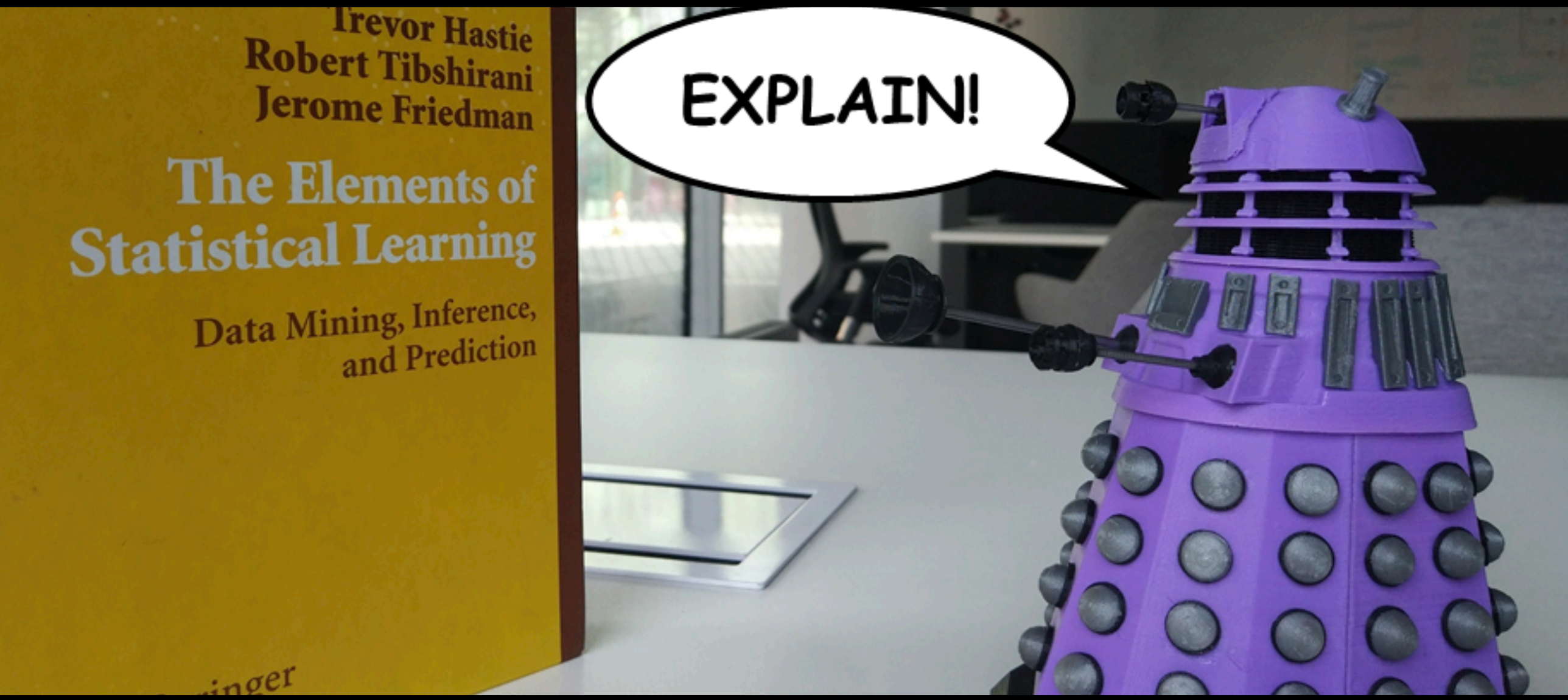
Black-box openers:
How to explain predictions from complex ML models?

Predictive Models: Visualisation, Exploration and Explanation



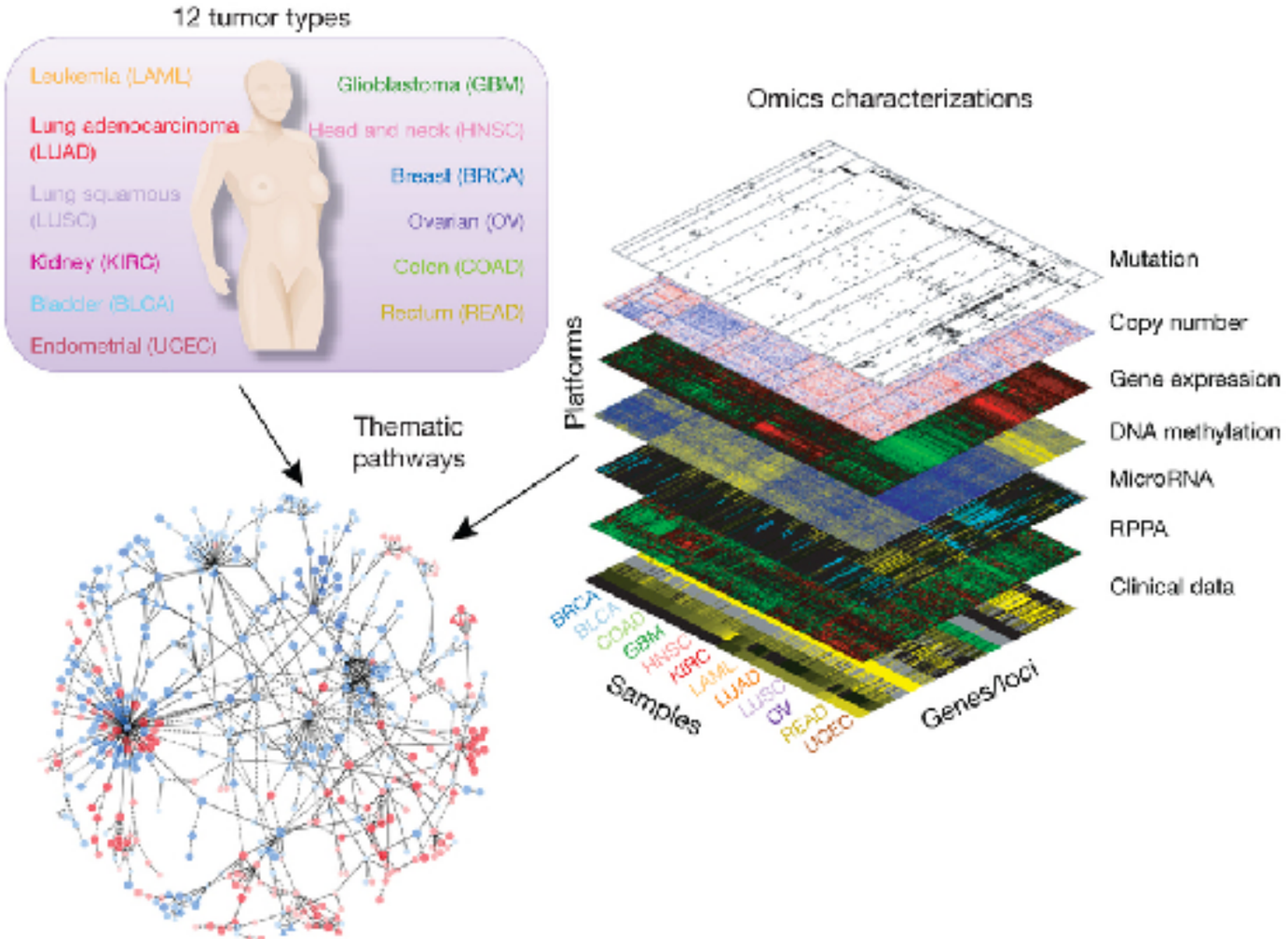
Przemysław Biecek Warsaw University of Technology

Background
- MSc in Software Engineering and Mathematical Statistics
- PhD in Mathematical Statistics

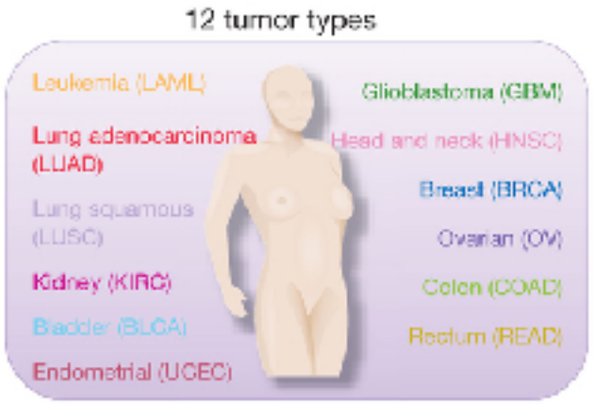
Head of MI^2 DataLab
- at Warsaw University of Technology
- Group of data enthusiasts (MSc and PhD students)
- https://github.com/MI2DataLab/

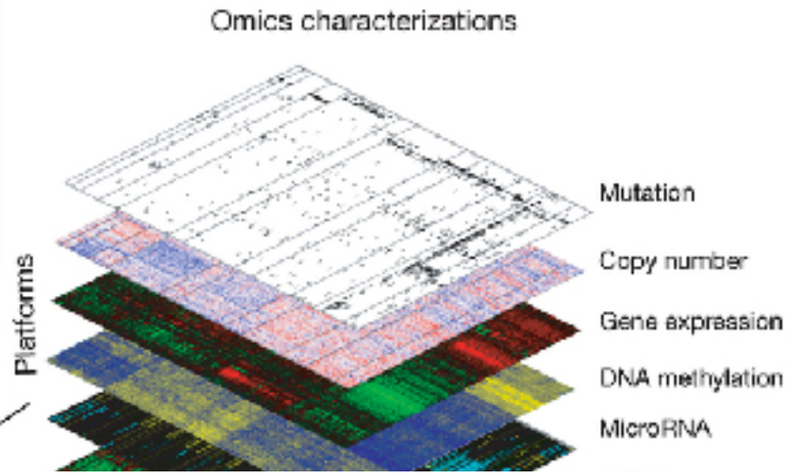
Research area
- Model interpretability
- Machine Learning / Predictive Modeling
- Data visualization

Coorganizer of R meetups and conferences in Poland
- Spotkania Entuzjastów R



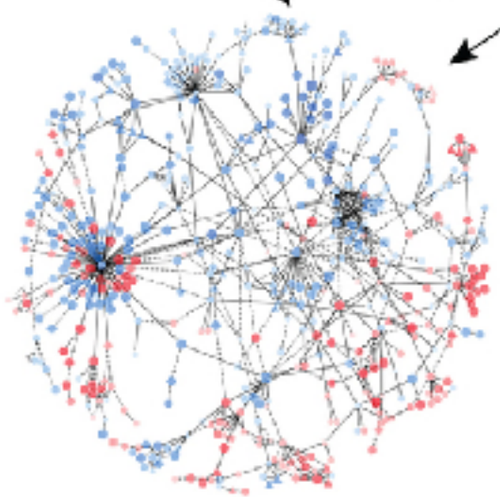
Kyle Chang et al,
The Cancer Genome Atlas
Pan-Cancer analysis project,
Nature Genetics 45(10):1113-20



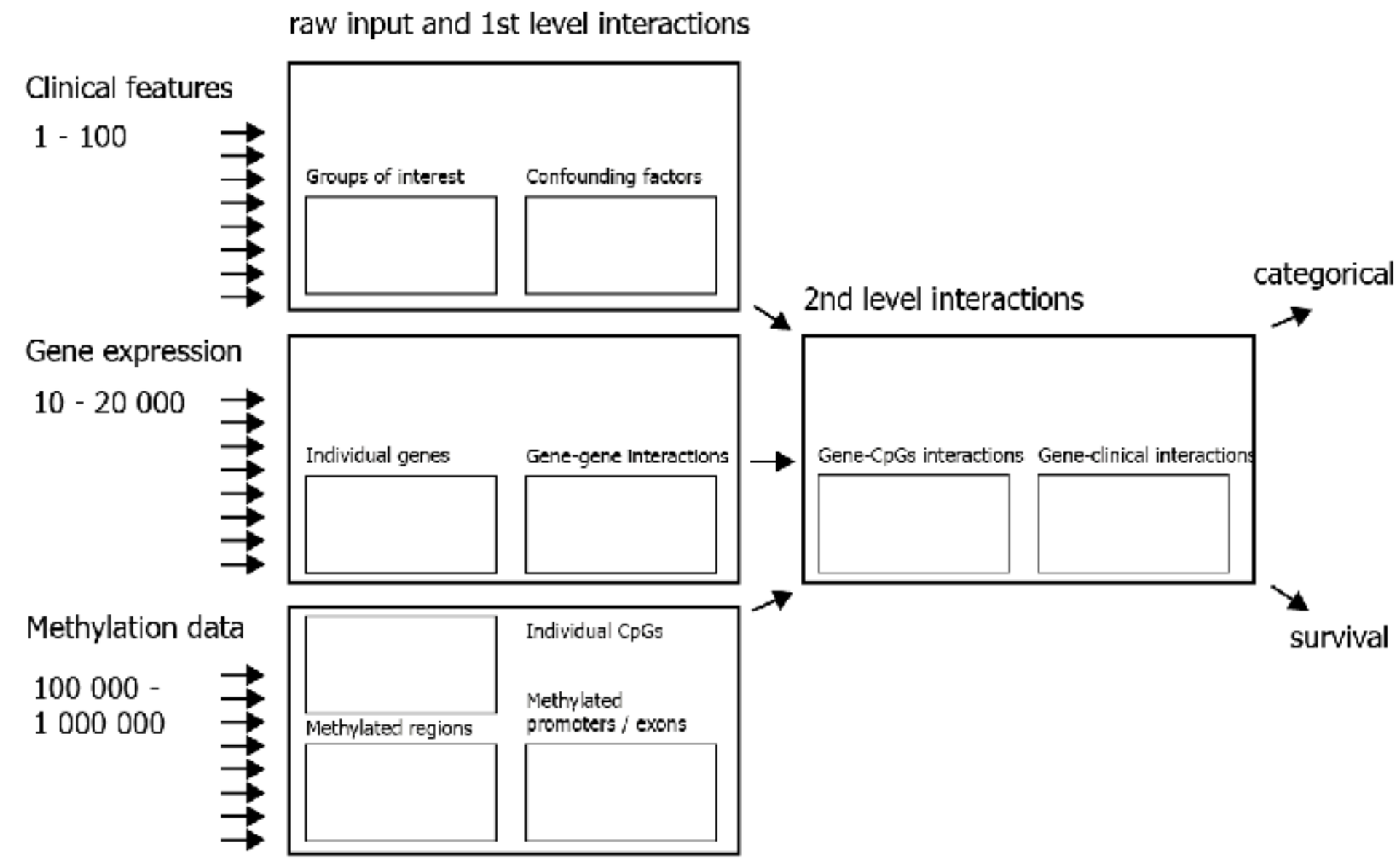


Kyle Chang et al,
The Cancer Genome Atlas
Pan-Cancer analysis project,
Nature Genetics 45(10):1113-20

Thematic pathways



Learning with Structure: MLGenSig (Machine Learning Genetic Signatures)



Complex models

(black-boxes: random forest, boosting, neural nets)

- elastic,
- high performance,
- hard to understand.

Simple models

(white-boxes: linear models, decision trees)

- easy to understand,
- average performance.

Complex models

(black-boxes: random forest, boosting, neural nets)

- elastic,
- high performance,
- hard to understand,
- *physicians may not trust/not use them.*

Simple models

(white-boxes: linear models, decision trees)

- easy to understand,
- average performance,
- *is it ethical to use them if we can do better?*

Model Averaging

Classification trees can be simple, but often produce noisy (bushy) or weak (stunted) classifiers.

- Bagging (Breiman, 1996): Fit many large trees to bootstrap-resampled versions of the training data, and classify by majority vote.
- Boosting (Freund & Shapire, 1996): Fit many large or small trees to reweighted versions of the training data. Classify by weighted majority vote.
- Random Forests (Breiman 1999): Fancier version of bagging.

In general Boosting $\succ$ Random Forests $\succ$ Bagging $\succ$ Single Tree.

- http://jessica2.msri.org/attachments/10778/10778-boost.pdf

Model Averaging

Classification trees can be simple, but often produce noisy (bushy) or weak (stunted) classifiers.

- Bagging (Breiman, 1996): Fit many large trees to bootstrap-resampled versions of the training data, and classify by majority vote.
- Boosting (Freund & Shapire, 1996): Fit many large or small trees to reweighted versions of the training data. Classify by weighted majority vote.
- Random Forests (Breiman 1999): Fancier version of bagging.

In general Boosting $\succ$ Random Forests $\succ$ Bagging $\succ$ Single Tree.

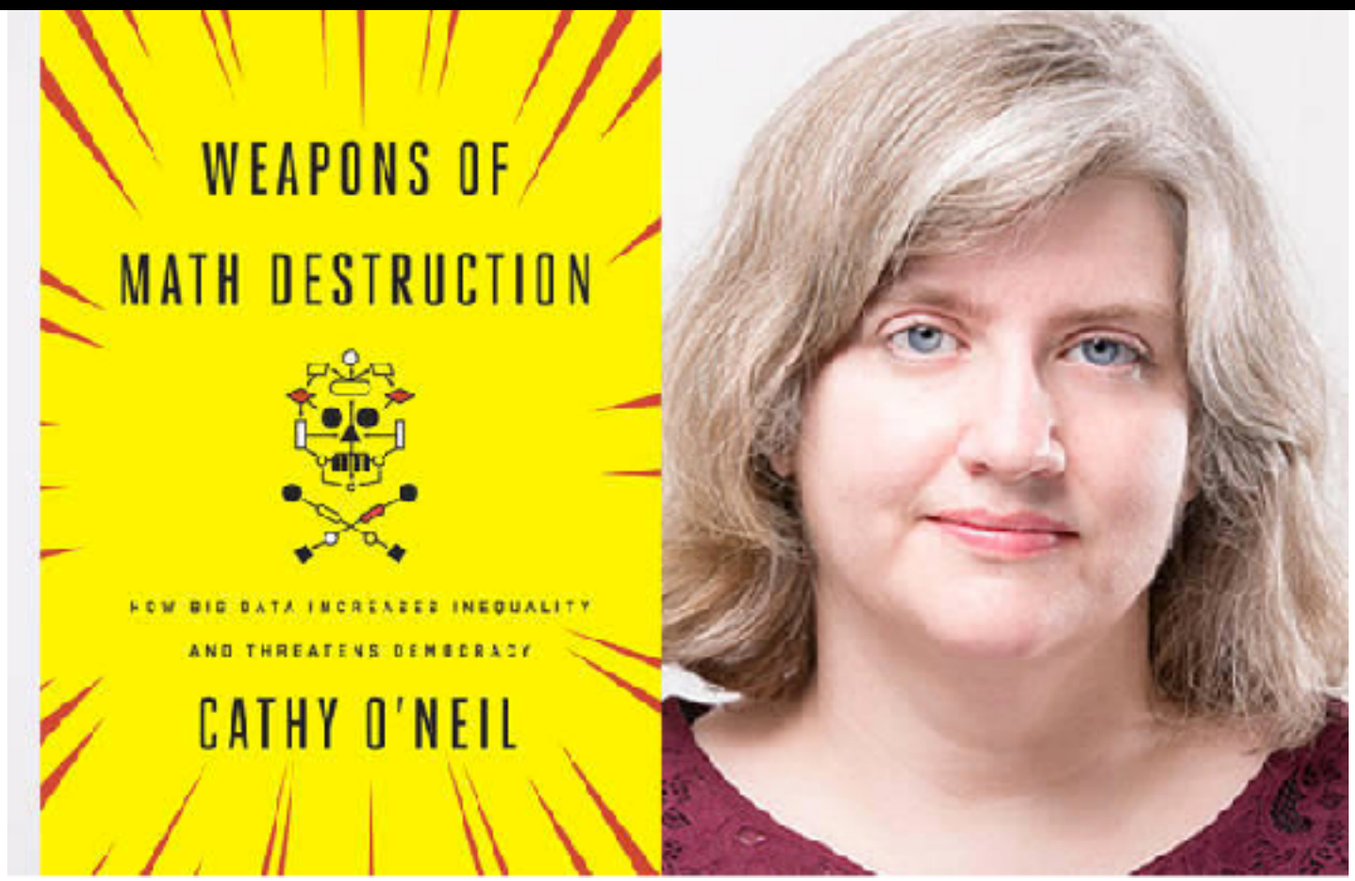
Make these easy to interpret

- http://jessica2.msri.org/attachments/10778/10778-boost.pdf

Why do we need explanations for complex models?

Cathy O'Neil:

The era of blind faith in ~~big data~~ black boxes must end



- "You don't see a lot of skepticism," she says. "The algorithms are like shiny new toys that we can't resist using. We trust them so much that we project meaning on to them."
- Ultimately algorithms, according to O'Neil, reinforce discrimination and widen inequality, "using people's fear and trust of mathematics to prevent them from asking questions".

https://www.theguardian.com/books/2016/oct/27/cathy-oneil-weapons-of-math-destruction-algorithms-big-data

Why do we need explanations for complex models?

Article | Talk | Read | Edit | View history | Search Wikipedia

Right to explanation

From Wikipedia, the free encyclopedia

In the regulation of algorithms, particularly artificial intelligence and its subfield of machine learning, a **right to explanation** (or **right to explanation**) is a right to be given an explanation for an output of the algorithm. Such rights primarily refer to individual rights to be giv explanation for decisions that significantly affect an individual, particularly legally or financially. For example, a person who applies for is denied may ask for an explanation, which could be "Credit bureau X reports that you declared bankruptcy last year; this is the main considering you too likely to default, and thus we will not give you the loan you applied for."

Some such legal rights already exist, while the scope of a general "right to explanation" is a matter of ongoing debate.

Contents [hide]

Local or global explanations?

Global explanations
(scope: population)

- How good is the model?
- How does it work?
- Is model A better than model B?
- Which variables are important?

Local explanations
(scope: single prediction)

- Which variables influence this single prediction?
- What would happen if input is slightly different?
- Is this single prediction accurate?

Local or global explanations?

Global explanations
(scope: population)

- How good is the model?
- How does it work?
- Is model A better than model B?
- Which variables are important?

Local explanations
(scope: single prediction)

- Which variables influence this single prediction?
- What would happen if input is slightly different?
- Is this single prediction accurate?

Today I will talk only about local explanations.

Model agnostic or model specific?

Model specific

- Linear models: diagnostic tools (qq plots, RESET test, many other), variable selection operators, etc.
- Neural networks: Layer-wise Relevance Propagation, Saliency Maps technique, etc.
- Random Forest / trees ensembles: variable importance, randomForestExplainer, xgboostExplainer.
- …

Model agnostic

- Partial Dependency Plots.
- LIME: Local Interpretable Model Agnostic Explanations.
- SHAP (SHapley Additive exPlanations).
- Accumulated Local Effects (ALE).
- Permutation based variable importance.
- …

Model agnostic or model specific?

Model specific

- Linear models: diagnostic tools (qq plots, RESET test, many other), variable selection operators, etc.
- Neural networks: Layer-wise Relevance Propagation, Saliency Maps technique, etc.
- Random Forest / trees ensembles: variable importance, randomForestExplainer, xgboostExplainer.
- …

Model agnostic

- Partial Dependency Plots.
- LIME: Local Interpretable Model Agnostic Explanations.
- SHAP (SHapley Additive exPlanations).
- Accumulated Local Effects (ALE).
- Permutation based variable importance.
- …

Today I will talk only about model agnostic tools.

How model vis is different from data vis?

Data vis / exploration

- Learning relations
- Data may be noisy
- Sampling may be biased
- Sampling results in randomness

Model vis / exploration

- Extraction of relations
- Models may be inaccurate
- Models may be biased
- No randomness in deterministic models

How model vis is different from data vis?

Data vis / exploration

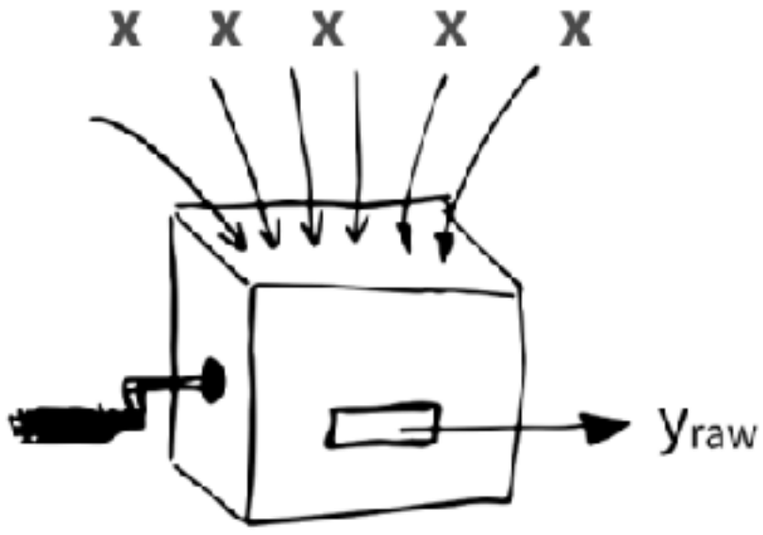
- Learning relations
- Data may be noisy
- Sampling may be biased
- Sampling results in randomness

Model vis / exploration

- Extraction of relations
- Models may be inaccurate
- Models may be biased
- No randomness in deterministic models

Today I will talk only about model vis / exploration.

How can we explain model predictions?

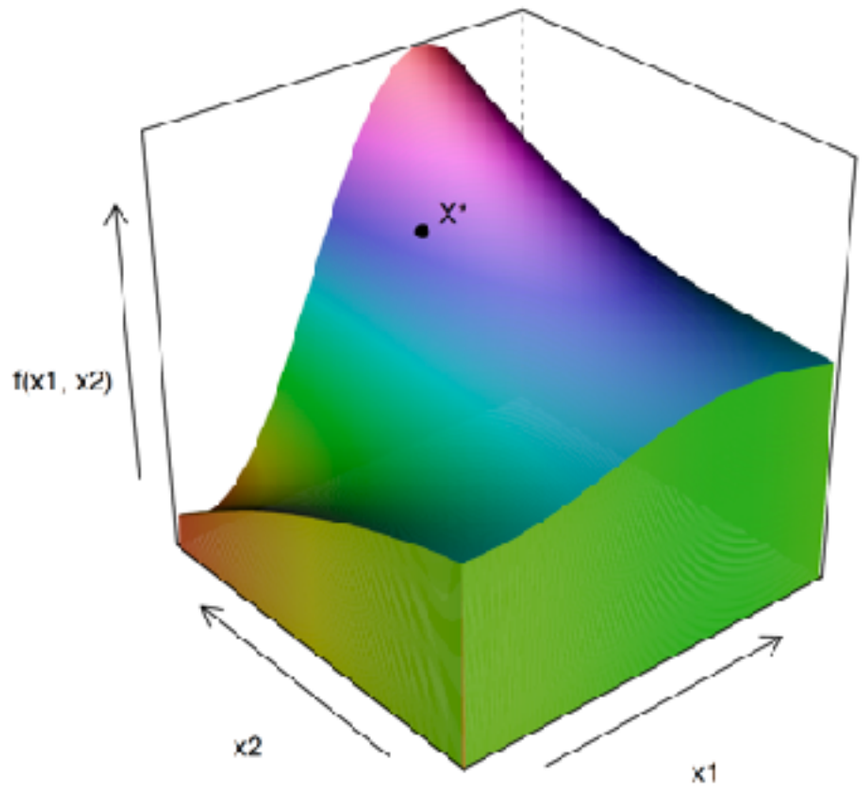


Black box model is a function

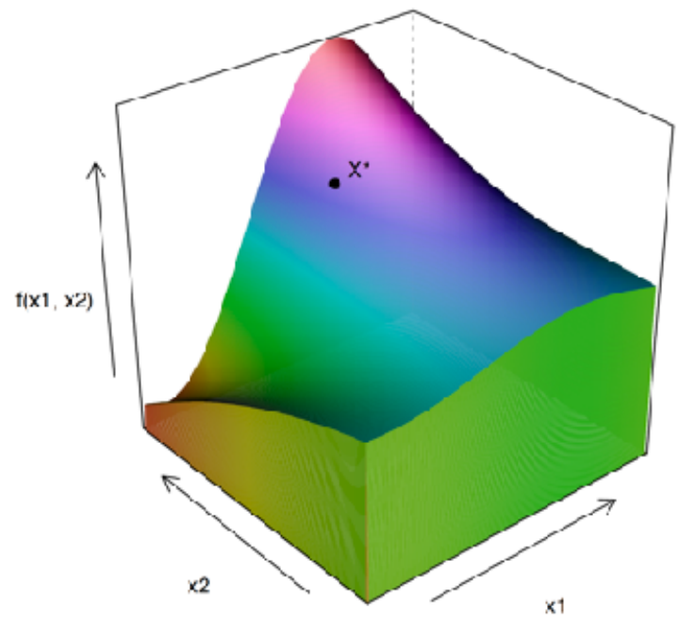
$$f : R^p \rightarrow R$$

For p > 2 we cannot plot this function directly

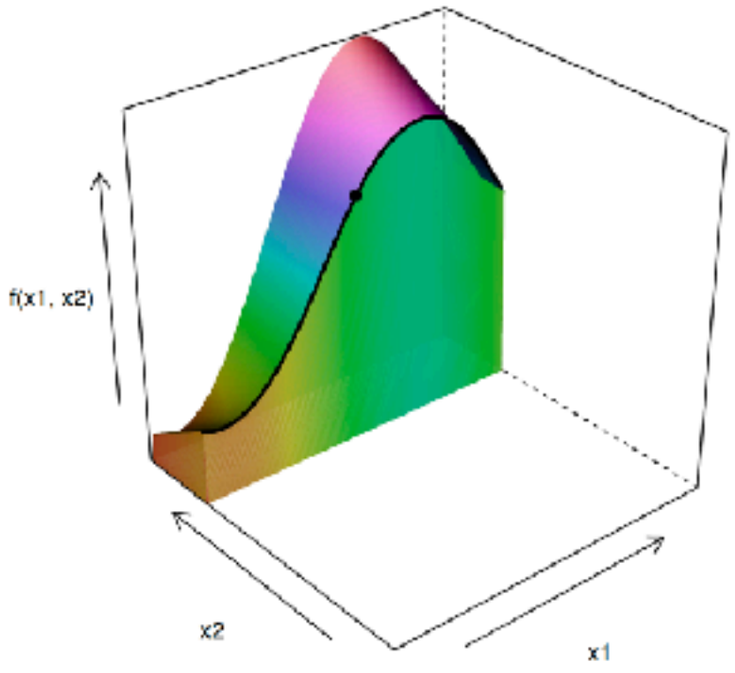
In many applications p is very large
(hundreds, thousands, …)



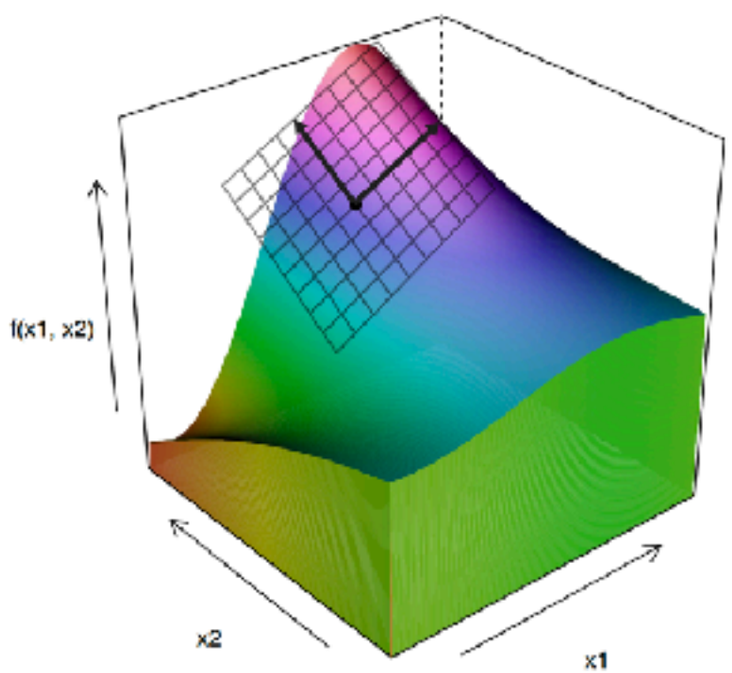
Three approaches to local model explanations



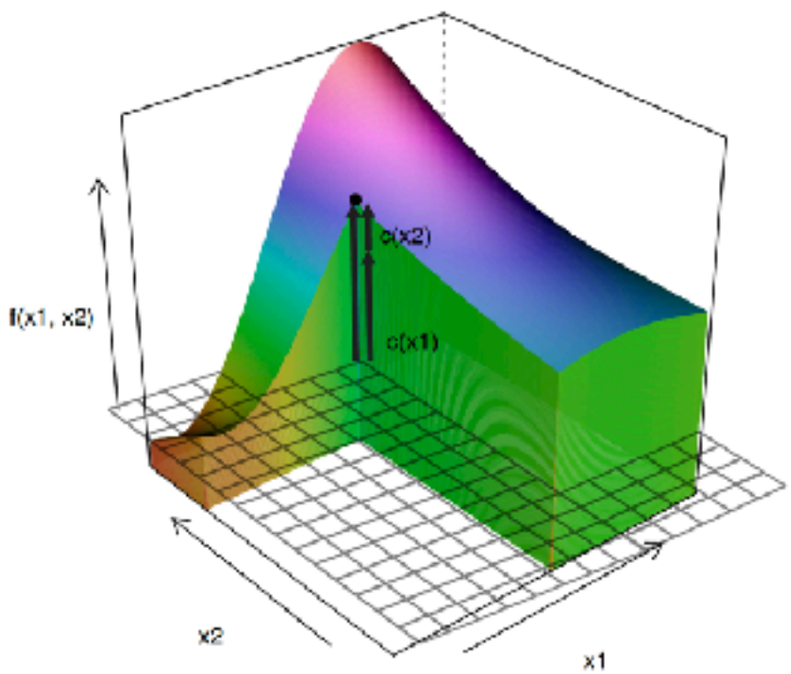
A)



B)



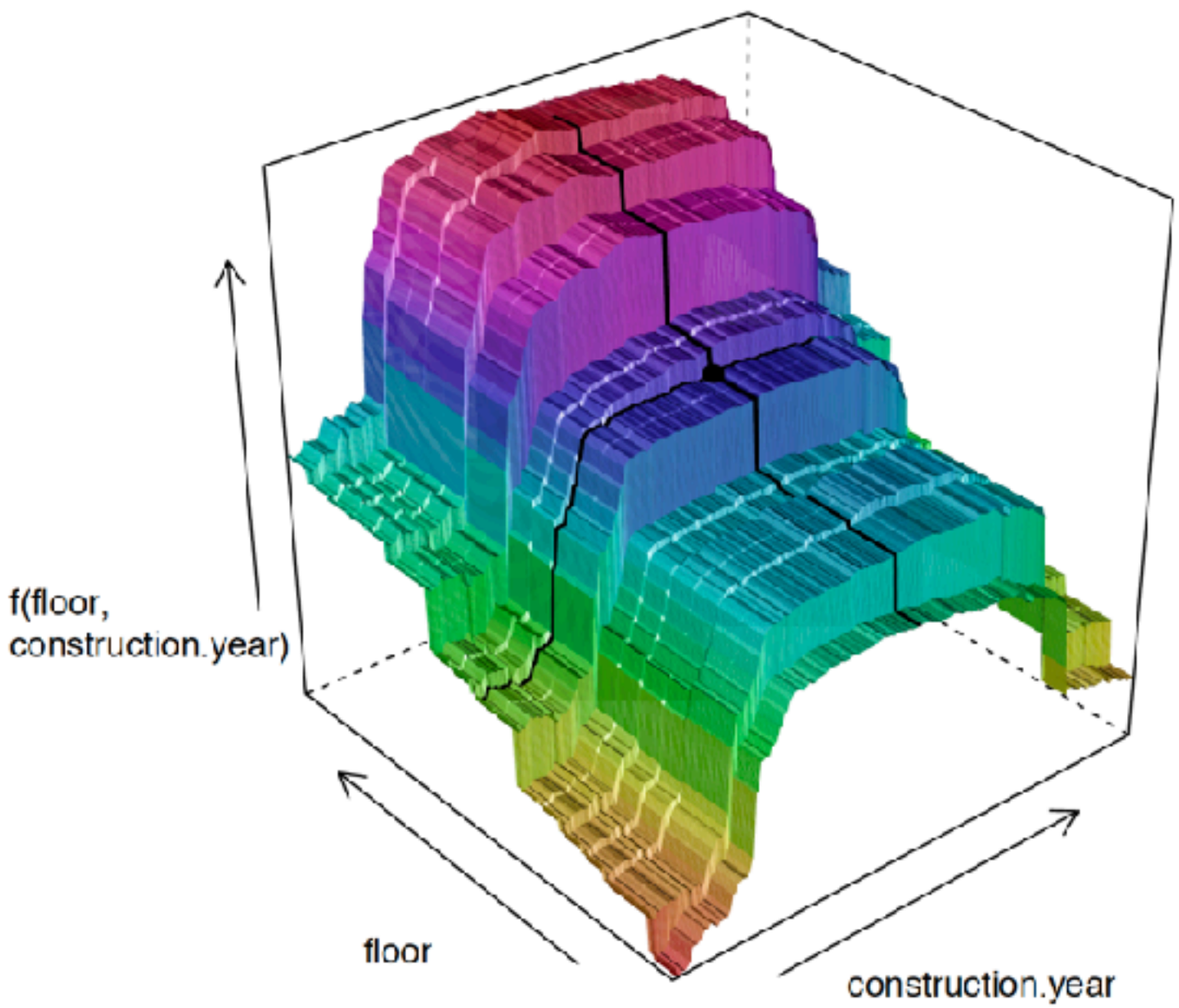
C)



Ceteris Paribus Profiles

Ceteris paribus is a Latin phrase meaning “other things held constant” or “all else unchanged”.

It is a method for exploration of model responses given only a single variable is changed.



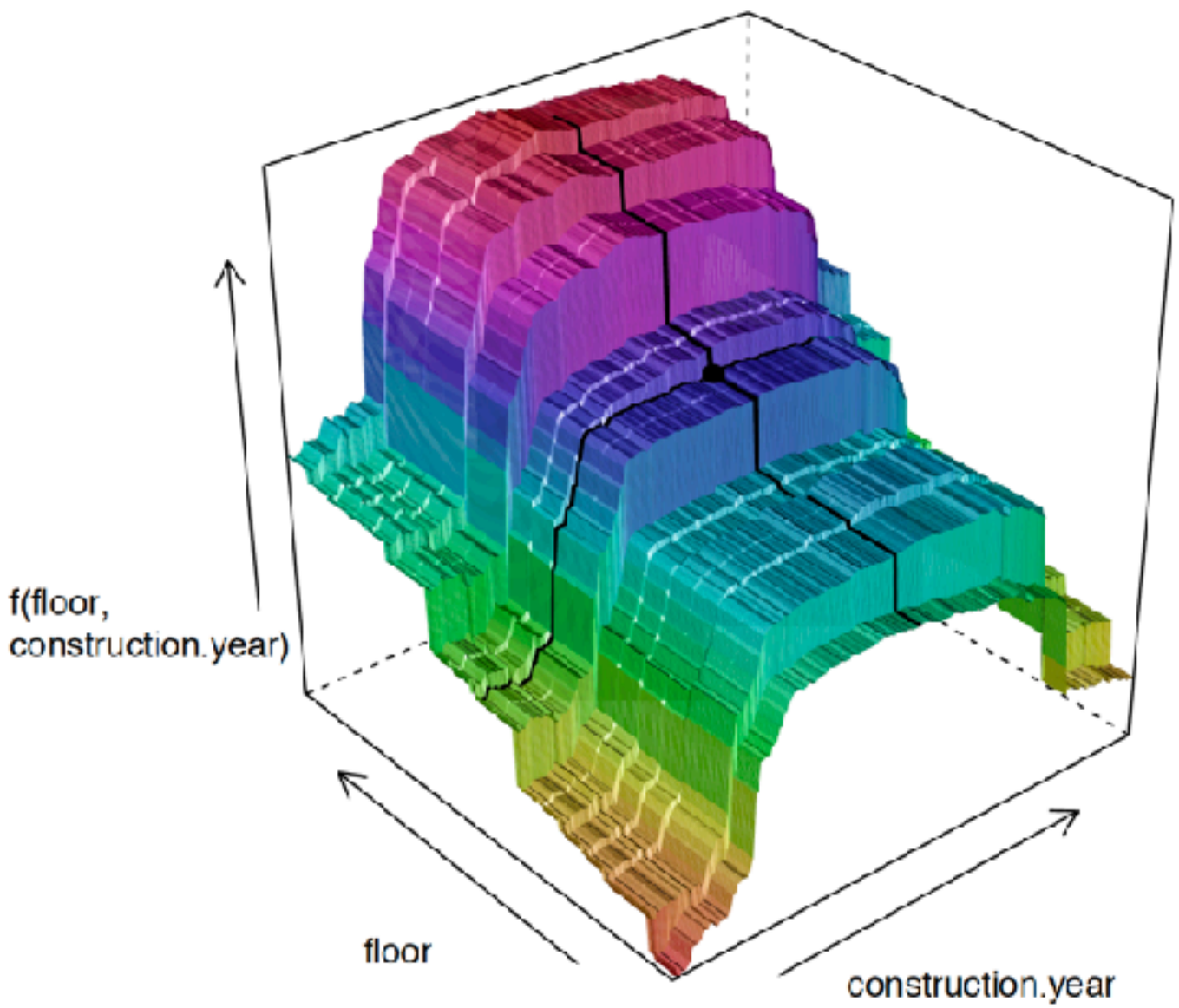
Ceteris Paribus Profiles

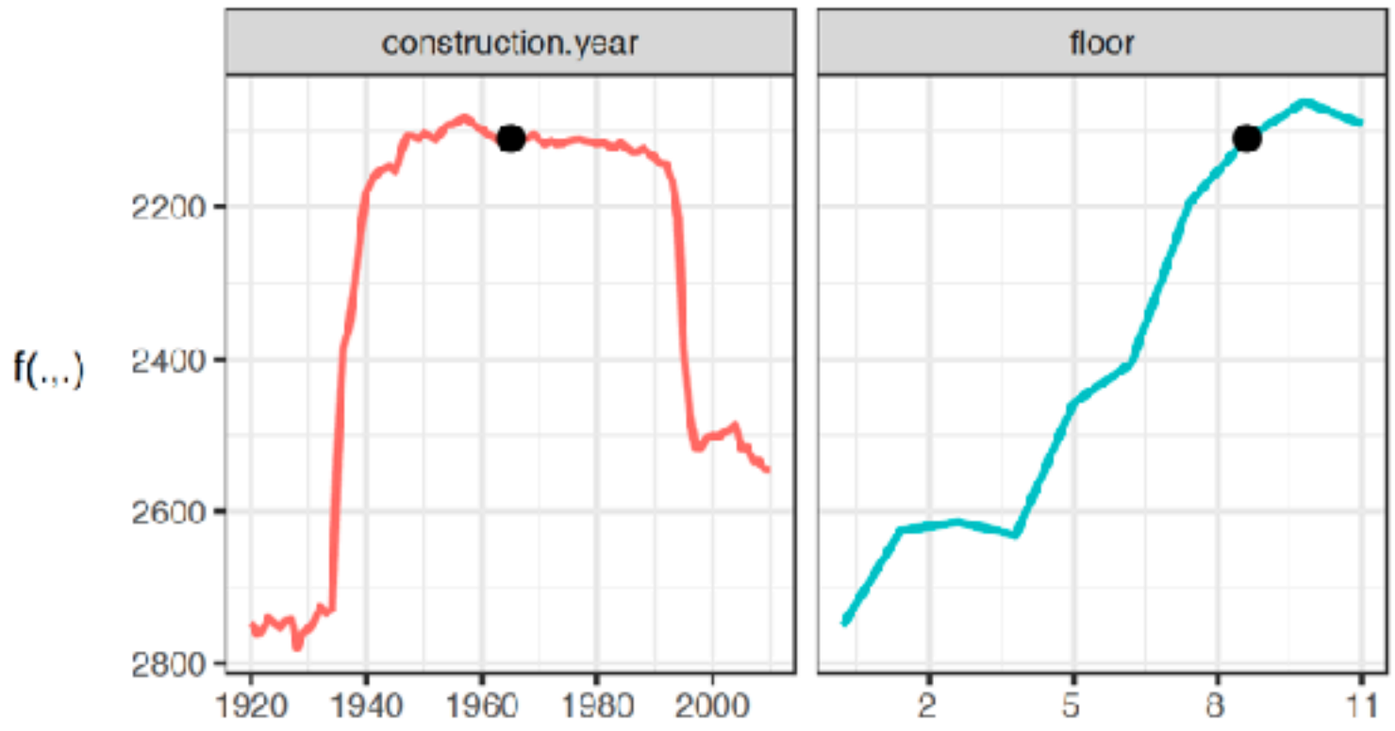
Ceteris paribus is a Latin phrase meaning “other things held constant” or “all else unchanged”.

It is a method for exploration of model responses given only a single variable is changed.

More formally

$$CP^{f,j,x}(z) := f(x|^j = z).$$



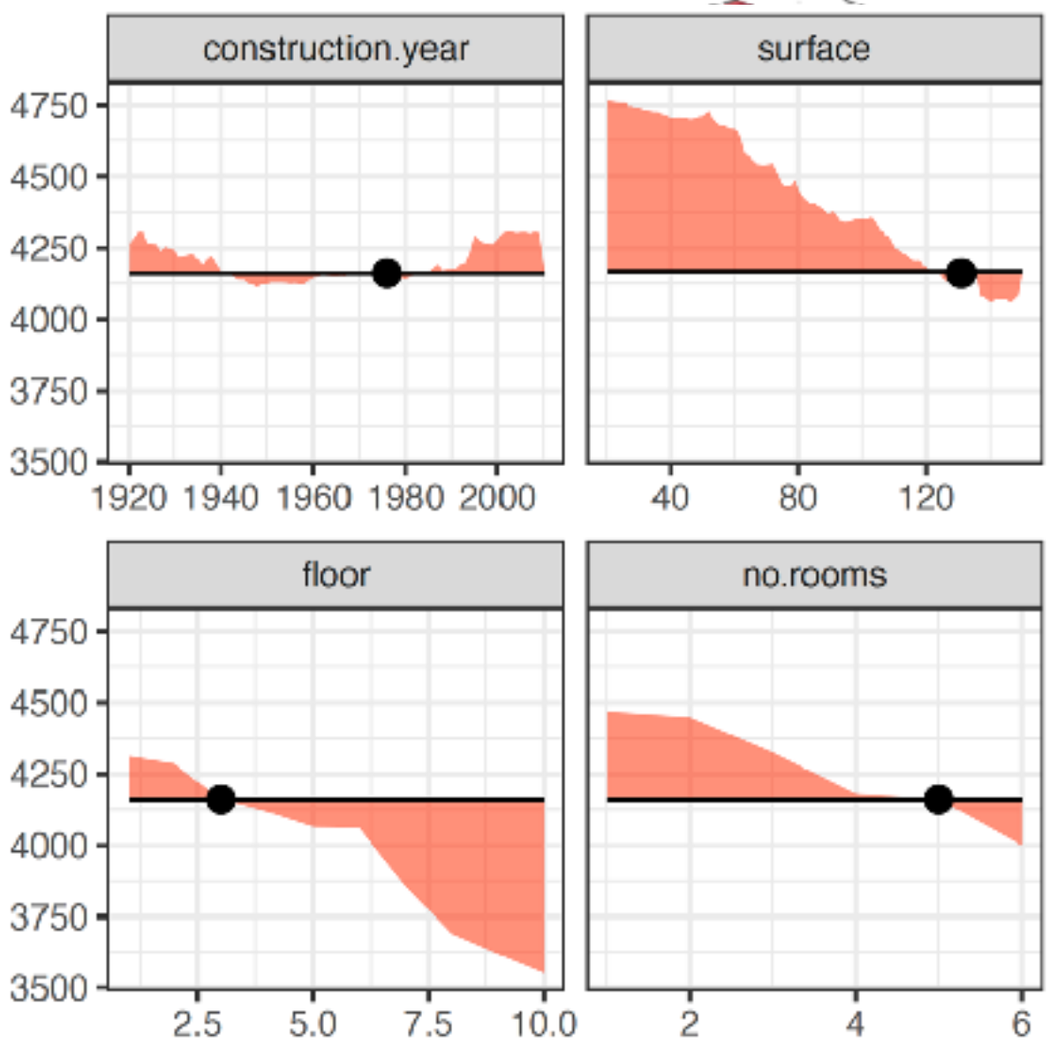


Ceteris Paribus Profiles

Based on Ceteris Paribus Profiles you may calculate local variable importance.

One way to do so is to integrate CP oscillations over model predictions.

$$vip_j^{CP}(x) = \int_{-\inf}^{inf} |CP^{f,j,x}(z) - f(x)|dz$$



Ceteris Paribus Profiles

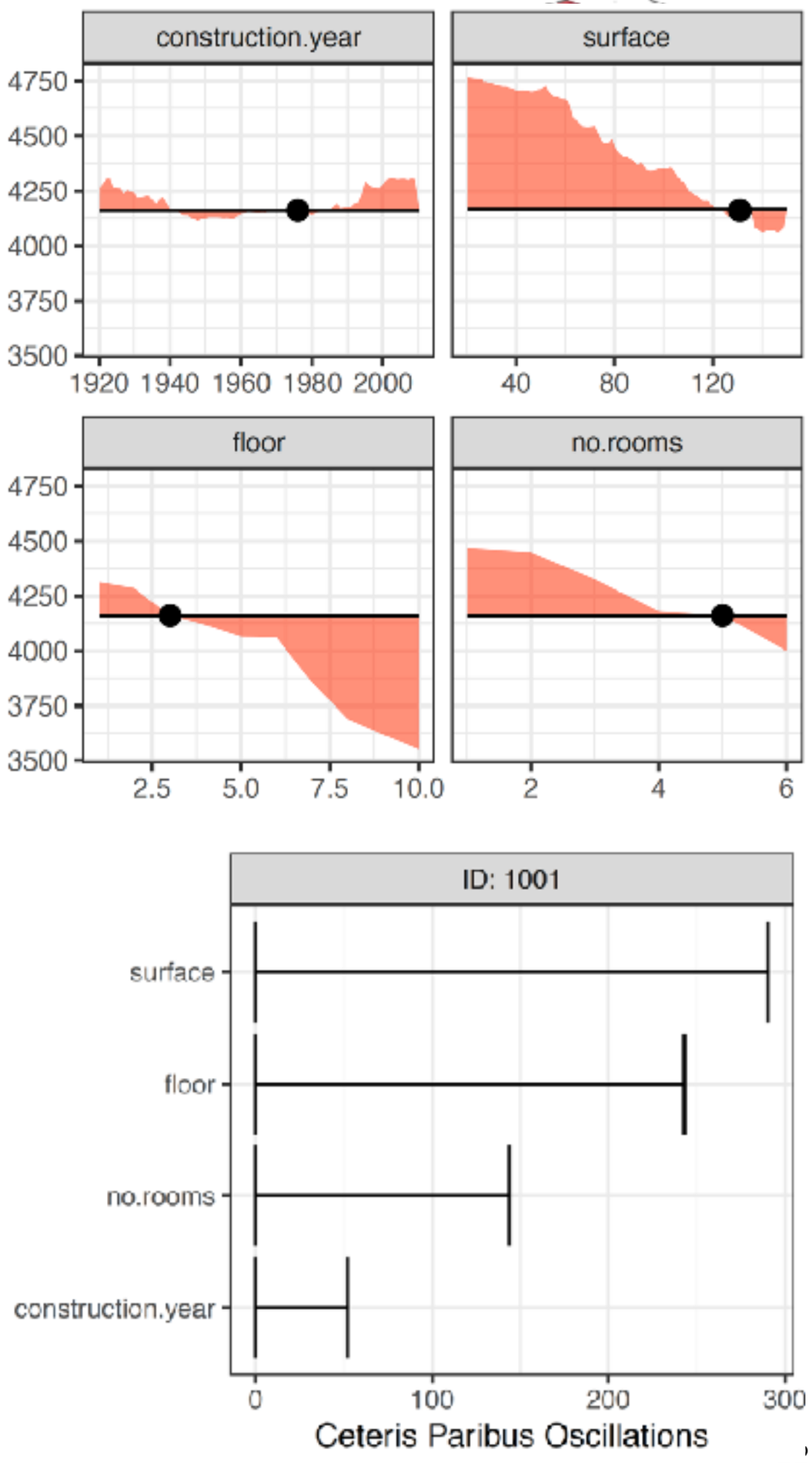
Based on Ceteris Paribus Profiles you may calculate local variable importance.

One way to do so is to integrate CP oscillations over model predictions.

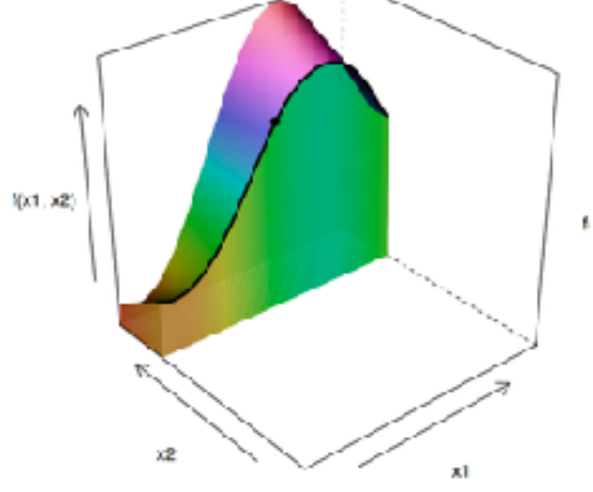
$$vip_j^{CP}(x) = \int_{-\inf}^{inf} |CP^{f,j,x}(z) - f(x)|dz$$

This leads to a straightforward estimator for variable importance

$$\widehat{vip_j^{CP}(x)} = \frac{1}{n}\sum_{i=1}^{n} |CP^{f,j,x}(x_i) - f(x)|$$



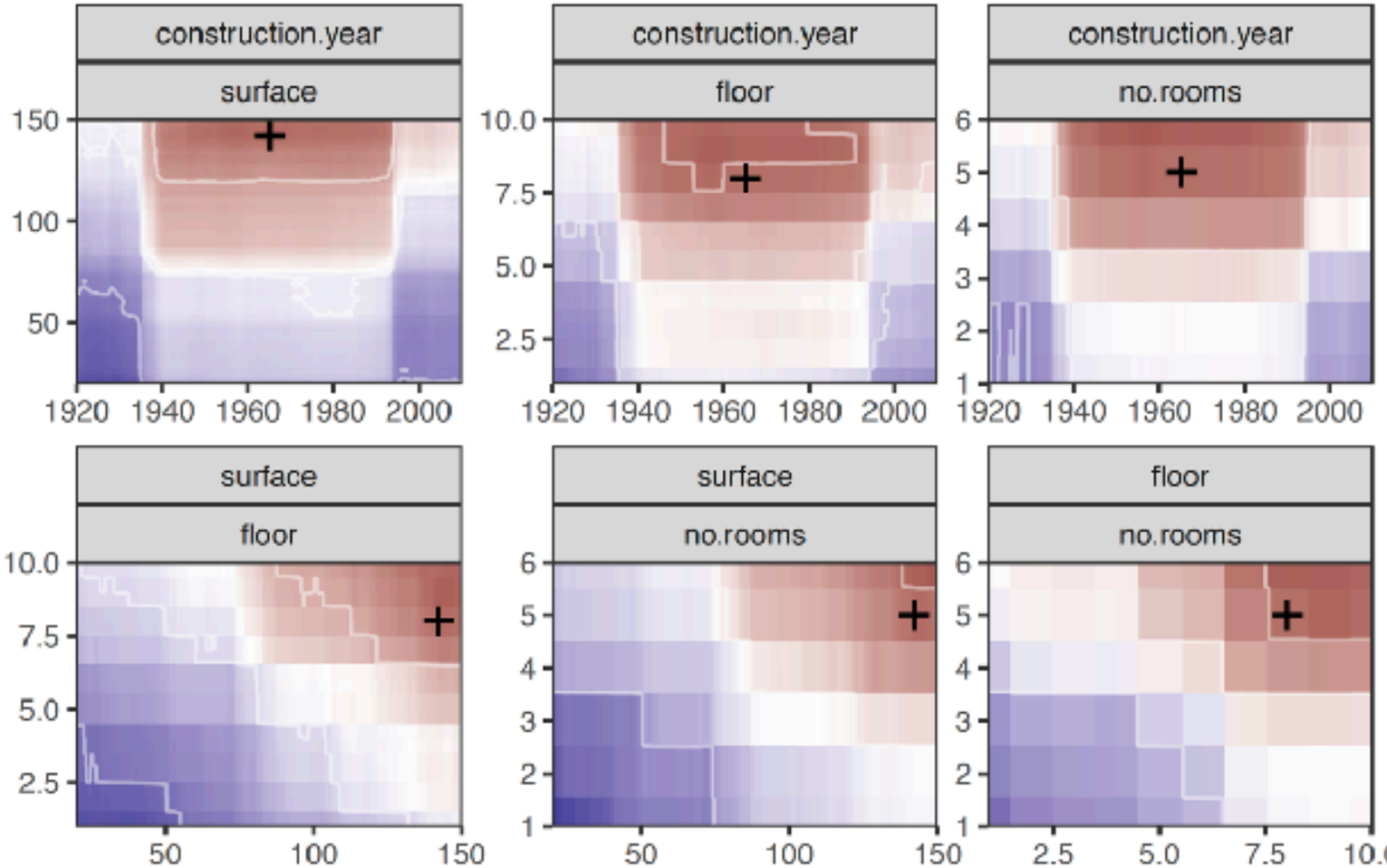
Ceteris Paribus Profiles 2D



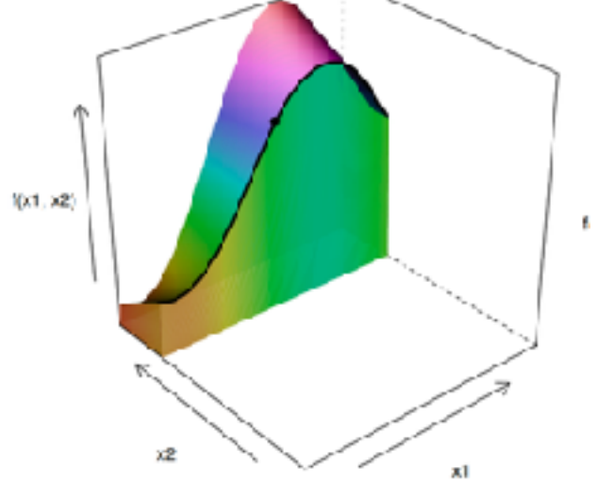
Ceteris Paribus Profiles may be easily calculated for larger number of variables.

Analysis of such profiles help to identify interaction between pairs of variables.

$$CP^{f,(j,k),x}(z_1, z_2) := f(x|^{(j,k)} = (z_1, z_2))$$

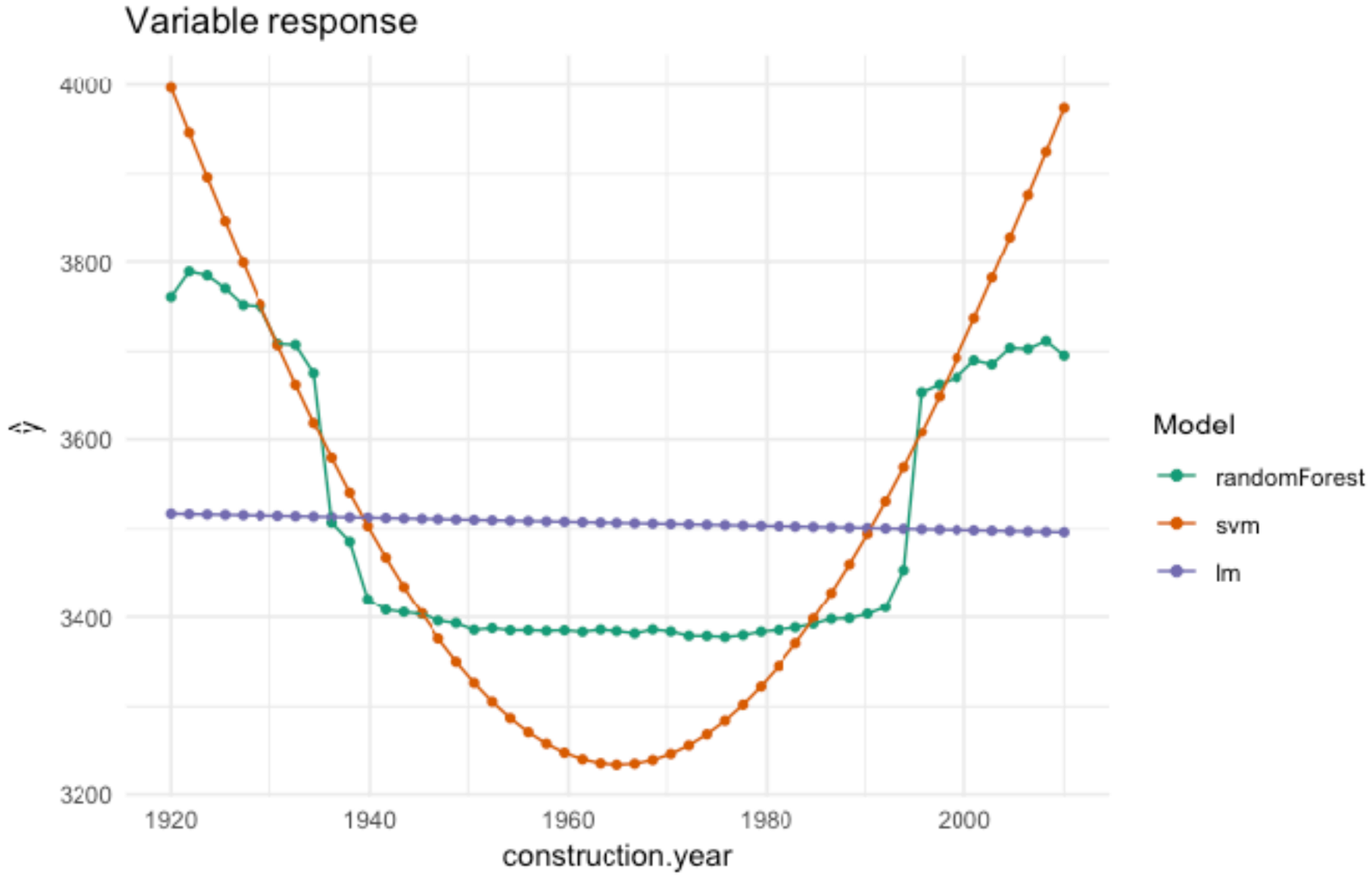


Ceteris Paribus Profiles

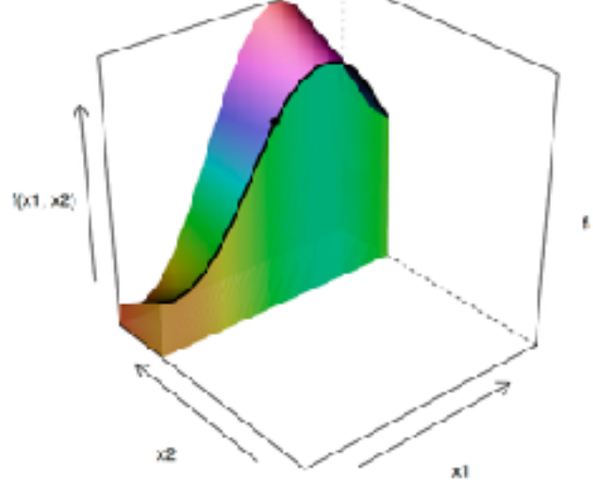


Ceteris Paribus Profiles may be used for comparisons of different models. Having few competing models one may compare their CP profiles.

Agreement (or lack of it) between competing models shows how stable are model responses.

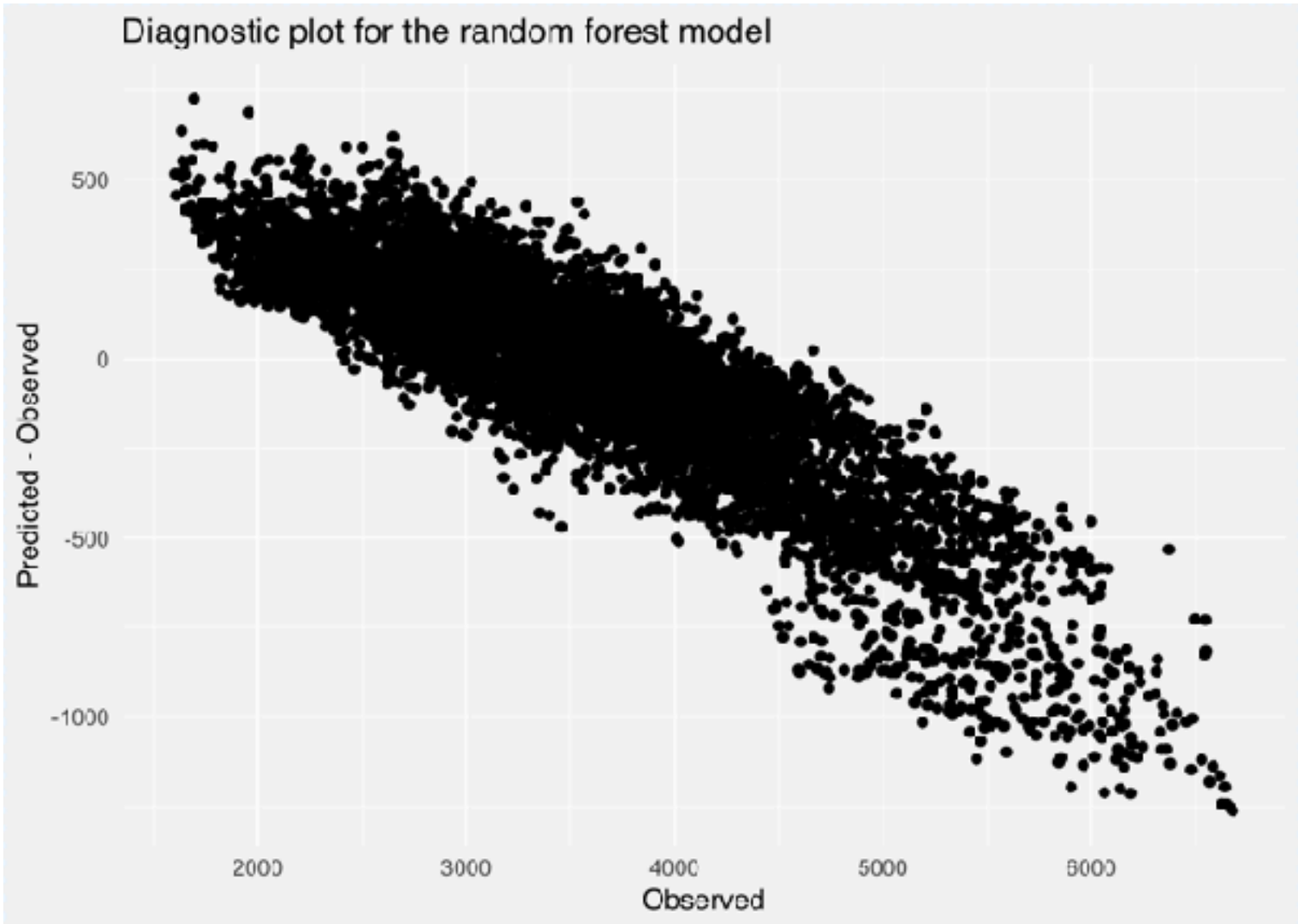


Ceteris Paribus Profiles

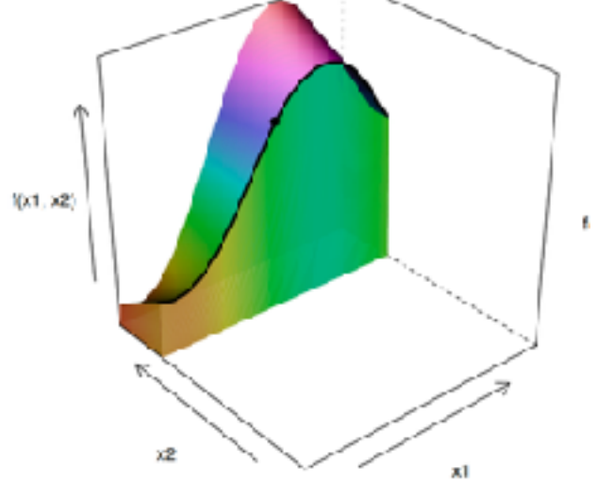


Ceteris Paribus Profiles may be used for validation of local fidelity of a model.

If model predictions are biased (e.g. for Random Forest) then we may additionally perform some checks.



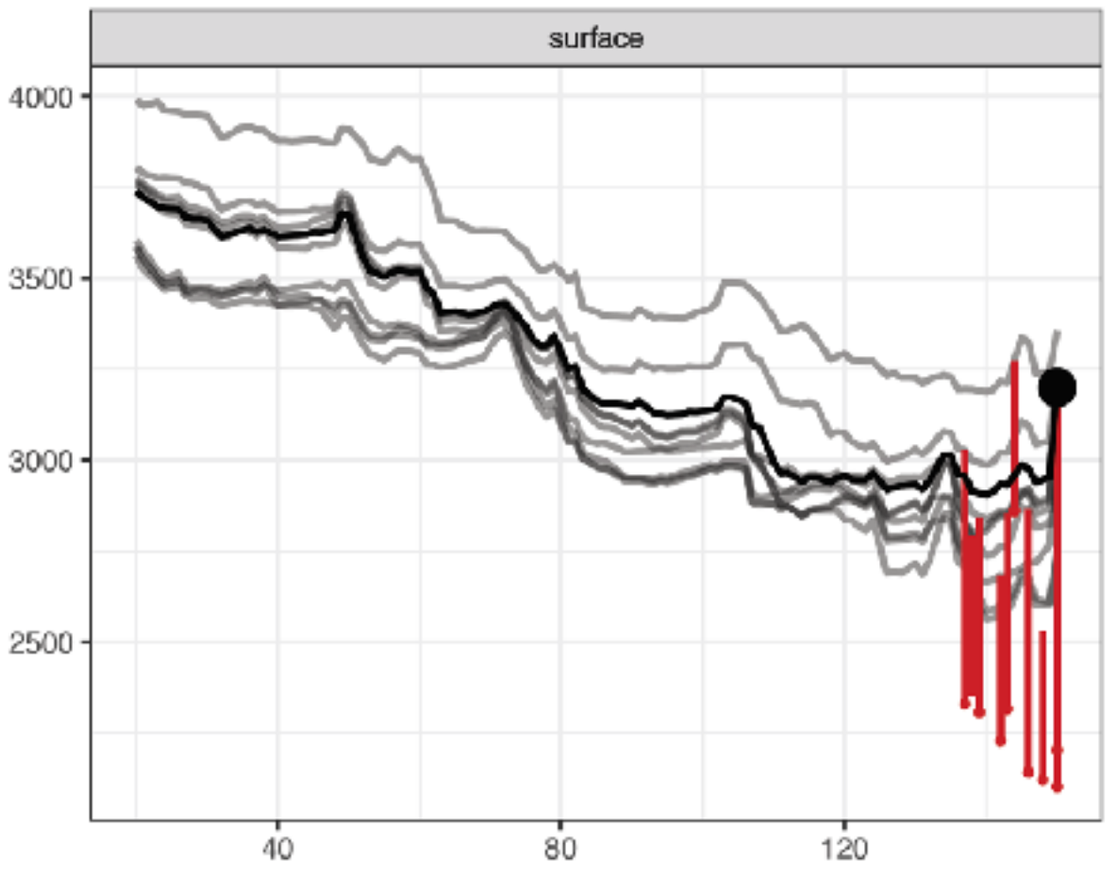
Ceteris Paribus Profiles

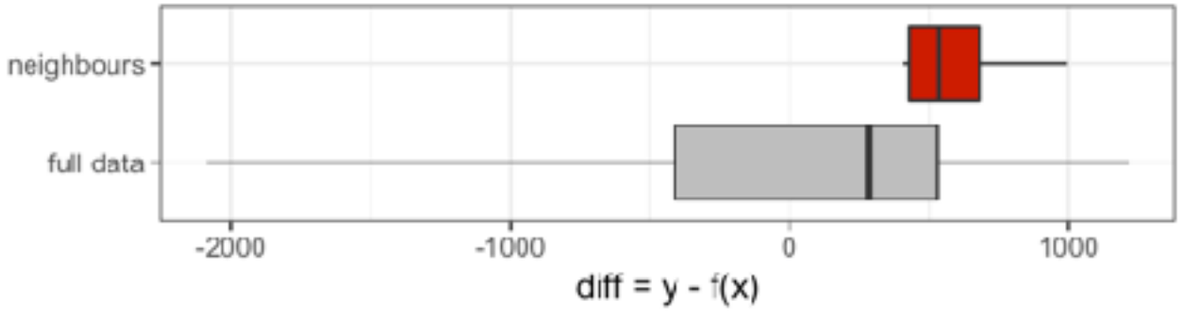


Ceteris Paribus Profiles may be used for validation of local fidelity of a model.

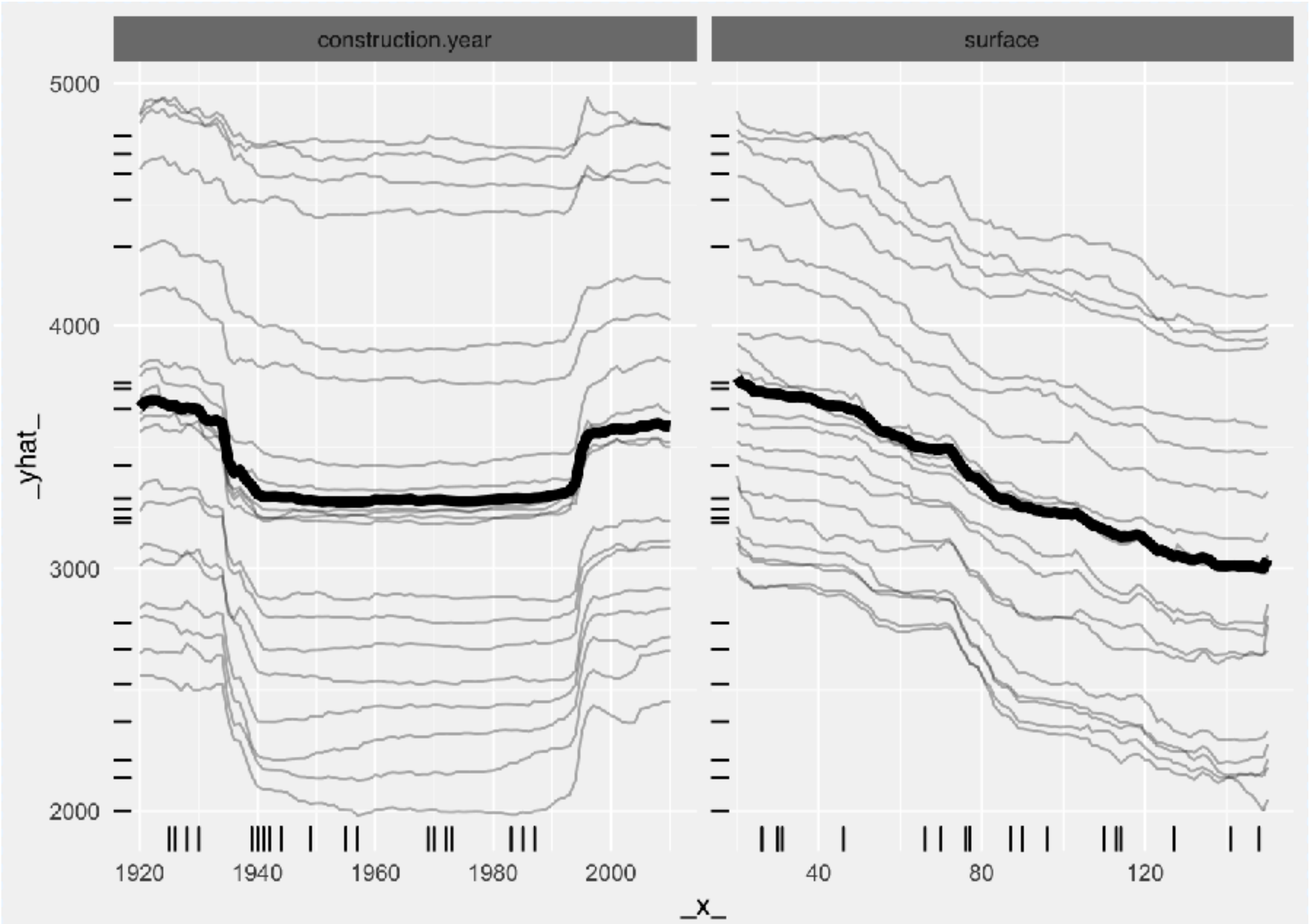
If model predictions are biased (e.g. for Random Forest) then we may additionally perform some checks.

Analysis of profiles for labeled neighbours help to assess model smoothness and stability around the point of interest.

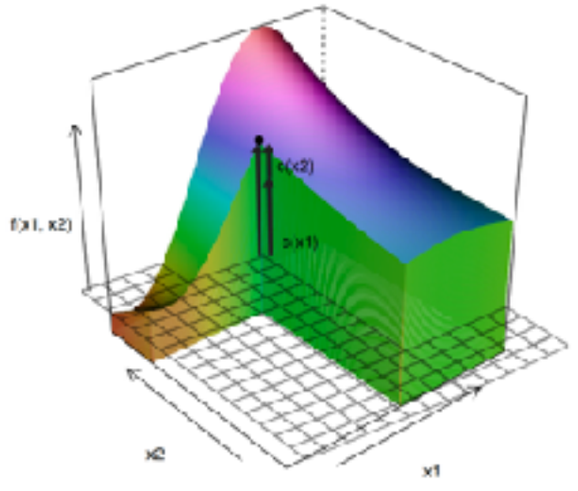




Partial Dependency Plots



Variable attributions for lm



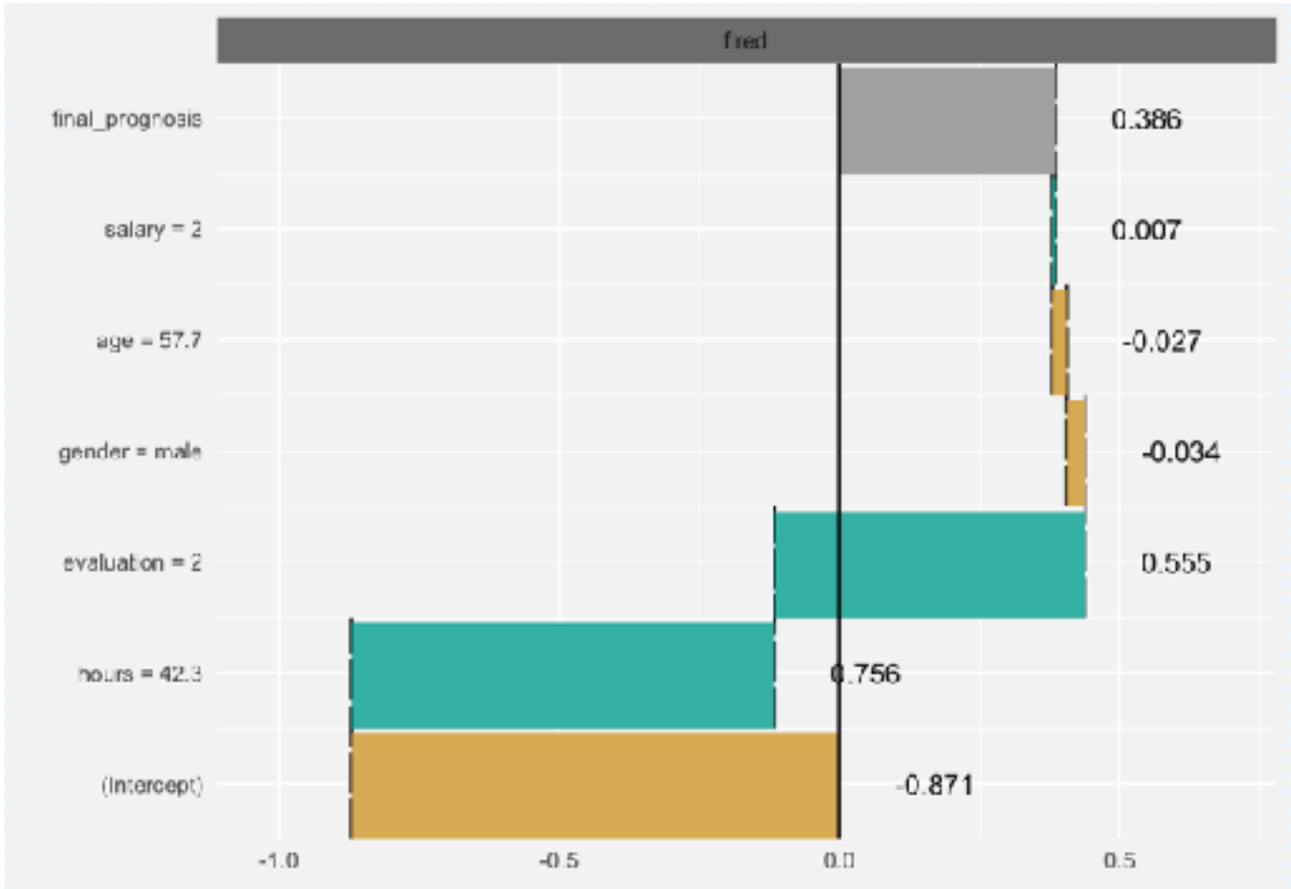
Sometimes, instead of What-If analysis we are more interested in contributions/effects of selected variables on a model prediction. There are model agnostic tools for this.

For linear models the variable attribution is relatively easy.

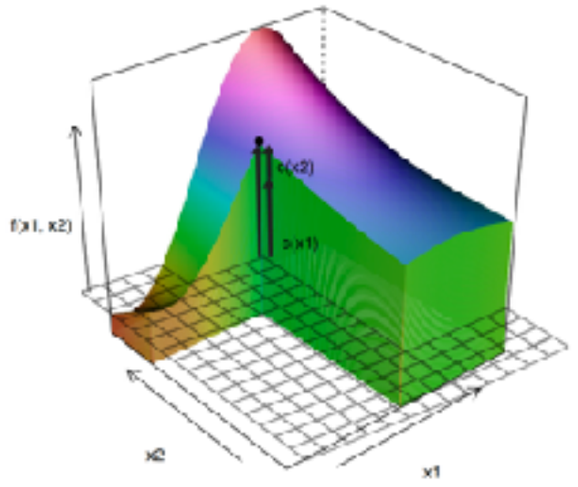
$$f(x) = \beta_0 + x_1\beta_1 + \ldots + x_p\beta_p$$

$$f(x) = baseline + (x_1 - \bar{x}_1)\beta_1 + \ldots + (x_p - \bar{x}_p)\beta_p$$

$$baseline = \mu + \bar{x}_1\beta_1 + \ldots + \bar{x}_p\beta_p$$

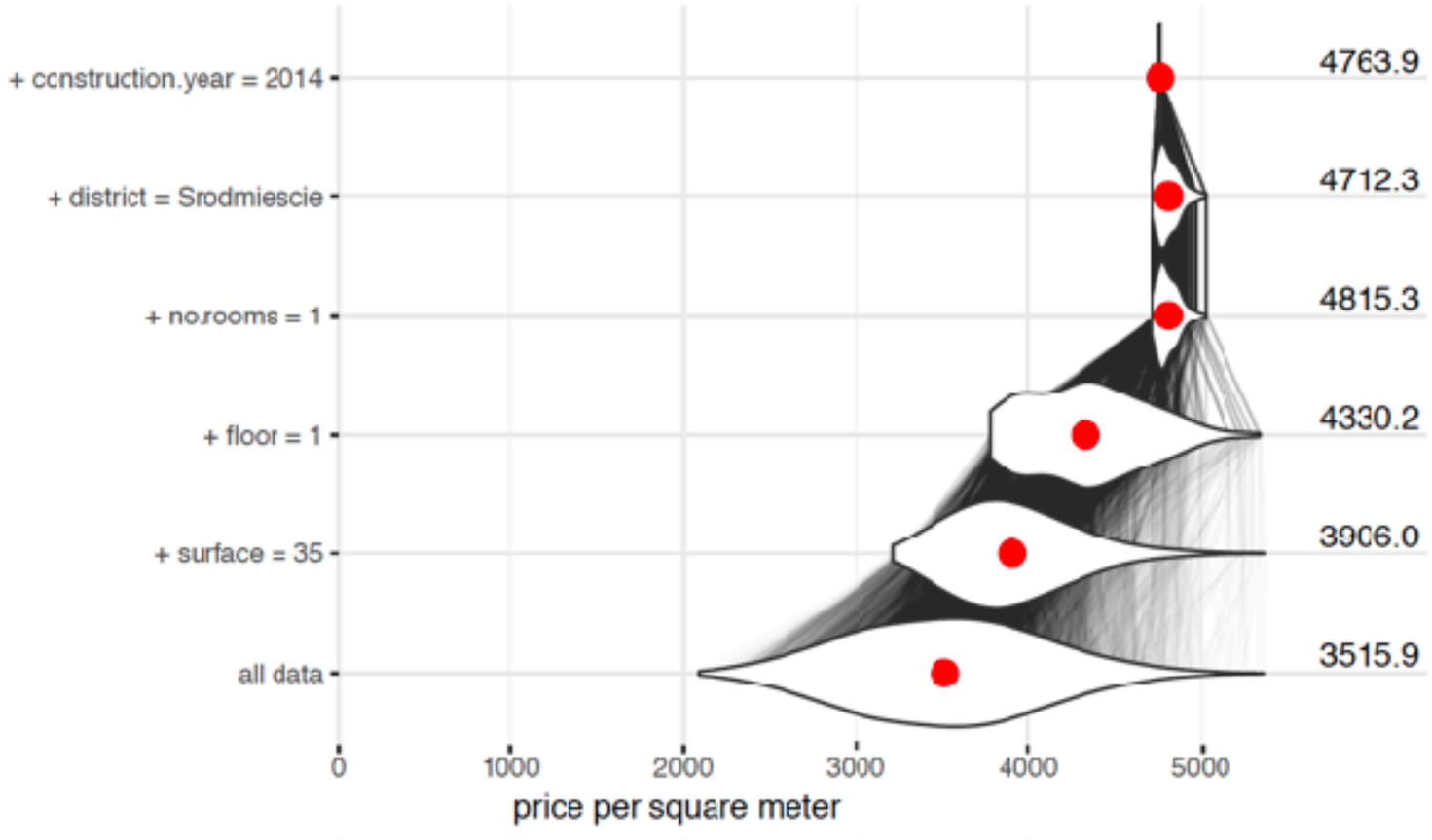


Variable attributions



For non-additive models one solution is to consider sequence of conditionings that explains how model response changes from average (baseline) to the prediction for the selected observation.

$$f(x^*) = baseline + \sum_{i=1}^{p} v(f, x^*, i)$$



Variable attributions

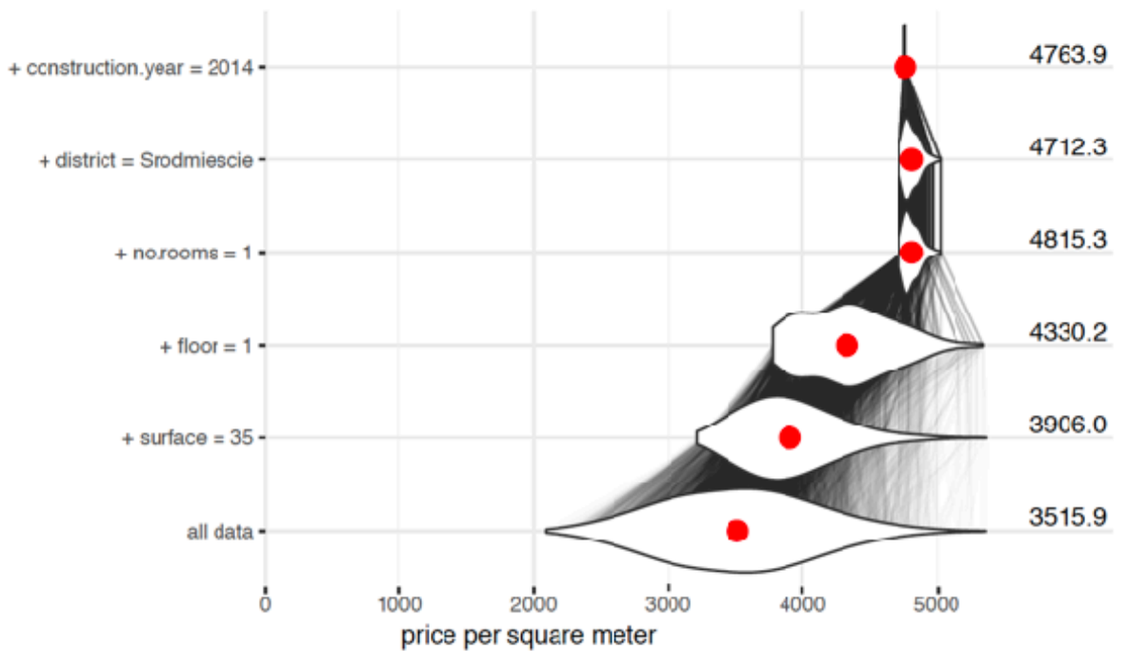
$$f(x^*) = baseline + \sum_{i=1}^{p} v(f, x^*, i)$$

$$E[f(X)|X_1 = x_1^*, \ldots, X + p = x_p^*] = E[f(X)] + \sum_{i=1}^{p} v(f, x^*, i)$$

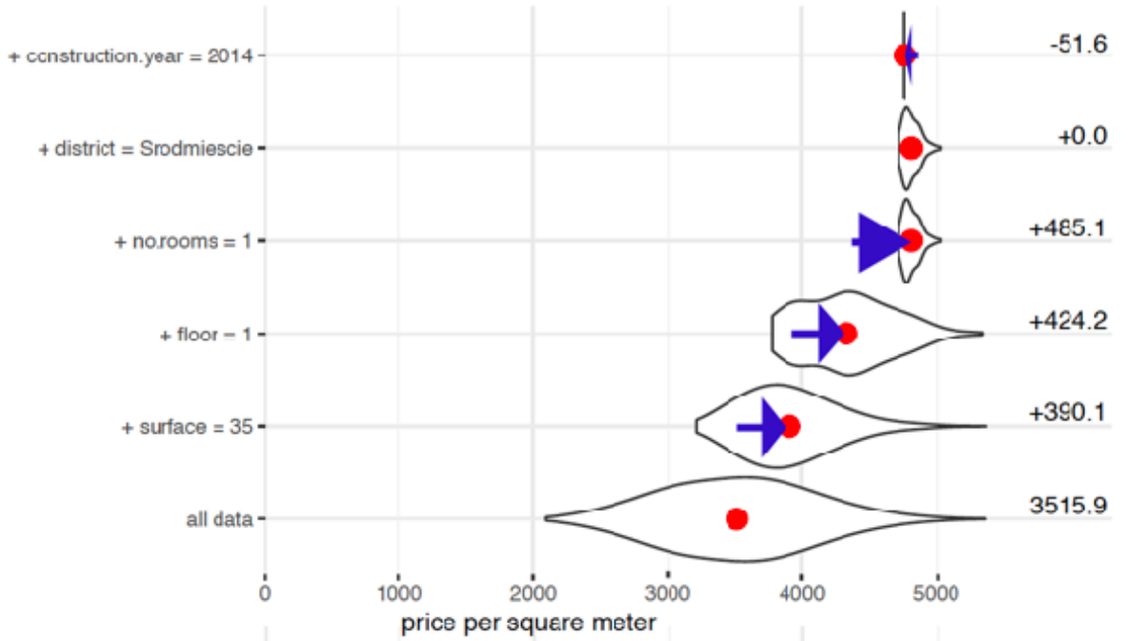
$$v(f, x_i^*, i) = E[f(X)|X_1 = x_1^*, \ldots, X_i = x_i^*] -$$

$$E[f(X)|X_1 = x_1^*, \ldots, X_{i-1} = x_{i-1}^*]$$

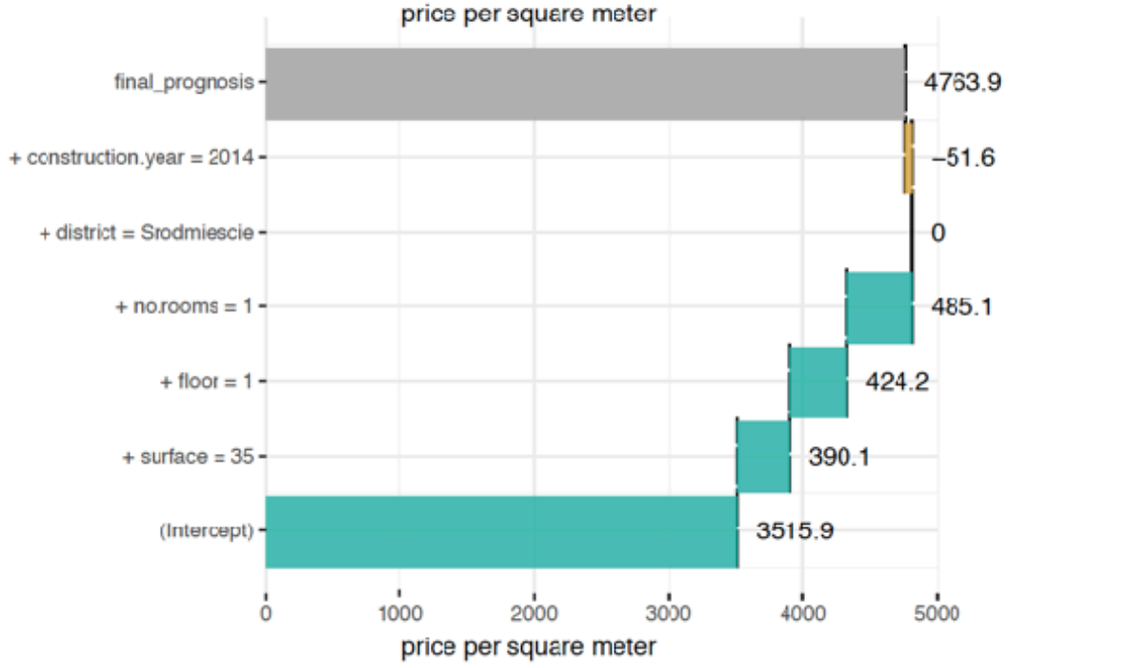
A)



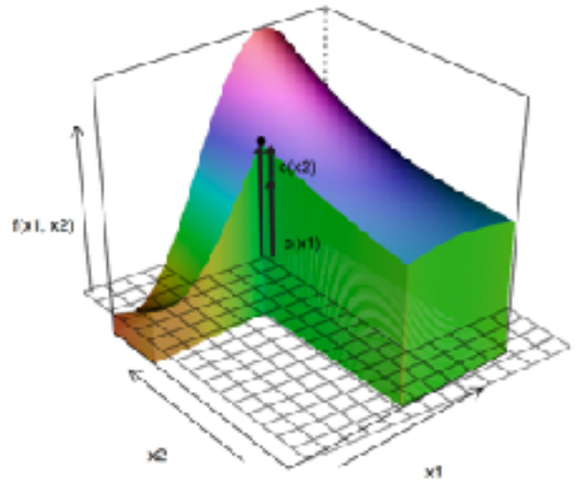
B)



C)



Variable attributions

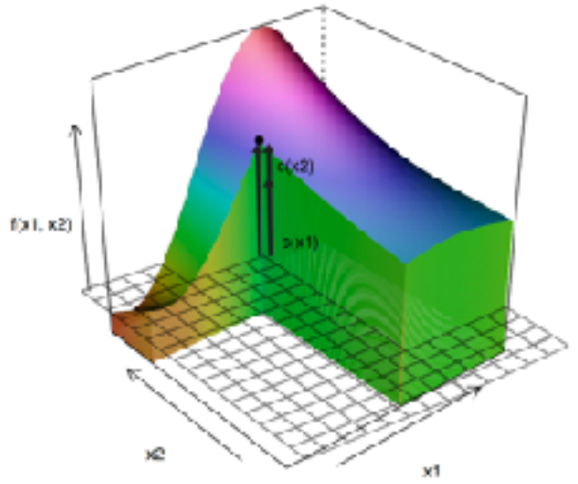


Various greedy strategies are possible for identification of the „best" ordering.

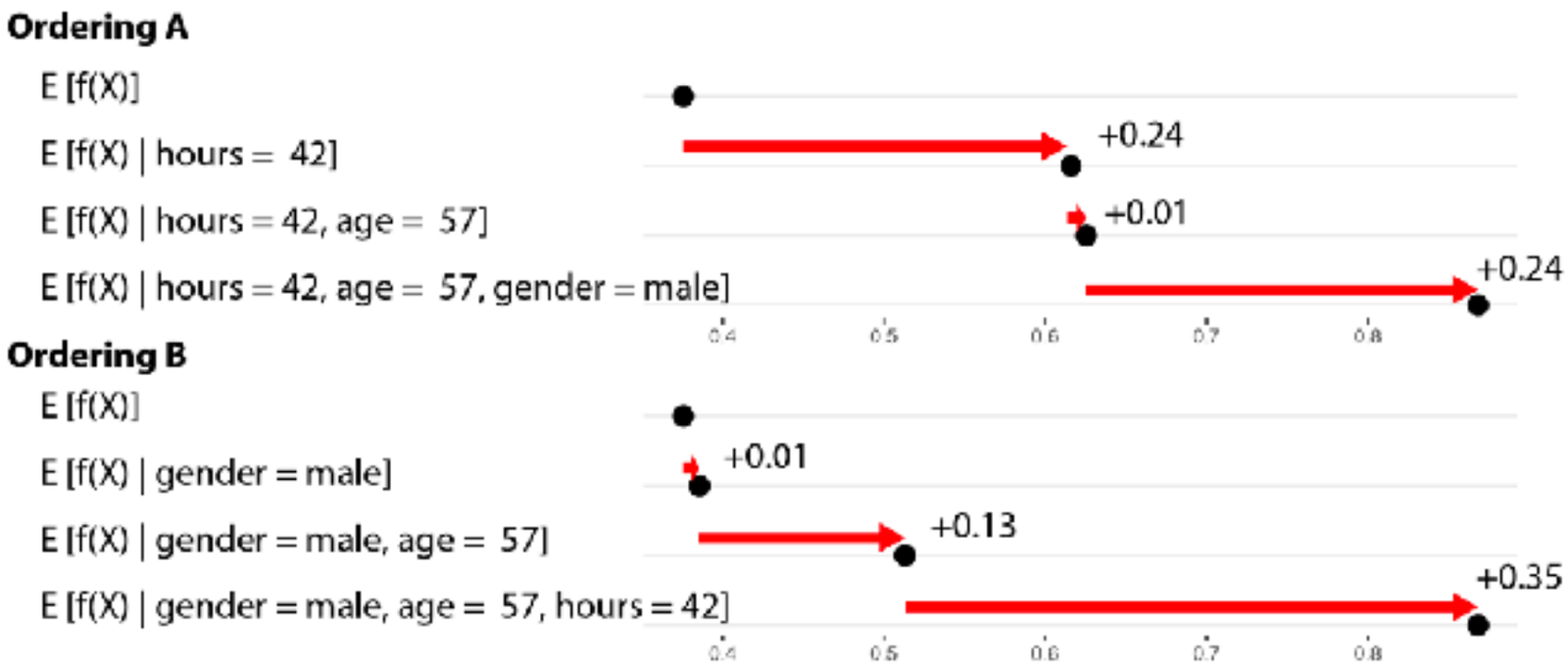
Algorithm Model agnostic break down of model predictions.

$p \leftarrow$ number of variables
$IndSet \leftarrow \emptyset$ empty set
for i in $\{1, \ldots, p\}$ **do**
    Find new variable that can be relaxed with large distance to $f^{\emptyset}(x^{new})$
    for j in $\{1, \ldots, p\} \setminus IndSet$ **do**
        Calculate relaxed distance with j added
        $dist(j) \leftarrow d(x^{new}, IndSet \cup \{j\})$
    end for
    Find and add j that maximize distance
    $j_{max} \leftarrow \arg\max_j dist(j)$
    $Contribution^{IndSet}(i) \leftarrow f^{IndSet \cup \{j_{max}\}}(x^{new}) - f^{IndSet}(x^{new})$
    $Variables(i) \leftarrow j_{max}$
    $IndSet \leftarrow IndSet \cup \{j_{max}\}$
end for

Variable attributions



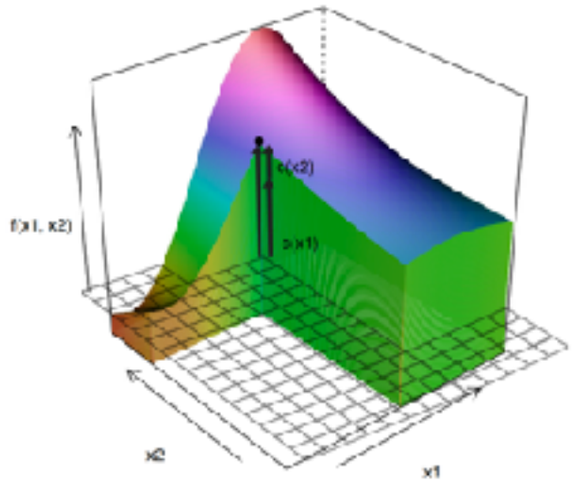
Order is important. It affects calculated attributions.



Possible solutions:

- Show a single conditioning,
- Show an average attribution (average from all possible orderings -> Shapley values),
- Use these differences to identify interactions between variables.

Variable attributions



SHAP (SHapley Additive exPlanations) is a unified framework for interpretation of model predictions. It has desired properties (Local accuracy, Missingness, Consistency)

$$v(f, x^*, i) = \frac{1}{|P|} \sum_{S \subseteq P \setminus \{i\}} \binom{|P|-1}{|S|}^{-1} \left(E[f(X)|X_{S \cup \{i\}} = x^*_{S \cup \{i\}}] - E[f(X)|X_S = x^*_S] \right)$$

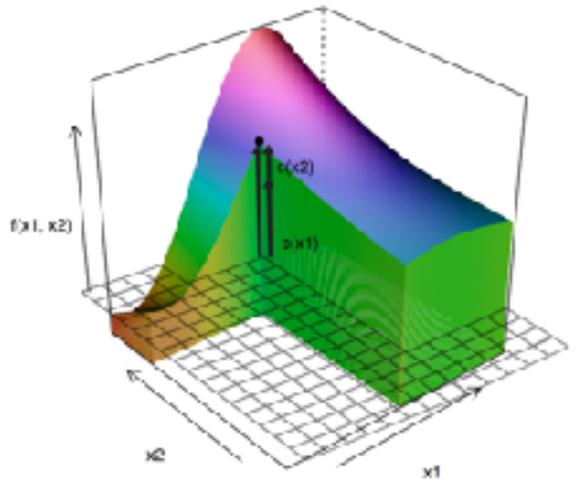
A Unified Approach to Interpreting Model Predictions

Scott M. Lundberg
Paul G. Allen School of Computer Science
University of Washington
Seattle, WA 98105
slund1@cs.washington.edu

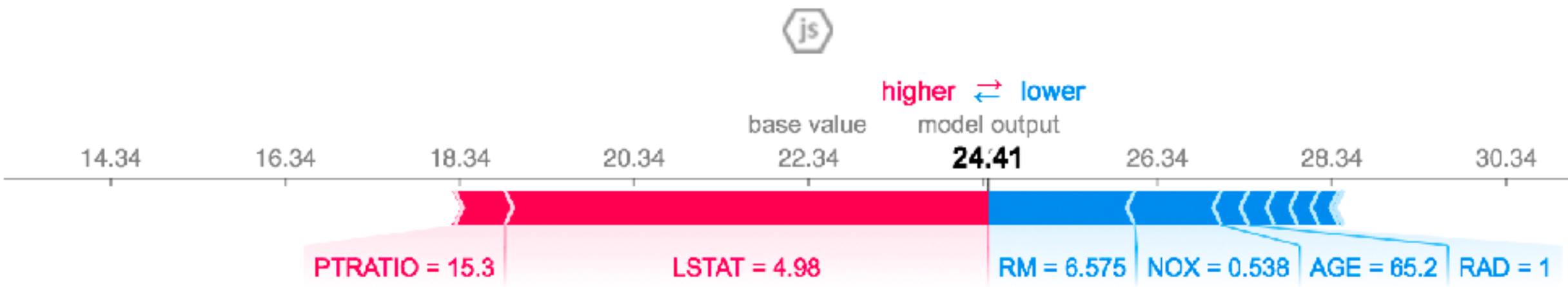
Su-In Lee
Paul G. Allen School of Computer Science
Department of Genome Sciences
University of Washington
Seattle, WA 98105
suinlee@cs.washington.edu

http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions

Variable attributions



SHAP (SHapley Additive exPlanations) is a unified framework for interpretation of model predictions. It has desired properties (Local accuracy, Missingness, Consistency) and may be seen as unification of other approaches like DeepLIFT, Layer-Wise Relevance Propagation, LIME.



A Unified Approach to Interpreting Model Predictions

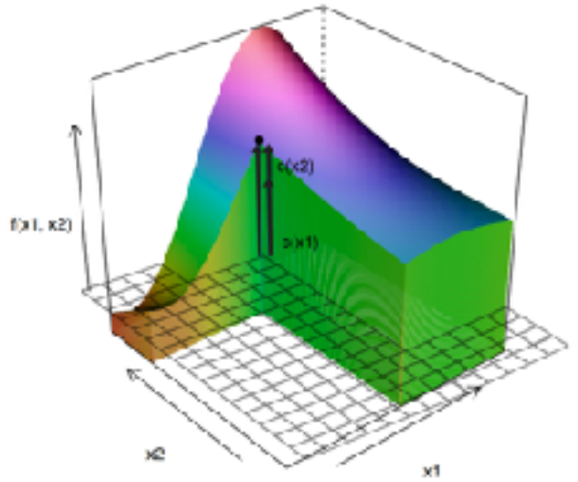
Scott M. Lundberg
Paul G. Allen School of Computer Science
University of Washington
Seattle, WA 98105
slund1@cs.washington.edu

Su-In Lee
Paul G. Allen School of Computer Science
Department of Genome Sciences
University of Washington
Seattle, WA 98105
suinlee@cs.washington.edu

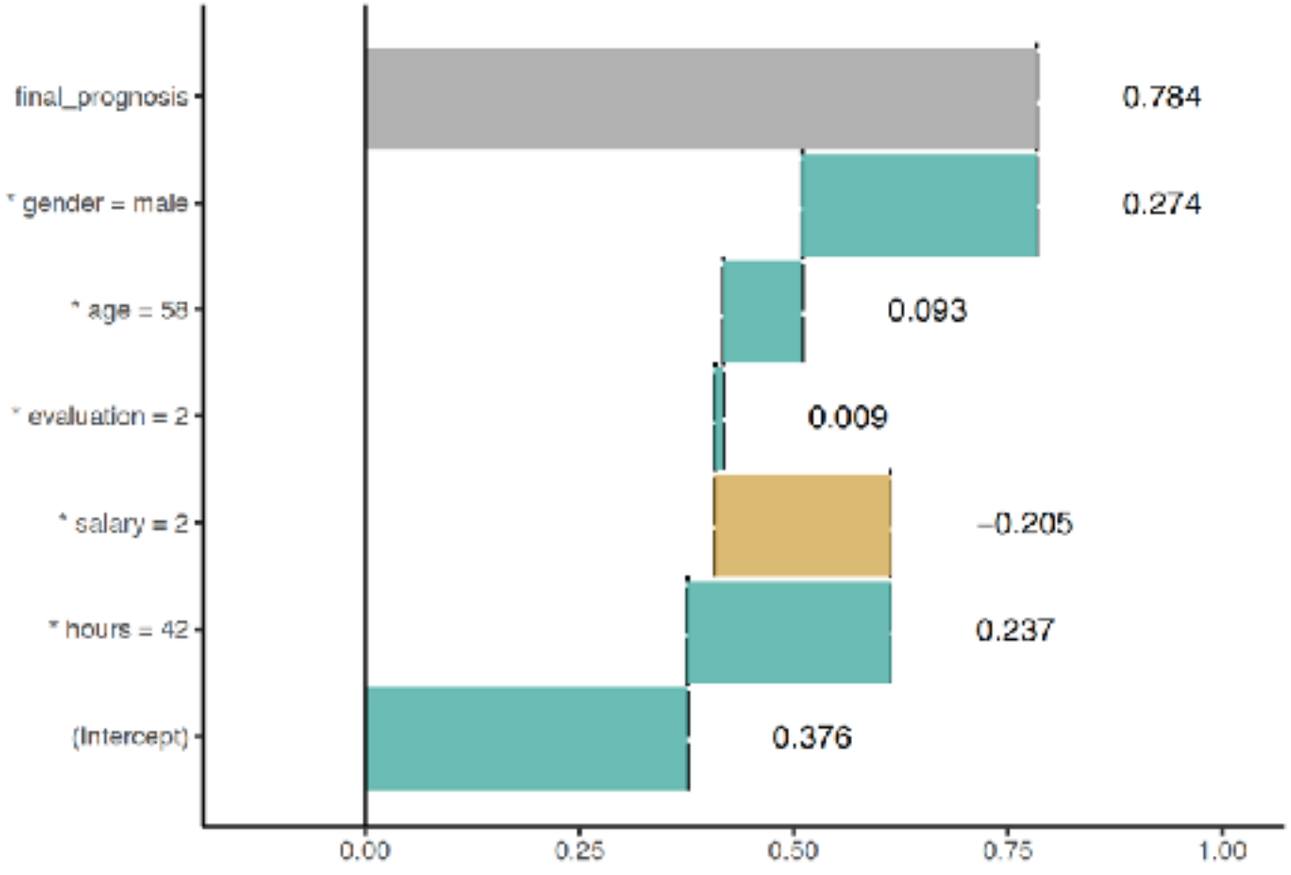
http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions

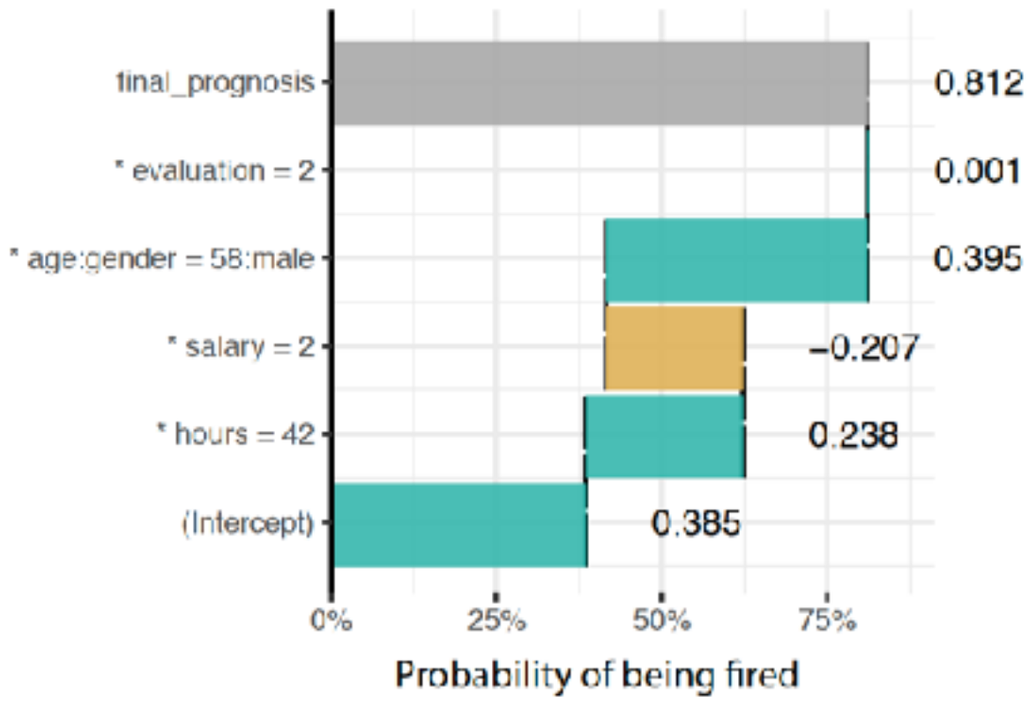
Variable attributions

Interactions may be identified when effect of two variables is different than sum of effects for each single variable.

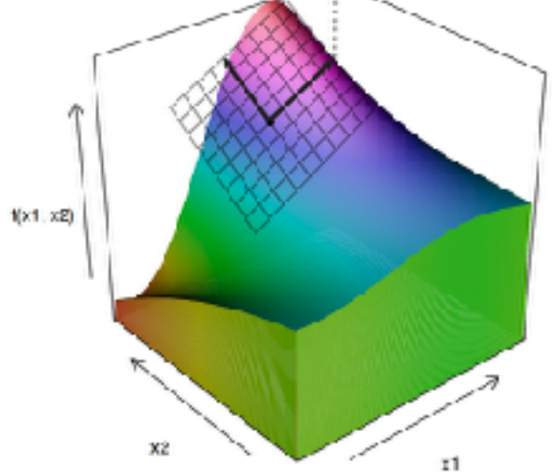


$$score_2(f, x^*, (i,j)) = \left| E[f(X)|X_i = x_i^*, X_j = x_j^*] - score_1(f, x^*, i) - score_1(f, x^*, j) + baseline \right.$$

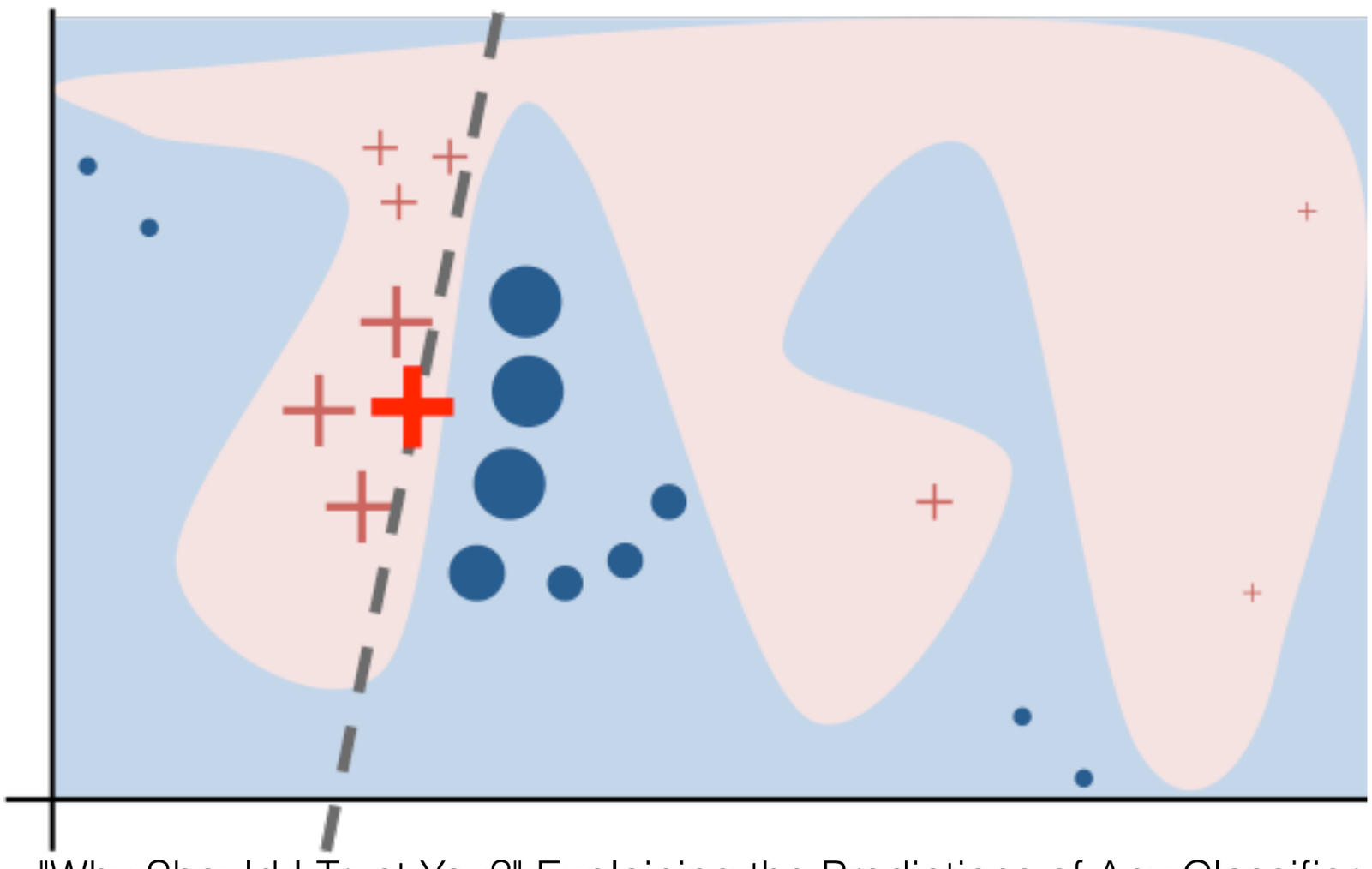




Local Model approximations



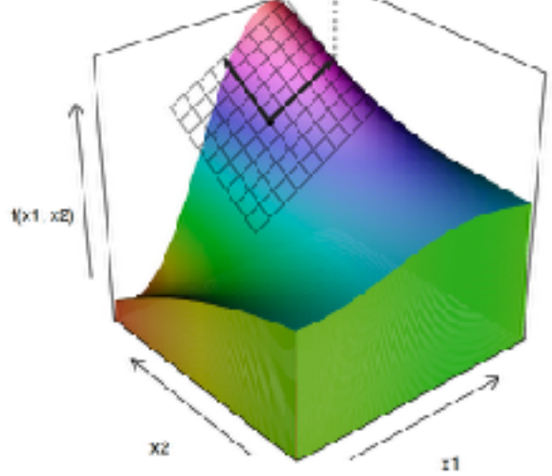
A different approach to model explanation is to locally approximate the complex black-box model with an easier to interpret white-box model constructed on interpretable features.



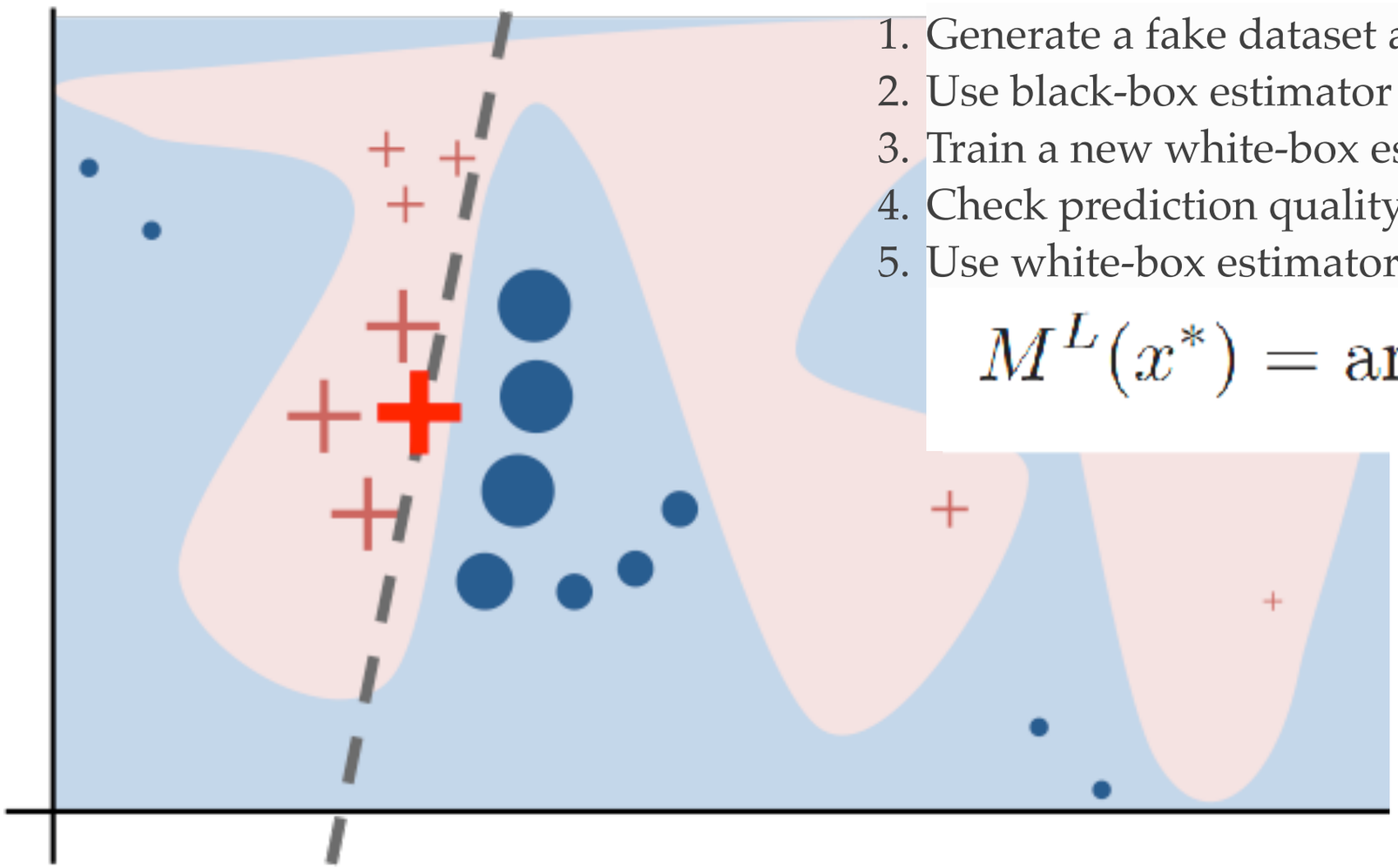
"Why Should I Trust You?" Explaining the Predictions of Any Classifier.
Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin (2016). https://arxiv.org/pdf/1602.04938.pdf

Port to R: Thomas Lin Pedersen (2017) https://github.com/thomasp85/lime
Other implementations: live (Staniak, Biecek 2018) and iml (Molnar 2018)

Local Model approximations



A different approach to model explanation is to locally approximate the complex black-box model with an easier to interpret white-box model constructed on interpretable features.



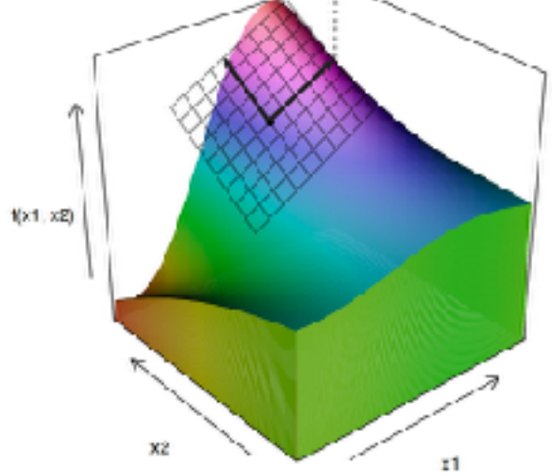
1. Generate a fake dataset around x.
2. Use black-box estimator to get target values y.
3. Train a new white-box estimator for (y,x).
4. Check prediction quality of a white-box classifier.
5. Use white-box estimator as an explanation of black-box model.

$$M^L(x^*) = \arg\min_{g \in G} L(f, g, \Pi_{x^*}) + \Omega(g$$

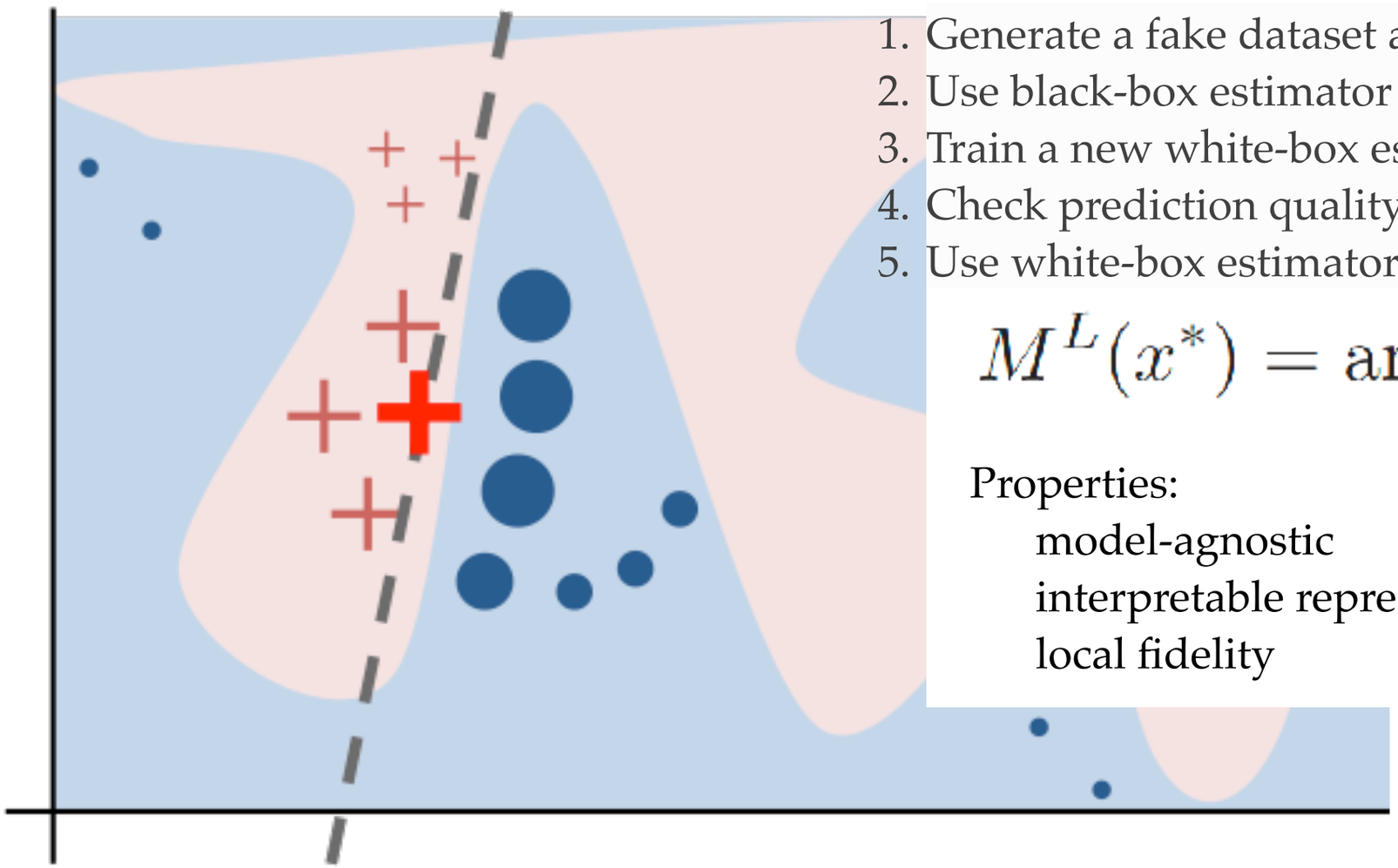
"Why Should I Trust You?" Explaining the Predictions of Any Classifier.
Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin (2016). https://arxiv.org/pdf/1602.04938.pdf

Port to R: Thomas Lin Pedersen (2017) https://github.com/thomasp85/lime
Other implementations: live (Staniak, Biecek 2018) and iml (Molnar 2018)

Local Model approximations



A different approach to model explanation is to locally approximate the complex black-box model with an easier to interpret white-box model constructed on interpretable features.



1. Generate a fake dataset around x.
2. Use black-box estimator to get target values y.
3. Train a new white-box estimator for (y,x).
4. Check prediction quality of a white-box classifier.
5. Use white-box estimator as an explanation of black-box model.

$$M^L(x^*) = \arg\min_{g \in G} L(f, g, \Pi_{x^*}) + \Omega(g$$

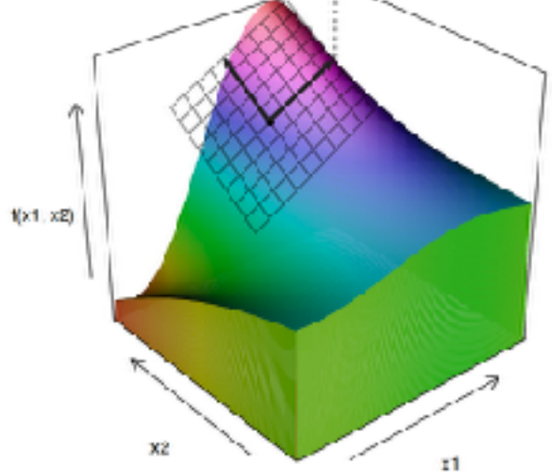
Properties:

- model-agnostic
- interpretable representation
- local fidelity

"Why Should I Trust You?" Explaining the Predictions of Any Classifier.
Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin (2016). https://arxiv.org/pdf/1602.04938.pdf

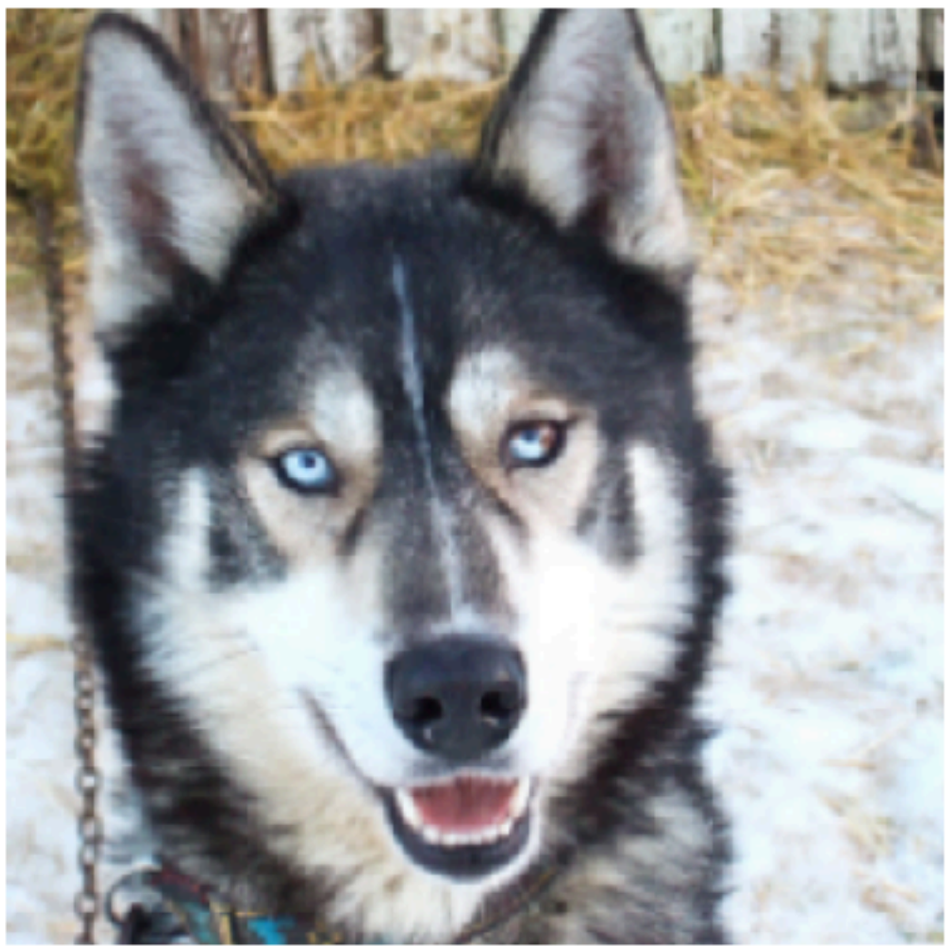
Port to R: Thomas Lin Pedersen (2017) https://github.com/thomasp85/lime
Other implementations: live (Staniak, Biecek 2018) and iml (Molnar 2018)

Local Model approximations



Spectacular use-cases for image data and text data.

Not that developed for tabular data (what are interpretable features?
do we have to reduce continuous variables to sets of binary features).



(a) Husky classified as wolf

(b) Explanation

"Why Should I Trust You?" Explaining the Predictions of Any Classifier.
Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin (2016). https://arxiv.org/pdf/1602.04938.pdf

How to understand a black-box model?

Choose the right visual explainer in 2.875 simple steps

1. Want to understand a model or a single prediction?
- entire model
- prediction for a single observation

2. Is it *how to change it* or *why it happened*?
- interested in *what-if* scenarios
- how variables affected this single prediction

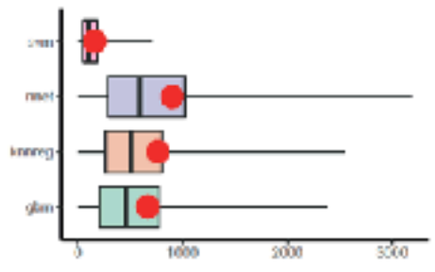
3. Variable attribution or importance?
- decompose prediction (breakDown, Shapley)
- identify key features (live, LIME)

3. Evaluate performance or validate fit?
- compare models performance
- audit residuals and goodness of fit

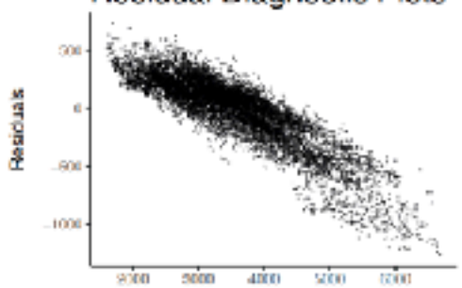
2. Interested in model performance or structure?
- how good is the model
- how does it work

3. Which variable are you interested in?
- all
- a categorical
- a continuous

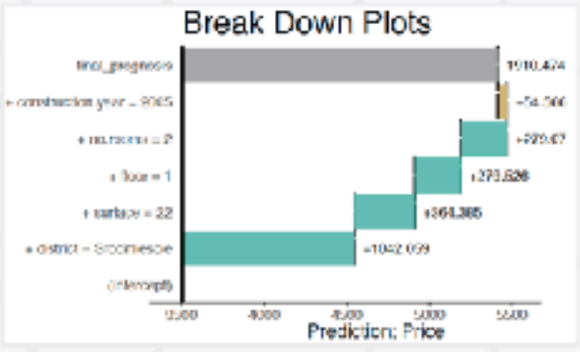
Model Performance Plots



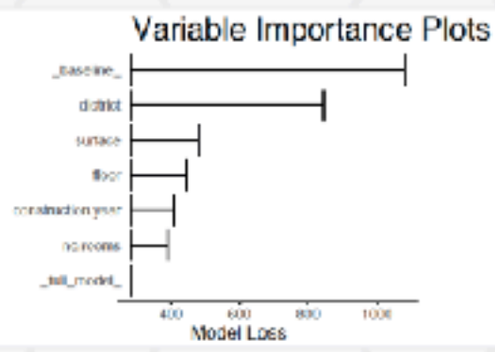
Residual Diagnostic Plots



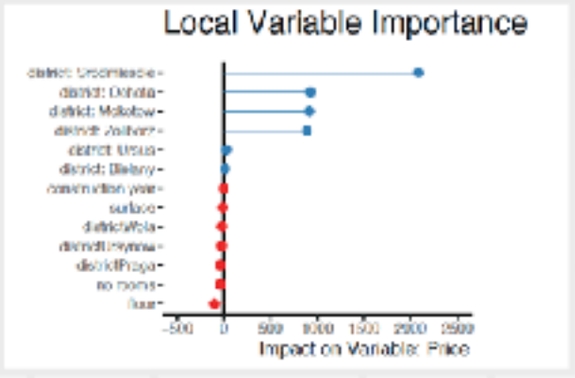
Break Down Plots



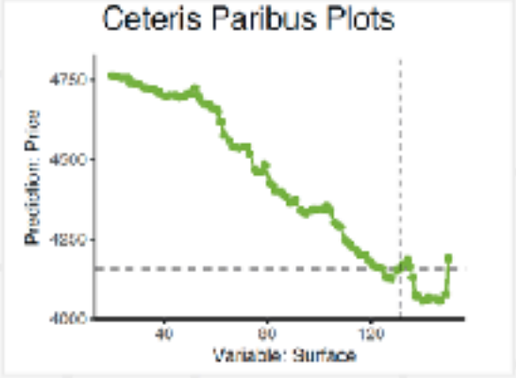
Variable Importance Plots



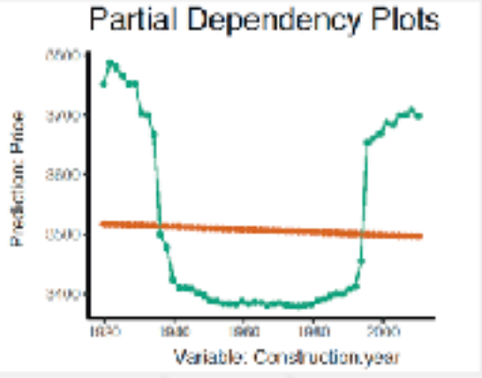
Local Variable Importance



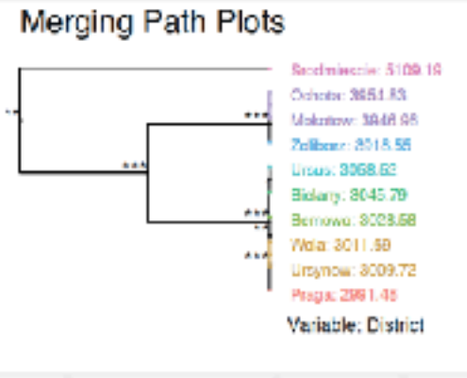
Ceteris Paribus Plots



Partial Dependency Plots

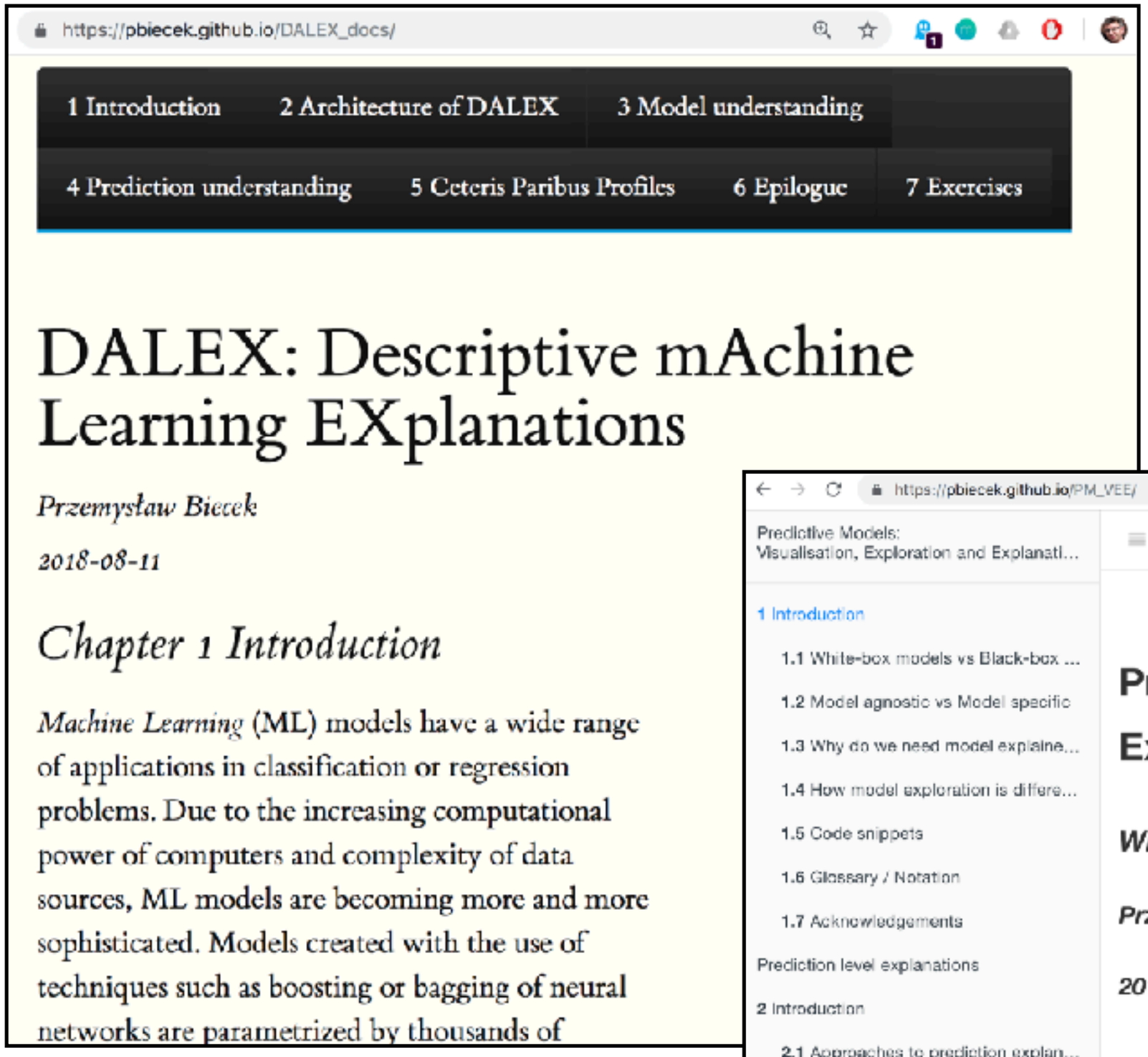


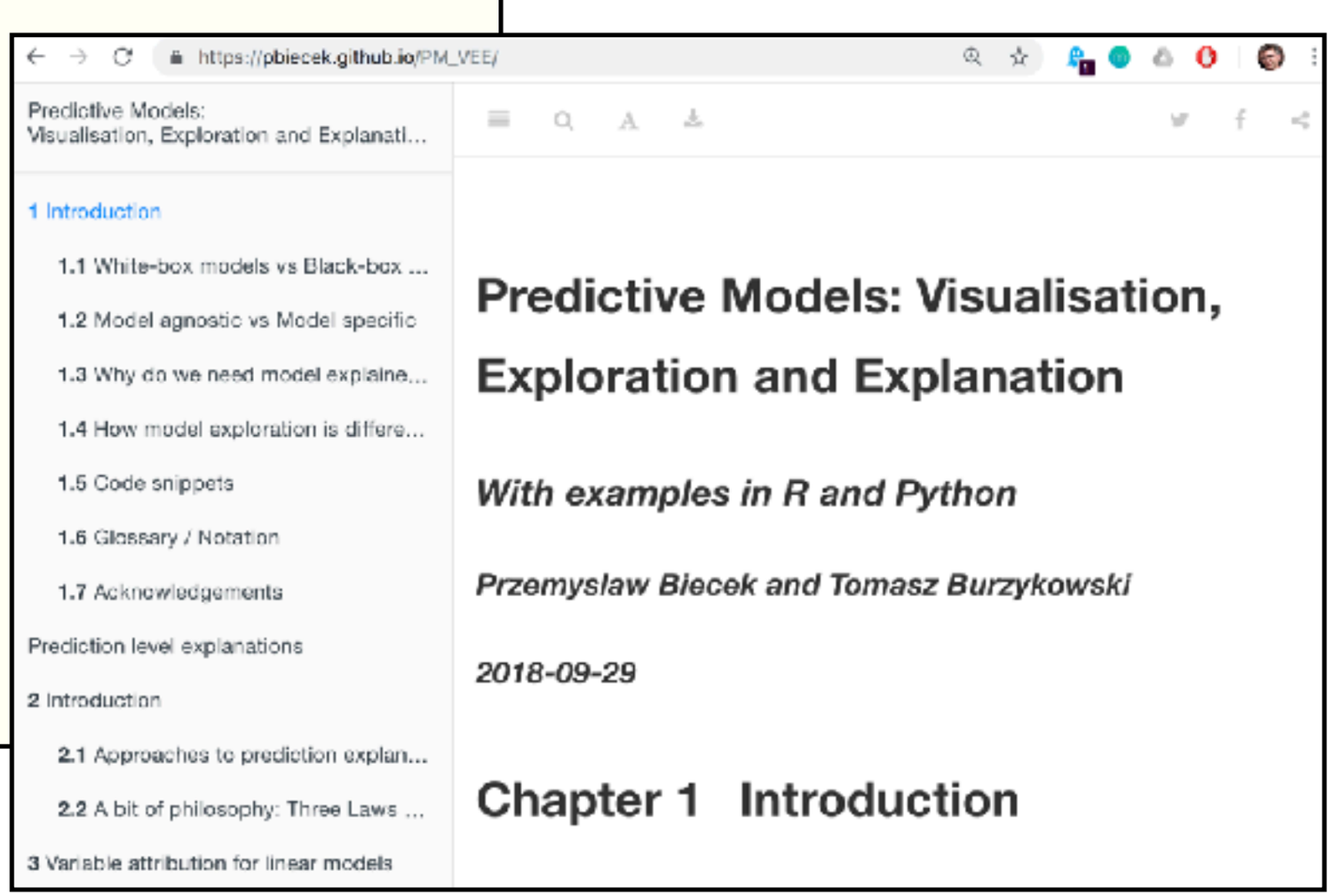
Merging Path Plots



Find more at:
https://github.com/pbiecek/DALEX

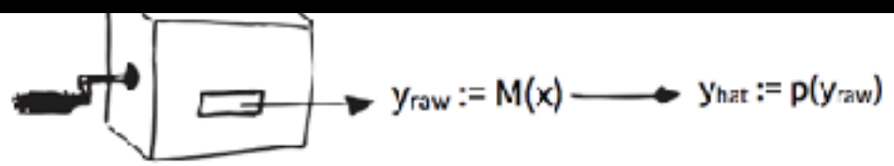
Find more about DALEX and XAI





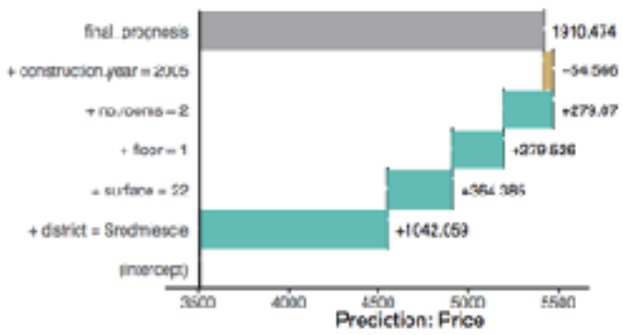
DALEX architecture

A) model

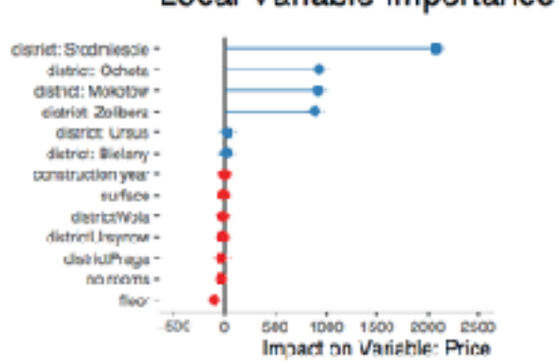


B) explain(model; data; y; predict_function)

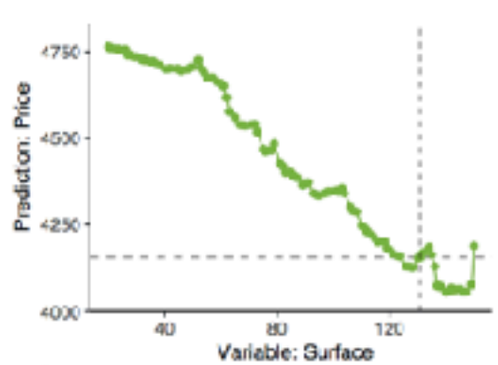
C)



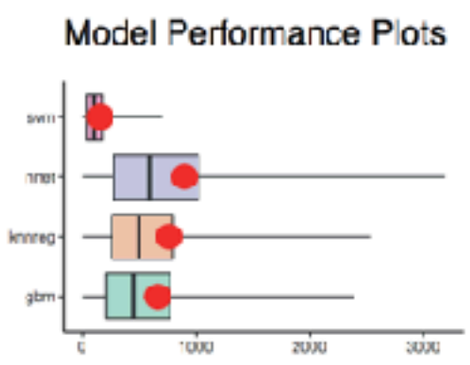
D)



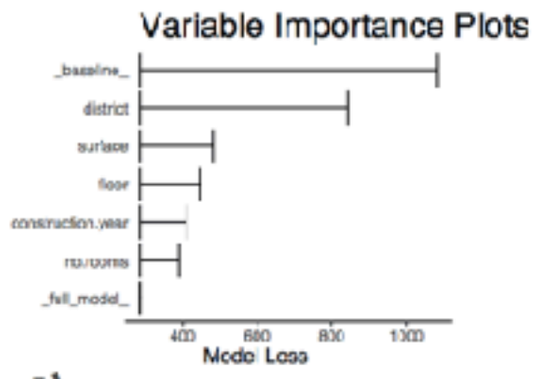
E)



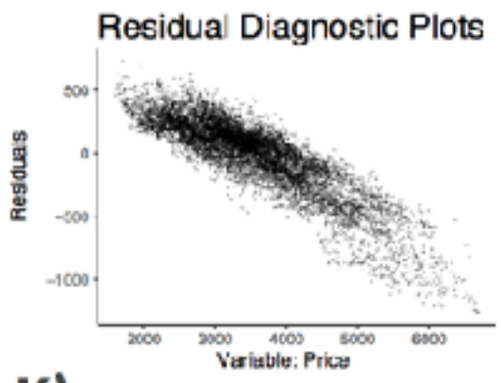
F)



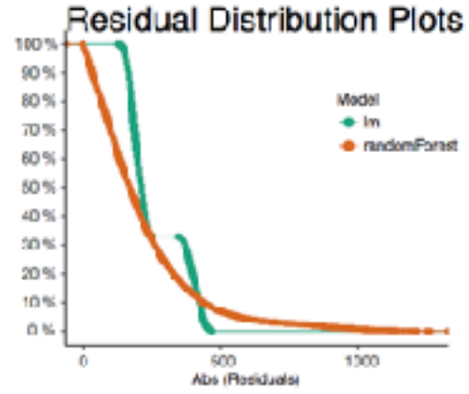
G)



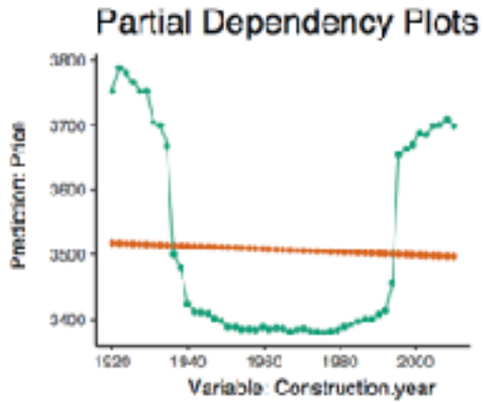
H)



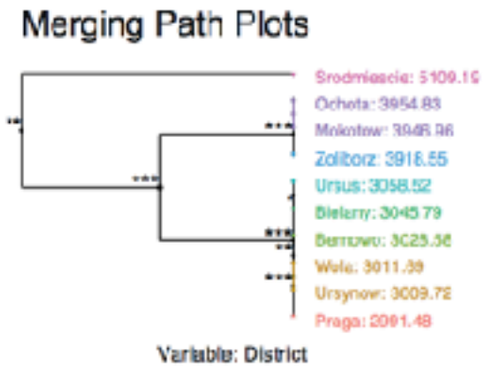
I)



J)



K)



Wrap a model

↓

Create an explainer

↓

Plot the explainer

modelDown: pkgdown for models

https://github.com/MI2DataLab/modelDown

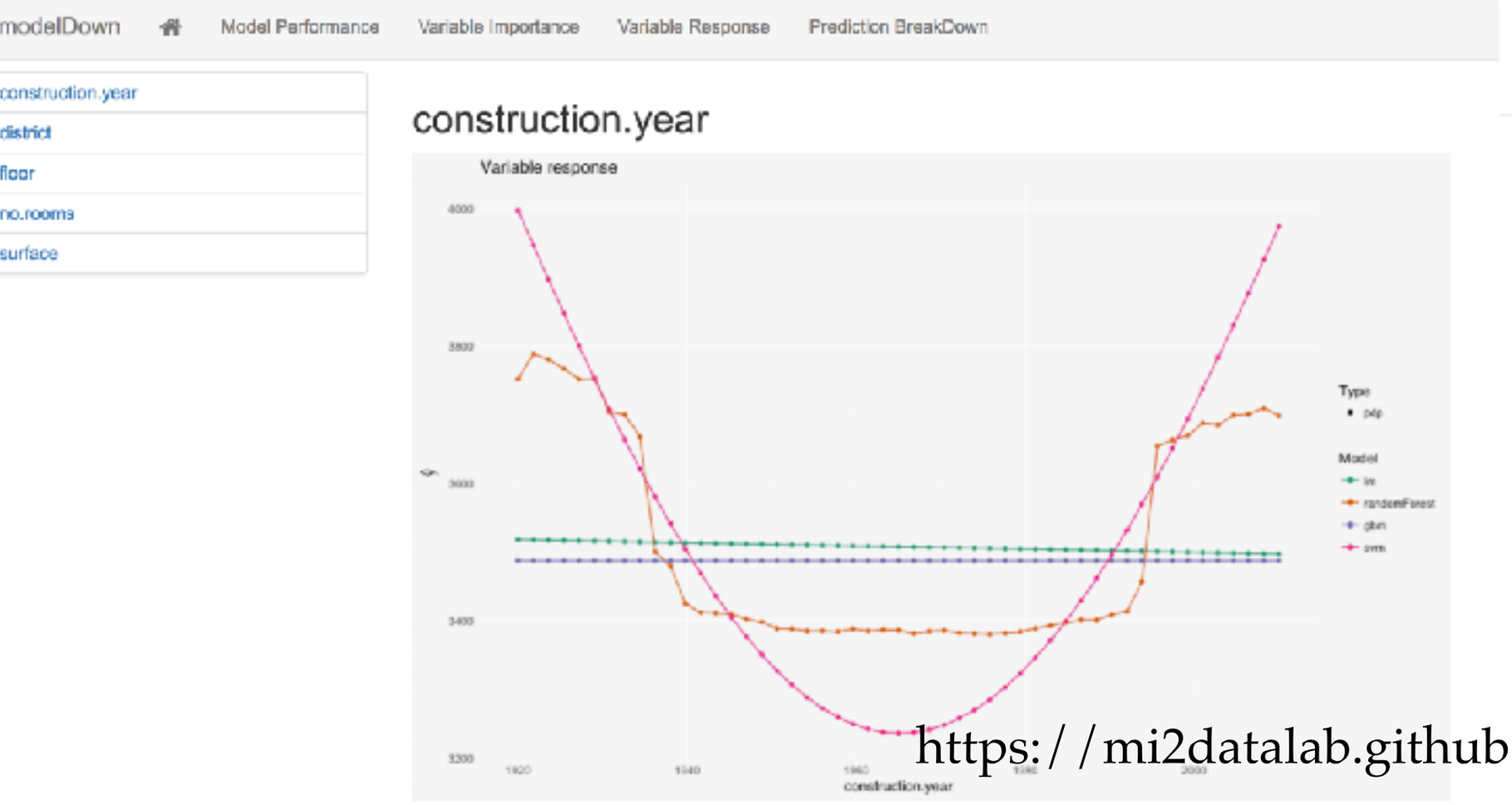
modelDown

build passing

`modelDown` generates a website with HTML summaries for predictive models. Is uses DALEX explainers to compute and plot summaries of how given models behave. We can see how exactly scores for predictions were calculated (Prediction BreakDown), how much each variable contributes to predictions (Variable Response), which variables are the most important for a given model (Variable Importance) and how well out models behave (Model Performance).

`pkgdown` documentation: https://mi2datalab.github.io/modelDown/

An example website for regression models: https://mi2datalab.github.io/modelDown_example/



https://mi2datalab.github.io/modelDown_example/

References

Apley, Dan. 2018. ALEPlot: Accumulated Local Effects (Ale) Plots and Partial Dependence (Pd) Plots.

Bach, Sebastian, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. 2015. “On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation.” PLOS ONE 10 (7):e0130140. https://doi.org/10.1371/journal.pone.0130140.

Edwards, Lilian, and Michael Veale. 2018. “Enslaving the Algorithm: From a ‘Right to an Explanation’ to a ‘Right to Better Decisions’?” IEEE Security & Privacy 16 (3):46–54. https://doi.org/10.1109/MSP.2018.2701152.

Fisher, A., C. Rudin, and F. Dominici. 2018. “Model Class Reliance: Variable Importance Measures for any Machine Learning Model Class, from the ‘Rashomon’ Perspective.”

Goldstein, Alex, Adam Kapelner, Justin Bleich, and Emil Pitkin. 2015. “Peeking Inside the Black Box.” Journal of Computational and Graphical Statistics 24 (1):44–65. https://doi.org/10.1080/10618600.2014.907095.

Greenwell, Brandon M. 2017. “pdp: An R Package for Constructing Partial Dependence Plots.” The R Journal 9 (1):421–36.

Lipton, Z. C., A. Chouldechova, and J. McAuley. 2017. “Does mitigating ML’s impact disparity require treatment disparity?”

Lundberg, Scott M., Gabriel G. Erion, and Su-In Lee. 2018. “Consistent Individualized Feature Attribution for Tree Ensembles.” CoRR abs/1802.03888.

Lundberg, Scott M, and Su-In Lee. 2017. “A Unified Approach to Interpreting Model Predictions.” In Advances in Neural Information Processing Systems, 4765–74. Curran Associates, Inc.

Molnar, Christoph. 2018. Interpretable Machine Learning. https://christophm.github.io/interpretable-ml-book/.

O’Connell, Mark, Catherine Hurley, and Katarina Domijan. 2017. “Conditional Visualization for Statistical Models: An Introduction to the Condvis Package in R.” Journal of Statistical Software, Articles 81 (5):1–20.

O’Neil, Cathy. 2016. Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy. New York, NY, USA: Crown Publishing Group.

Paluszynska, Aleksandra, and Przemyslaw Biecek. 2017. RandomForestExplainer: Explaining and Visualizing Random Forests in Terms of Variable Importance.

Puri, Nikaash, Piyush Gupta, Pratiksha Agarwal, Sukriti Verma, and Balaji Krishnamurthy. 2017. “MAGIX: Model Agnostic Globally Interpretable Explanations.” CoRR abs/1706.07160.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. “Why Should I Trust You?: Explaining the Predictions of Any Classifier.” In, 1135–44. ACM Press.

Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman. 2013. “Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps.” CoRR abs/1312.6034.

Sitko, Agnieszka, Aleksandra Grudziąż, and Przemyslaw Biecek. 2018. FactorMerger: The Merging Path Plot.

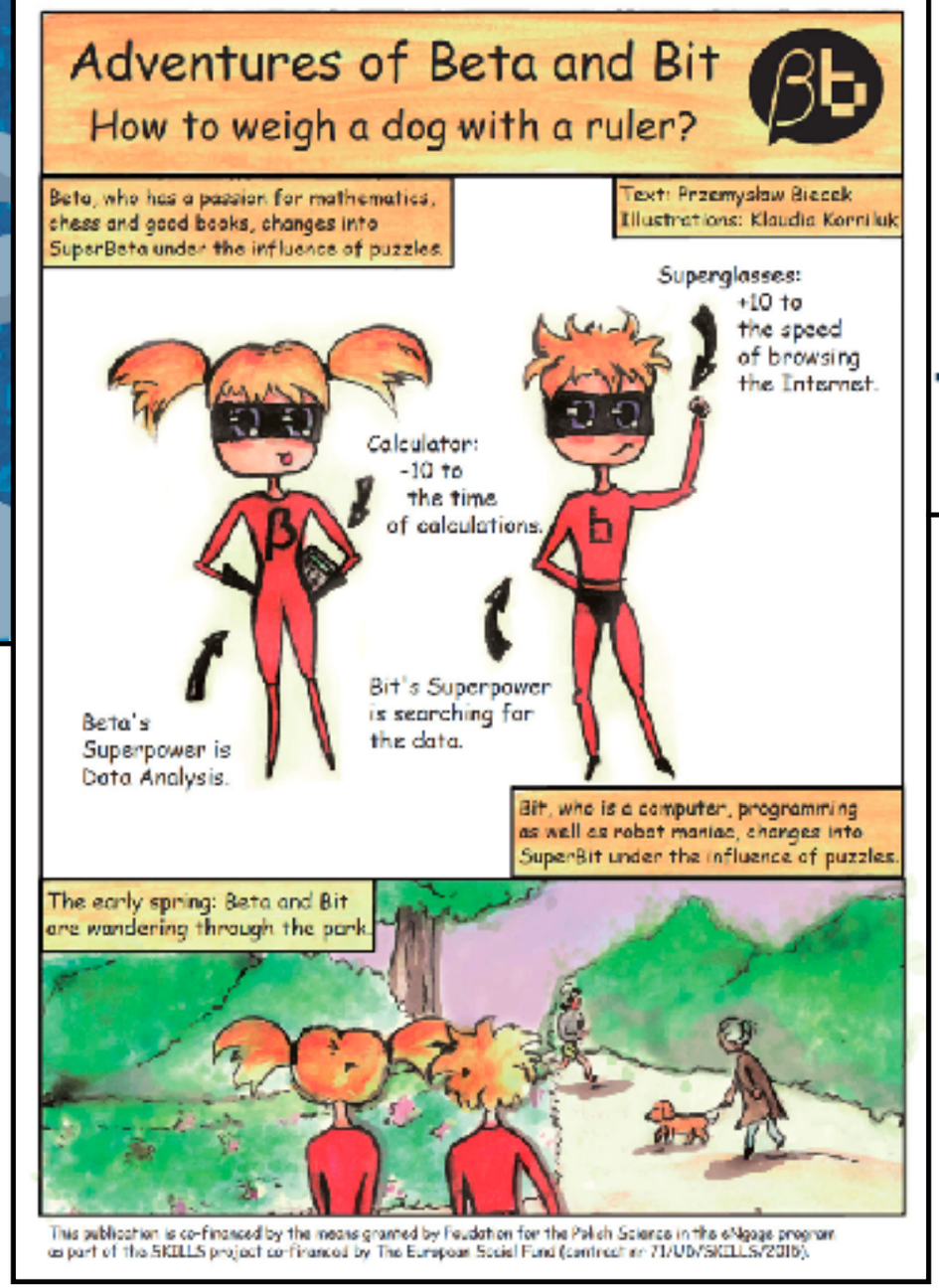
Strobl, Carolin, Anne-Laure Boulesteix, Thomas Kneib, Thomas Augustin, and Achim Zeileis. 2008. “Conditional Variable Importance for Random Forests.” BMC Bioinformatics 9 (1):307.

Strobl, Carolin, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. 2007. “Bias in Random Forest Variable Importance Measures: Illustrations, Sources and a Solution.” BMC Bioinformatics 8 (1): 25.

Štrumbelj, Erik, and Igor Kononenko. 2014. “Explaining Prediction Models and Individual Predictions with Feature Contributions.” Knowledge and Information Systems 41 (3):647–65.

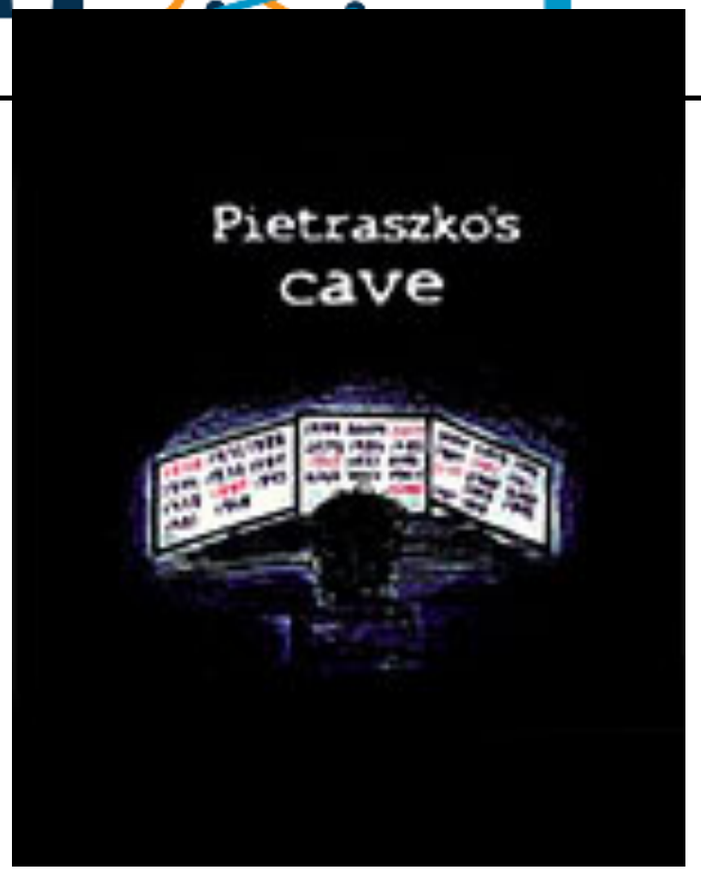
Tatarynowicz, Magda, Kamil Romaszko, and Mateusz Urbański. 2018. ModelDown: Make Static Html Website for Predictive Models. https://github.com/MI2DataLab/


W POGONI ZA
NIESKOŃCZONOŚCIĄ


Adventures of Beta and Bit
How to weigh a dog with a ruler?
Beta, who has a passion for mathematics, chess and good books, changes into SuperBeta under the influence of puzzles.
Text: Przemysław Biecek
Illustrations: Klaudia Kornijuk
Superglasses: +10 to the speed of browsing the Internet.
Calculator: -10 to the time of calculations.
Beta's Superpower is Data Analysis.
Bit's Superpower is searching for the data.
Bit, who is a computer, programming as well as robot maniac, changes into SuperBit under the influence of puzzles.
The early spring: Beta and Bit are wandering through the park.
This publication is co-financed by the means granted by Fundation for the Polish Science in the eNgage program as part of the SKILLS project co-financed by The European Social Fund (contract nr 71/UDWSKILLS/2018).


Jak długo żyją Muffinki?
Beta bit


Wykresy unplugged


Pietraszko's cave