Uczenie maszynowe w R

Zaglądamy do czarnej skrzynki

Bałtycki Festiwal Nauki

Uczenie maszynowe

Dziedzina informatyki i matematyki, w której komputery dzięki stworzony przez człowieka algorytmom, potrafią “uczyć się” na danym zbiorze danych.

W przypadku niektórych algorytmów , szczególnie tych bardziej skomplikowanych, trudno jest zrozumieć, jak dokładnie działają, stąd określenie “czarna skrzynka”.

Uczenie maszynowe a AI

Język programowania

Język programowania, który używa się głównie do analizowania danych .
Jest bardzo popularny wśród naukowców i statystyków.

Środowisko

Program, w którym pracujemy z językiem R

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.

Zestaw danych

Zestaw danych, który analizujemy, zawiera informacje o ponad 2200 pasażerach i członkach załogi Titanica.

Dane obejmują:

- wiek, płeć, klasę podróży
- miejsce, z którego zaokrętowano
- kraj pochodzenia
- liczbę bliskich osób na pokładzie
    - sibsp – liczba rodzeństwa i/lub małżonków na pokładzie
    - parch – liczba rodziców i/lub dzieci na pokładzie
- opłatę za bilet

oraz najważniejsze

- czy dana osoba przeżyła katastrofę

	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
16	male	35.0000000	2nd	Southampton	England	13.0000	0	0	no

Celem analizy jest odkrycie, jakie czynniki mogły wpłynąć na przeżycie pasażerów i zrozumienie, jak wyglądała struktura społeczna na pokładzie Titanica.

Modele, których użyjemy

Model regresji liniowej

Model gradient boosting

Model regresji logistycznej

Model lasów losowych

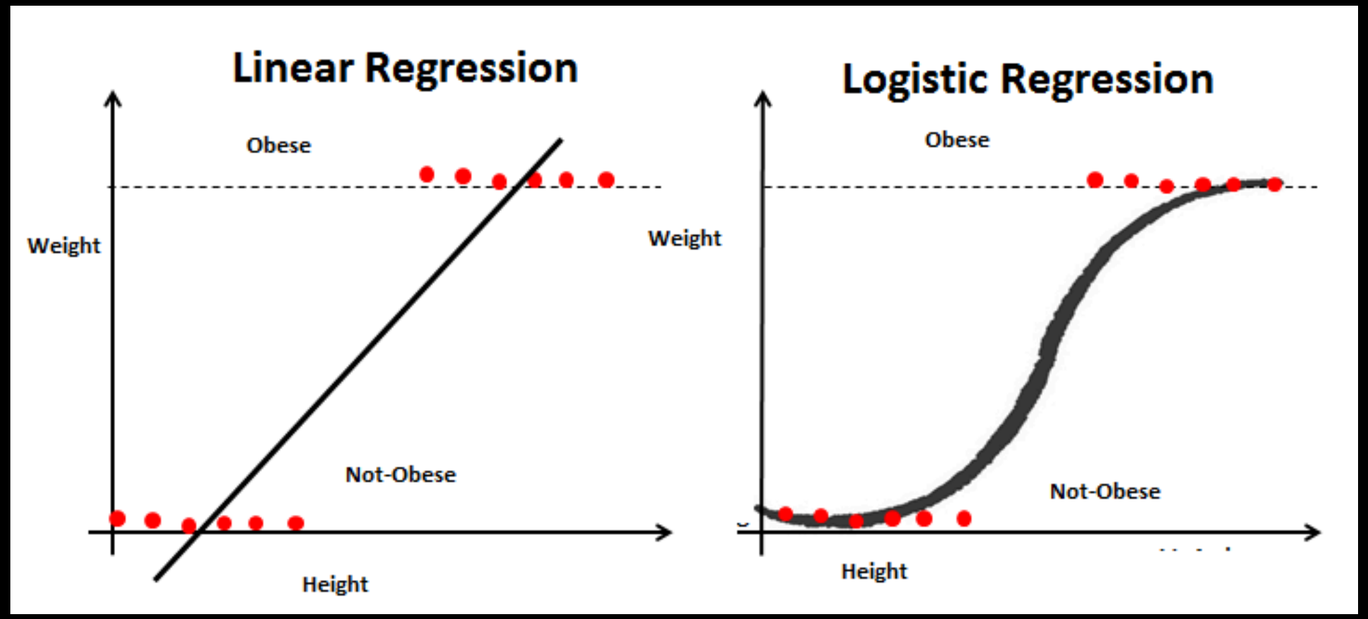
Model maszyny wektorów nośnych (SVM)

Model regresji liniowej

Przewiduje wartość liczbową zmiennej zależnej na podstawie jednej lub więcej zmiennych niezależnych. Zakłada istnienie liniowej zależności między tymi zmiennymi. Na przykład kiedy na podstawie wieku i klasy pasażera chcemy oszacować cenę biletu.

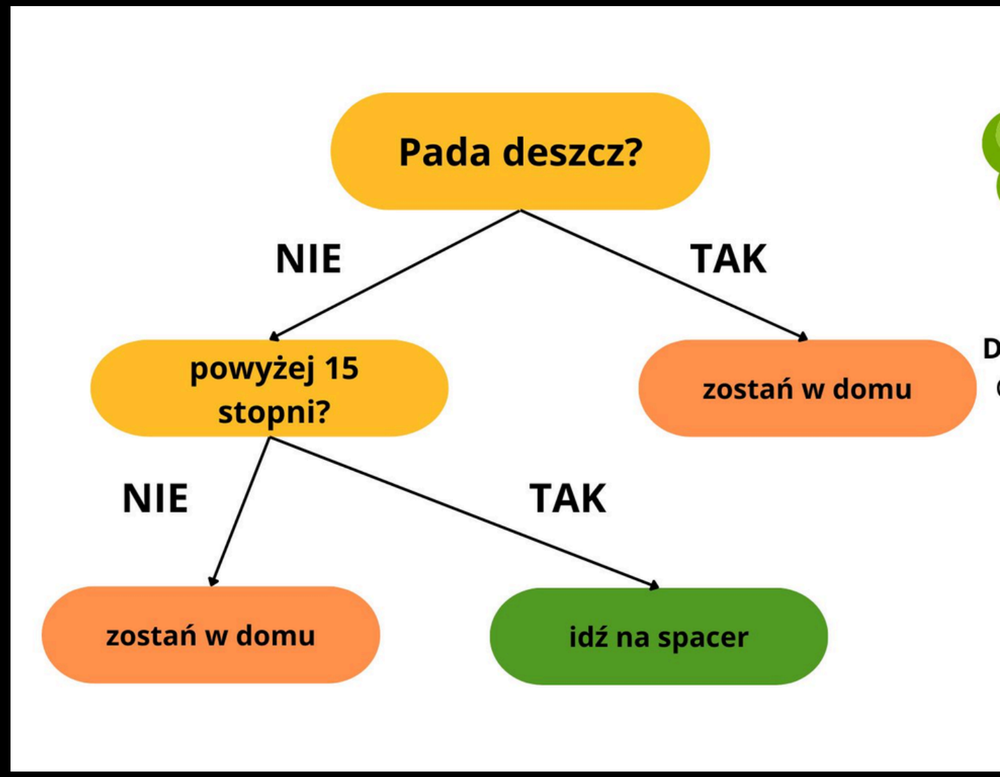
Model regresji logistycznej

Służy do przewidywania prawdopodobieństwa wystąpienia konkretnego zdarzenia, które ma tylko dwie możliwości (np. tak/nie, przeżył/nie przeżył). Zamiast prostej linii (jak w regresji liniowej), używa krzywej w kształcie litery S, która dobrze oddziela dwie grupy.



Model lasów losowych

Tworzy i łączy wiele drzew decyzyjnych, cechuje się wysoką dokładnością i stabilnością – przewidywania dla zadań klasyfikacji i regresji.

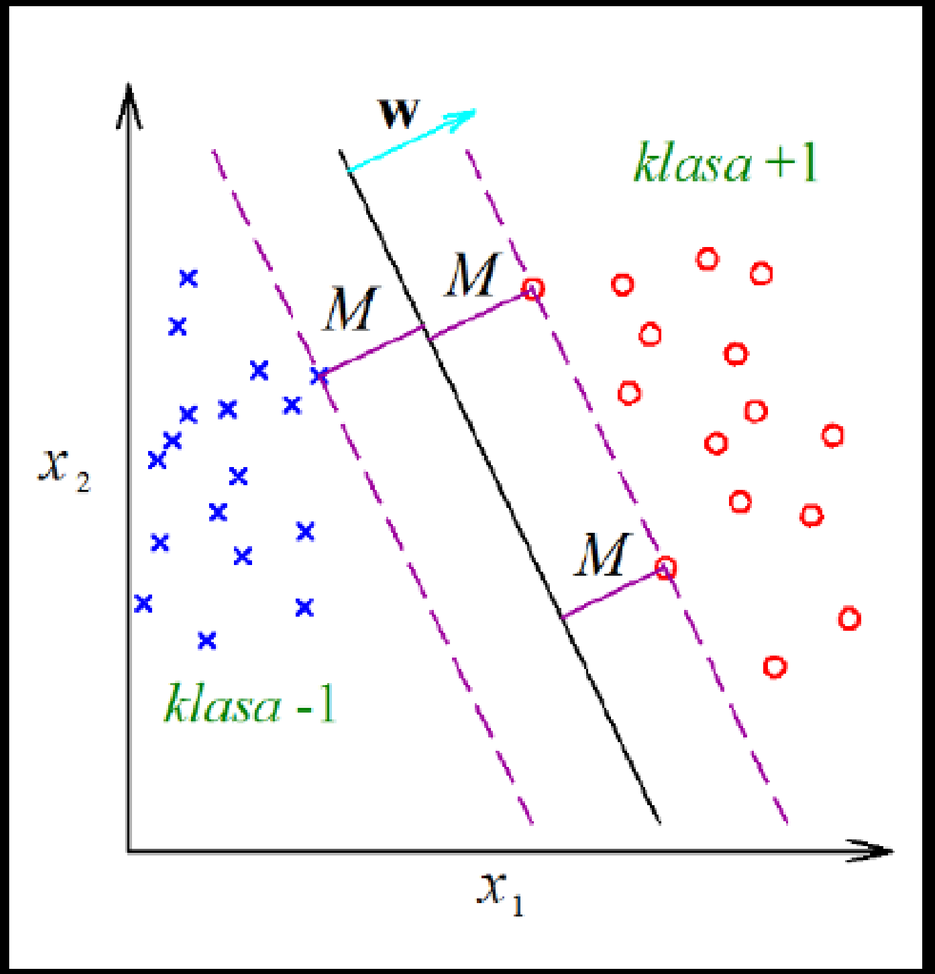


Model gradient boosting

Polega na stopniowym dodawaniu nowych, prostych modeli (często drzew decyzyjnych) do zestawu już istniejących, aby coraz lepiej przewidywać wyniki, poprawiając błędy krok po kroku

Model maszyny wektorów nośnych (SVM)

Szuka najlepszej granicy oddzielającej różne klasy danych, maksymalizując odstęp (margines) między nimi – bardzo skuteczna w klasyfikacji, zwłaszcza gdy dane są dobrze rozdzielne.



Tworzenie modeli

Model regresji logistycznej:

```{r}
titanic_lmr <- rms::lrm(survived == "1" ~ gender + rms::rcs(age) + class + sibsp + parch + fare + embarked, titanic)
```

Model lasów losowych

```{r}
set.seed(1313)
titanic_rf <- randomForest::randomForest(survived ~ class + gender + age + sibsp + parch + fare + embarked, data = titanic,
na.action=na.roughfix)
```

Model gradient boosting

```{r}
set.seed(1313)
titanic_gbm <- gbm(
  survived ~ class + gender + age + sibsp + parch + fare + embarked, data = titanic, n.trees = 15000, distribution = "bernoulli")
```

Model maszyny wektorow nosnych:

```{r}
set.seed(1313)
titanic_svm <- e1071::svm(survived == "1" ~ class + gender + age + sibsp + parch + fare + embarked, data = titanic,
                          type = "C-classification", probability = TRUE)
```

Zapakowanie modeli do podobnych opakowań

Model regresji logistycznej:

```{r}
titanic_lmr_exp <- explain(model = titanic_lmr,
                           data = titanic[, -9],
                           y = titanic$survived == "1",
                           label = "Logistic Regression",
                           type = "classification")
```

Model lasów losowych

```{r}
titanic_rf_exp <- explain(model = titanic_rf,
                          data = titanic[, -9],
                          y = titanic$survived == "1",
                          label = "Random Forest")
```

Model gradient boosting

```{r}
titanic_gbm_exp <- explain(model = titanic_gbm,
                           data = titanic[, -9],
                           y = titanic$survived == "1",
                           label = "Generalized Boosted Regression")
```

Model maszyny wektorow nośnych:

```{r}
titanic_svm_exp <- explain(model = titanic_svm,
                           data = titanic[, -9],
                           y = titanic$survived == "1",
                           label = "Support Vector Machine")
```

Porównanie siły predykcyjnej modeli

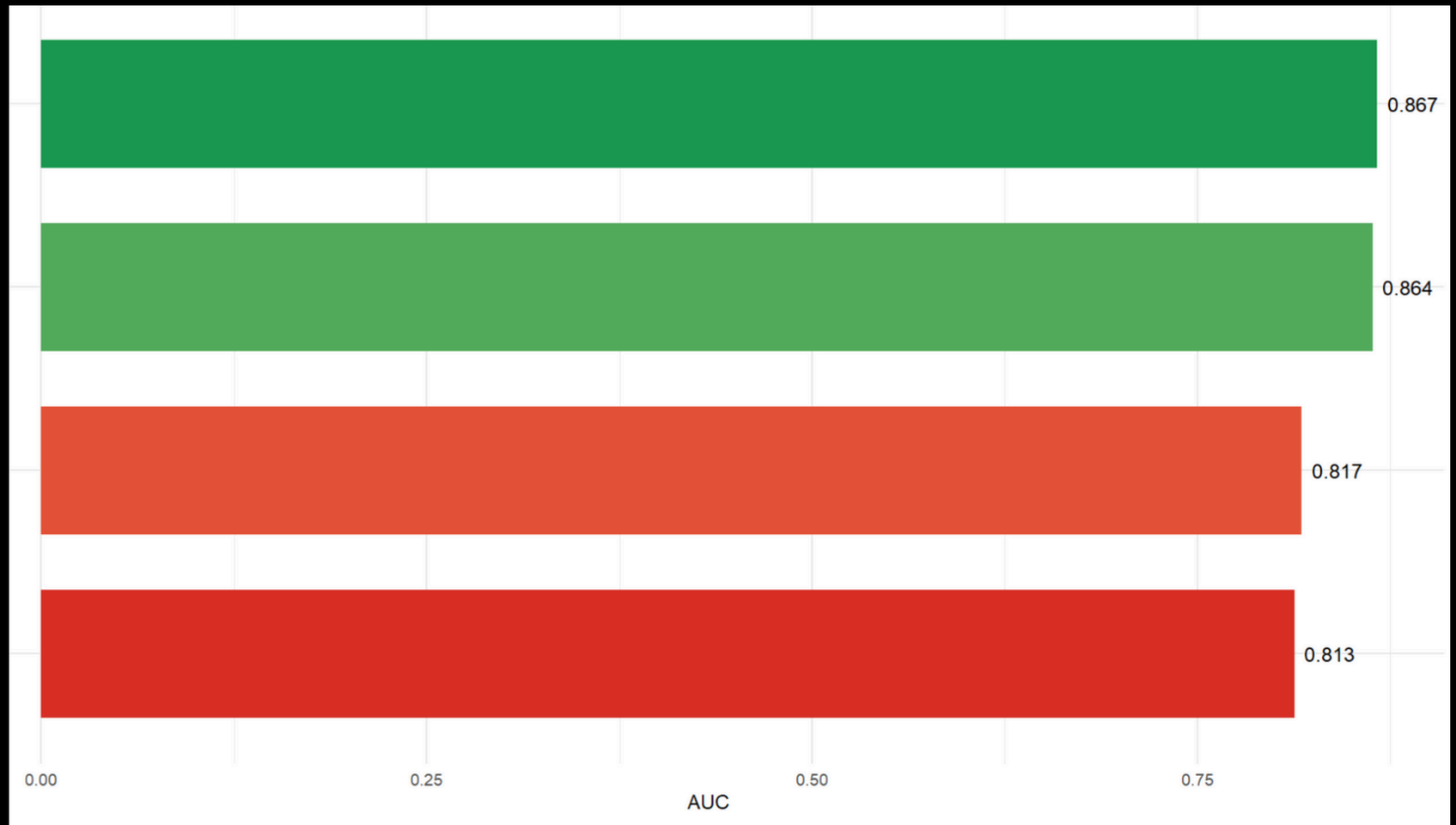
```r
auc_lmr<-model_performance(titanic_lmr_exp)$measures$auc
auc_rf<-model_performance(titanic_rf_exp)$measures$auc
auc_gbm<-model_performance(titanic_gbm_exp)$measures$auc
auc_svm<-model_performance(titanic_svm_exp)$measures$auc
```

Gradient Boosting

Random Forest

Regresja logistyczna

Maszyna wektorów nośnych



Przykładowi pasażerowie

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

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

Sprawdźmy czy Rose i Jack przeżyliby na prawdziwym Titanicu

```
#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: 96,82%

Model gradient boosting: 96,89%

Model maszyn wektorów nośnych: 43,57%

Jack

Model regresji logistycznej: 9,75%

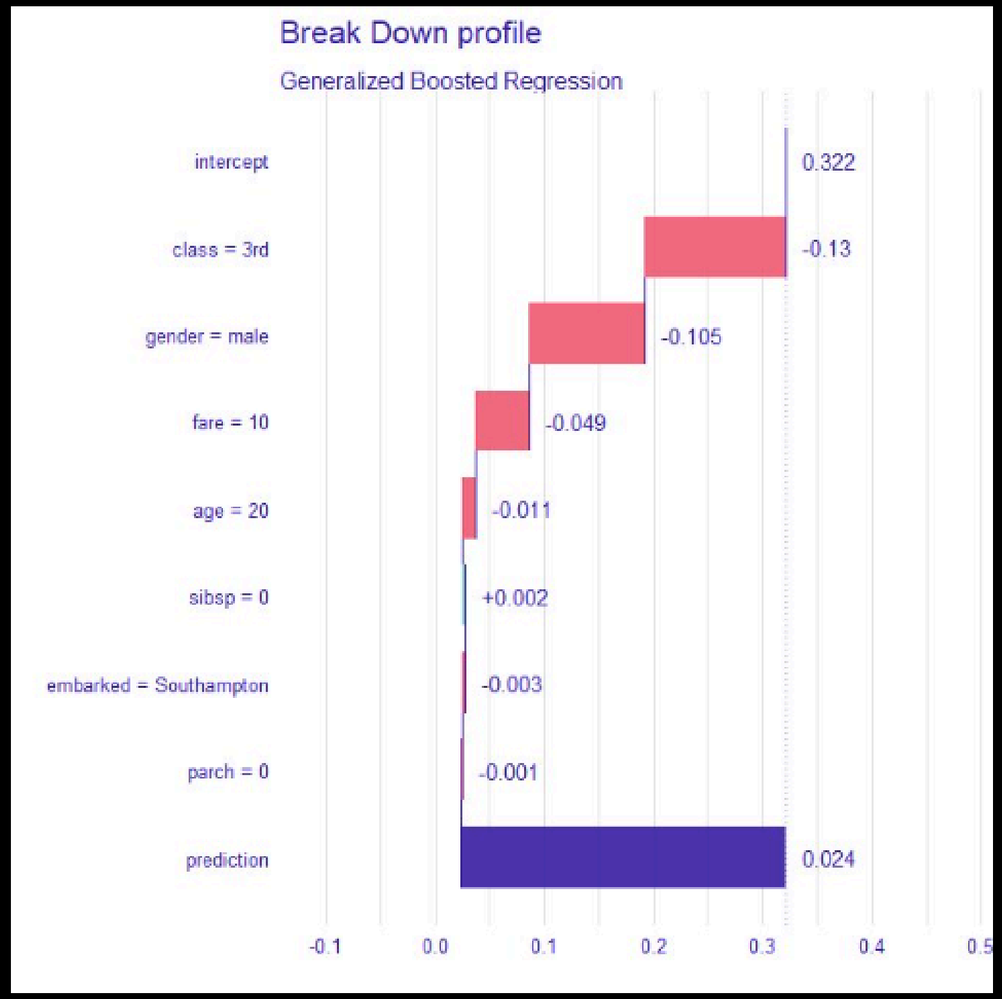
Model lasów losowych: 1,2%

Model gradient boosting: 2,37%

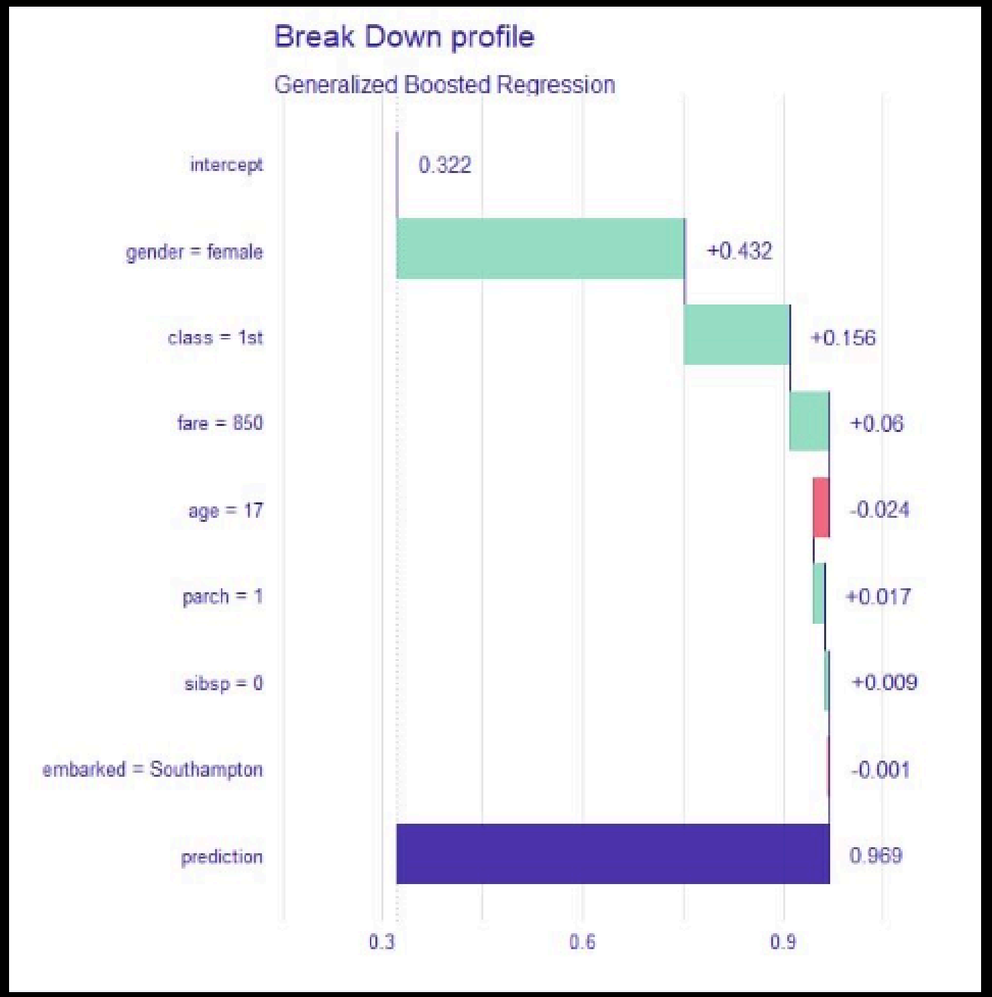
Model maszyn wektorów nośnych: 1,72%

Co wpływa na wynik?

Jack



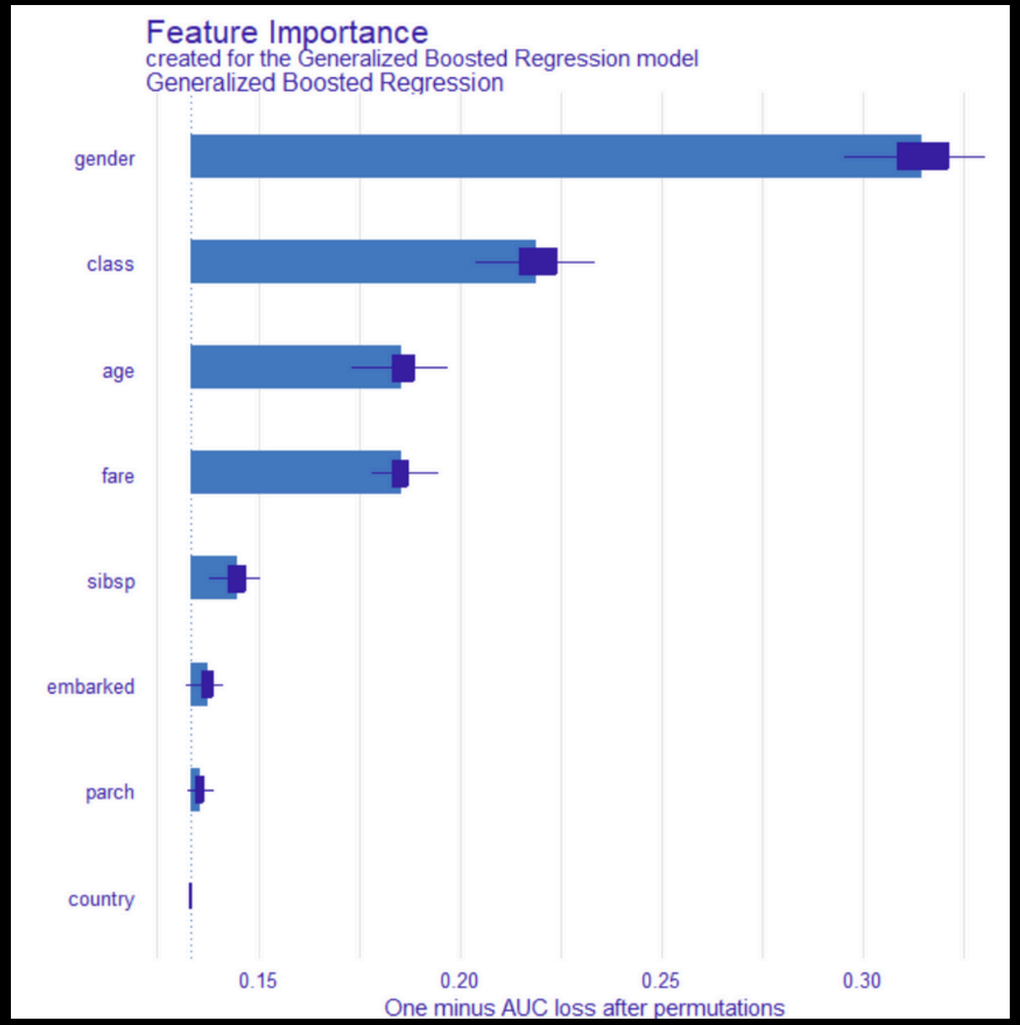
Rose



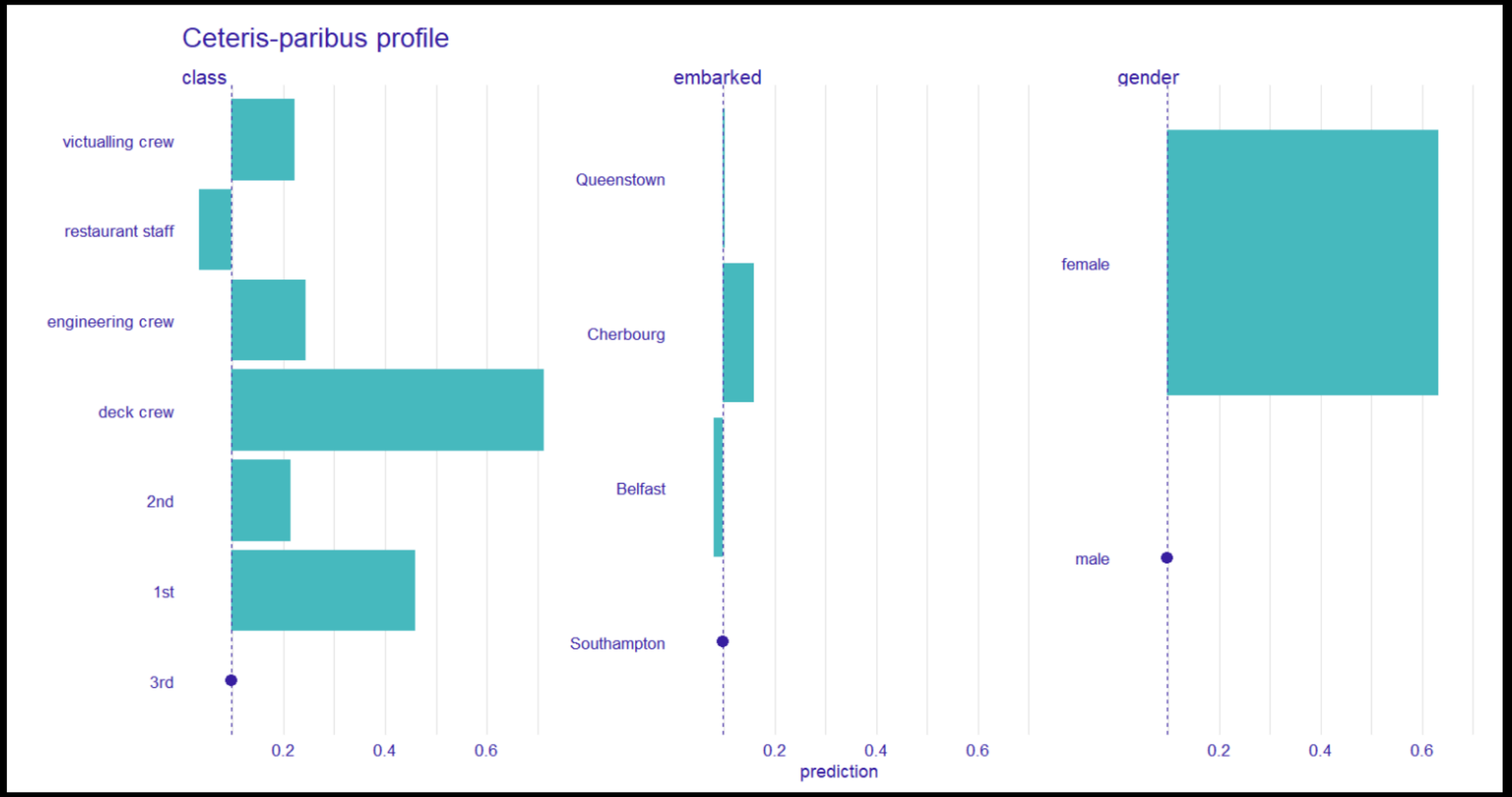
Które zmienne najbardziej wpływały na decyzje modelu?

Największy wpływ miały:

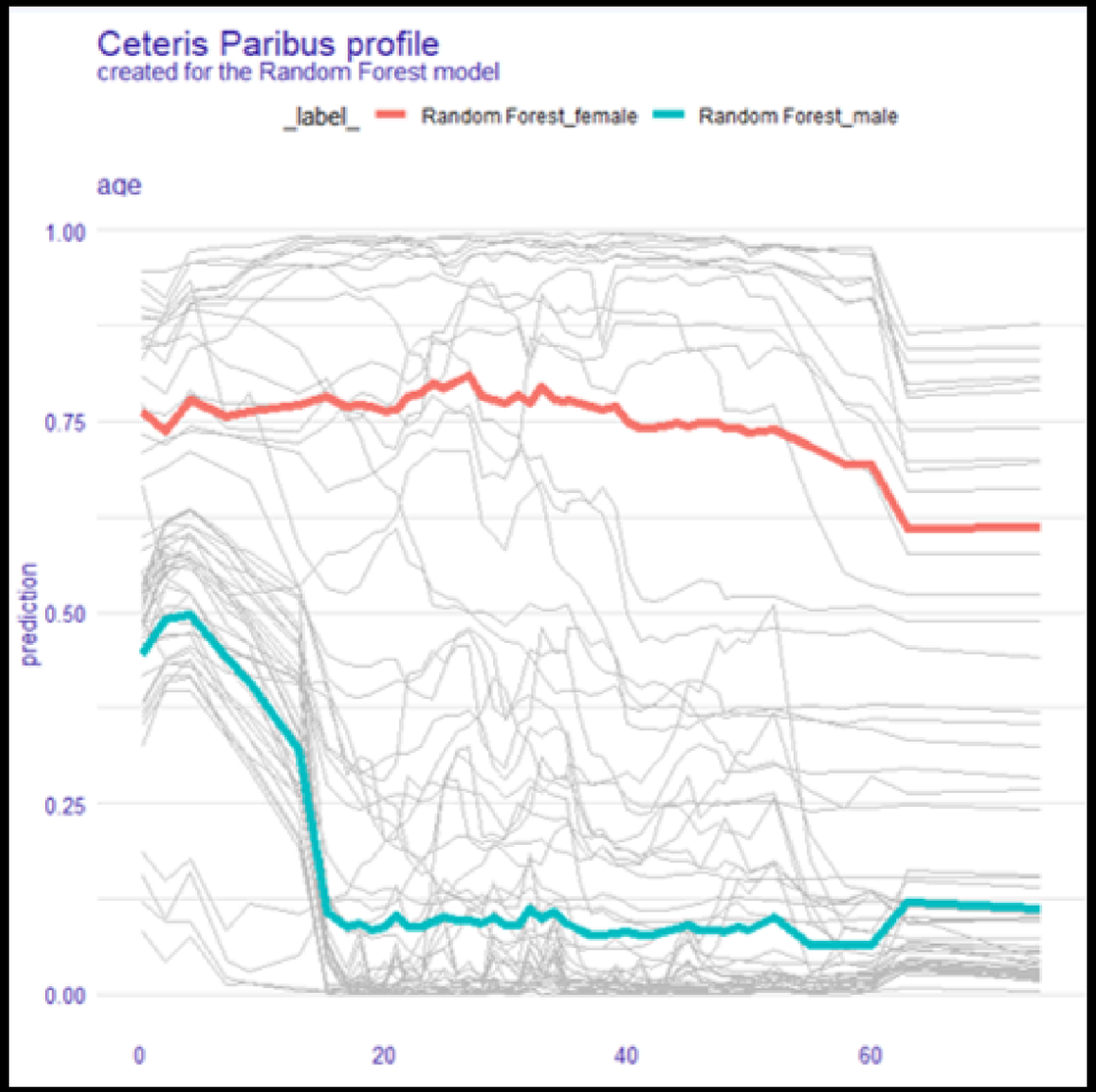
1. **Płeć**
2. **Klasa**
3. **Wiek**



Które cechy najbardziej wpływają na przeżycie pasażerów?



Jak płeć i wiek wpływa na przeżywalność?



Teraz pora na Ciebie!

```r
#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"))

#)
```