

dalex: Responsible Machine Learning with Interactive Explainability and Fairness in Python

Hubert Baniecki

Warsaw University of Technology, Poland

Joint Statistical Meetings

August 8, 2022





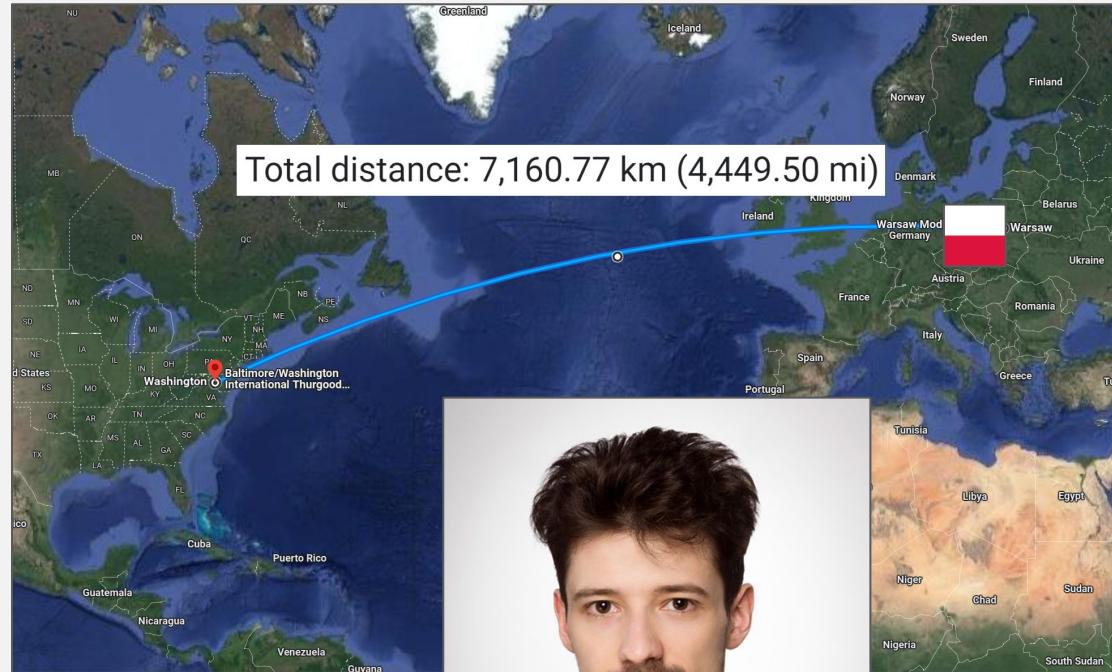
Hi!

Finishing my master's in **Data Science** at
Warsaw University of Technology, Poland

Research in explainable and
interpretable **machine learning**

Main author of the **dalex** Python package
John M. Chambers Statistical Software Award (2022)

<https://hbaniecki.com/jsm2022>





John M. Chambers Statistical Software Award

Many thanks to organizers and the review panel: **Raymond Wong** (Texas A&M University),
Yixuan Qiu (Shanghai University of Finance and Economics), **Samantha Tyner** (Tritura),
Philip Waggoner (YouGov America, Northwestern University, University of Virginia).

Congratulations to **Vittorio Orlandi** (Duke University) for the award's honorable mention,
for the R package **FLAME**: interpretable matching methods for performing causal inference
on observational data with discrete covariates.



dalex team

Przemysław Biecek



Wojciech Kretowicz, Mateusz Krzyziński, Piotr Piątyszek, Jakub Wiśniewski, Artur Żółkowski



Faculty of Mathematics and Information Science, Warsaw University of Technology



National Science Centre (Poland) grants No. 2017/27/B/ST6/0130 and 2019/34/E/ST6/00052





Outline

1. Responsible machine learning - Why should I care?
2. dalex: the main idea and software - What and how?
3. Educational materials - Where to start?
4. What's next?

This article was published more than 3 years ago

BUSINESS

Apple Card algorithm sparks gender bias allegations against Goldman Sachs

Entrepreneur David Heinemeier Hansson says his credit limit was 20 times that of his wife, even though he had a higher credit score

By Taylor Telford

November 11, 2019 at 10:44 a.m. EST

Many articles have been published in 2020 describing new machine learning-based models for [detection and prognostication of COVID-19], but it is unclear which are of potential clinical utility. [...] Our review finds that none of the models identified are of potential clinical use due to methodological flaws and/or underlying biases.

2019

1. Decision making: from credit scoring to precision diagnostics in bio-medicine

nature machine intelligence

Explore content ▾ About the journal ▾ Publish with us ▾

2021

[nature](#) > [nature machine intelligence](#) > [analyses](#) > [article](#)

Analysis | Open Access | Published: 15 March 2021

Common pitfalls and recommendations for using machine learning to detect and prognosticate for COVID-19 using chest radiographs and CT scans

Michael Roberts✉, Derek Driggs, Matthew Thorpe, Julian Gilbey, Michael Yeung, Stephan Ursprung, Angelica I. Aviles-Rivero, Christian Etmann, Cathal McCague, Lucian Beer, Jonathan R. Weir-McCall, Zhongzhao Teng, Effrossyni Gkrania-Klotsas, AIX-COVNET, James H. F. Rudd, Evis Sala & Carola-Bibiane Schönlieb

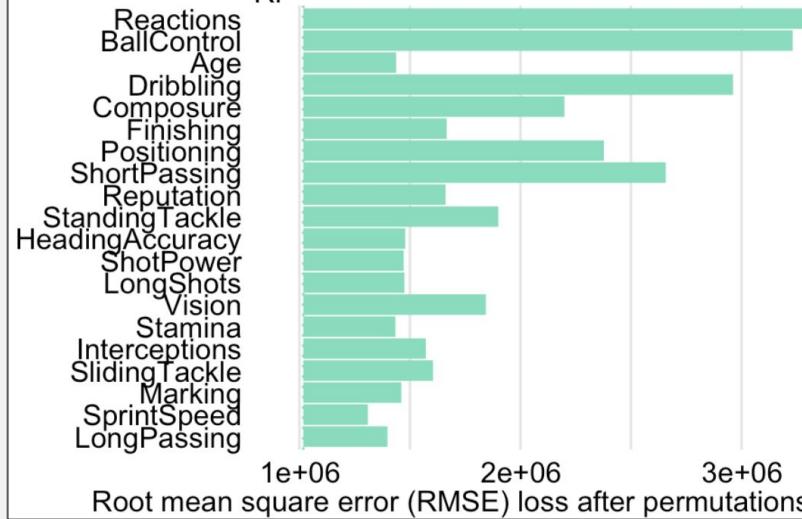
[Nature Machine Intelligence](#) 3, 199–217 (2021) | [Cite this article](#)

74k Accesses | 237 Citations | 1159 Altmetric | [Metrics](#)

HOMER

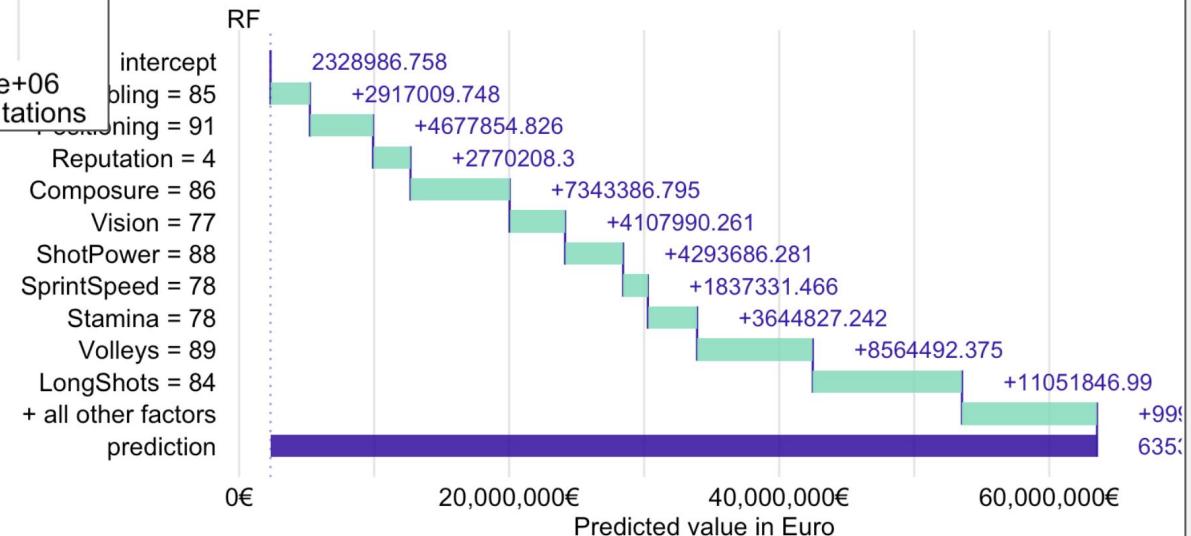
Feature Importance

RF



2. Knowledge discovery through explanatory model analysis

Break-down plot for Robert Lewandowski



Sponsor

Edit Pins

Unwatch 43

Fork 142

Starred 1.1k

Code

Issues 20

Pull requests 2

Discussions

Actions

Projects

...

master

Go to file

Add file

Code

README.md



moDel Agnostic Language for Exploration and eXplanation

R-CMD-check passing

coverage 85% CRAN 2.4.2

downloads 169K

DrWhy BackBone

Python-check passing

python 3.6 | 3.7 | 3.8 | 3.9

pypi package 1.4.1 downloads 187k



About

moDel Agnostic Language for Exploration and eXplanation

Readme

GPL-3.0 license

Cite this repository

1.1k stars

43 watching

142 forks

Used by 49



Contributors 20

DALEX project in numbers:

- 4+ years
- 50+ package releases
- 350K+ downloads
- 1000+ stars on GitHub
- 50+ dependant software projects

JMLR, 2021

dalex: Responsible Machine Learning with Interactive Explainability and Fairness in Python

Hubert Baniecki¹

HUBERT.BANIECKI.STUD@PW.EDU.PL

Wojciech Kretowicz¹

Submitted 6/18; Revised 10/18; Published 11/18

Piotr Piatyszek¹

Journal of Machine Learning Research 19 (2018) 1-5

Jakub Wisniewski¹

Przemysław Biecek^{1,2}

JMLR, 2018

¹Faculty of Mathematics and Informati

²Samsung Research & Development Ins

Editor: Joaquin Vanschoren

DALEX: Explainers for Complex Predictive Models in R

Przemysław Biecek

PRZEMYSŁAW.BIECEK@GMAIL.COM

Faculty of Mathematics and Information Science, Warsaw University of Technology

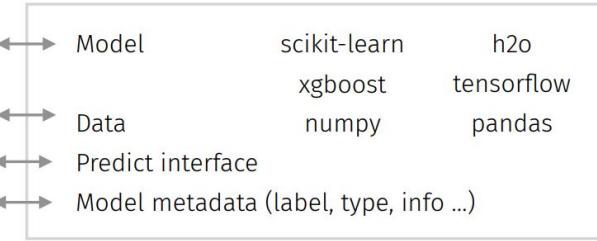
75 Koszykowa Street, Warsaw, Poland

Samsung Research Poland

Editor: Alexandre Gramfort



A. Explainer: Uniform abstraction over predictive models



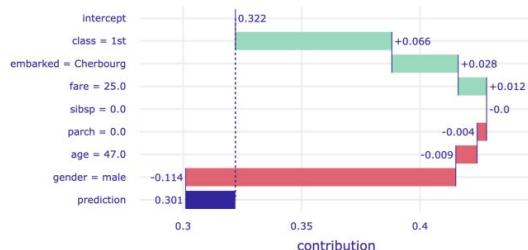
B. Consistent grammar for model analysis

```
import dalex as dx
explainer = dx.Explainer(model, X, y)

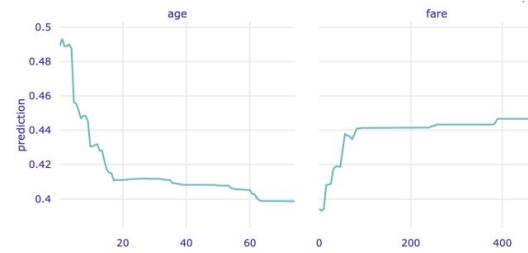
explanation = explainer.model_parts()
explanation.result
explanation.plot()

explainer.predict_parts(new_observation).result
explainer.predict_parts(new_observation).plot()
```

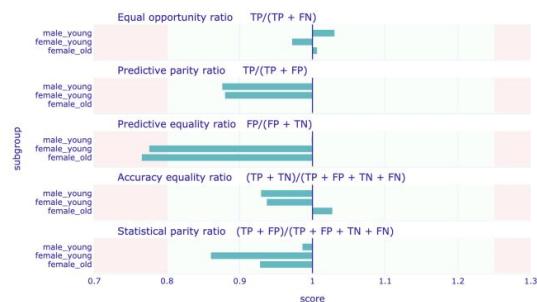
C. Predict-level explanations



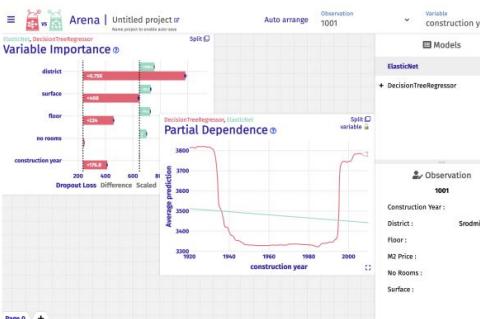
D. Model-level explanations



E. Fairness checks



F. Arena: Interactive comparative model analysis

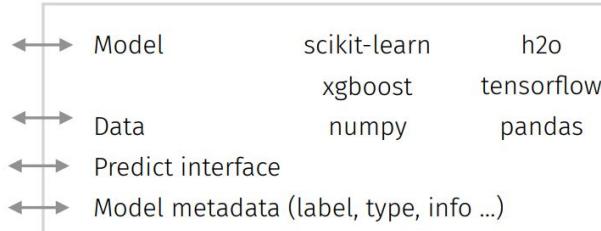


Baniecki et al. *dalex: Responsible Machine Learning with Interactive Explainability and Fairness in Python*. JMLR, 2021.



General idea

A. Explainer: Uniform abstraction over predictive models



B. Consistent grammar for model analysis

```
import dalex as dx
explainer = dx.Explainer(model, x, y)

explanation = explainer.model_parts()
explanation.result
explanation.plot()

explainer.predict_parts(new_observation).result
explainer.predict_parts(new_observation).plot()
```

MODEL

- scikit-learn
- tensorflow, keras
- xgboost, lightgbm
- ANY

DATA

- pandas
- numpy

pip install dalex

import dalex as dx

dx.Explainer

EXPLANATIONS

- `result` attribute (pandas)
- `plot` method (plotly)

METHODS

predict/model + parts/profile/diagnostics
/surrogate/performance

Exemplary machine learning predictive task

The screenshot shows a dataset page for the "World Happiness Report". The background image features a person riding a bicycle through a field of tall grass under a blue sky with white clouds. In the top left corner, there's a "Dataset" icon with the word "Dataset" next to it. In the top right corner, there's a yellow circular icon with a compass-like symbol, an upward-pointing arrow, and the number "2318". Below the image, the title "World Happiness Report" is displayed in large, bold, white font. Underneath the title, a subtitle reads "Happiness scored according to economic production, social support, etc." A small logo for the Sustainable Development Solutions Network is on the left, followed by the text "Sustainable Development Solutions Network • updated a year ago (Version 2)". Below this, a navigation bar includes tabs for "Data" (which is highlighted with a blue underline), "Tasks (2)", "Notebooks (775)", "Discussion (10)", "Activity", and "Metadata". To the right of the navigation bar are two buttons: "Download (79 KB)" and "New Notebook". On the far right of the navigation bar is a vertical ellipsis (...). At the bottom of the page, there are three sections: "Usability 8.5", "License CC0: Public Domain", and "Tags arts and entertainment, news, social science, economics, religion and belief systems".

GDP, life expectancy, freedom, social => country happiness score [0, 10]



<https://hbaniecki.com/jsm2022>



Explainer

```
# 0. package
import dalex as dx

# 1. data
X, y = ...

# 2. model
model = ...
model.fit(X, y)

# 3. explainer
explainer = dx.Explainer(model, X, y)
```

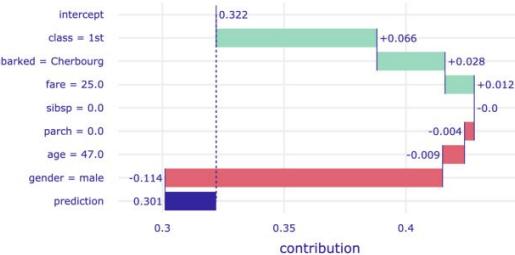
Preparation of a new explainer is initiated

```
-> data : 156 rows 6 cols
-> target variable : Argument 'y' was a pandas.Series. Converted to a numpy.ndarray.
-> target variable : 156 values
-> model_class : tensorflow.python.keras.engine.sequential.Sequential (default)
-> label : custom label
-> predict function : <function yhat_tf_regression at 0x000001D7649554C0> will be used
-> predict function : accepts pandas.DataFrame and numpy.ndarray
-> predicted values : min = 2.86, mean = 5.42, max = 7.73
-> model type : regression will be used (default)
-> residual function : difference between y and yhat (default)
-> residuals : min = -0.616, mean = -0.0103, max = 0.555
-> model_info : package tensorflow
```

A new explainer has been created!

Local & global explanations

C. Predict-level explanations



D. Model-level explanations



predict

model_performance

AUC
RMSE
geom = ecdf, boxplot, gain,
lift, histogram

predict_parts

type = break_down_interactions,
break_down, shap
lime, oscillations



model_parts

type = variable_importance



predict_profile

type = ceteris_paribus



model_profile

type = partial, accumulated,
conditional
geom = aggregates, profiles,
points



predict_diagnostics



model_diagnostics



INSTANCE LEVEL

DATASET LEVEL

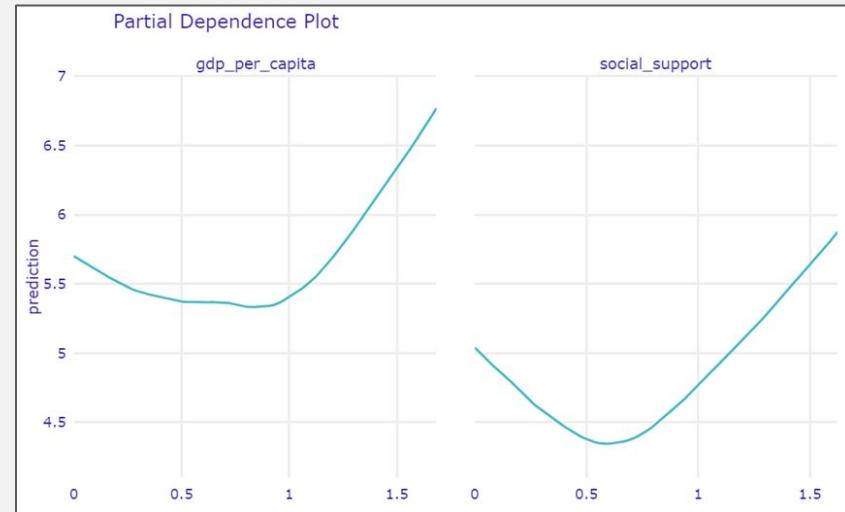
what to calculate
type - how to calculate
geom - how to plot

Model-level explanation

```
# 4. examine  
explainer.model_performance()  
  
# 5. explain  
explainer.model_parts().result  
  
# 6. explore  
explainer.model_profile().plot()
```

mse	rmse	r2	mae	mad
0.017569	0.132549	0.985729	0.072329	0.03636

	variable	dropout_loss	label
0	_full_model_	0.132549	custom label
1	generosity	0.567029	custom label
2	perceptions_of_corruption	0.572801	custom label
3	freedom_to_make_life_choices	0.665235	custom label
4	gdp_per_capita	0.888245	custom label
5	healthy_life_expectancy	0.917414	custom label
6	social_support	1.046778	custom label
7	_baseline_	1.557307	custom label



Predict-level



```
# 7. observation
```

```
obs = ...
```

```
explainer.predict(obs)
```

```
# 8. why?
```

```
explanation = explainer.predict_parts(obs)
```

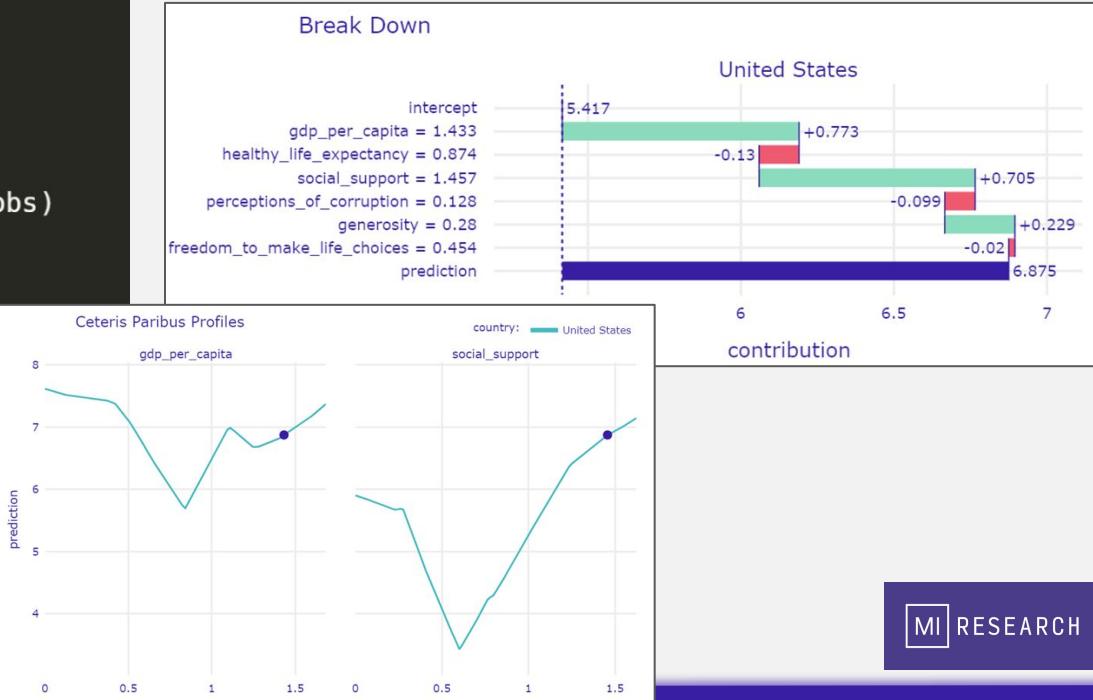
```
explanation.result
```

```
explanation.plot()
```

```
# 9. what if?
```

```
explainer.predict_profile(obs).plot()
```

	variable_name	variable_value	variable	cumulative	contribution	sign	position	label
0	intercept	1	intercept	5.417360	5.417360	1.0	7	custom label
1	gdp_per_capita	1.433	gdp_per_capita = 1.433	6.189979	0.772619	1.0	6	custom label
2	healthy_life_expectancy	0.874	healthy_life_expectancy = 0.874	6.059744	-0.130235	-1.0	5	custom label
3	social_support	1.457	social_support = 1.457	6.764811	0.705067	1.0	4	custom label
4	perceptions_of_corruption	0.128	perceptions_of_corruption = 0.128	6.666029	-0.098782	-1.0	3	custom label
5	generosity	0.28	generosity = 0.28	6.894894	0.228865	1.0	2	custom label
6	freedom_to_make_life_choices	0.454	freedom_to_make_life_choices = 0.454	6.874513	-0.020381	-1.0	1	custom label
7			prediction	6.874512	6.874512	1.0	0	custom label



Interactive explainability

Arena | useRI 2021

Auto arrange

Observation Australia

1 Select one or more models to create plots for them

Models ranger, gbm

2 Hold any of generated plots to open it

Plots

Observation Details Options

Partial Dependence

Average prediction vs gdp per capita

Variable Importance

Variable	Dropout Loss	Difference	Scaled
social support	0.402	0.2911	0.402
healthy life expectancy	0.338	0.1899	0.338
gdp per capita	0.371	0.2007	0.371
freedom to make life choices	0.167	0.0994	0.167
perceptions of corruption	0.148	0.078	0.148
generosity	0.148	0.0371	0.148

Shapley Values

Variable	contribution	observation
social support	1.548	+0.524
gdp per capita	1.372	+0.447
healthy life expectancy	1.036	+0.466
freedom to make life choices	0.557	+0.489
perceptions of corruption	0.29	+0.317
generosity	0.332	+0.1899
	0.0525	+0.362
	0.0371	+0.1655

Ceteris Paribus

prediction vs freedom to make life choices

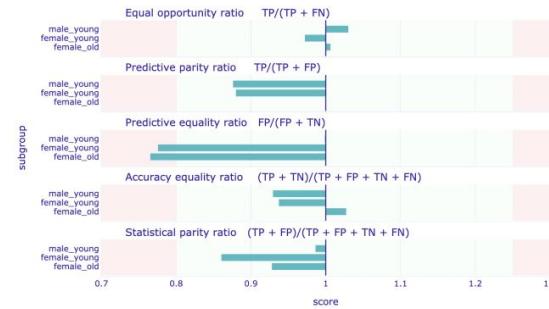
F. Arena: Interactive comparative model analysis



Fairness (German Credit Data)

```
# 1. protected variable with subgroups  
protected = [sex + age for ...]  
  
# 2. privileged subgroup  
privileged = "male_young"  
  
# 3. fairness  
explanation = explainer.model_fairness(  
    protected, privileged  
)  
  
# 4. check  
explanation.fairness_check()  
  
# 5. explain  
explanation.result  
explanation.plot()
```

E. Fairness checks



No bias was detected!

Conclusion: your model is fair in terms of checked fairness criteria.

Ratios of metrics, based on 'male_old'. Parameter 'epsilon' was set to 0.8 and therefore metrics should be within (0.8, 1.25)

	TPR	ACC	PPV	FPR	STP
female_old	1.018828	1.000000	0.971963	0.921525	0.964677
female_young	0.981172	0.938824	0.891355	0.872197	0.855055
male_young	1.019874	0.977647	0.929907	0.896861	0.918392

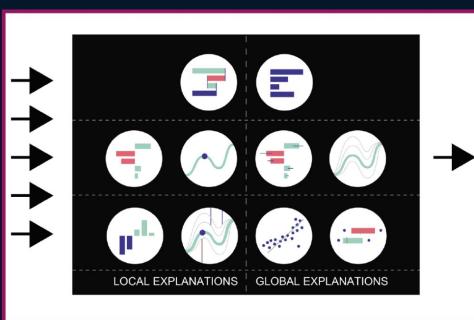
Bias detected in 3 metrics: PPV, FPR, STP

Conclusion: your model is not fair because 2 or more criteria exceeded acceptable limits set by epsilon.

DATA SCIENCE SERIES

EXPLANATORY MODEL ANALYSIS

Explore, Explain, and
Examine Predictive Models



PRZEMYSŁAW BIECEK
TOMASZ BURZYKOWSKI

CRC
Taylor & Francis Group
A CHAPMAN & HALL BOOK

Where to start?



<https://hbaniecki.com/jsm2022>

The HITCHHIKER'S GUIDE TO
RESPONSIBLE MACHINE LEARNING
WITH BETA AND BIT





dalex: Responsible Machine Learning in Python

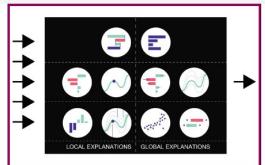
Star 1,085

Python-check passing python 3.6 | 3.7 | 3.8 | 3.9 pypi package 1.4.1 downloads 187k

Install from PyPI

pip install dalex

Explainability



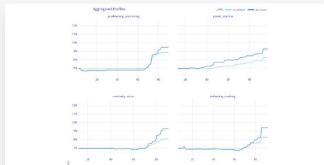
Explanatory Model Analysis

Book with examples in Python



Introduction to dalex

Titanic: tutorial and examples



Key features explained

FIFA 20: explain default vs tuned model with dalex



Aspect module in dalex

Case study - German Credit data



How to use dalex with XGBoost

Titanic classification example

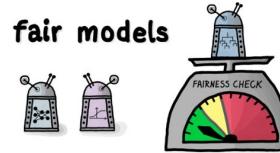


How to use dalex with TensorFlow

Happiness regression example

Fairness

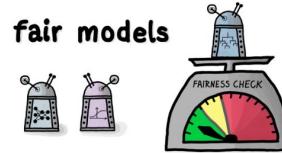
fair models



Fairness module in dalex

Case study - German Credit data

fair models



Tutorial on bias detection

Case study - COMPAS Recidivism data

Interactive analysis



Arena module in dalex

Introduction to the Arena dashboard features



Getting Started & Demos

Arena documentation



<https://hbaniecki.com/jsm2022>

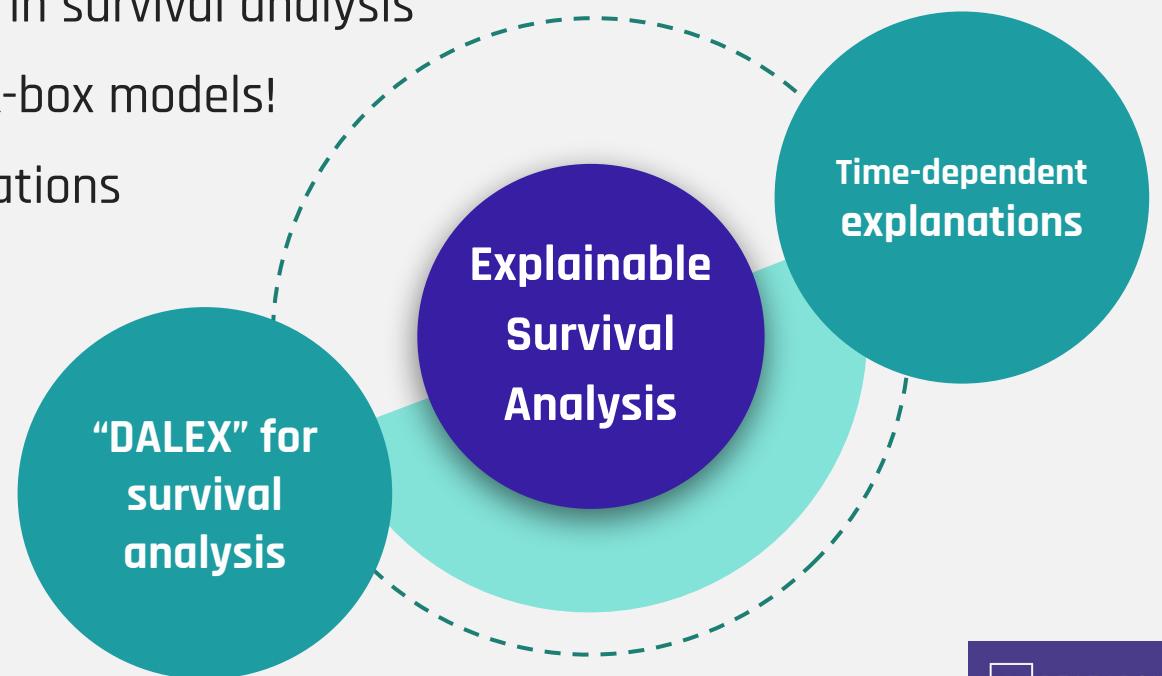


Spoiler: what's next?

Explainable machine learning in survival analysis

We want to understand black-box models!

1. Time-dependent explanations
2. Software... (R & Python)





John M. Chambers Statistical Software Award

Many thanks to organizers and the review panel: **Raymond Wong** (Texas A&M University),
Yixuan Qiu (Shanghai University of Finance and Economics), **Samantha Tyner** (Tritura),
Philip Waggoner (YouGov America, Northwestern University, University of Virginia).

Congratulations to **Vittorio Orlandi** (Duke University) for the Award's honorable mention,
for the R package **FLAME**: interpretable matching methods for performing causal inference
on observational data with discrete covariates.

Questions? Feel free to reach out!