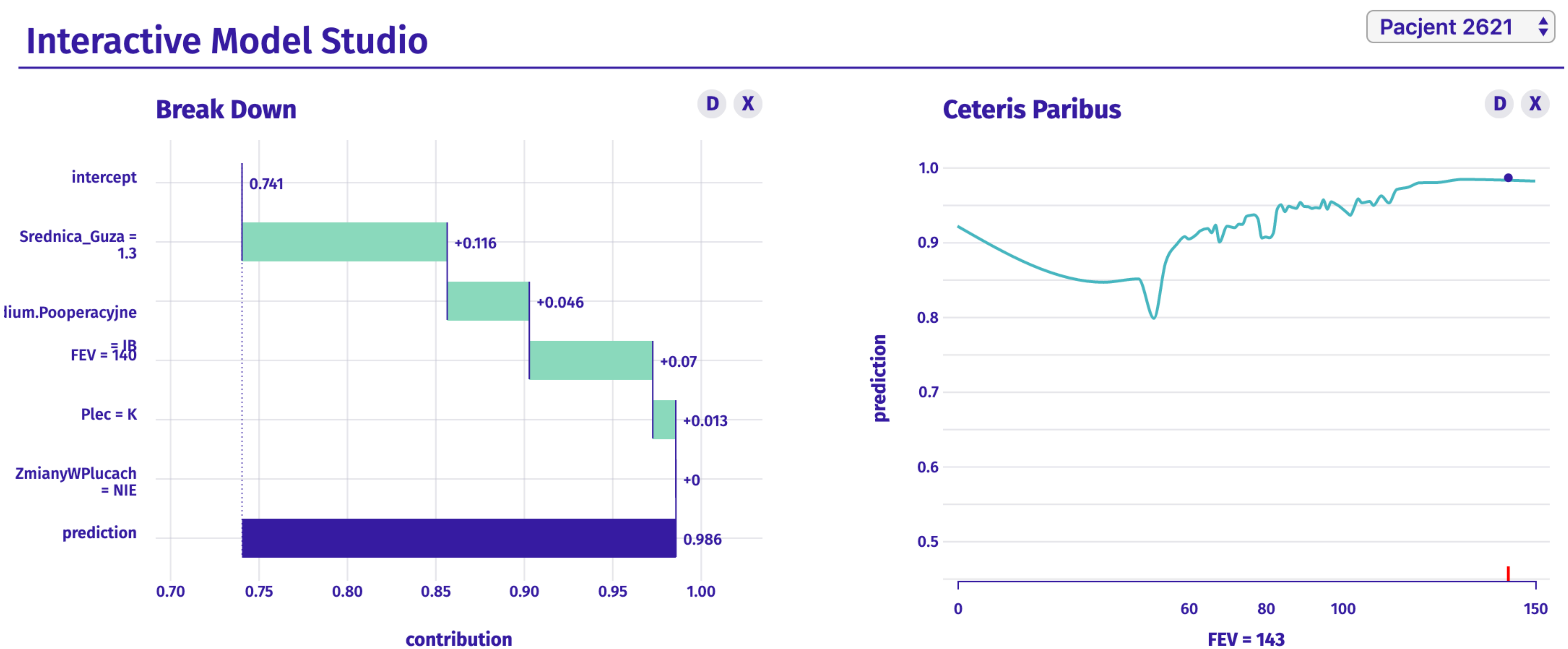
Explainable machine learning for modeling of early postoperative mortality in lung cancer*

Katarzyna Kobylińska[1][0000−0002−0292−4982], Tomasz Mikołajczyk[4], Mariusz Adamek[2][0000−0002−1885−9257], Tadeusz Orłowski[3], and Przemysław Biecek[1,4][0000−0001−8423−1823]

[1] University of Warsaw,Faculty of Mathematics, Informatics and Mechanics, Poland
[2] Faculty of Medicine and Dentistry, Medical University of Silesia
[3] National Institute of Tuberculosis and Lung Diseases
[4] Faculty of Mathematics and Information Science, Warsaw University of Technology

Abstract. In recent years we see an increasing interest in applications of complex machine learning methods to medical problems. Models based on deep neural networks or ensembles are more and more popular in diagnostic, personalized medicine [Hamet and Tremblay, 2017] or screening studies [Scheeder et al., 2018]. Partially because they are accurate and easy to train. But such models may be hard to understand and interpret. In high stake decisions, especially in medicine, the understanding of factors that drive model decisions is crucial. Lack of model understanding creates a serious risk in applications.


Interactive Model Studio
Pacjent 2105
Break Down
D X
intercept
lium.Pooperacyjne = IV
FEV = 0
Plec = M
Srednica_Guza = 9
ZmianyWPlucach = NIE
prediction
0.741
-0.267
-0.074
-0.078
-0.279
-0.001
0.043
0.0
0.2
0.4
0.6
0.8
contribution
Ceteris Paribus
D X
prediction
0.8
0.6
0.4
0.2
0.0
0
4
6
8
10
Srednica_Guza = 9


Interactive Model Studio
Pacjent 2621
Break Down
D X
intercept
Srednica_Guza = 1.3
lium.Pooperacyjne = IB
FEV = 140
Plec = K
ZmianyWPlucach = NIE
prediction
0.741
+0.116
+0.046
+0.07
+0.013
+0
0.986
0.70
0.75
0.80
0.85
0.90
0.95
1.00
contribution
Ceteris Paribus
D X
prediction
1.0
0.9
0.8
0.7
0.6
0.5
0
60
80
100
150
FEV = 143