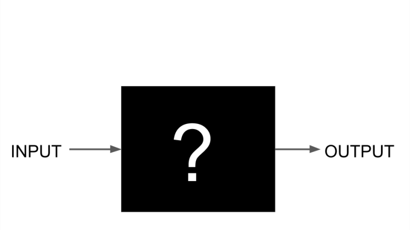
iml - Model Agnostic Interpretable ML [bit.ly/user-iml]

Christoph Molnar, Bernd Bischl

2018-07-12

IML Methods

INTERPRETABLE MACHINE LEARNING



- Machine learning (ML) has huge potential to improve research, products and processes
- ML models usually operate as intransparent black boxes
- The lack of explanation hurts trust and creates barrier for adoption

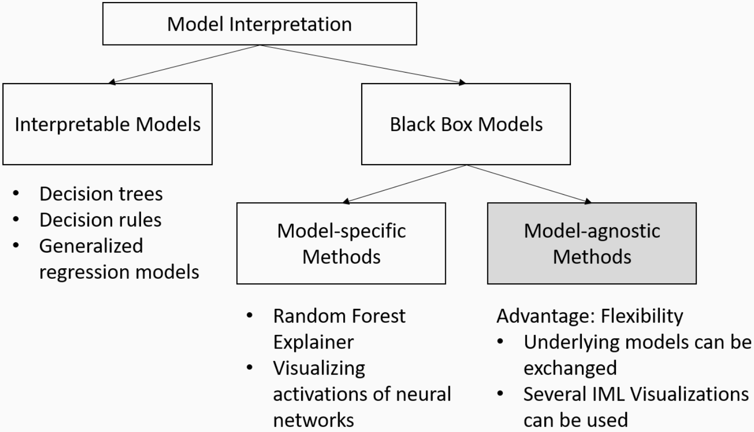
$\Rightarrow$ We need interpretability for machine learning models

WHEN DO WE NEED INTERPRETABILITY?

- Debugging the models
- Increasing trust
- Newly developed systems with unknown consequences
- Decisions about humans
- Critical applications that decide about life and death
- Models using proxies instead of causal inputs
- When the loss function does not cover all constraints

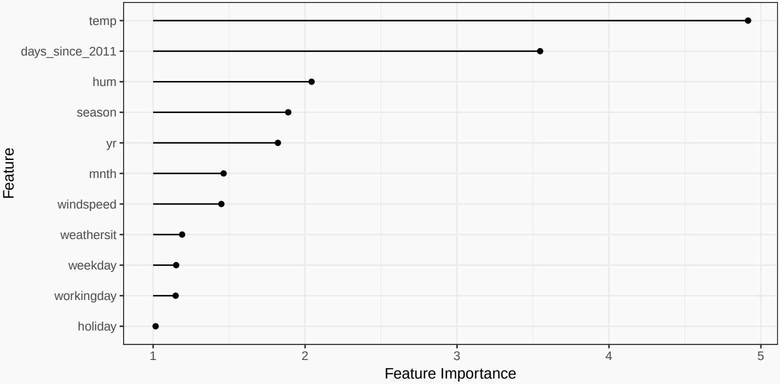
Doshi-Velez, F., and Kim, B. (2017). Towards A Rigorous Science of Interpretable Machine Learning, (Ml), 1-13. Retrieved from http://arxiv.org/abs/1702.08608

WHAT TOOLS DO WE HAVE?



PERMUTATION FEATURE IMPORTANCE

- Calculates the increase of the model's prediction error after permuting the feature
- Features are important if permuting one feature's value increases the model error



Fisher, A., Rudin, C., and Dominici, F. (2018). Model Class Reliance.

PERMUTATION FEATURE IMPORTANCE

1. Estimate model error on test data
2. For each feature x_j

- Shuffle the feature

original

x_1	$\dots$	x_j	$\dots$	x_p
3		1.4		6.0
5		1.2		7.2
$\dots$		$\dots$		$\dots$
6		2.0		8.9

$\Rightarrow$

shuffled x_j

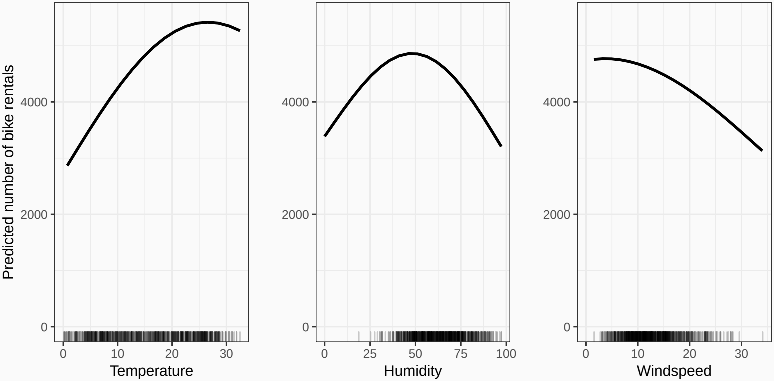
x_1	$\dots$	x_j	$\dots$	x_p
3		2.0		6.0
5		1.4		7.2
$\dots$		$\dots$		$\dots$
6		1.2		8.9

- Estimate the error of the model after shuffling
- Calculate importance as increase in error
- Average the feature importance over multiple repetitions

PARTIAL DEPENDENCE PLOTS

Show the marginal effect of a feature on the predicted outcome of a fitted model

$$f_{x_S}(x_S) = \mathbb{E}_{x_C} f(x_S, x_C)$$



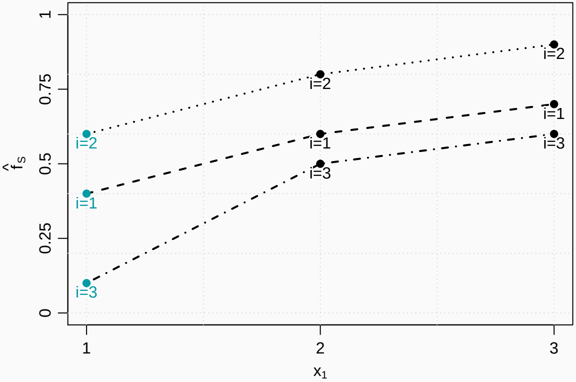
Friedman, J.H. 2001. "Greedy Function Approximation: A Gradient Boosting Machine." Annals of Statistics 29: 1189-1232.

PARTIAL DEPENDENCE PLOTS

- Select a feature x_j
- Choose grid points along x_j
- For each grid point:
  - Overwrite feature x_j in the dataset with the current grid value
  - Get the predictions for these points from the ML model
  - Average the predictions
- Draw a curve with the grid points on the x-axis and the average prediction on the y-axis.

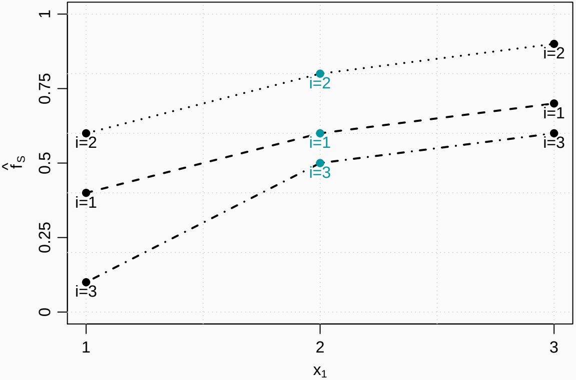
PARTIAL DEPENDENCE PLOTS

i	x_1	x_2	x_3	$\hat{f}^{(i)}_{\S}$
1	**1**	2	3	0.4
2	**1**	4	5	0.6
3	**1**	6	7	0.1
1	2	2	3	0.6
2	2	4	5	0.8
3	2	6	7	0.5
1	3	2	3	0.7
2	3	4	5	0.9
3	3	6	7	0.6



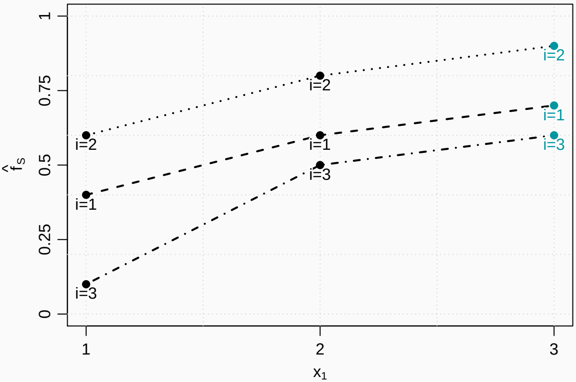
PARTIAL DEPENDENCE PLOTS

i	x_1	x_2	x_3	$\hat{f}_{\S}^{(i)}$
1	1	2	3	0.4
2	1	4	5	0.6
3	1	6	7	0.1
1	**2**	2	3	0.6
2	**2**	4	5	0.8
3	**2**	6	7	0.5
1	3	2	3	0.7
2	3	4	5	0.9
3	3	6	7	0.6



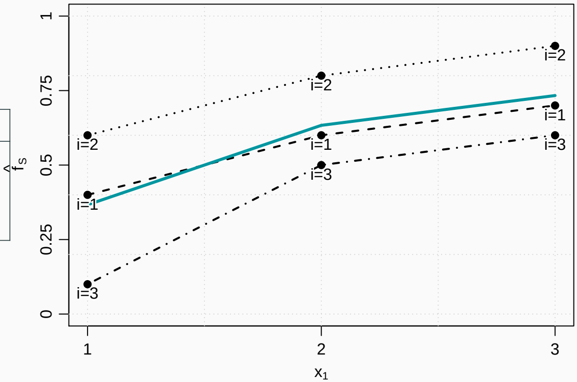
PARTIAL DEPENDENCE PLOTS

i	x_1	x_2	x_3	$\hat{f}_{\S}^{(i)}$
1	1	2	3	0.4
2	1	4	5	0.6
3	1	6	7	0.1
1	2	2	3	0.6
2	2	4	5	0.8
3	2	6	7	0.5
1	**3**	2	3	0.7
2	**3**	4	5	0.9
3	**3**	6	7	0.6



PARTIAL DEPENDENCE PLOTS

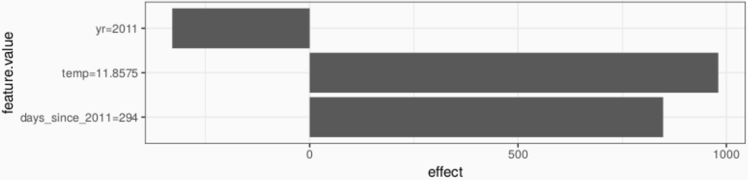
x_1	x_2	x_3	$\hat{f}_S$
1	2	3	$\frac{1}{3}\sum_{i=1}^{3}\hat{f}_S^{(i)}(1)$
2	4	5	$\frac{1}{3}\sum_{i=1}^{3}\hat{f}_S^{(i)}(2)$
3	6	7	$\frac{1}{3}\sum_{i=1}^{3}\hat{f}_S^{(i)}(3)$



LIME

Local Interpretable model-agnostic Explanations

- Fits local, interpretable models that can explain single predictions of any black-box model
- Local surrogate models, that are interpretable like a LM or CART and are learned on predictions of original model

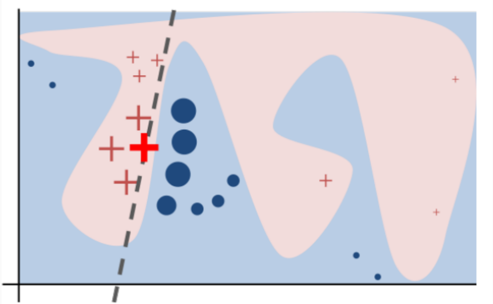


Ribeiro, M. T., (2016, August). Why should i trust you?: Explaining the predictions of any classifier

LIME

How to fit local surrogate model

1. Choose instance of interest x
2. Perturb data and get black box predictions for them
3. Weight new samples by their proximity to x
4. Fit a weighted, interpretable model on this new data set



Ribeiro, M. T., (2016, August). Why should i trust you?: Explaining the predictions of any classifier

iml Package

THE IML PACKAGE

- R6 package for **model-agnostic** Interpretable Machine Learning methods
- Analyses a fixed machine learning model
- Model can be from mlr, caret, or anything else (for the latter you might have to write a few line of glue code)
- Available on CRAN and Github: https://github.com/christophM/iml
- Detailed explanations for the methods can be found in the book "Interpretable Machine Learning": https://christophm.github.io/interpretable-ml-book/agnostic.html

Molnar et al., (2018). iml: An R package for Interpretable Machine Learning . Journal of Open Source Software, 3(26), 786, https://doi.org/10.21105/joss.00786

PACKAGE `IML`

The `iml` package contains the following IML tools

- Permutation Feature Importance (`FeatureImp`)
- Feature Interactions (`Interaction`)
- Partial Dependence Plots (`Partial`)
- LIME (`LocalModel`)
- Shapley Values (`Shapley`)
- Tree Surrogates (`TreeSurrogate`)

EXAMPLE

- Load neccessary packages

```r
library(mlr)
library(iml)
```

- Import data:

```r
load("bike.RData")
```

UCI BIKE SHARING DATA SET

Hourly and daily count of rental bikes between years 2011 and 2012 in Capital bikeshare system with the corresponding weather and seasonal information.

https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset

(Excluded year and day info)

```
##     season         mnth           holiday    weekday           workingday
##  SPRING:181   JAN    : 62   NO HOLIDAY:710   SUN:105   NO WORKING DAY:231
##  SUMMER:184   MAR    : 62   HOLIDAY   : 21   MON:105   WORKING DAY   :500
##  FALL  :188   MAY    : 62                    TUE:104
##  WINTER:178   JUL    : 62                    WED:104
##               AUG    : 62                    THU:104
##               OKT    : 62                    FRI:104
##               (Other):359                    SAT:105
##           weathersit       temp          hum         windspeed         cnt
##  GOOD           :463   Min.   :-5.2   Min.   : 0.0   Min.   : 1.5   Min.   :  22
##  MISTY          :247   1st Qu.: 7.8   1st Qu.:52.0   1st Qu.: 9.0   1st Qu.:3152
##  RAIN/SNOW/STORM: 21   Median :15.4   Median :62.7   Median :12.1   Median :4548
##                        Mean   :15.3   Mean   :62.8   Mean   :12.8   Mean   :4504
##                        3rd Qu.:22.8   3rd Qu.:73.0   3rd Qu.:15.6   3rd Qu.:5956
##                        Max.   :32.5   Max.   :97.2   Max.   :34.0   Max.   :8714
##
```

FIT MLR MODEL AND CREATE IML PREDICTOR

- We have to fit a ML model first

```
task = makeRegrTask(data = bike, target = "cnt")
lrn = makeLearner("regr.randomForest")
mod = train(lrn, task)
```

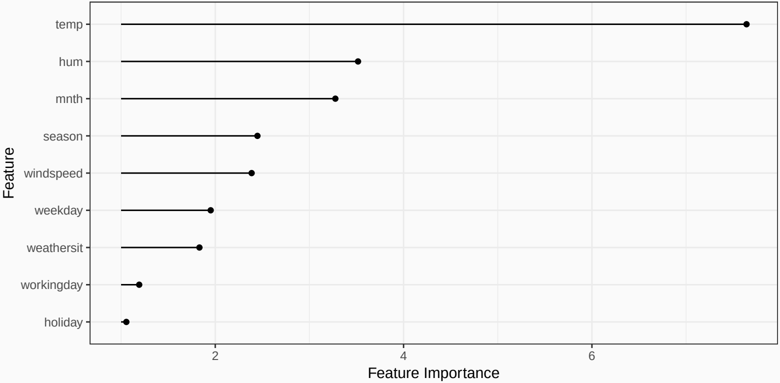
- We can use one IML model for all methods

```
# Create data frame without target column
bike.x = bike[names(bike) != 'cnt']

predictor = Predictor$new(mod, data = bike.x, y = bike$cnt)
```

PERMUTATION FEATURE IMPORTANCE PLOT

```r
importance = FeatureImp$new(predictor, loss = 'mse')
plot(importance)
```



ACCESS RESULTS IN TABLE FORMAT

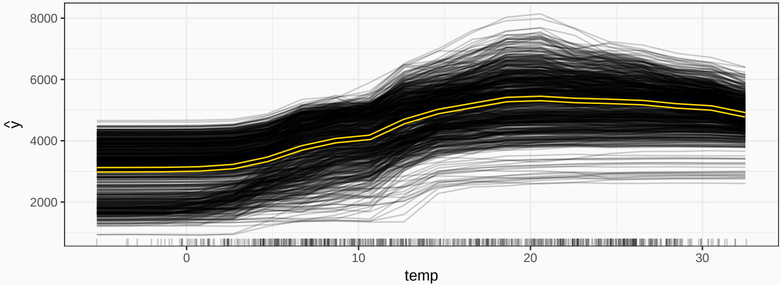
- All results can be viewed in table form

```
importance$results

##      feature original.error permutation.error importance
## 1       temp         336711           2573065       7.64
## 2        hum         336711           1183987       3.52
## 3       mnth         336711           1102907       3.28
## 4     season         336711            824275       2.45
## 5  windspeed         336711            803497       2.39
## 6    weekday         336711            657163       1.95
## 7 weathersit         336711            616884       1.83
## 8 workingday         336711            401488       1.19
## 9    holiday         336711            355549       1.06
```

PARTIAL DEPENDENCE PLOT

```r
pdp = Partial$new(predictor, "temp", ice = TRUE)
pdp$plot()
```

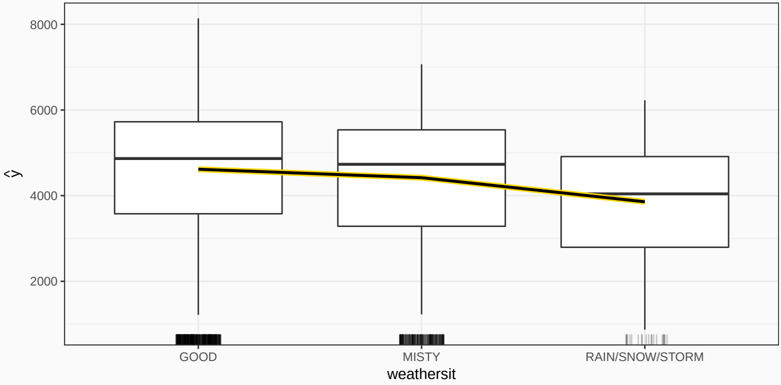


- `ice = TRUE`: Individual Conditional Expectation (ICE) Plots visualizes the relationship between the predicted response and the feature for *individual* observations

REUSE PD OBJECTS

- PD objects can be reused, e.g. for fitting other features

```
pdp$set.feature("weathersit")
pdp$plot()
```



LIME PLOT

- Select one instance

```
id = 726
pred = predictor$predict(bike.x[id,])[1,1]
cbind(bike[id,], yhat = pred)

##      season mnth    holiday weekday  workingday      weathersit
## 726 SPRING  DEZ NO HOLIDAY     WED WORKING DAY RAIN/SNOW/STORM
##      temp  hum windspeed cnt yhat
## 726 3.44 82.3      21.2 441  955
```

```
lim = LocalModel$new(predictor, x.interest = bike.x[id,], k = 3)
plot(lim)
```