Uczenie maszynowe w R.

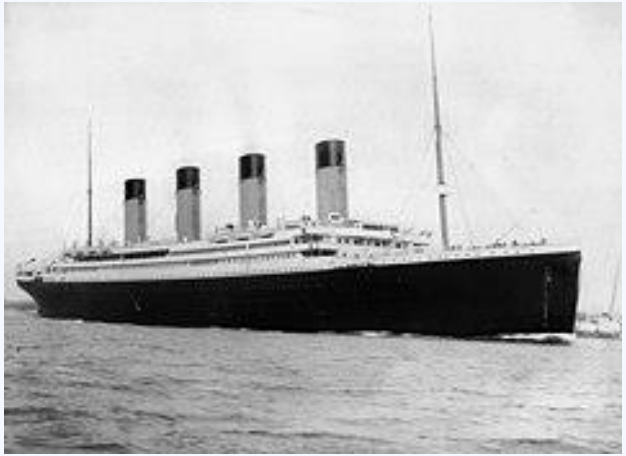
Zaglądamy do czarnej skrzynki

Bałtycki Festiwal Nauki

Uczenie maszynowe

Dziedzina sztucznej inteligencji zajmująca się tworzeniem algorytmów, które potrafią uczyć się ze zbioru danych. W przypadku niektórych algorytmów, szczególnie tych bardziej skomplikowanych, trudno jest zrozumieć, jak dokładnie działają, stąd określenie **“czarna skrzynka”.**

RMS Titanic



Był to największy i najbardziej luksusowy statek pasażerski swoich czasów, wyruszył w dziewiczy rejs z **Southampton** do **Nowego Jorku** *10 kwietnia 1912 roku*. Niestety, *14 kwietnia* statek zderzył się z górą lodową na północnym Atlantyku, co spowodowało poważne uszkodzenia i zatonięcie statku kilka godzin później. Tragedia ta pochłonęła życie **ponad 1500 osób** i stała się jednym z najbardziej tragicznych wydarzeń w historii morskiej.

Tabela *titanic* to dane przygotowane przez prof. P. Biecka opisujące pasażerów Titanika.

```
titanic<-archivist::aread("pbiecek/models/27e5c")
```

```
View(titanic)
```

	gender	age	class	embarked	country	fare	sibsp	parch	survived
1	male	42.0000000	3rd	Southampton	United States	7.1100	0	0	no
2	male	13.0000000	3rd	Southampton	United States	20.0500	0	2	no
3	male	16.0000000	3rd	Southampton	United States	20.0500	1	1	no
4	female	39.0000000	3rd	Southampton	England	20.0500	1	1	yes
5	female	16.0000000	3rd	Southampton	Norway	7.1300	0	0	yes
6	male	25.0000000	3rd	Southampton	United States	7.1300	0	0	yes
7	male	30.0000000	2nd	Cherbourg	France	24.0000	1	0	no
8	female	28.0000000	2nd	Cherbourg	France	24.0000	1	0	yes
9	male	27.0000000	3rd	Cherbourg	Lebanon	18.1509	0	0	yes
10	male	20.0000000	3rd	Southampton	Finland	7.1806	0	0	yes
11	male	30.0000000	3rd	Southampton	Sweden	7.0500	0	0	no
12	male	27.0000000	3rd	Southampton	England	8.0100	0	0	no
13	female	40.0000000	3rd	Southampton	Sweden	9.0906	1	0	no
14	male	0.8333333	3rd	Southampton	England	9.0700	0	1	yes
15	female	18.0000000	3rd	Southampton	England	9.0700	0	1	yes

Modele, których użyjemy

Model regresji liniowej

przewidywanie prawdopodobieństwa wystąpienia zdarzenia oparte na zależności między zmienną zależną a jedną lub więcej zmiennymi niezależnymi

Model lasów losowych

tworzy i łączy wiele drzew decyzyjnych, aby zwiększyć dokładność i stabilność przewidywań dla zadań klasyfikacji i regresji

Model *gradient boosting*

stopniowe dodawanie nowych, prostych modeli (często drzew decyzyjnych) do zestawu już istniejących, aby coraz lepiej przewidywać wyniki, poprawiając błędy krok po kroku

Model maszyny wektorów nośnych

znajduje linię lub płaszczyznę, która najlepiej oddziela różne grupy danych, pomagając w podejmowaniu decyzji o ich klasyfikacji

```r
#model regresji logistycznej:
titanic_lmr <- rms::lrm(survived == "yes" ~ gender + rms::rcs(age) + class +
                          sibsp + parch + fare + embarked, titanic)

#model lasów losowych
set.seed(1313)
titanic_rf <- randomForest::randomForest(survived ~ class + gender + age +
                                           sibsp + parch + fare + embarked, data = titanic, na.action=na.roughfix)

#model gradient boosting
set.seed(1313)
titanic_gbm <- gbm::gbm(survived == "yes" ~ class + gender + age +
                          sibsp + parch + fare + embarked, data = titanic,
                        n.trees = 15000, distribution = "bernoulli")

#model maszyny wektorow nosnych:
set.seed(1313)
titanic_svm <- e1071::svm(survived == "yes" ~ class + gender + age +
                            sibsp + parch + fare + embarked, data = titanic,
                          type = "C-classification", probability = TRUE)
```

Zapakowane modeli do podobnych opakowań

```
titanic_lmr_exp <- explain(model = titanic_lmr,
                           data = titanic[, -9],
                           y = titanic$survived == "yes",
                           label = "Logistic Regression",
                           type = "classification")

titanic_rf_exp <- explain(model = titanic_rf,
                          data = titanic[, -9],
                          y = titanic$survived == "yes",
                          label = "Random Forest")

titanic_gbm_exp <- explain(model = titanic_gbm,
                           data = titanic[, -9],
                           y = titanic$survived == "yes",
                           label = "Generalized Boosted Regression")

titanic_svm_exp <- explain(model = titanic_svm,
                           data = titanic[, -9],
                           y = titanic$survived == "yes",
                           label = "Support Vector Machine")
```

Porównywanie siły predykcyjnej modeli

```
model_performance(titanic_lmr_exp)$measures$auc

model_performance(titanic_rf_exp)$measures$auc

model_performance(titanic_gbm_exp)$measures$auc

model_performance(titanic_svm_exp)$measures$auc
```

Model regresji logistycznej:	Model lasów losowych:	Model gradient boosting:	Model maszyny wektorów nośnych:
0,8174447	0,8636533	**0,8666712**	***0,8129198***

Przykładowi pasażerowie

Rose

class: 1st
gender: female
age: 17
sibsp: 0
parch: 1
fare: 850$
embarked: Southampton

Jack

class: 3rd
gender: male
age: 20
sibsp: 0
parch: 0
fare: ok 10$
embarked: Southampton

```r
#przewidywania modeli dla pasazerow

{print(predict(titanic_lmr_exp, jack))
  print(predict(titanic_rf_exp, jack))
  print(predict(titanic_gbm_exp, jack))
  print(predict(titanic_svm_exp, jack))
  }

{print(predict(titanic_lmr_exp, rose))
  print(predict(titanic_rf_exp, rose))
  print(predict(titanic_gbm_exp, rose))
  print(predict(titanic_svm_exp, rose))}
```

Przewidywania modeli

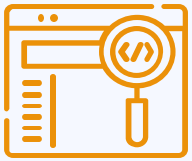
Rose

- Model regresji logistycznej: 98,04%
- Model lasów losowych: 96,8%
- Model gradient boosting: 96,899%
- Model maszyny wektorów nośnych: 43,57%

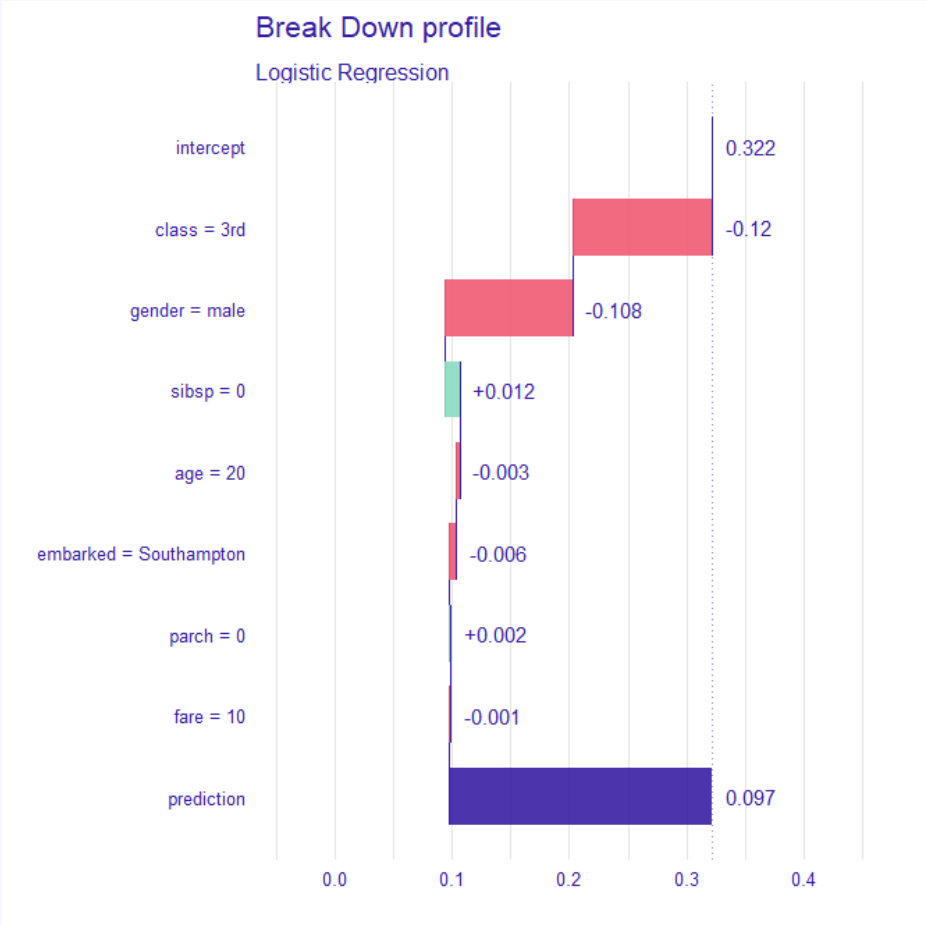
Jack

- Model regresji logistycznej: 9,75%
- Model lasów losowych: 1,2%
- Model gradient boosting: 2,37%
- Model maszyny wektorów nośnych: 1,72%

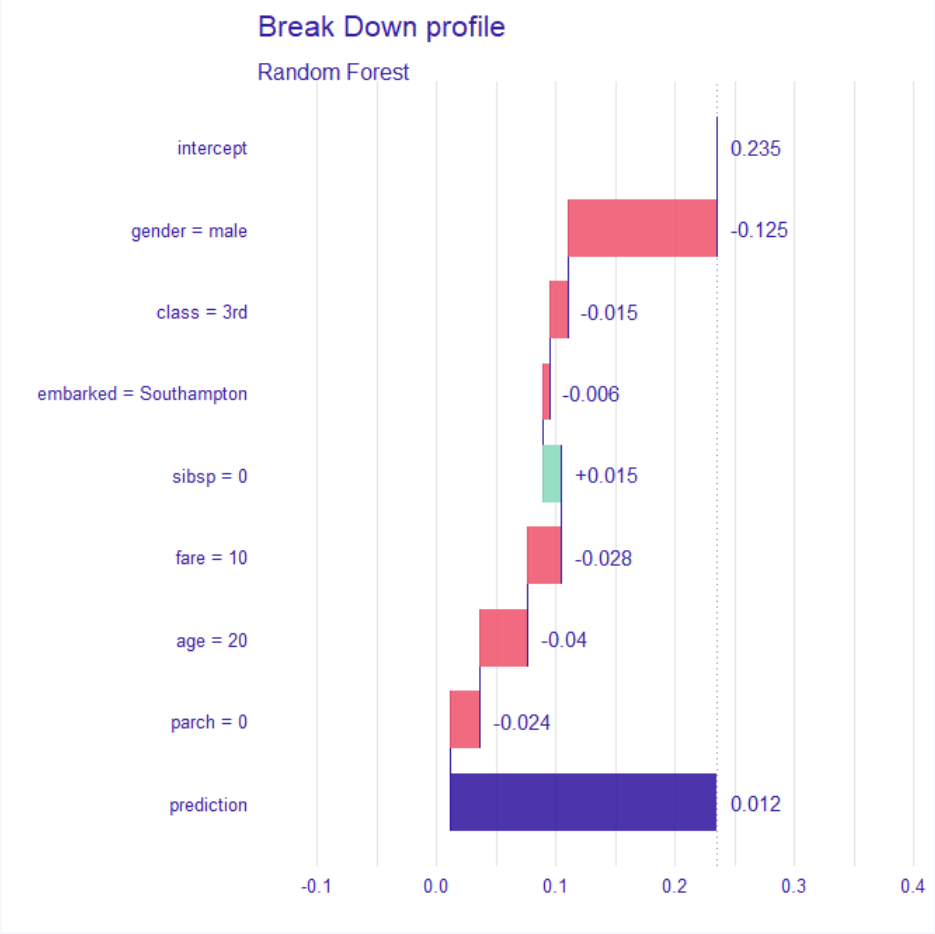
Co wpływa na wynik w danym modelu?



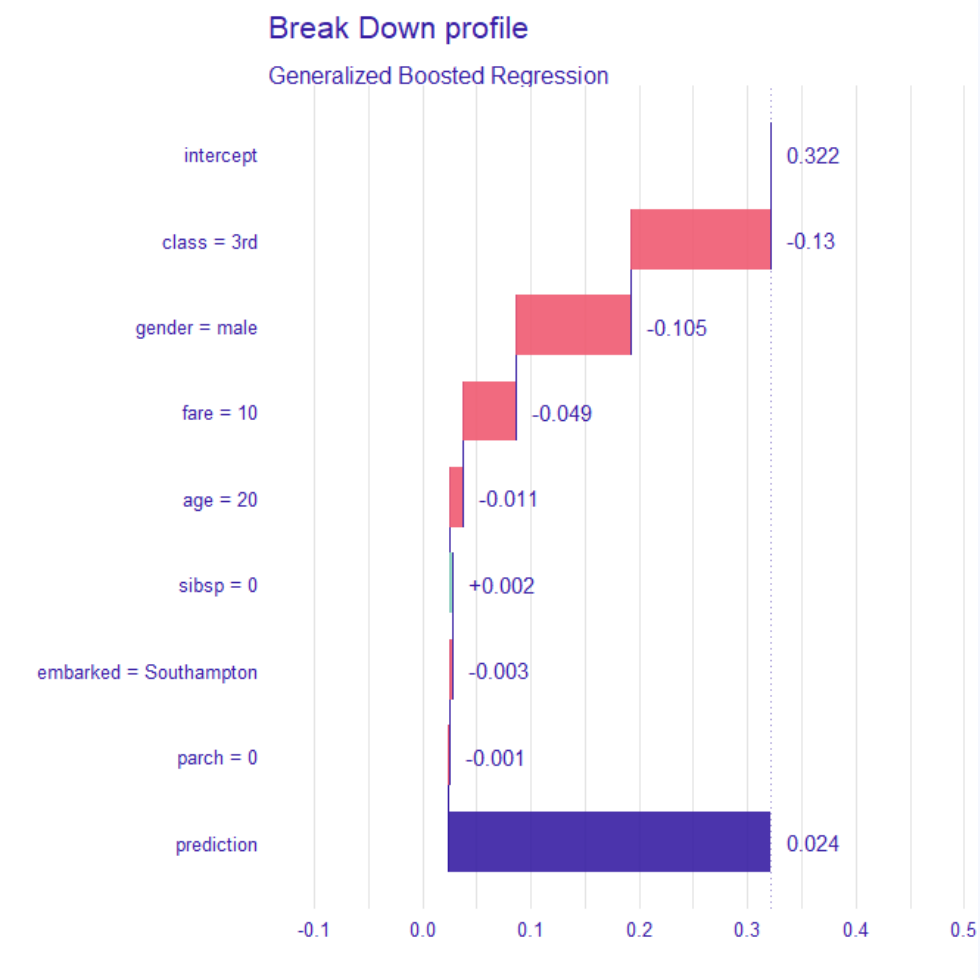
Jack


Break Down profile
Logistic Regression
intercept
class = 3rd
gender = male
sibsp = 0
age = 20
embarked = Southampton
parch = 0
fare = 10
prediction
0.322
-0.12
-0.108
+0.012
-0.003
-0.006
+0.002
-0.001
0.097
0.0
0.1
0.2
0.3
0.4

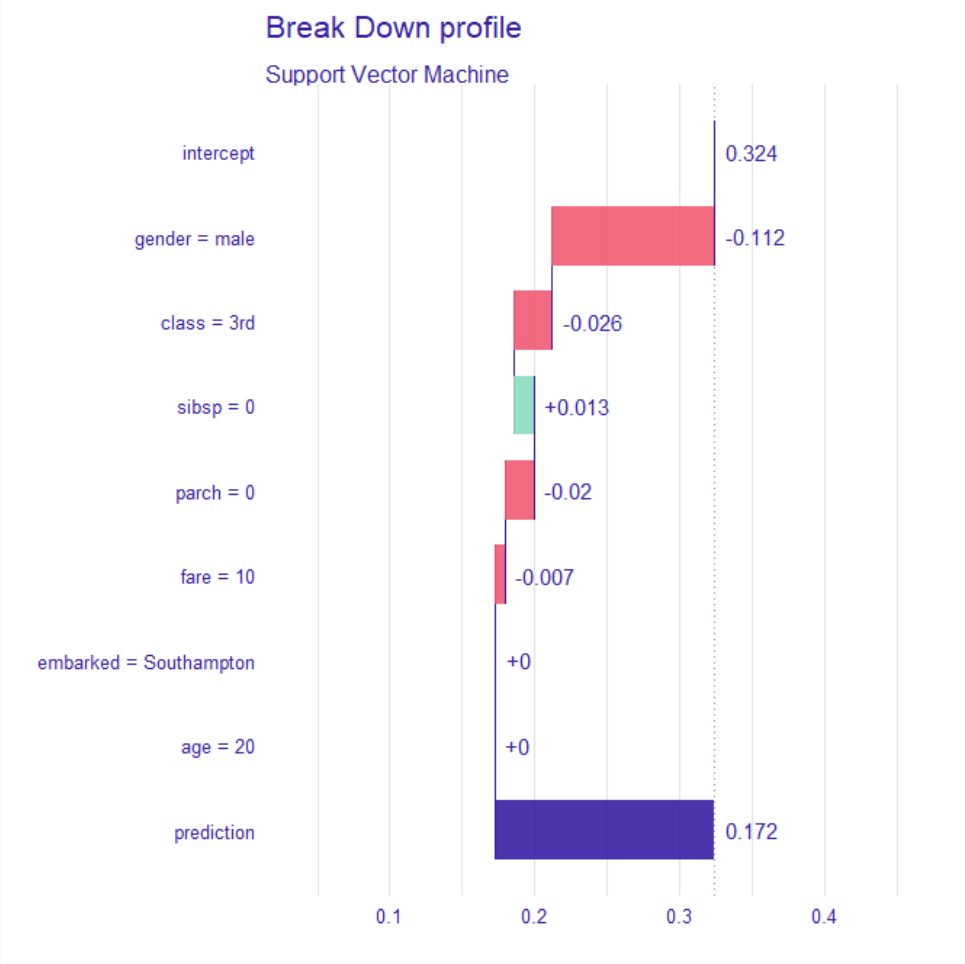
Jack


Break Down profile
Random Forest
intercept
gender = male
class = 3rd
embarked = Southampton
sibsp = 0
fare = 10
age = 20
parch = 0
prediction
0.235
-0.125
-0.015
-0.006
+0.015
-0.028
-0.04
-0.024
0.012
-0.1
0.0
0.1
0.2
0.3
0.4

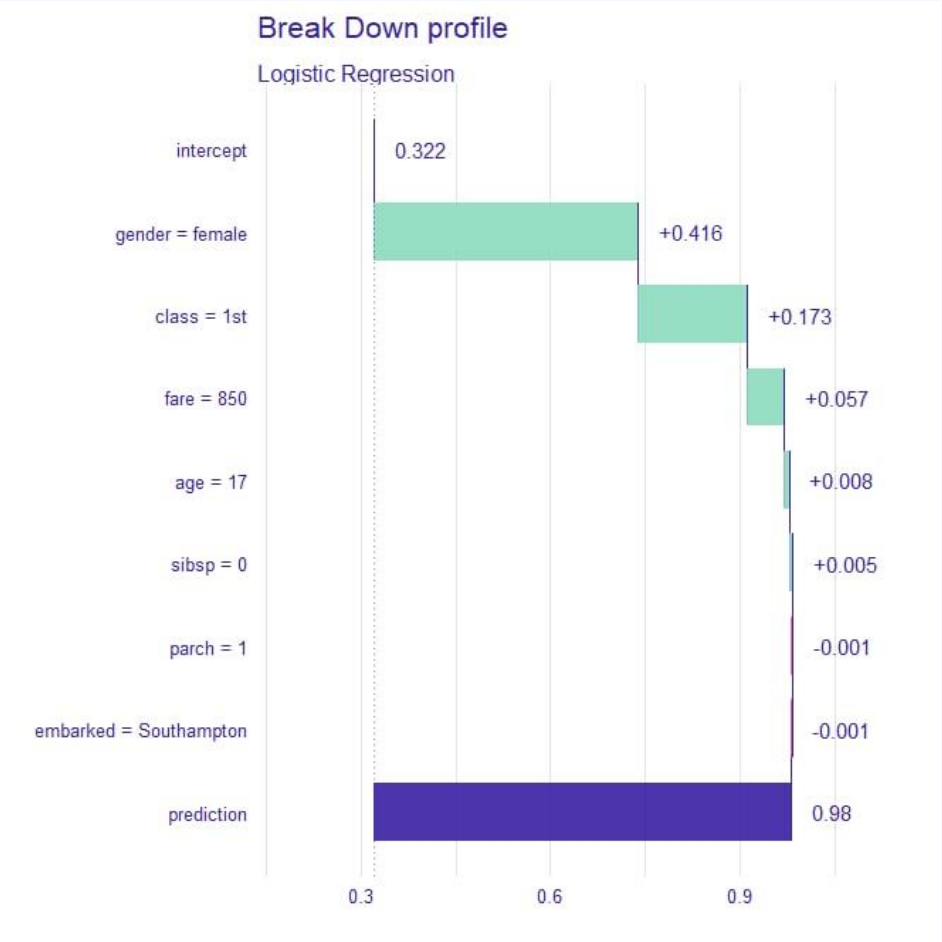
Jack


Break Down profile
Generalized Boosted Regression
intercept
class = 3rd
gender = male
fare = 10
age = 20
sibsp = 0
embarked = Southampton
parch = 0
prediction
0.322
-0.13
-0.105
-0.049
-0.011
+0.002
-0.003
-0.001
0.024
-0.1
0.0
0.1
0.2
0.3
0.4
0.5

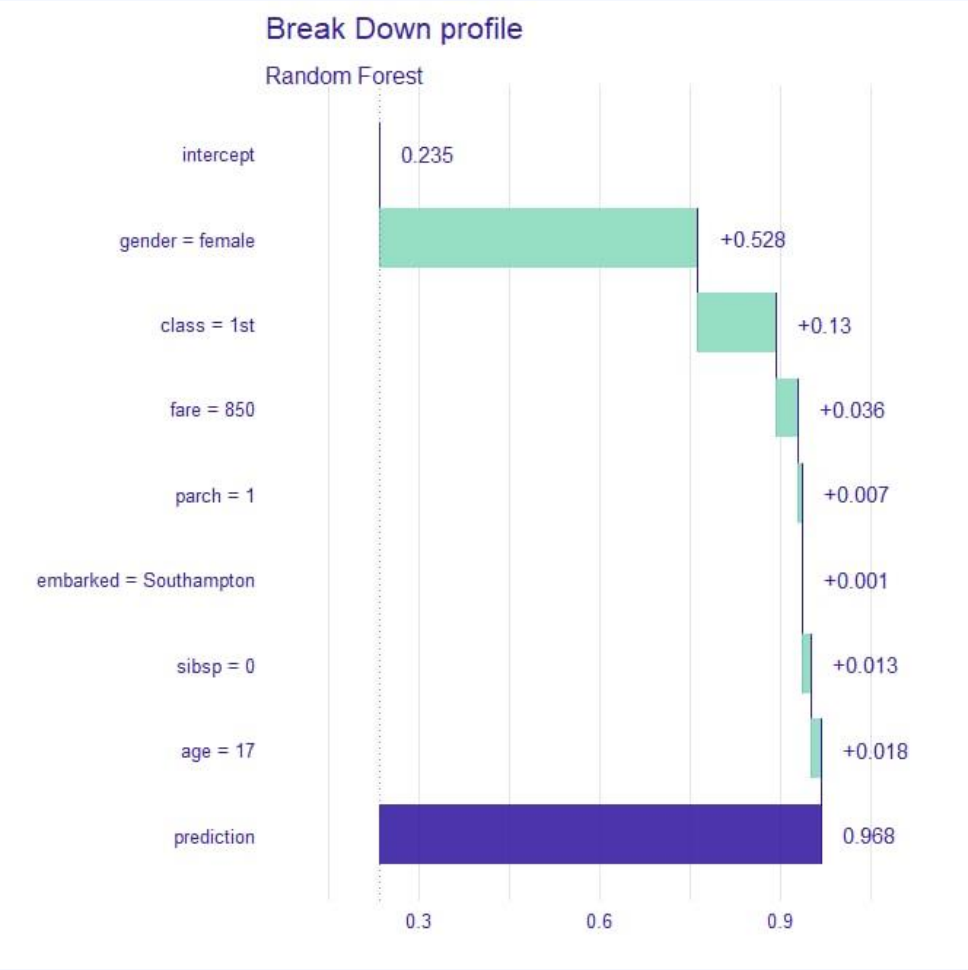
Jack


Break Down profile
Support Vector Machine
intercept
gender = male
class = 3rd
sibsp = 0
parch = 0
fare = 10
embarked = Southampton
age = 20
prediction
0.324
-0.112
-0.026
+0.013
-0.02
-0.007
+0
+0
0.172
0.1
0.2
0.3
0.4

Rose


Break Down profile
Logistic Regression
intercept
0.322
gender = female
+0.416
class = 1st
+0.173
fare = 850
+0.057
age = 17
+0.008
sibsp = 0
+0.005
parch = 1
-0.001
embarked = Southampton
-0.001
prediction
0.98
0.3
0.6
0.9

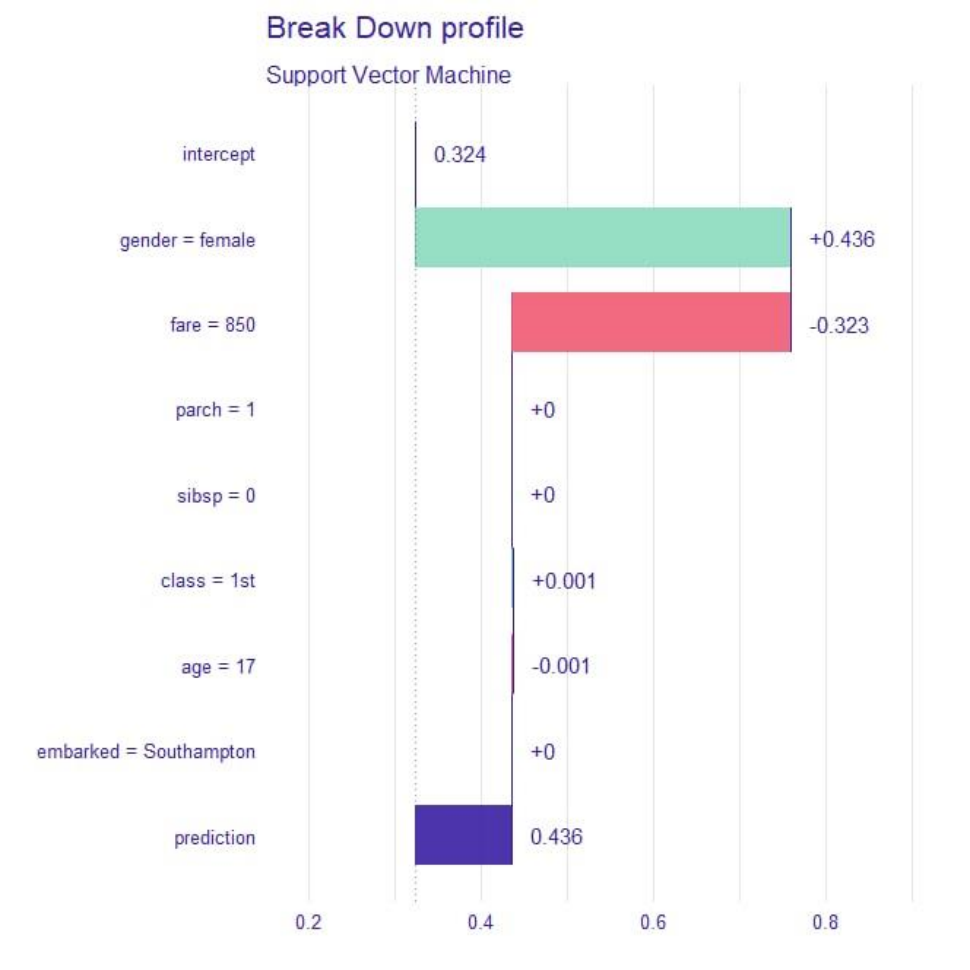
Rose


Break Down profile
Random Forest
intercept
0.235
gender = female
+0.528
class = 1st
+0.13
fare = 850
+0.036
parch = 1
+0.007
embarked = Southampton
+0.001
sibsp = 0
+0.013
age = 17
+0.018
prediction
0.968
0.3
0.6
0.9

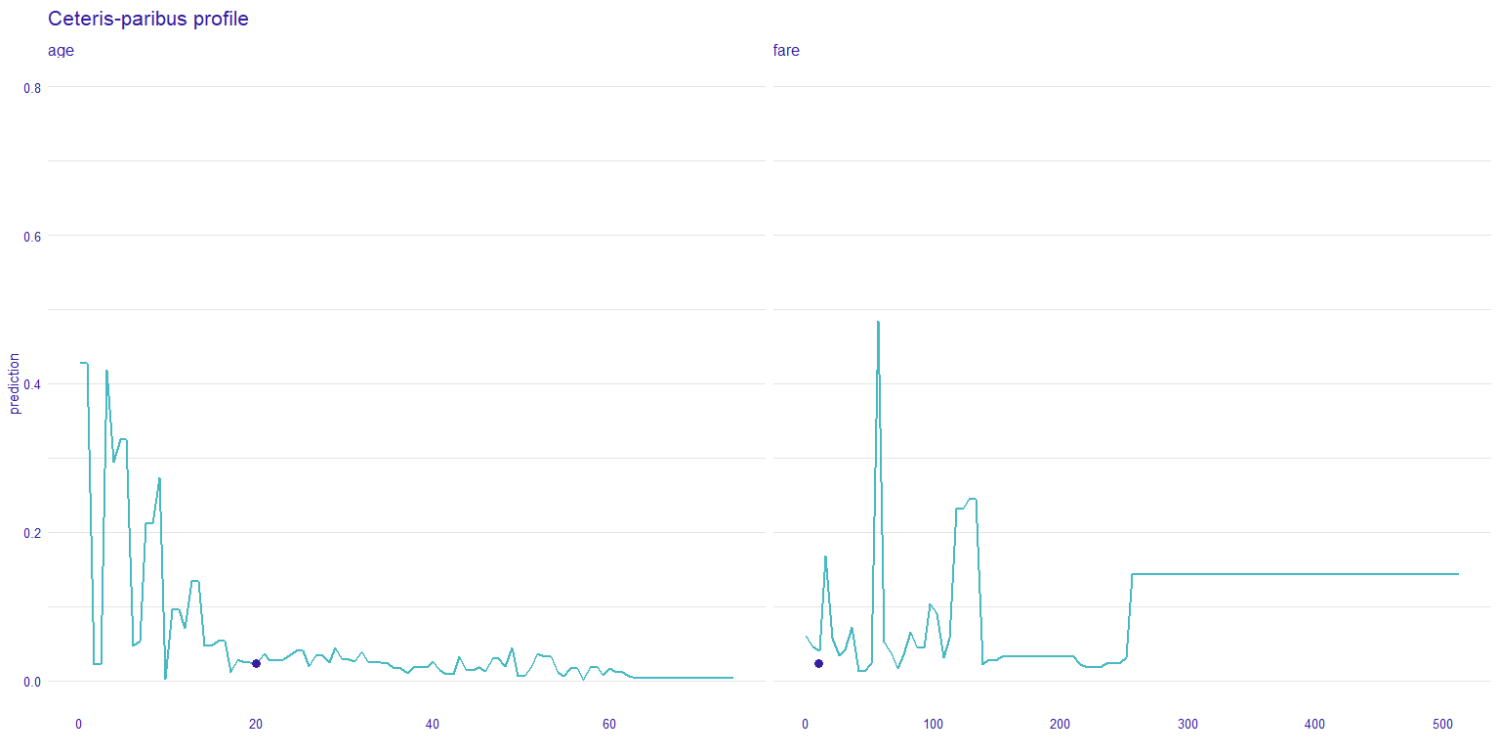
Rose


Break Down profile
Generalized Boosted Regression
intercept
0.322
gender = female
+0.432
class = 1st
+0.156
fare = 850
+0.06
age = 17
-0.024
parch = 1
+0.017
sibsp = 0
+0.009
embarked = Southampton
-0.001
prediction
0.969
0.3
0.6
0.9

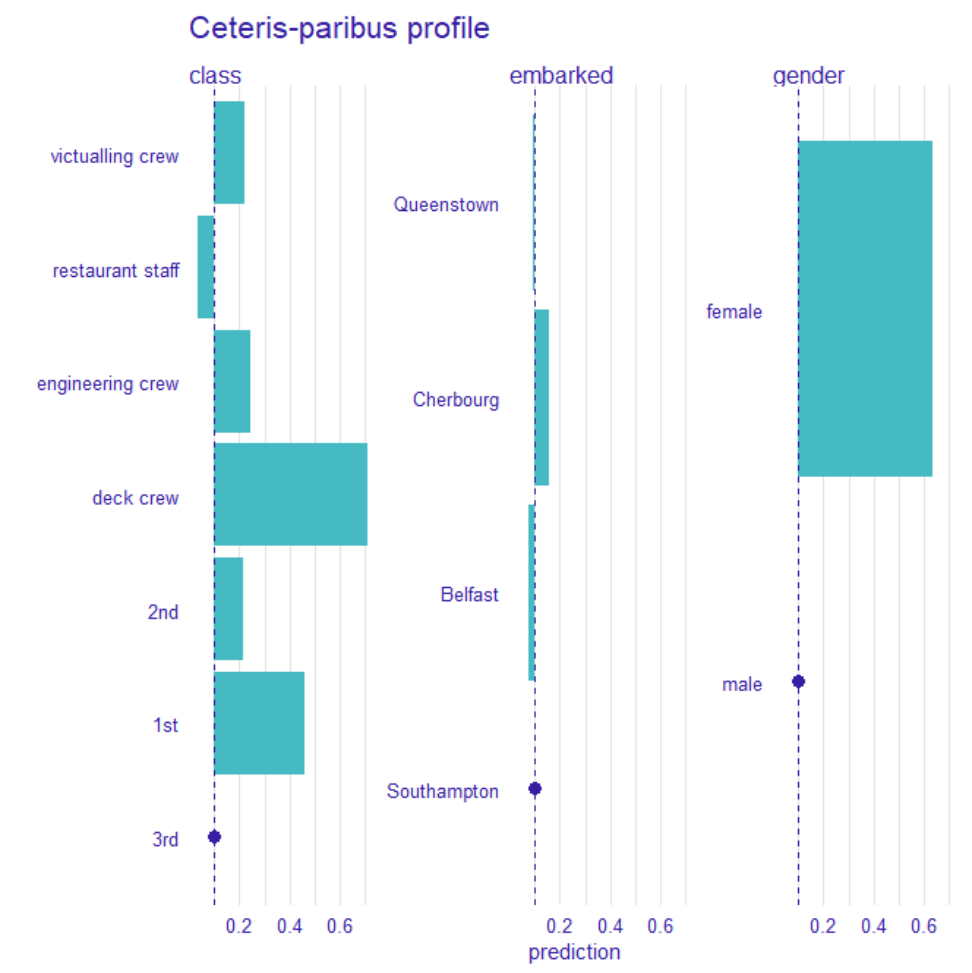
Rose


Break Down profile
Support Vector Machine
intercept
0.324
gender = female
+0.436
fare = 850
-0.323
parch = 1
+0
sibsp = 0
+0
class = 1st
+0.001
age = 17
-0.001
embarked = Southampton
+0
prediction
0.436
0.2
0.4
0.6
0.8

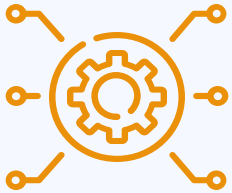
```
cp_titanic_gbm <- predict_profile(explainer = titanic_gbm_exp,
                                  new_observation = jack)

plot(cp_titanic_gbm, variables = c("age", "fare")) +
  ggtitle("Ceteris-paribus profile", "") + ylim(0, 0.8)
```

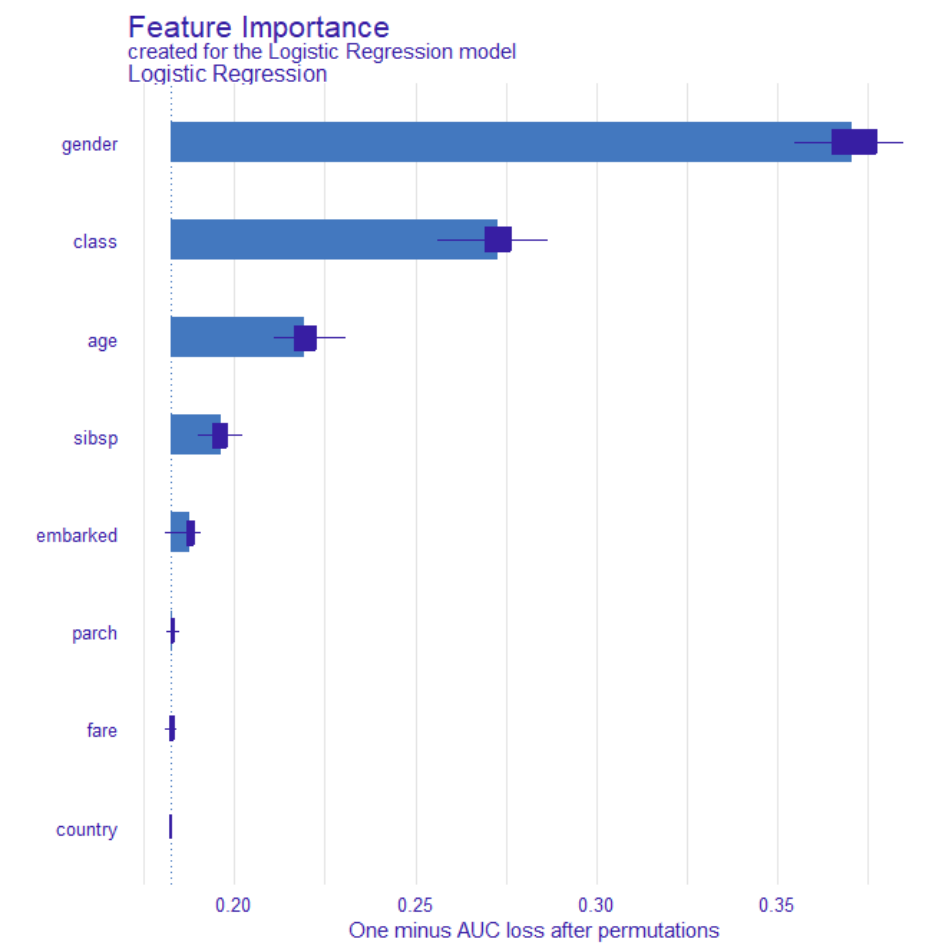


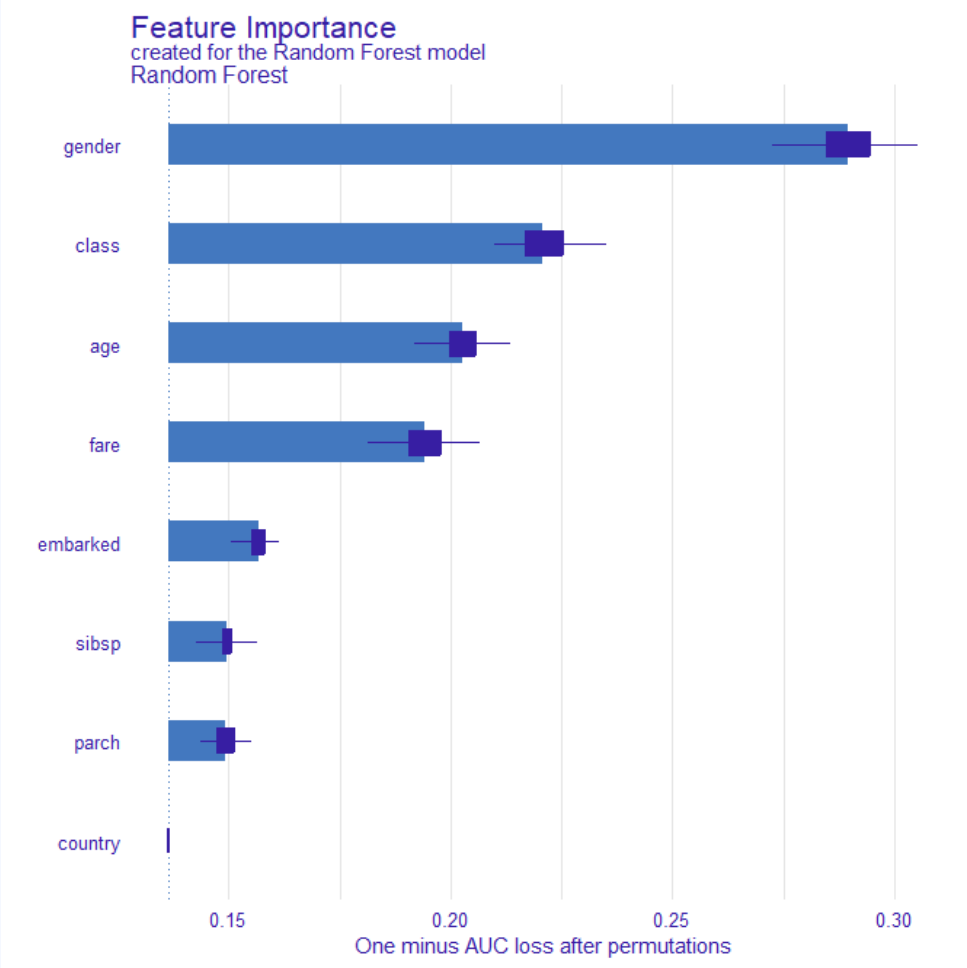
```r
cp_titanic_lmr <- predict_profile(explainer = titanic_lmr_exp,
                                  new_observation = jack)

plot(cp_titanic_lmr, variables = c("class", "embarked", "gender"),
     variable_type = "categorical", categorical_type = "bars") +
  ggtitle("Ceteris-paribus profile", "")
```

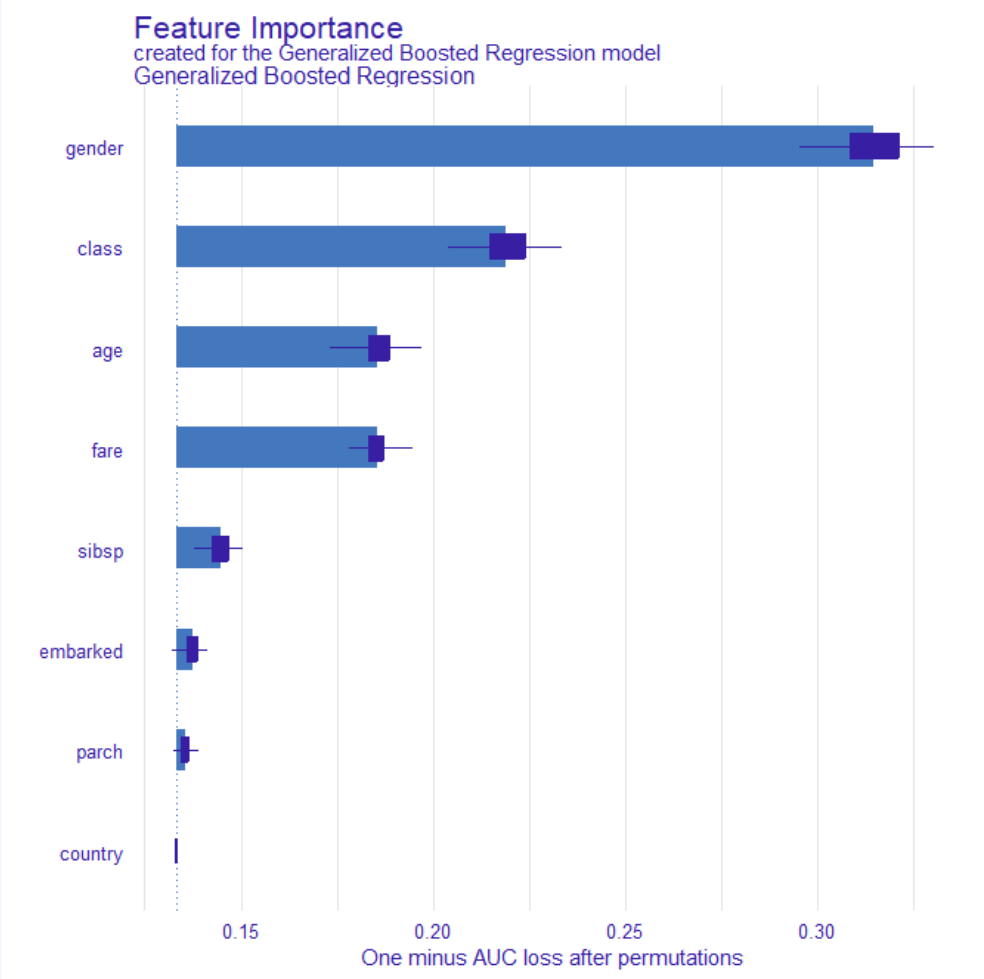

Ceteris-paribus profile
class
victualling crew
restaurant staff
engineering crew
deck crew
2nd
1st
3rd
0.2
0.4
0.6
embarked
Queenstown
Cherbourg
Belfast
Southampton
0.2
0.4
0.6
prediction
gender
female
male
0.2
0.4
0.6

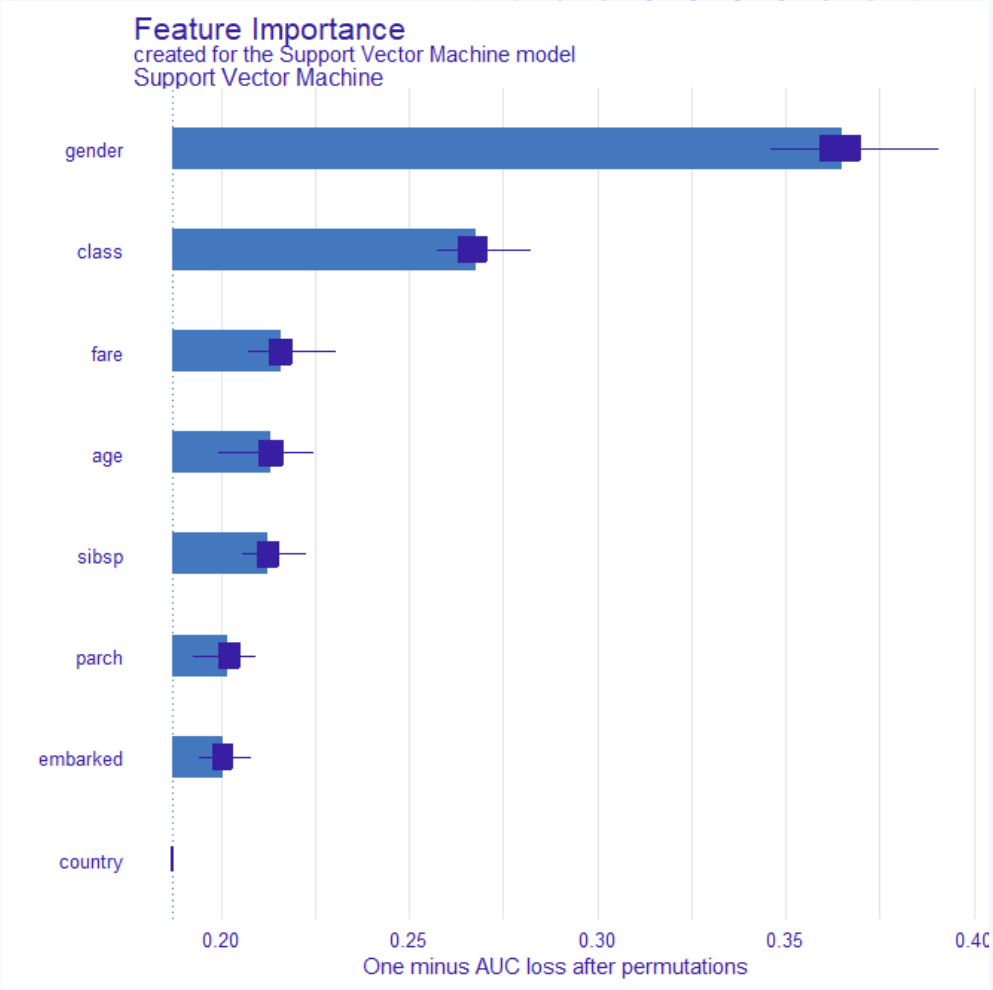
Od jakich zmiennych zależy predykcja?



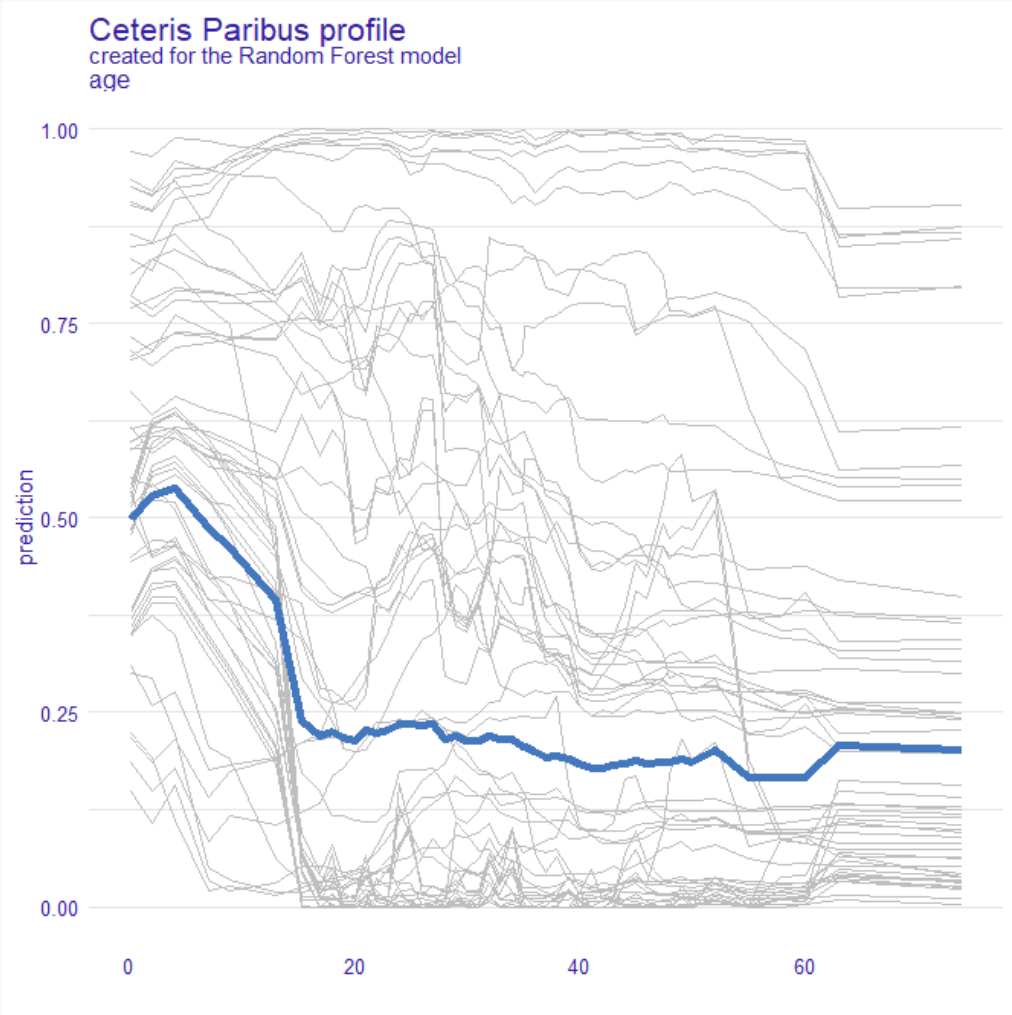

Feature Importance
created for the Logistic Regression model
Logistic Regression
gender
class
age
sibsp
embarked
parch
fare
country
0.20
0.25
0.30
0.35
One minus AUC loss after permutations


Feature Importance
created for the Random Forest model
Random Forest
gender
class
age
fare
embarked
sibsp
parch
country
0.15
0.20
0.25
0.30
One minus AUC loss after permutations

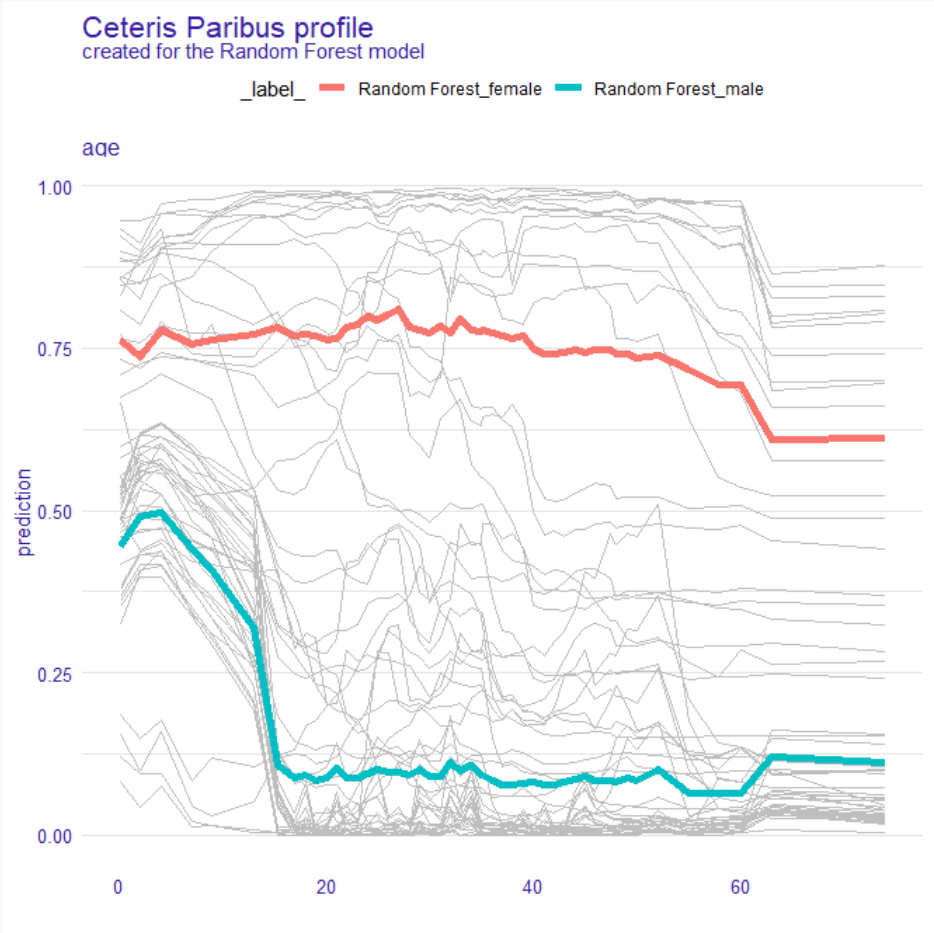

Feature Importance
created for the Generalized Boosted Regression model
Generalized Boosted Regression
gender
class
age
fare
sibsp
embarked
parch
country
0.15
0.20
0.25
0.30
One minus AUC loss after permutations


Feature Importance
created for the Support Vector Machine model
Support Vector Machine
gender
class
fare
age
sibsp
parch
embarked
country
0.20
0.25
0.30
0.35
0.40
One minus AUC loss after permutations

Jak wiek wpływa na przeżywalność?



Jak płeć wpływa na przeżywalność?



Teraz pora na Ciebie ☺

```
#twoja kolej
#twoje_imie <- data.frame(
  #class = factor(, levels = c("1st", "2nd", "3rd", "deck crew", "engineering crew", "restaurant staff", "victualling crew")),
  #gender = factor(, levels = c("female", "male")),
  #age = ,
  #sibsp = ,
  #parch = ,
  #fare = ,
  #embarked = factor(, levels = c("Belfast", "Cherbourg", "Queenstown", "Southampton"))
#)
```