Explaining single prediction

Mateusz Staniak

Why?

- to increase *trust* in important predictions
- to *understand* which factors drive the prediction
- to inspect unusual/outlier predictions
- to improve the model

How?

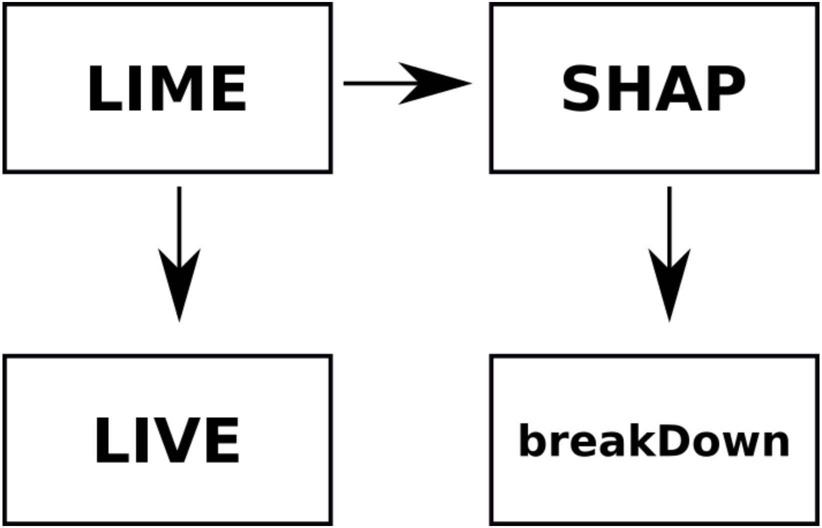


Figure 1:

Now details!

LIME

LIME: general idea

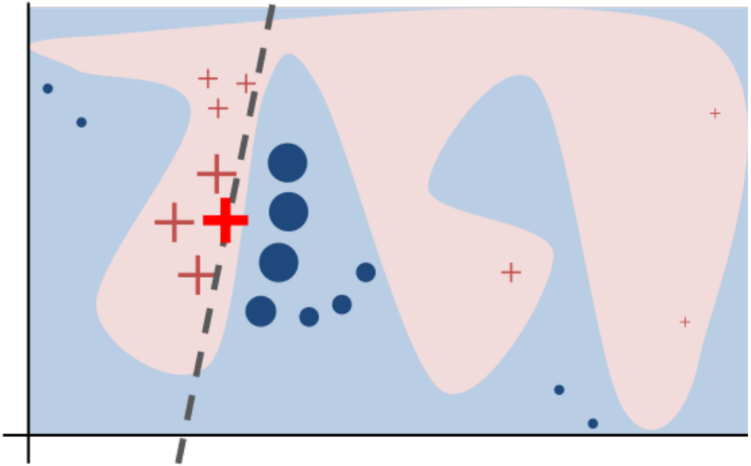


Figure 2:

LIME: how it's done

- Gaussian sampling for tabular, uniform sampling from interpretable inputs for image/text.
- Scores for new observation are weighted by the distance from original observation.
- Variable selection is usually based on ridge/lasso regression.
- Weights are assigned to interpretable inputs to decide if they *vote* for or against a given label.

LIME: summary

`LIME` approximates the complex model with a simple model and discovers which features in *interpretable representation* of data drive the prediction.

Pros:

- intuitive & suitable for different types of data
- easy to understand & interpret

Cons:

- results can be inconsistent (see: Lundberg)
- depends on many *hyperparameters* (kernel, size, . . .)

Shapley Values

Shapley values: general idea

- The goal is a decomposition of prediction into sum of scores related to (simplified) features.
- The problem is solved using game theory: *Shapley values*.
- This approach unifies several methods (including `LIME`).

Shapley values: how it's done

- Different approximate algorithms were proposed.
- Exact methods exist for linear models and tree ensemble methods.
- Classic way: sample permutations of variables, then average contributions.
- Better way: approximation based on LIME and Shapley values for regression.

Shapley values: summary

`Shapley values` decomposition prediction into feature contributions in a rigorous way rooted in game theory.

Pros:

- good theoretical properties
- comes with good visual diagnostic tools

Cons:

- won't produce sparse explanations
- assumes *additivity*, uses simplified inputs (exception: trees)
- computational issues

LIVE

LIVE: general idea

- Modification of `LIME` for tabular data and regression problems.
- Similar observation for *fake* dataset are sampled from empirical distributions.
- Variable selection is possible.
- Focused on model visualization.

LIVE: how it's done

- All new observations are treated as similar (identity kernel & equal weights) (Because of the use of empirical distributions)
- Variables are selected by LASSO (`glmnet` package).
- All models from `mlr` are supported both as black box and white box.
- Model visualization for linear models in particular.

LIVE: summary

`live` fits a simple model to approximate the black box locally and allows model visualization for this model.

Pros:

- flexible & focused on model visualization
- local exploration in data space (no binary inputs)

Cons:

- comes with no theoretical guarantees (though *works* at least in simple cases and is pretty stable)
- inherits strengts/weaknesses of white box model

breakDown

breakDown: general idea

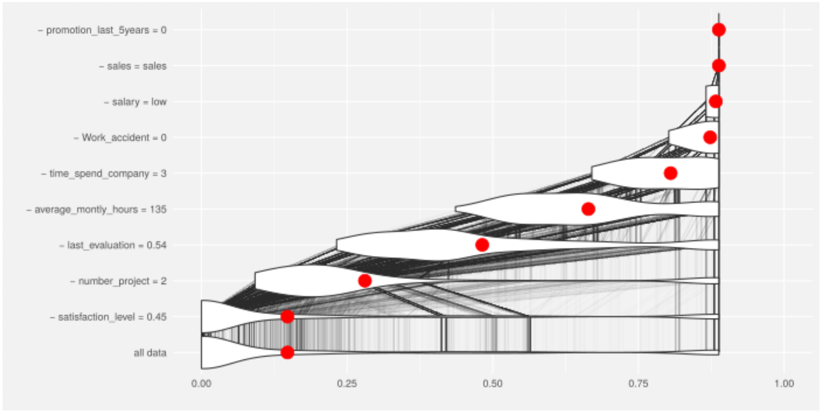


Figure 3:

breakDown: how it's done

- Contributions are assigned in a greedy way.
- Waterfall plots are important tool as a visual representation.

breakDown: summary

`breakDown` decompositions the prediction into feature contributions and visualizes them.

Pros:

- simple computation
- easy to interpret (in particular compared to SHAP)

Cons:

- limited to additive effects
- again, no theoretical guarantees (though can be thought of as a rough Shapley values estimate)

Let's see some examples!

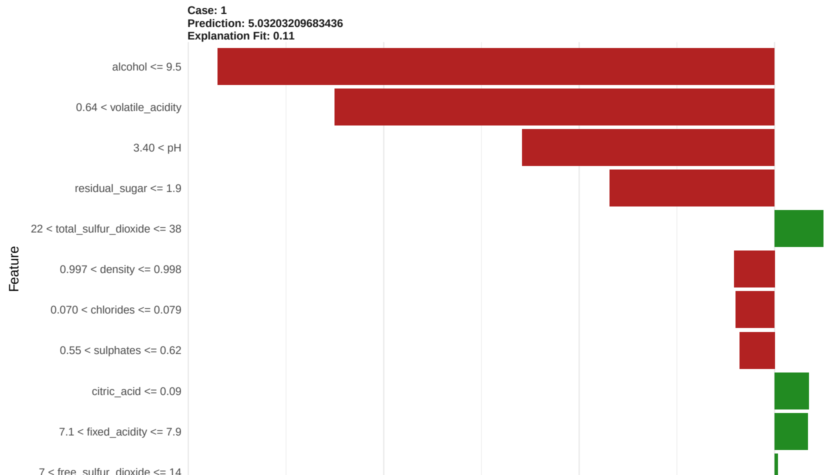
Dataset

```r
library(DALEX)
library(live)
library(lime)
library(shapleyr)
library(mlr)
library(tidyverse)
library(e1071)
load("wine.rda")
glimpse(wine)
```

```
## Observations: 1,599
## Variables: 12
## $ fixed_acidity         <dbl> 7.4, 7.8, 7.8, 11.2, 7.4, 7
## $ volatile_acidity      <dbl> 0.700, 0.880, 0.760, 0.280
## $ citric_acid           <dbl> 0.00, 0.00, 0.04, 0.56, 0.
## $ residual_sugar        <dbl> 1.9, 2.6, 2.3, 1.9, 1.9, 1
## $ chlorides             <dbl> 0.076, 0.098, 0.092, 0.075
## $ free_sulfur_dioxide   <dbl> 11, 25, 15, 17, 11, 13, 15
```

LIME

```r
lime_obj <- lime(wine, wine_svm)
model_type.svm <- function(x, ...) "regression"
lime_expl <- lime::explain(wine[nobs, ], lime_obj, n_featu
plot_features(lime_expl)
```



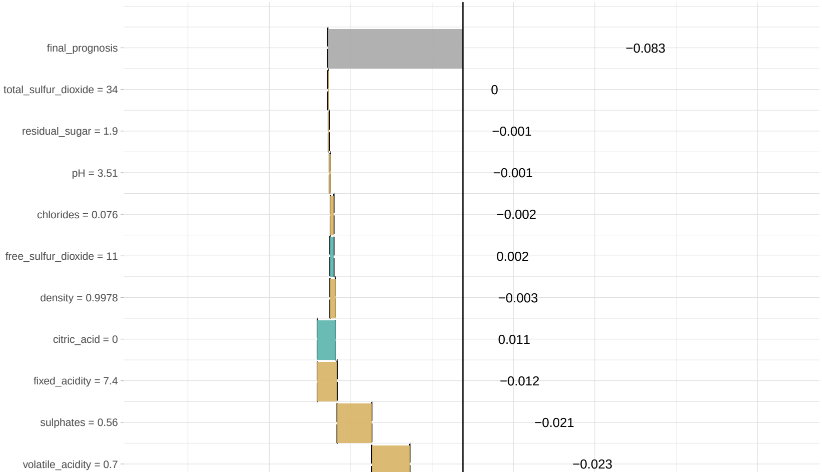
Shapley values

```r
tsk <- makeRegrTask(data = wine, target = "quality")
model <- train("regr.svm", tsk)
shap_expl <- shapley(nobs, model = model, task = tsk)
gather(shap_expl, "variable", "contribution") %>%
  arrange(desc(abs(contribution)))
```

```
##                variable contribution
## 1      volatile_acidity       -0.232
## 2               alcohol       -0.207
## 3             sulphates       -0.199
## 4           citric_acid        0.169
## 5         fixed_acidity       -0.085
## 6                    pH       -0.070
## 7   free_sulfur_dioxide        0.034
## 8        residual_sugar       -0.020
## 9  total_sulfur_dioxide        0.015
## 10            chlorides        0.012
## 11              density        0.001
```

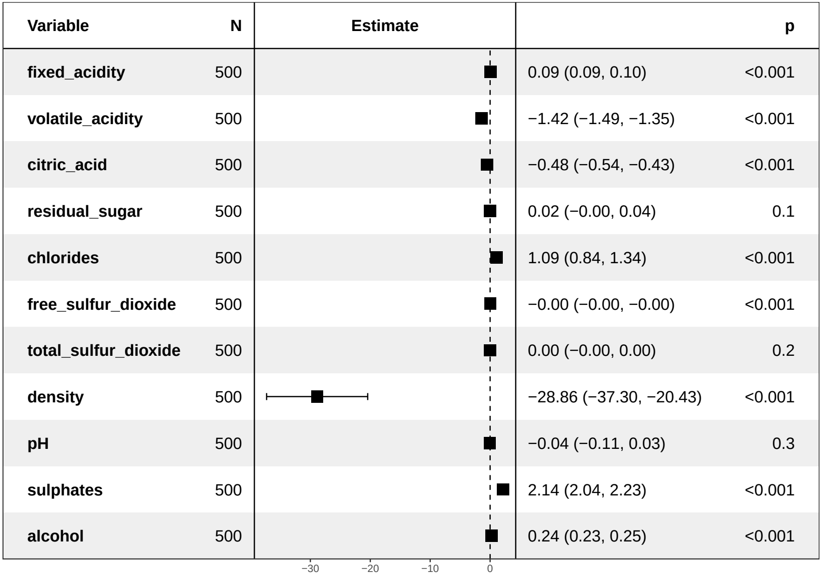
LIVE

```
local <- sample_locally(wine, wine[nobs, ], "quality", 500
local <- live::add_predictions(wine, local, wine_svm)
live_expl <- fit_explanation(local, "regr.lm")
plot_explanation(live_expl, "waterfallplot", wine[nobs, ])
```



LIVE cont.

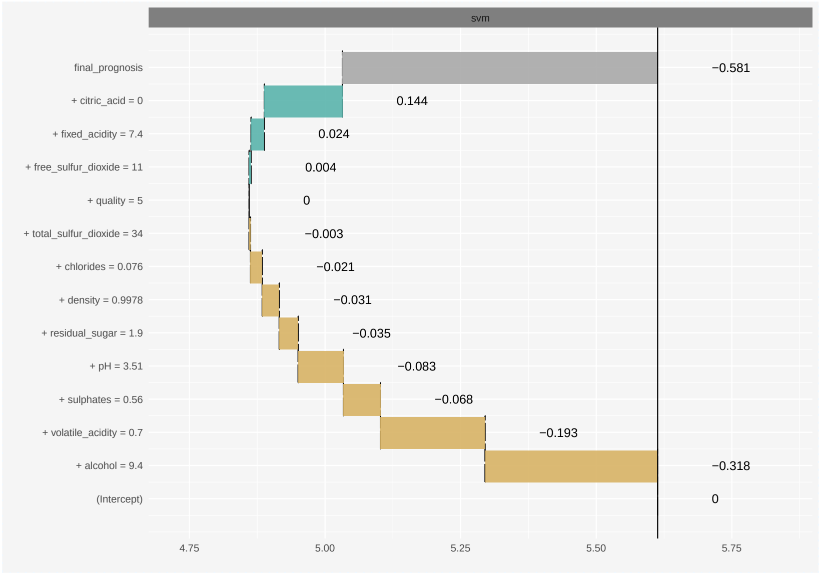
```
plot_explanation(live_expl, "forestplot")
```



Variable	N	Estimate		p
fixed_acidity	500		0.09 (0.09, 0.10)	<0.001
volatile_acidity	500		−1.42 (−1.49, −1.35)	<0.001
citric_acid	500		−0.48 (−0.54, −0.43)	<0.001
residual_sugar	500		0.02 (−0.00, 0.04)	0.1
chlorides	500		1.09 (0.84, 1.34)	<0.001
free_sulfur_dioxide	500		−0.00 (−0.00, −0.00)	<0.001
total_sulfur_dioxide	500		0.00 (−0.00, 0.00)	0.2
density	500		−28.86 (−37.30, −20.43)	<0.001
pH	500		−0.04 (−0.11, 0.03)	0.3
sulphates	500		2.14 (2.04, 2.23)	<0.001
alcohol	500		0.24 (0.23, 0.25)	<0.001

breakDown

```
wine_explainer <- DALEX::explain(wine_svm, data = wine, y =
wine_expl <- single_prediction(wine_explainer, wine[nobs, ]
plot(wine_expl)
```


svm
final_prognosis
+ citric_acid = 0
+ fixed_acidity = 7.4
+ free_sulfur_dioxide = 11
+ quality = 5
+ total_sulfur_dioxide = 34
+ chlorides = 0.076
+ density = 0.9978
+ residual_sugar = 1.9
+ pH = 3.51
+ sulphates = 0.56
+ volatile_acidity = 0.7
+ alcohol = 9.4
(Intercept)
−0.581
0.144
0.024
0.004
0
−0.003
−0.021
−0.031
−0.035
−0.083
−0.068
−0.193
−0.318
0
4.75
5.00
5.25
5.50
5.75

Time for discussion!