Model Explainability

Presenter: Kshitij Srivastava

September 23, 2020

SOCIETY OF ACTUARIES®

21

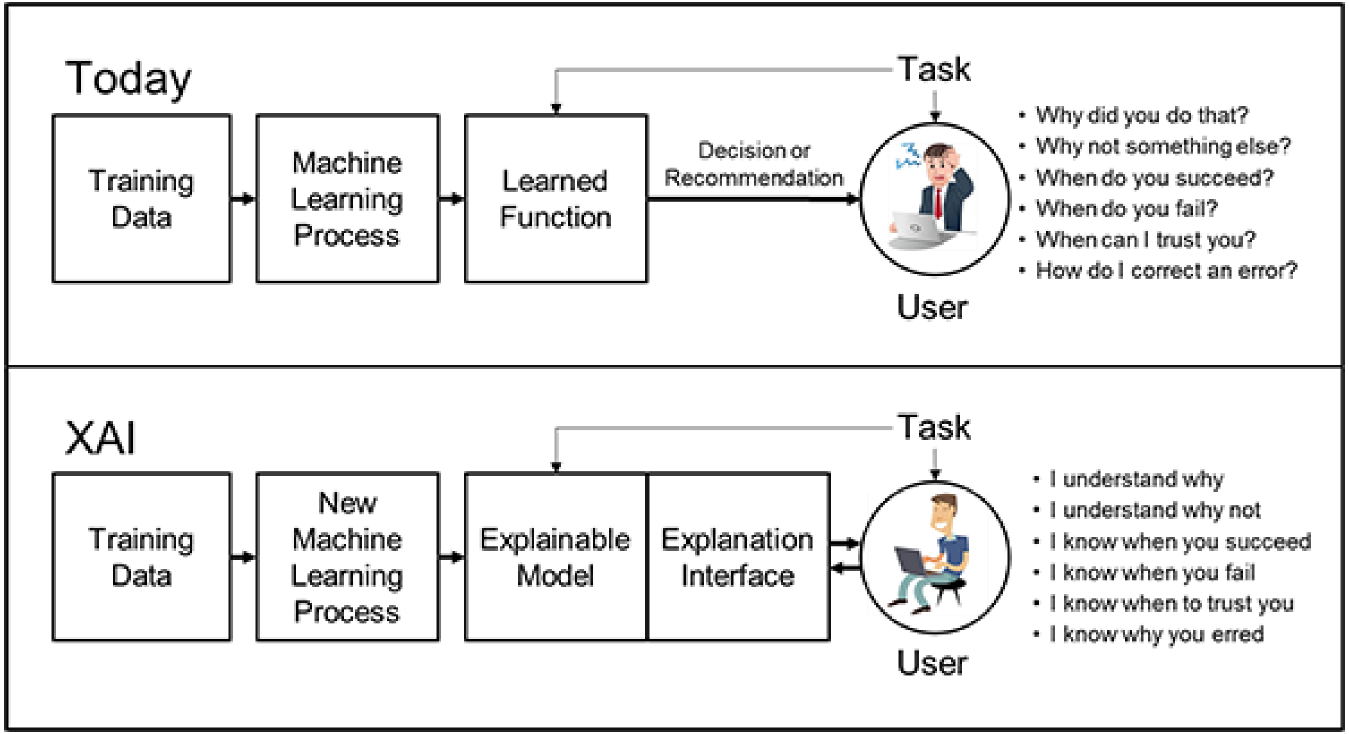
Amazon reportedly scraps internal AI recruiting tool that was biased against women

The secret program penalized applications that contained the word "women's"

Microsoft's news AI publishes stories about its own racist failures

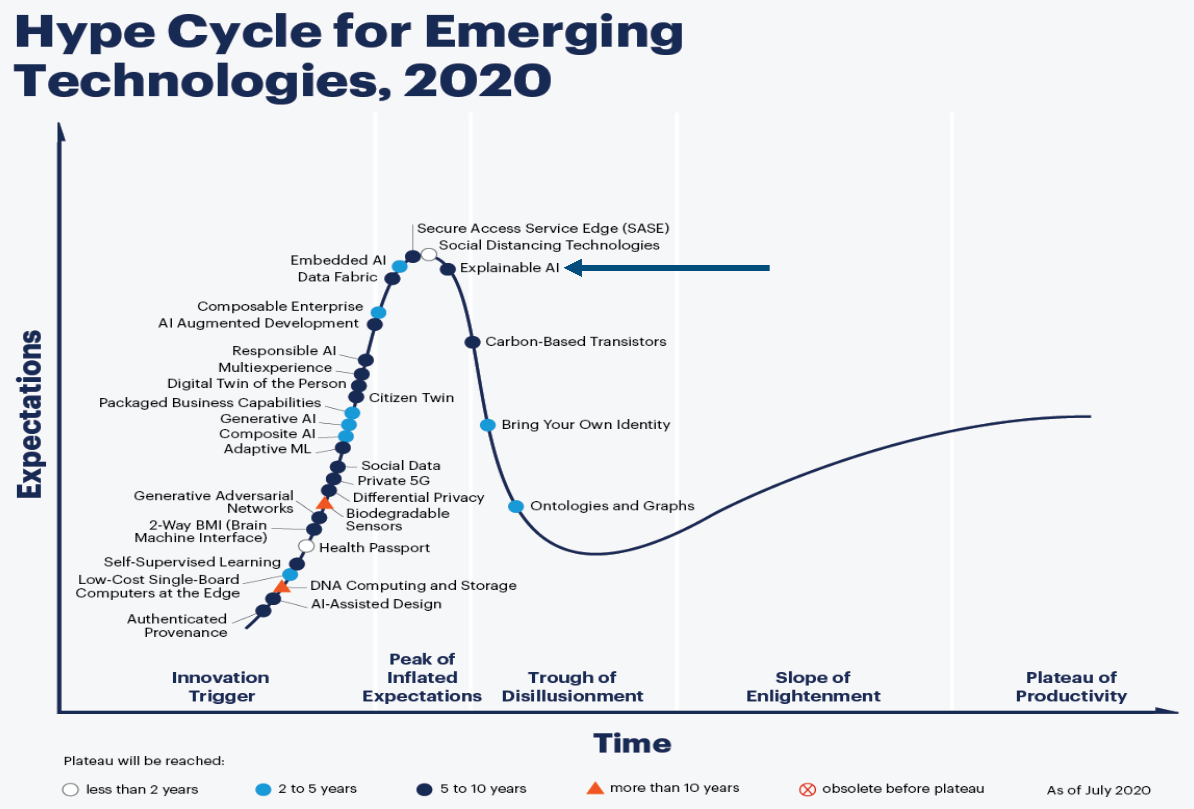
An Artificial Intelligence (AI) tool developed by Google failed during real-world testing. It was supposed to detect signs of blindness.

Need for Explainable ML

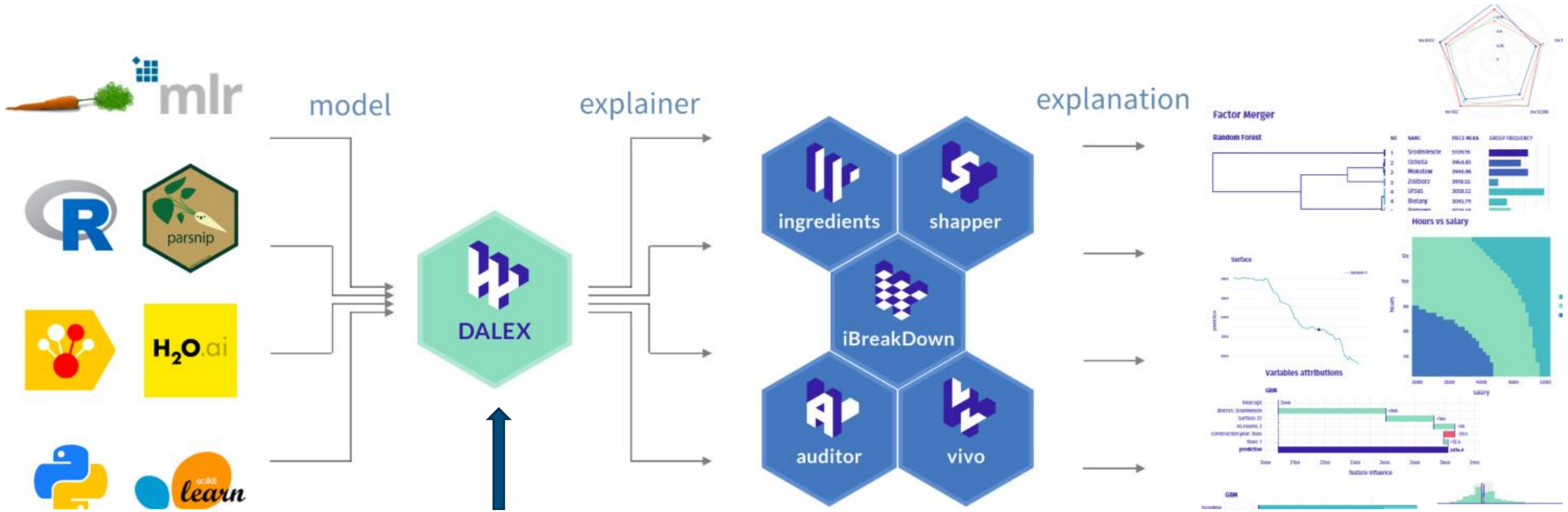


Picture Credits: https://www.darpa.mil/program/explainable-artificial-intelligence

Gartner Hype Cycle 2020



Package ecosystem for Explainable ML



DALEX wraps models created by different factories into a uniform structure that can be then used by model explainers

Credits:
https://github.com/ModelOriented/DrWhy/blob/master/README.md

Explaining ML predictions with Titanic dataset

```
## 2. Load the dataset

```{r}
head(titanic_imputed)
```
```

	gender <fctr>	age <dbl>	class <fctr>	embarked <fctr>	fare <dbl>	sibsp <dbl>	parch <dbl>	survived <dbl>
1	male	42	3rd	Southampton	7.11	0	0	0
2	male	13	3rd	Southampton	20.05	0	2	0
3	male	16	3rd	Southampton	20.05	1	1	0
4	female	39	3rd	Southampton	20.05	1	1	1
5	female	16	3rd	Southampton	7.13	0	0	1
6	male	25	3rd	Southampton	7.13	0	0	1

6 rows

- gender a factor with levels `male` and `female`.
- age a numeric value with the persons age on the day of the sinking.
- class a factor specifying the class for passengers or the type of service aboard for crew members.
- embarked a factor with the persons place of of embarkment (Belfast/Cherbourg/Queenstown/Southampton).
- country a factor with the persons home country.
- fare a numeric value with the ticket price (`0` for crew members, musicians and employees of the shipyard company).
- sibsp an ordered factor specifying the number if siblings/spouses aboard;
- parch an ordered factor specifying the number of parents/children aboard;
- survived a factor with two levels (`no` and `yes`) specifying whether the person has survived the sinking.

Fit RF

```
## 3. Fit a random forest and a lr with splines

```{r}
#fits a simple random forest with default hp
model_ranger <- ranger(survived ~ ., data = titanic_imputed,
                       classification = TRUE, probability = TRUE)

# using restricted cubic splines. Frank Harrell, the creator of rms package notes here:
https://stats.stackexchange.com/questions/328545/reporting-the-effect-of-a-predictor-in-a-logistic-regress
ion-fitted-with-a-restr that the coefficients of a rcs shouldn't be interpreted like usual lr. Instead a
partial effect plot or a nomogram can be used.
model_rms <- lrm(survived ~ rcs(age)*gender + rcs(fare) +
                 class, data = titanic_imputed)
```
```

Fast implementation of Random Forests: https://cran.r-project.org/web/packages/ranger/ranger.pdf

Cubic Splines visualization: https://pclambert.net/interactivegraphs/spline_eg/spline_eg

Create a model explainer object

```
## 4. Create an explainer for random forest
```

```{r}
exp_ranger <- explain(model_ranger,
                      data = titanic_imputed[,1:7],
                      y = titanic_imputed$survived)

predict(exp_ranger, titanic_imputed[1,])
```

```
Preparation of a new explainer is initiated
  -> model label       :  ranger  (  default  )
  -> data              :  2207  rows  7  cols
  -> target variable   :  2207  values
  -> predict function  :  yhat.ranger  will be used (  default  )
  -> predicted values  :  numerical, min =  0.01304684 , mean =  0.3220136 , max =  0.9884273
  -> model_info        :  package ranger , ver. 0.12.1 , task classification (  default  )
  -> residual function :  difference between y and yhat (  default  )
  -> residuals         :  numerical, min =  -0.7811852 , mean =  0.0001431836 , max =  0.8837261
  A new explainer has been created!
        1
0.1037679
```

```
## 5. Create an explainer for lr with splines
```

```{r}
exp_rms <- explain(model_rms,
                   data = titanic_imputed[,1:7],
                   y = titanic_imputed$survived,
                   predict_function = function(m, x)
                     predict(m, x, type = "fitted"),
                   label = "Logistic with splines")
```

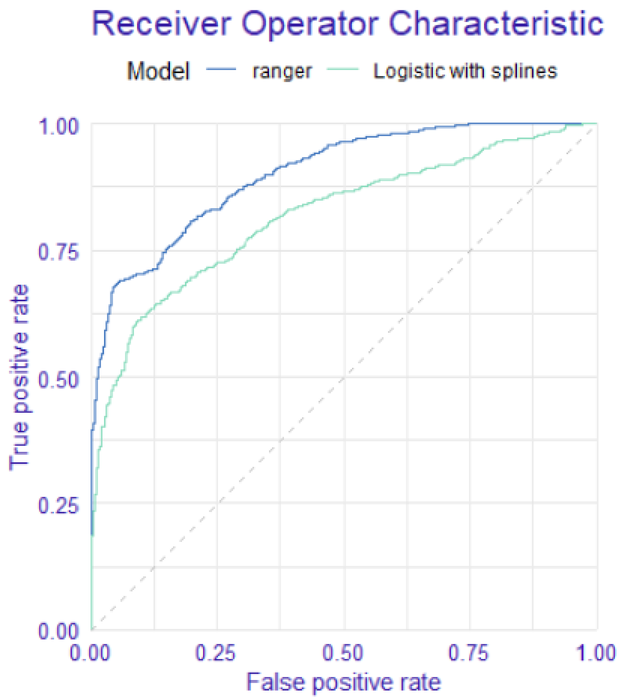
```
Preparation of a new explainer is initiated
  -> model label       :  Logistic with splines
  -> data              :  2207  rows  7  cols
  -> target variable   :  2207  values
  -> predict function  :  function(m, x) predict(m, x, type = "fitted")
  -> predicted values  :  numerical, min =  0.01182128 , mean =  0.3221568 , max =  0.9589928
  -> model_info        :  package rms , ver. 6.0.1 , task classification (  default  )
  -> residual function :  difference between y and yhat (  default  )
  -> residuals         :  numerical, min =  -0.9508948 , mean =  -2.68076e-09 , max =  0.9733383
  A new explainer has been created!
```

Measures of Performance

```
mp_ranger <- model_performance(exp_ranger)
```

```
## 7. Plot comparison charts
```

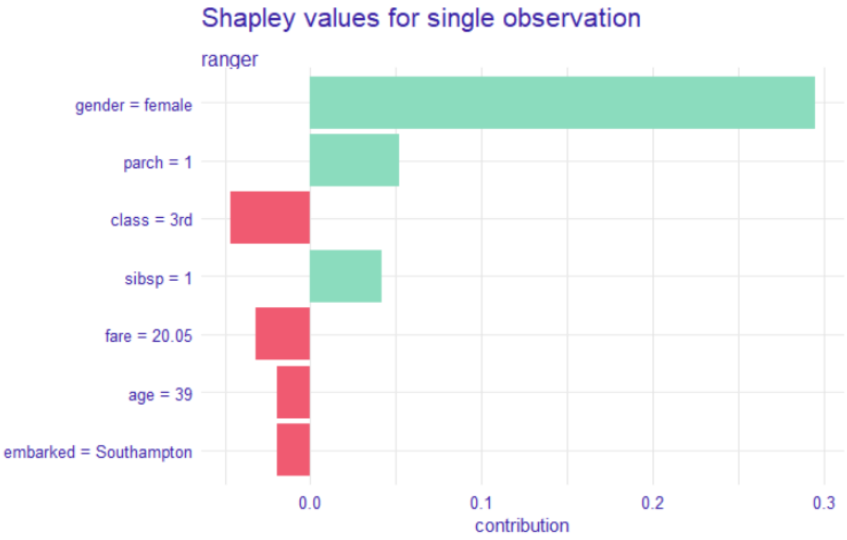
```
```{r}
plot(mp_ranger, mp_rms, geom = "roc")
```

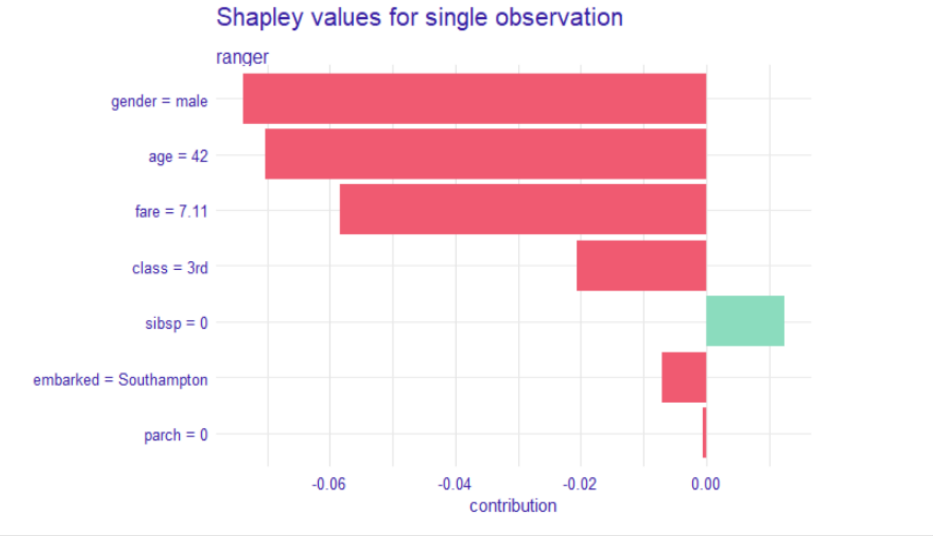

```

Shapley Values

```
# Shapley Values
sh_ranger <- predict_parts(exp_ranger, titanic_imputed[4,], type = "shap", B = 1)

plot(sh_ranger, show_boxplots = FALSE) +
   ggtitle("Shapley values for single observation","")
```

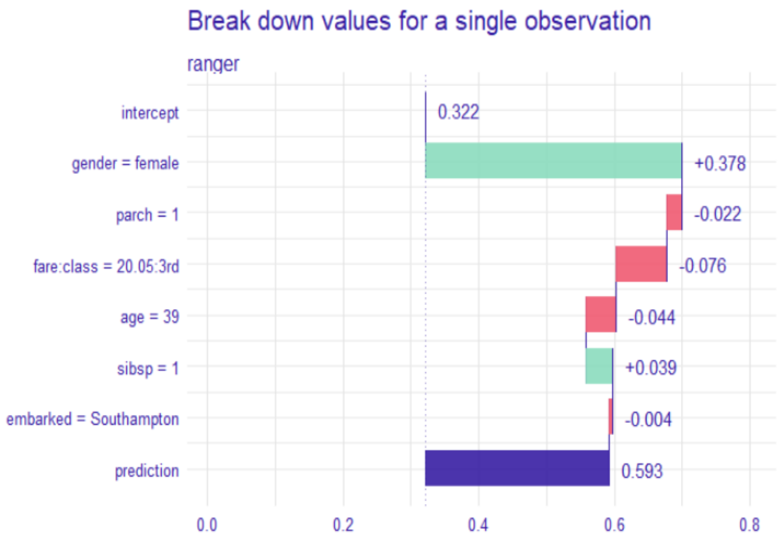


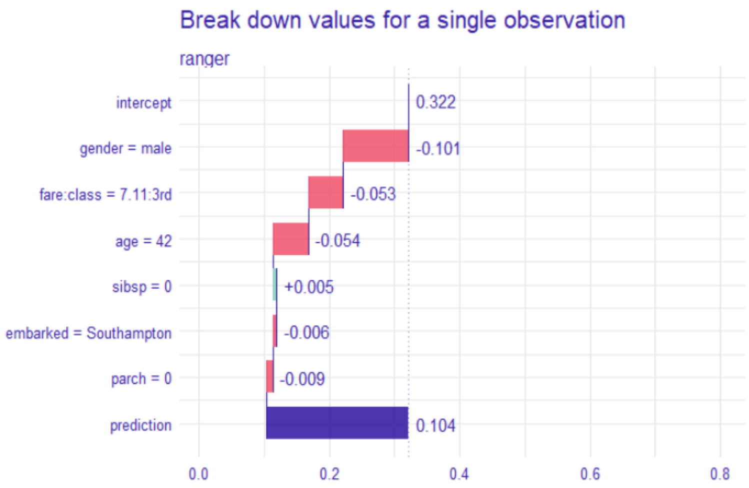


https://towardsdatascience.com/shap-explained-the-way-i-wish-someone-explained-it-to-me-ab81cc69ef30

Break down Values

```
bd_ranger <- predict_parts(exp_ranger, titanic_imputed[4,], type = "break_down_interactions")
bd_ranger
plot(bd_ranger, show_boxplots = FALSE) +
  ggtitle("Break down values for a single observation","") +
  scale_y_continuous("",limits = c(0.01,0.8))
```



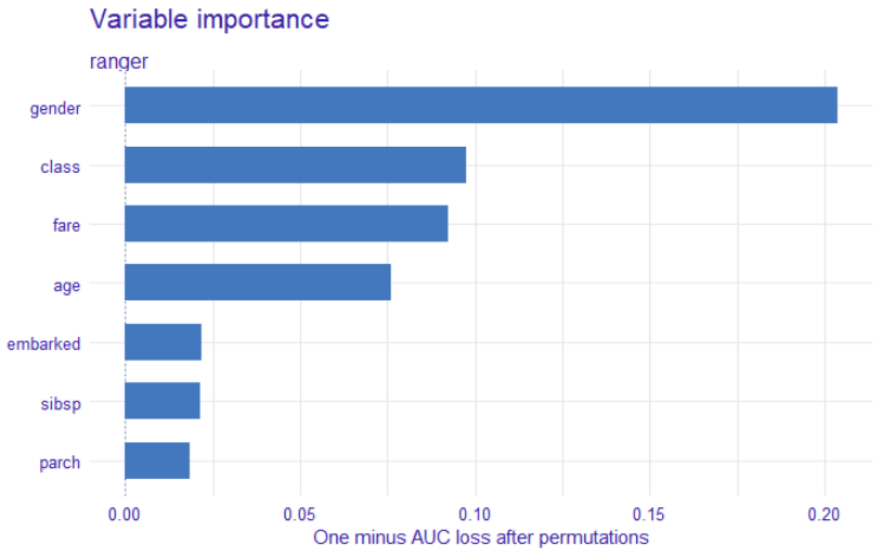


More about breakdown methodology: https://arxiv.org/abs/1804.01955

Variable Importance

```
mp_ranger <- model_parts(exp_ranger, type = "difference")

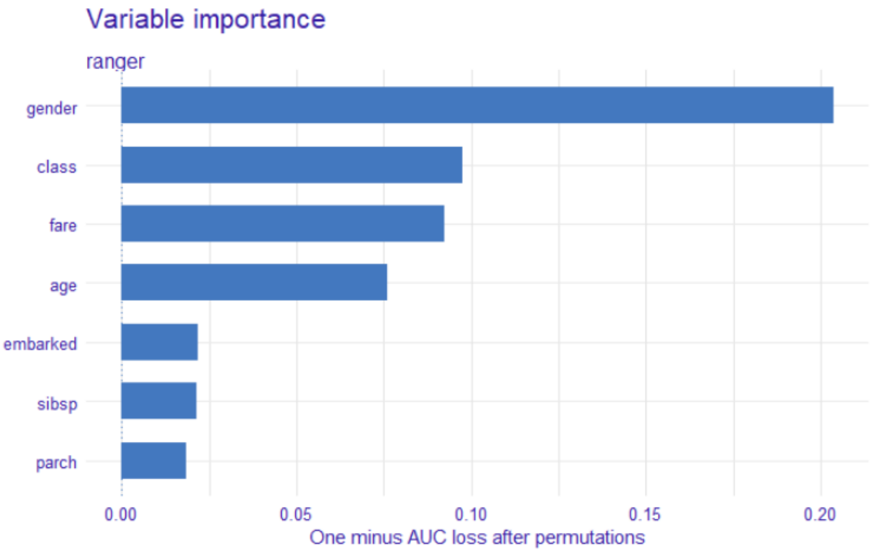
plot(mp_ranger, show_boxplots = FALSE) +
  ggtitle("Variable importance","")
```



Variable Importance

```
mp_ranger <- model_parts(exp_ranger, type = "difference")

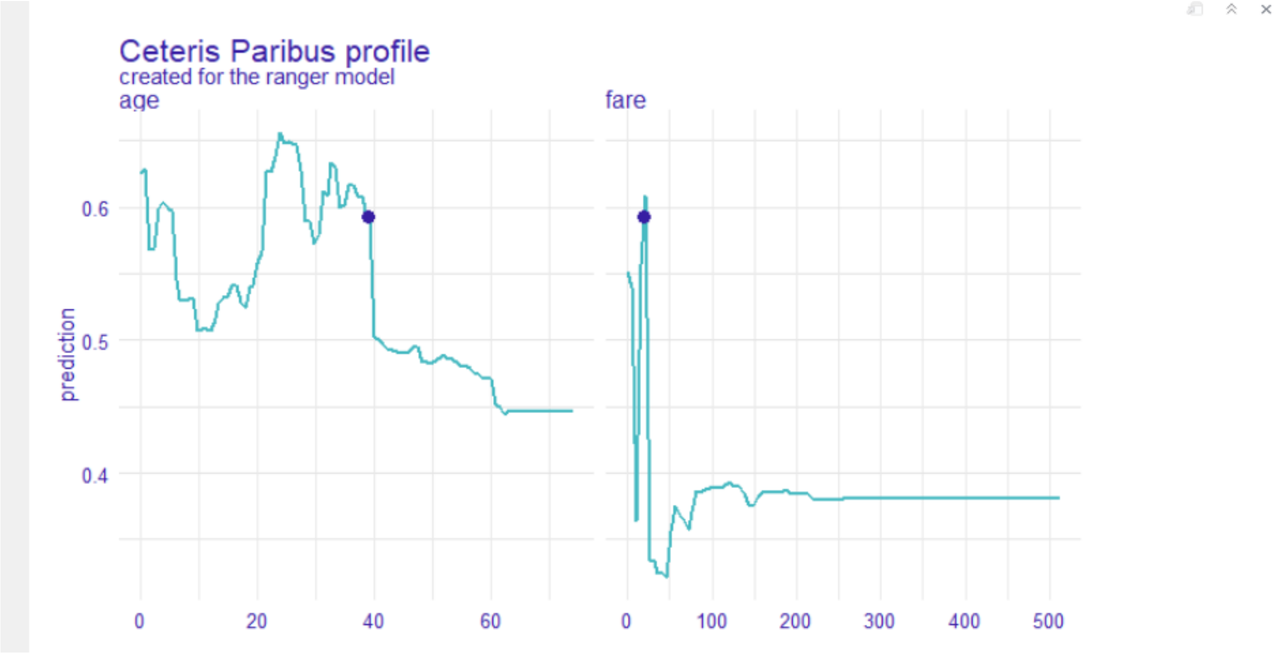
plot(mp_ranger, show_boxplots = FALSE) +
  ggtitle("Variable importance","")
```



Ceteris Paribus profile

```
cp_ranger <- predict_profile(exp_ranger, titanic_imputed[4,])

plot(cp_ranger, variables = c("age", "fare"))
```

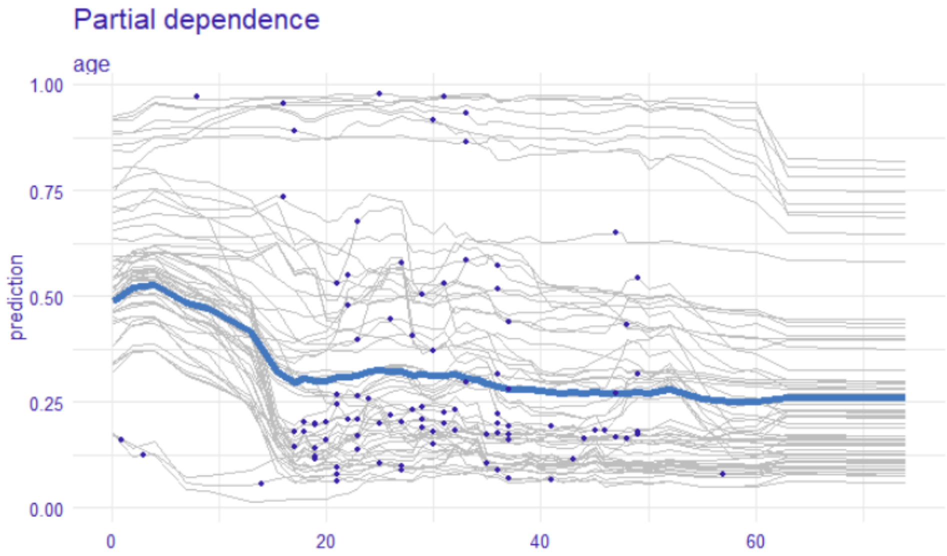


Partial Dependence plot

```
mp_ranger <- model_profile(exp_ranger)

plot(mp_ranger, variables = "age")

plot(mp_ranger, variables = "age", geom = "points") +
  ggtitle("Partial dependence","")
```



More resources

- https://modeloriented.github.io/DALEX/articles/vignette_titanic.html
- https://github.com/ModelOriented/DALEX
- https://github.com/ModelOriented/DrWhy/blob/master/README.md
- https://www.darpa.mil/program/explainable-artificial-intelligence

- Questions?