The Journal of Open Source Software

modelStudio: Interactive Studio with Explanations for ML Predictive Models

Hubert Baniecki[1] **and Przemyslaw Biecek**[1]

1 Faculty of Mathematics and Information Science, Warsaw University of Technology

DOI: 10.21105/joss.01798

Software

- Review
- Repository
- Archive

Submitted: 05 October 2019
Published: 05 November 2019

Introduction

Machine learning predictive models are widely used in many areas of business and research. Their rising popularity is due to them being effective but often leads to problems with explaining their prediction. This has led to development of many Interpretable Machine Learning tools, e.g., `DALEX` (Biecek, 2018) R package, `lime` (Ribeiro, Singh, & Guestrin, 2016) and `shap` (Lundberg & Lee, 2017) Python packages and `H2o.ai Driverless AI` (Hall, Gill, Kurka, & Phan, 2017).

Nowadays, we can see a huge demand for automation in many areas. This is how Automated Machine Learning and Automated Exploratory Data Analysis came to existence. AutoML (Truong et al., 2019) and AutoEDA (Staniak & Biecek, 2018) tools not only speed up the model development process but also often lead to new discoveries or higher quality of models.

Explaining predictive models might be a time consuming and tedious task. Libraries for interpretable machine learning (Biecek, 2018; Carme, 2019; Jenkins, Nori, Koch, & Caruana, 2019; Meudec, 2019; Molnar, Casalicchio, & Bischl, 2018) require high programing skills and endless exploration of different aspects of a predictive model.

There are tools for automation of the XAI process like `modelDown` (Romaszko, Tatarynowicz, Urbański, & Biecek, 2019) which produces static HTML site to compare and explain various models. Unfortunately, such tools are focused on global level explanations and deliver monotonous experience.

The `modelStudio` package

The `modelStudio` R package automates the process of model exploration. It generates advanced interactive and animated model explanations in the form of a serverless HTML site. It combines **R** (R Core Team, 2019) with **D3.js** (Bostock, 2016) to produce plots and descriptions for various local and global explanations. Tools for model exploration unite with tools for EDA to give a broad overview of the model behaviour.

The usage of `modelStudio` is meant to be intuitive and simple. The computation time needed to produce the output might not be short though. The main goal of this tool is to make model explaining more automated and achieve higher quality explanations by juxtaposition of complementary aspects of a model.

Comparing instance level explanations and model level explanations side by side adds wider context and allows for deeper understanding. `modelStudio` helps to study relations between various methods for model explanation like *Break Down*, *SHAP*, *Partial Dependency Plots*, *Feature Importance*, and others.

Example

The package `modelStudio` is available on CRAN. It can be installed using the `install.packages('modelStudio')` command. This package is based on `DALEX` explainers created with `DALEX::explain()`. Below there is a basic code example, which produces demo.

```
library("modelStudio")

# Create a model
model <- glm(survived ~., data = DALEX::titanic_imputed, family = "binomial")

# Wrap it into an explainer
explainer <- DALEX::explain(model, data = DALEX::titanic_imputed[,-8],
                            y = DALEX::titanic_imputed[,8], label = "glm")

# Pick some data points
new_observations <- titanic_small[1:4,]
rownames(new_observations) <- c("Lucas", "James", "Thomas", "Nancy")

# Make a studio for the model
modelStudio(explainer, new_observations)
```

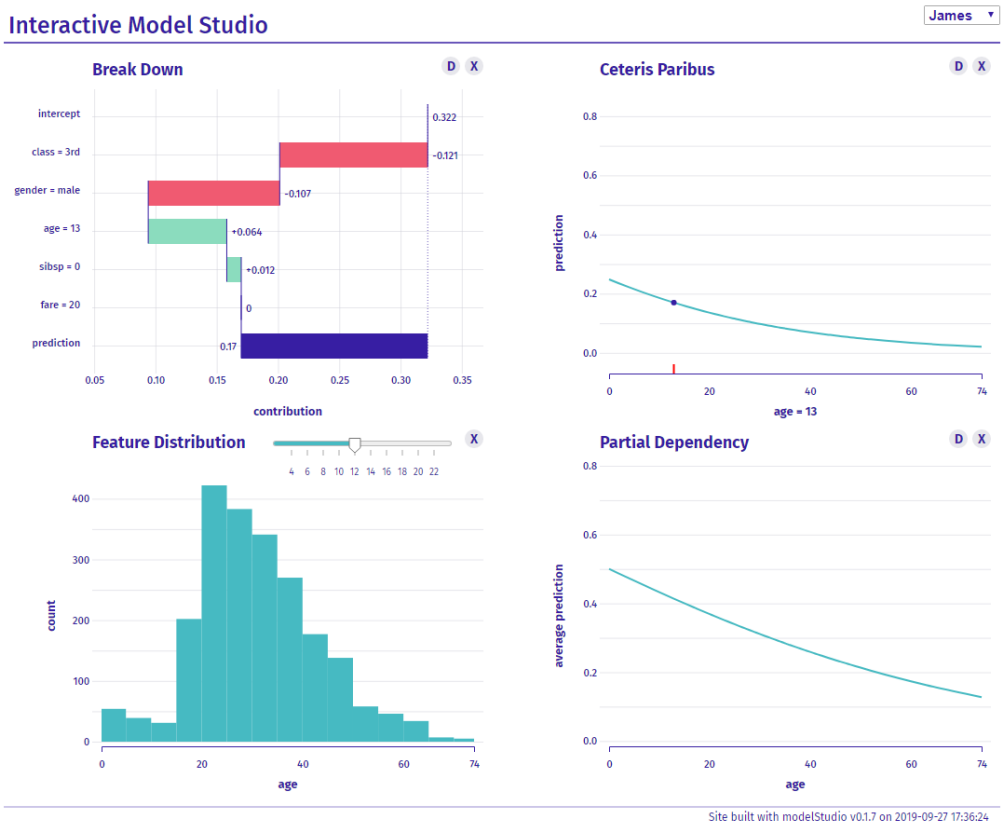

Figure 1: Examplary HTML output layout.

Key Features

The generated HTML site has many interactive features. One can choose which plots are displayed on the grid and change them at any given moment by clicking the X symbol. A drop down list may be used to pick the observation that will be considered for local explanation plots. One may manipulate plots having a variable-based dimension by selecting corresponding bars on the other plots. Mousing over the D symbol displays a description of the plot. Finally, mousing over lines and bars displays the tooltip.

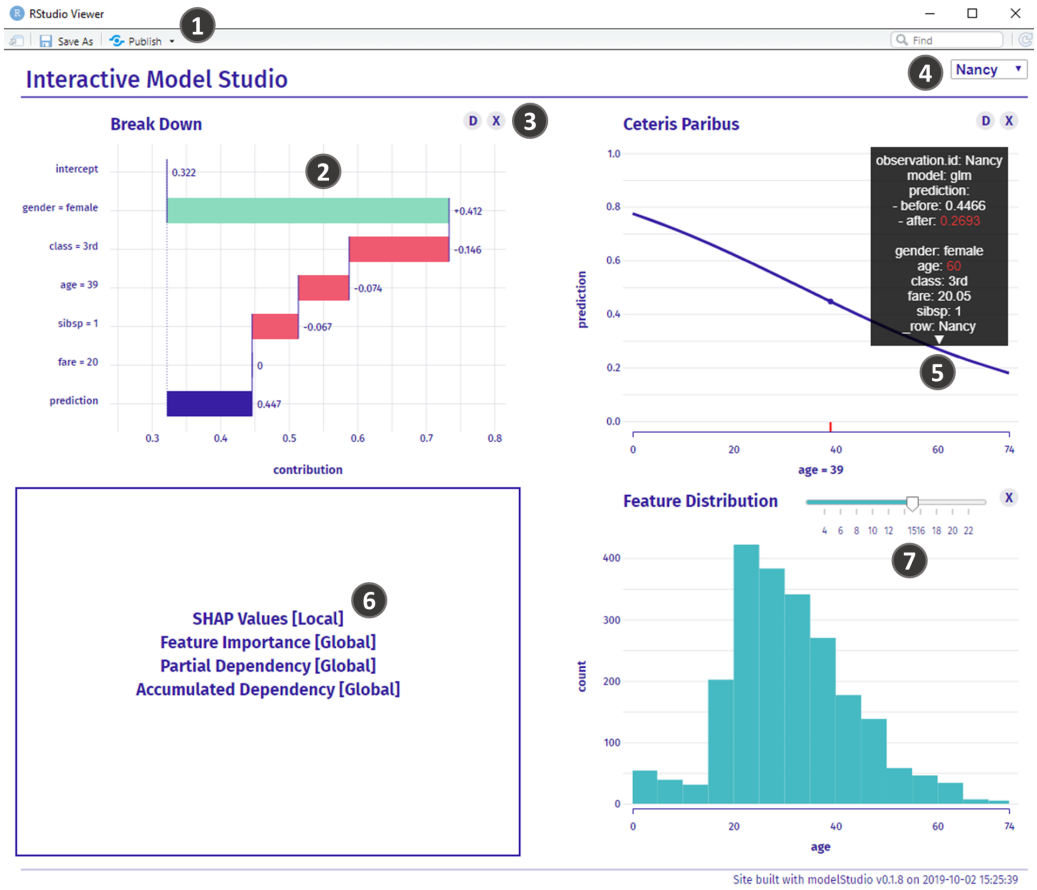

Figure 2: 1. Open in browser or save as HTML document or PNG image 2. Click on bars to choose which feature will be used for other plots 3. Mouse over the D symbol to display a description of the plot and click X to close the plot 4. Choose which observation will be used for local explanations 5. Mouse over lines and bars to display the tooltip 6. Click on the text to choose the plot 7. Interact with other elements like a slider

Explanations

Seven possible plots to choose from are implemented. There are three local explanation plots, three global explanation plots and a feature density plot.

Local explanations are designed to better understand model behaviour around a single observation.

- **Break Down** plot and **SHAP Values** (Lundberg & Lee, 2017) plot present variable

contributions to a model prediction (Gosiewska & Biecek, 2019). Both of them come from the `iBreakDown` (Biecek et al., 2019b) R package.

- **Ceteris Paribus** plot presents model responses around a single point in the feature space (Biecek, 2019).

Global explanations are designed to allow for better understanding of how the model works in general, for some population of interest.

- **Feature Importance** plot presents permutation based feature importance (Fisher, Rudin, & Dominici, 2018).
- **Partial Dependency** plot presents averages from N number of Ceteris Paribus Profiles (Greenwell, 2017).
- **Accumulated Dependency** plot presents accumulated local changes in Ceteris Paribus Profiles (Apley, 2016).

Detailed overview of these methods can be found in "Predictive Models: Explore, Explain, and Debug" (Biecek & Burzykowski, 2019). The last explanations are implemented in the `ingredients` (Biecek et al., 2019a) R package.

Conclusions

The `modelStudio` package is easy to use and its output is intuitive to explore. Automation is convenient and interactivity adds an another dimension to visualisations. All of this enhance explanation of machine learning predictive models. More features and examples can be found in the vignette: modelStudio - perks and features and on GitHub.

Acknowledgments

Work on this package was financially supported by the 'NCN Opus grant 2016/21/B/ST6/02176'.

References

Apley, D. W. (2016). Visualizing the effects of predictor variables in black box supervised learning models. *arXiv preprint arXiv:1612.08468*.

Biecek, P. (2018). DALEX: explainers for complex predictive models. Retrieved from http://arxiv.org/abs/1806.08915

Biecek, P. (2019). *CeterisParibus: Ceteris paribus profiles*. Retrieved from https://CRAN.R-project.org/package=ceterisParibus

Biecek, P., Baniecki, H., Izdebski, A., & Pekala, K. (2019a). *Ingredients: Effects and importances of model ingredients*. Retrieved from http://CRAN.R-project.org/package=ingredients

Biecek, P., & Burzykowski, T. (2019). *Predictive models: Explore, explain, and debug*. Retrieved from https://pbiecek.github.io/PM_VEE/

Biecek, P., Gosiewska, A., Baniecki, H., & Izdebski, A. (2019b). *IBreakDown: Model agnostic instance level variable attributions*. Retrieved from https://CRAN.R-project.org/package=iBreakDown

Bostock, M. (2016). D3.js-data-driven documents (2016). *URL: https://d3js.org*.

Carme, A. (2019). *Sklearn explain*. Retrieved from https://github.com/antoinecarme/sklearn_explain

Fisher, A., Rudin, C., & Dominici, F. (2018). All models are wrong but many are useful: Variable importance for black-box, proprietary, or misspecified prediction models, using model class reliance. *arXiv preprint arXiv:1801.01489*.

Gosiewska, A., & Biecek, P. (2019). IBreakDown: Uncertainty of model explanations for non-additive predictive models. *arXiv preprint arXiv:1903.11420*.

Greenwell, B. M. (2017). pdp: An R Package for Constructing Partial Dependence Plots. *The R Journal*, *9*(1), 421–436. doi:10.32614/rj-2017-016

Hall, P., Gill, N., Kurka, M., & Phan, W. (2017). *Machine learning interpretability with H2O driverless AI*. (A. Bartz, Ed.) (1st ed.). Mountain View, CA: H2O.ai, Inc. Retrieved from http://docs.h2o.ai

Jenkins, S., Nori, H., Koch, P., & Caruana, R. (2019). *InterpretML*. Retrieved from https://github.com/microsoft/interpret

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems 30* (pp. 4765–4774). Curran Associates, Inc. Retrieved from http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

Meudec, R. (2019). *Interpretability methods for tf.keras models with tensorflow 2.0*. Retrieved from https://tf-explain.readthedocs.io

Molnar, C., Casalicchio, G., & Bischl, B. (2018). iml: An R package for Interpretable Machine Learning. *Journal of Open Source Software*, *3*(26). doi:10.21105/joss.00786

R Core Team. (2019). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.R-project.org/

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, san francisco, ca, usa, august 13-17, 2016* (pp. 1135–1144). doi:10.18653/v1/n16-3020

Romaszko, K., Tatarynowicz, M., Urbański, M., & Biecek, P. (2019). modelDown: Automated website generator with interpretable documentation for predictive machine learning models. *Journal of Open Source Software*, *4*(38), 1444. doi:10.21105/joss.01444

Staniak, M., & Biecek, P. (2018). Explanations of Model Predictions with live and breakDown Packages. *The R Journal*, *10*(2), 395–409. doi:10.32614/RJ-2018-072

Truong, A., Walters, A., Goodsitt, J., Hines, K., Bruss, C. B., & Farivar, R. (2019). Towards automated machine learning: Evaluation and comparison of automl approaches and tools. *ArXiv*, *abs/1908.05557*.