# Opłacalność windykacji po 3 miesiącach obsługi

Filip Wolniewski, Tomasz Gładki

Uniwersytet Wrocławski

28 Stycznia 2025

# Problem

- Zagadnienie: Bazując na trzech pierwszych miesiącach obsługi (cechy behawioralne) oraz cechach aplikacyjnych zidentyfikuj sprawy, w których nie jest opłacalne wykonywanie działań procesowych.
- Cel: Budowa modelów klasyfikacyjnych.

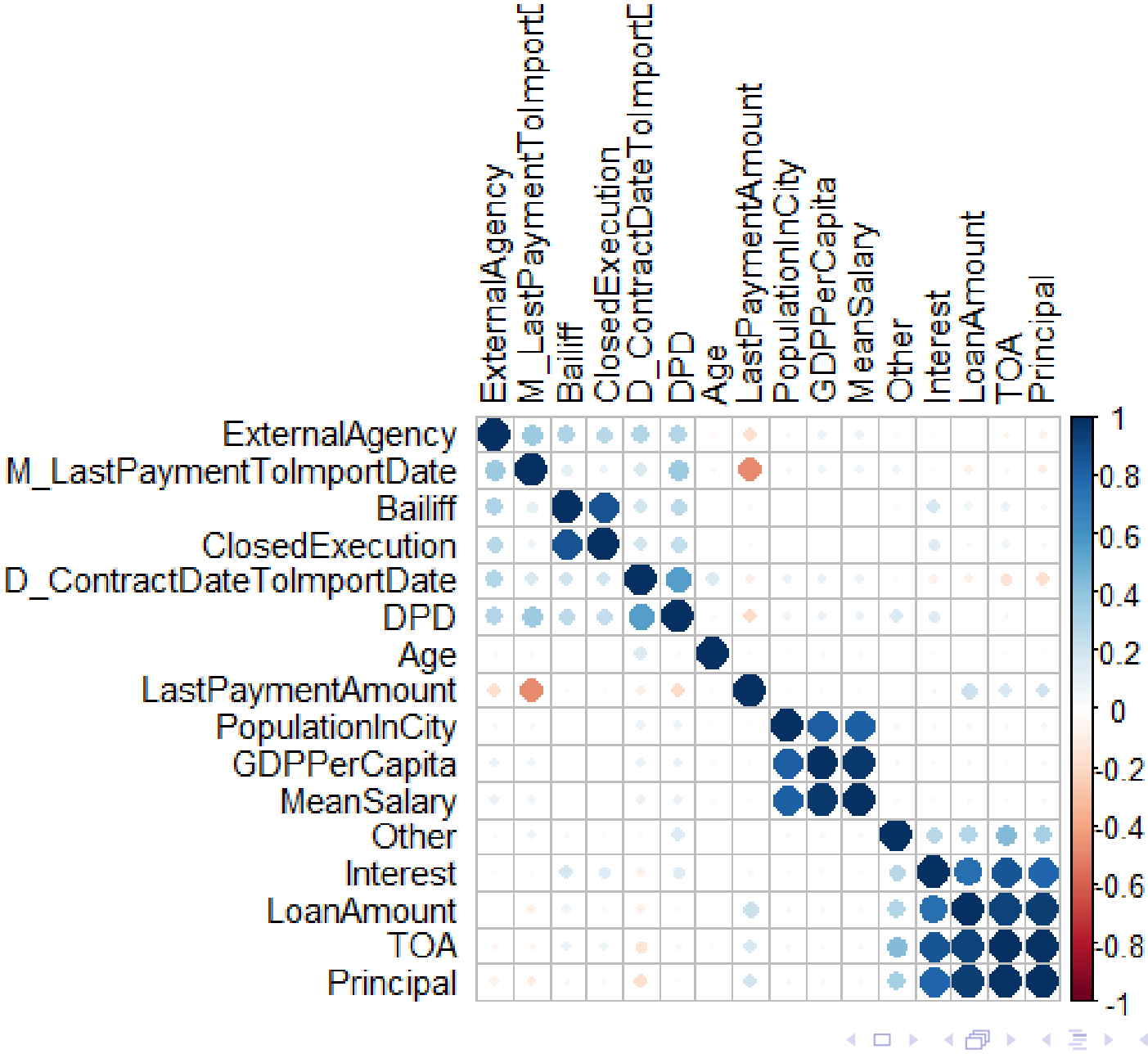
# Zmienna objaśniana

- Zmienna celu: kodujemy binarnie, że sprawa nie jest opłacalna w miesiącach 1-3 (tabela *events*).
- Zbiór testowy: kodujemy binarnie, że sprawa nie jest opłacalna w miesiącach 1-12.
- Opłacalność: czy sprawa spłaca dług (zysk), czy też poprzez różne czynności musimy się o spłatę dopominać (strata).
- Podejście: per sprawa.

# Obróbka danych w tabeli cases

- Dołączamy zmienną objaśnianą.
- Braki w zmiennych ciągłych uzupełniamy średnią; w zmiennych binarnych losowaniem z rozkładu dwupunktowego z progiem w postaci średniej.
- Obserwacje odstające: ze zmiennych *LoanAmount*, *DPD*, *LastPaymentAmount* usuwamy 0.001 obserwacji.
- Zmienne kategoryczne zamieniamy na *dummy variables*
- Analiza korelacji oraz selekcja zmiennych ze względu na złożoność obliczeniową modelu.
- Standaryzacja: jeśli konieczna dla modelu.

# Analiza korelacji

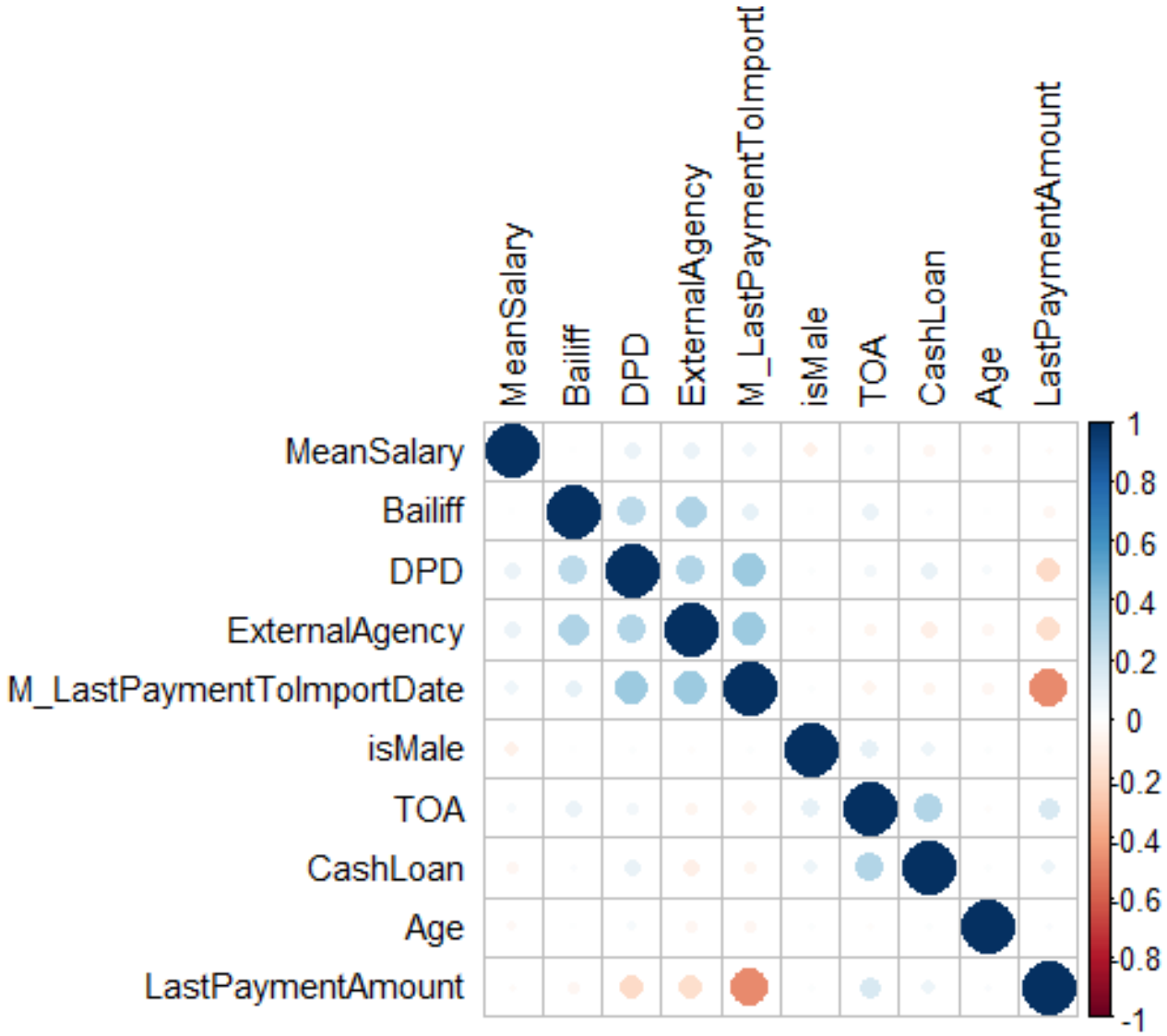


# Na co zwracać uwagę?

- Za zyski odpowiadają dobrze zidentyfikowane sprawy opłacalne (u nas: zera); za straty - źle zidentyfikowane (u nas: jedynki).
- Przy braku identyfikacji nic nie robimy.
- Średni zysk i średnia strata na sprawie.
- Jak zmaksymalizować *TNR* albo *profits*?
- Problem: 84% przypadków jest nieopłacalna.

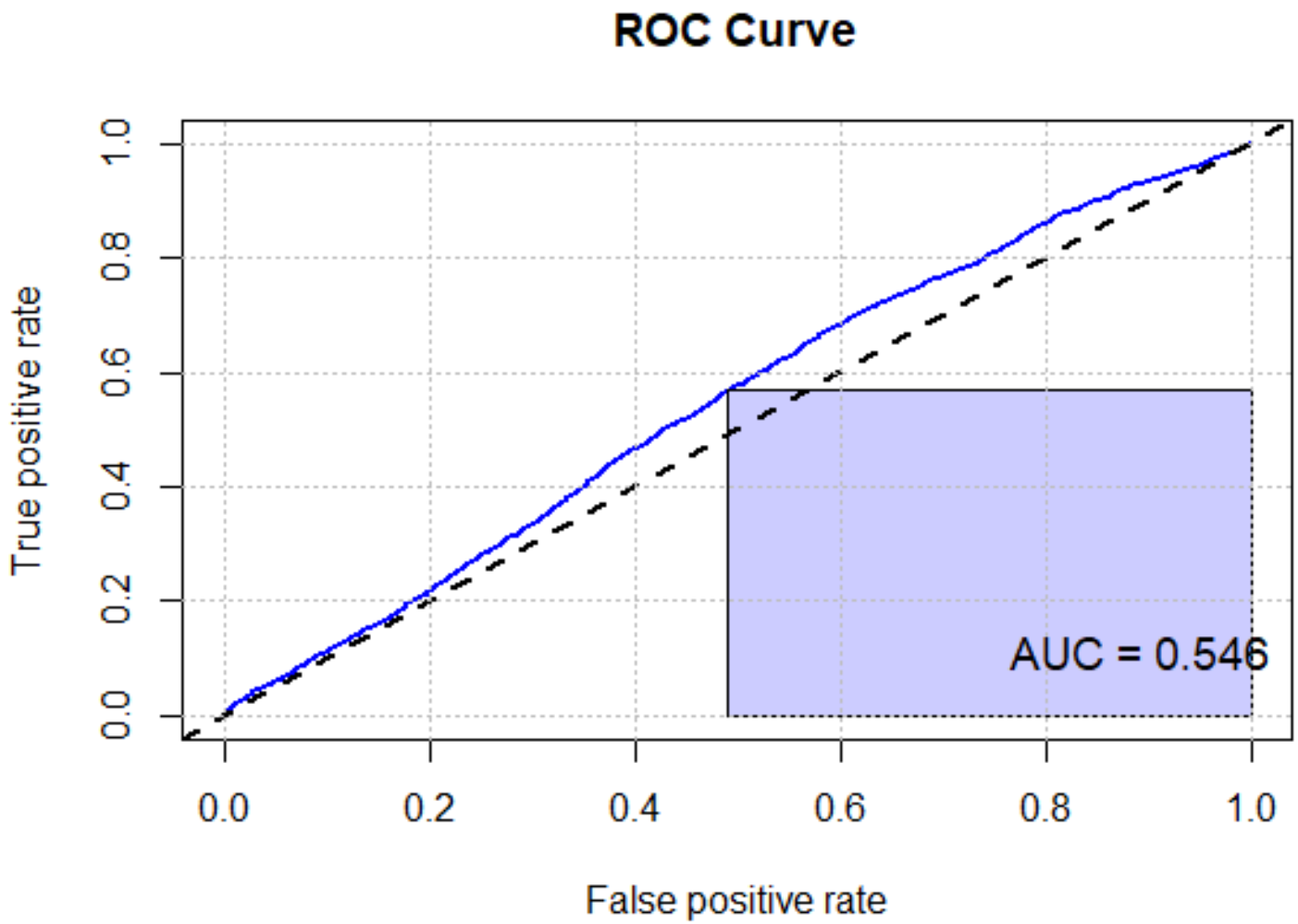
# Regresja logistyczna

- Budowa modelu: 10-krotna kroswalidacja + 1 zbiór testowy.

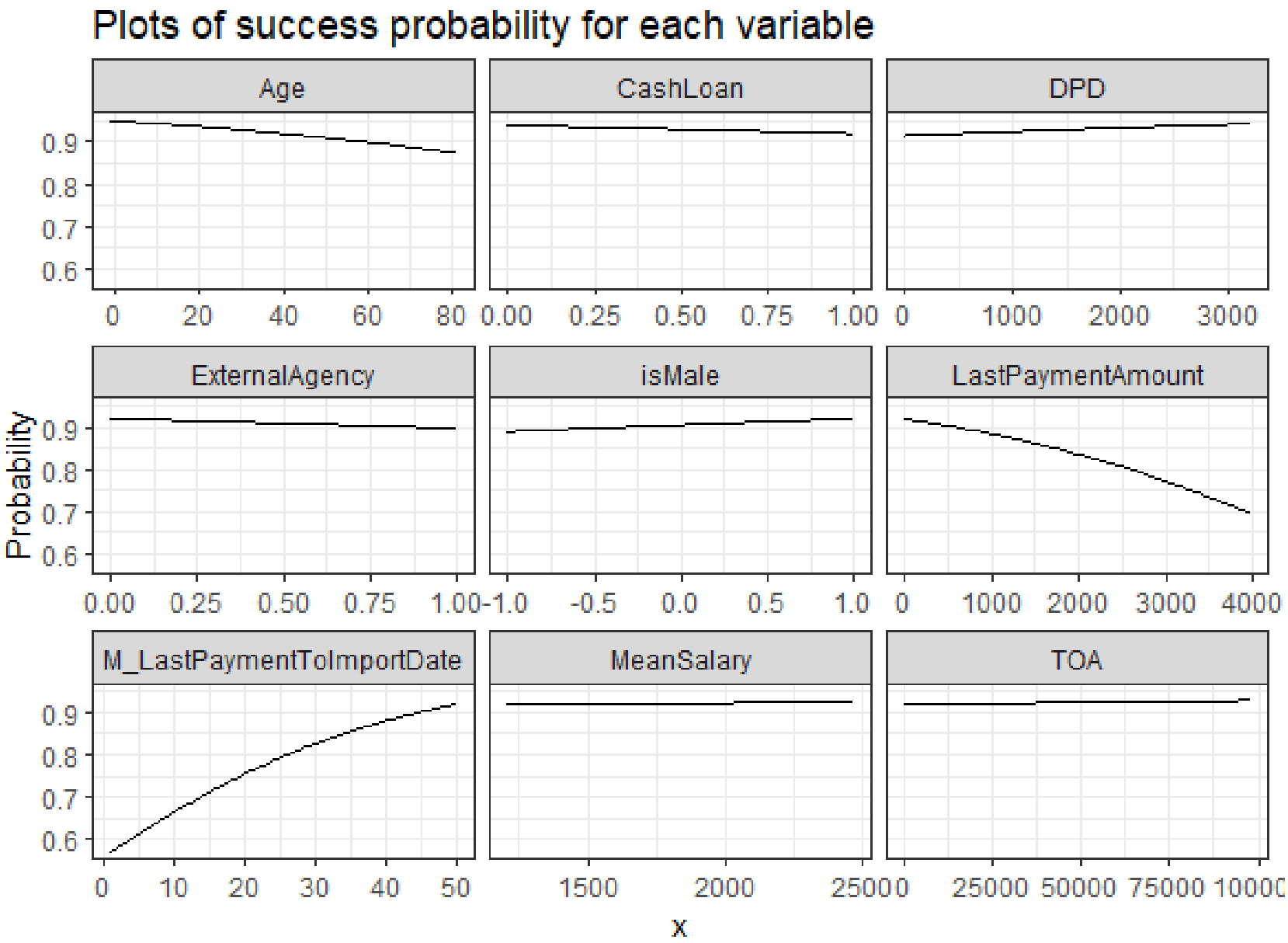


# Regresja logistyczna

- Selekcja: krzywa ROC.
- Wyniki: Tabela w R dla zbioru walidacyjnego.

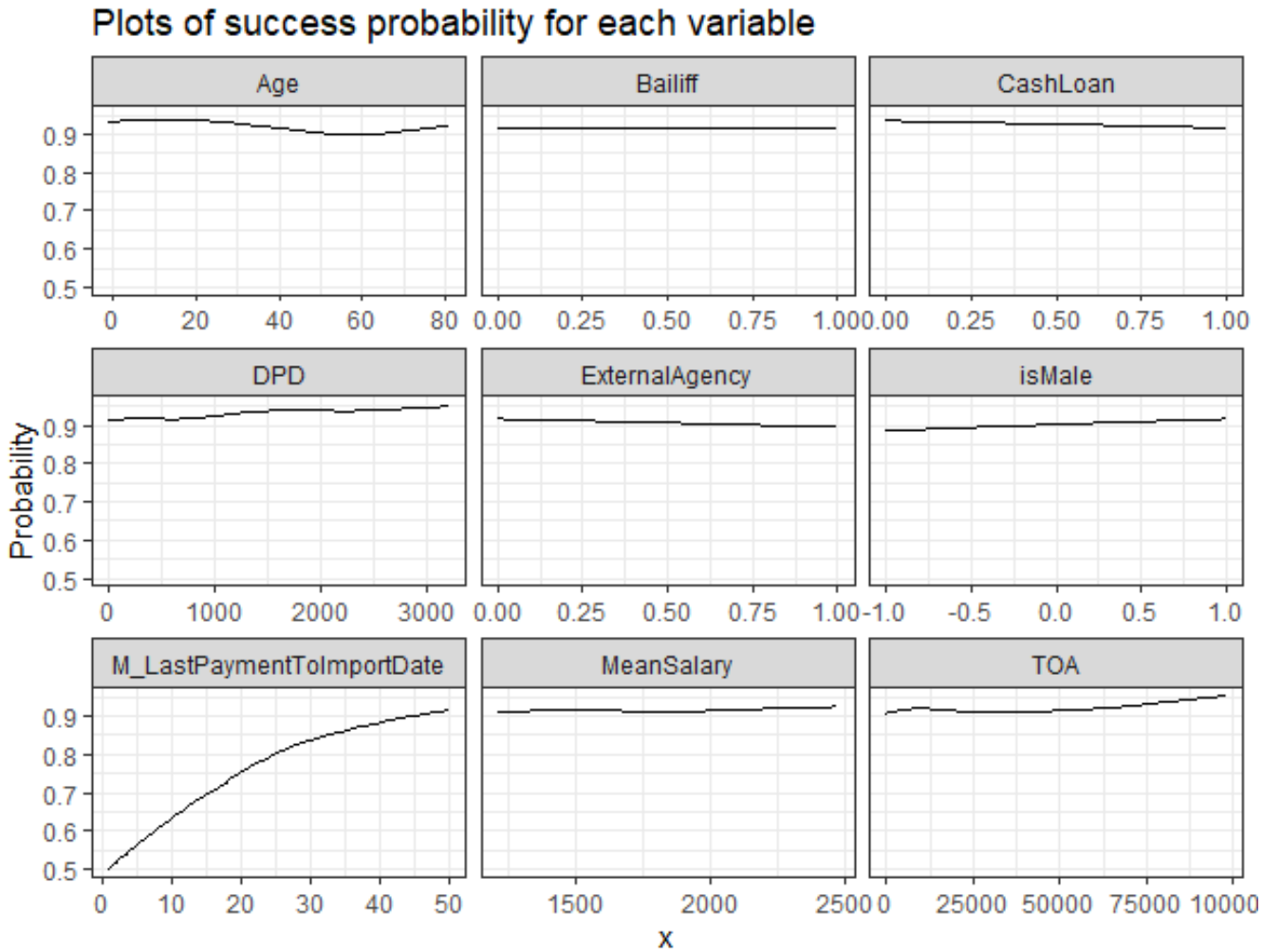


# Regresja logistyczna: istotność zmiennych



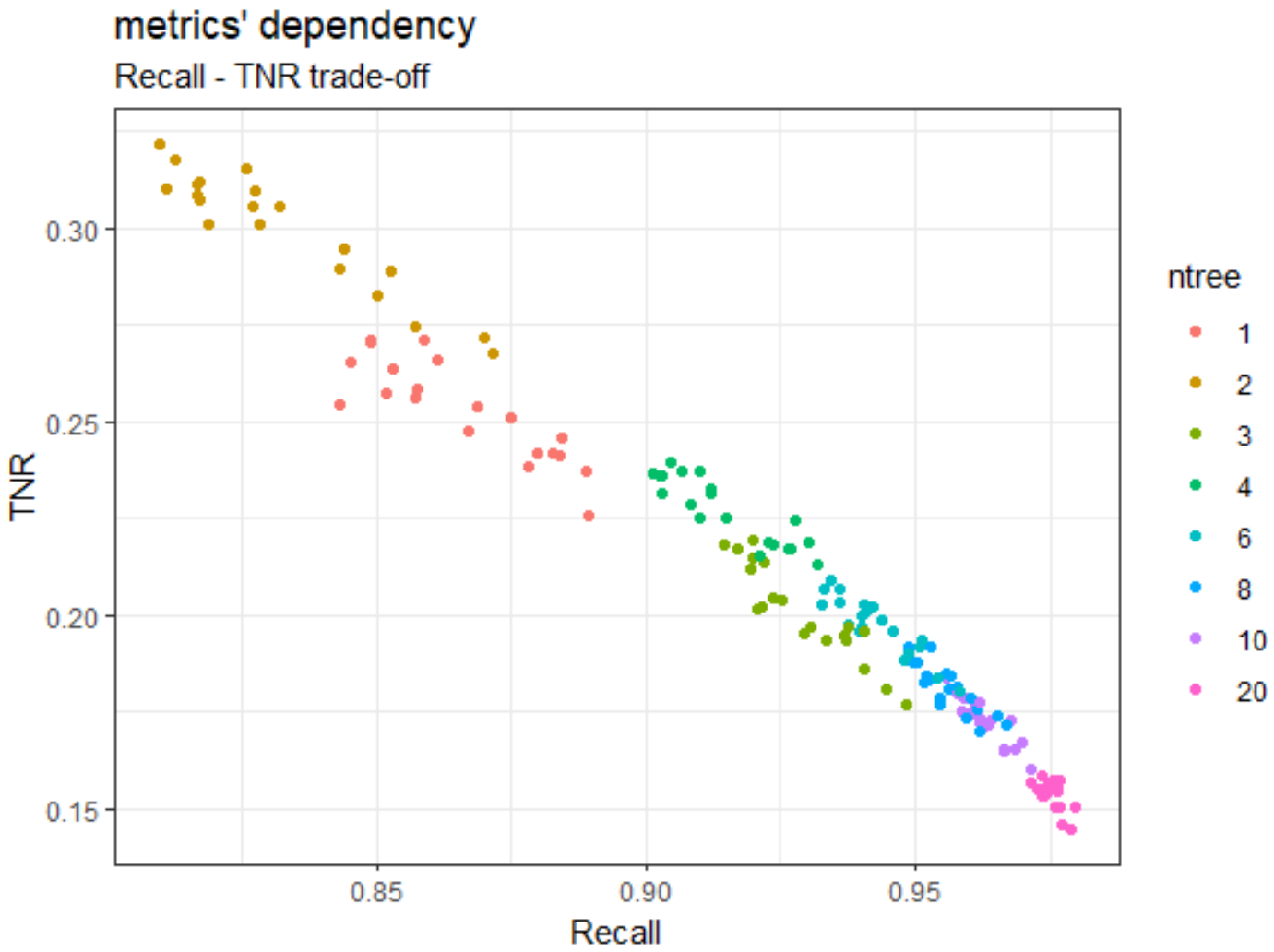
# GAM: istotność zmiennych

- Budowa modelu: 10-krotna kroswalidacja + 1 zbiór testowy.
- Wyniki: Tabela w R dla zbioru walidacyjnego.

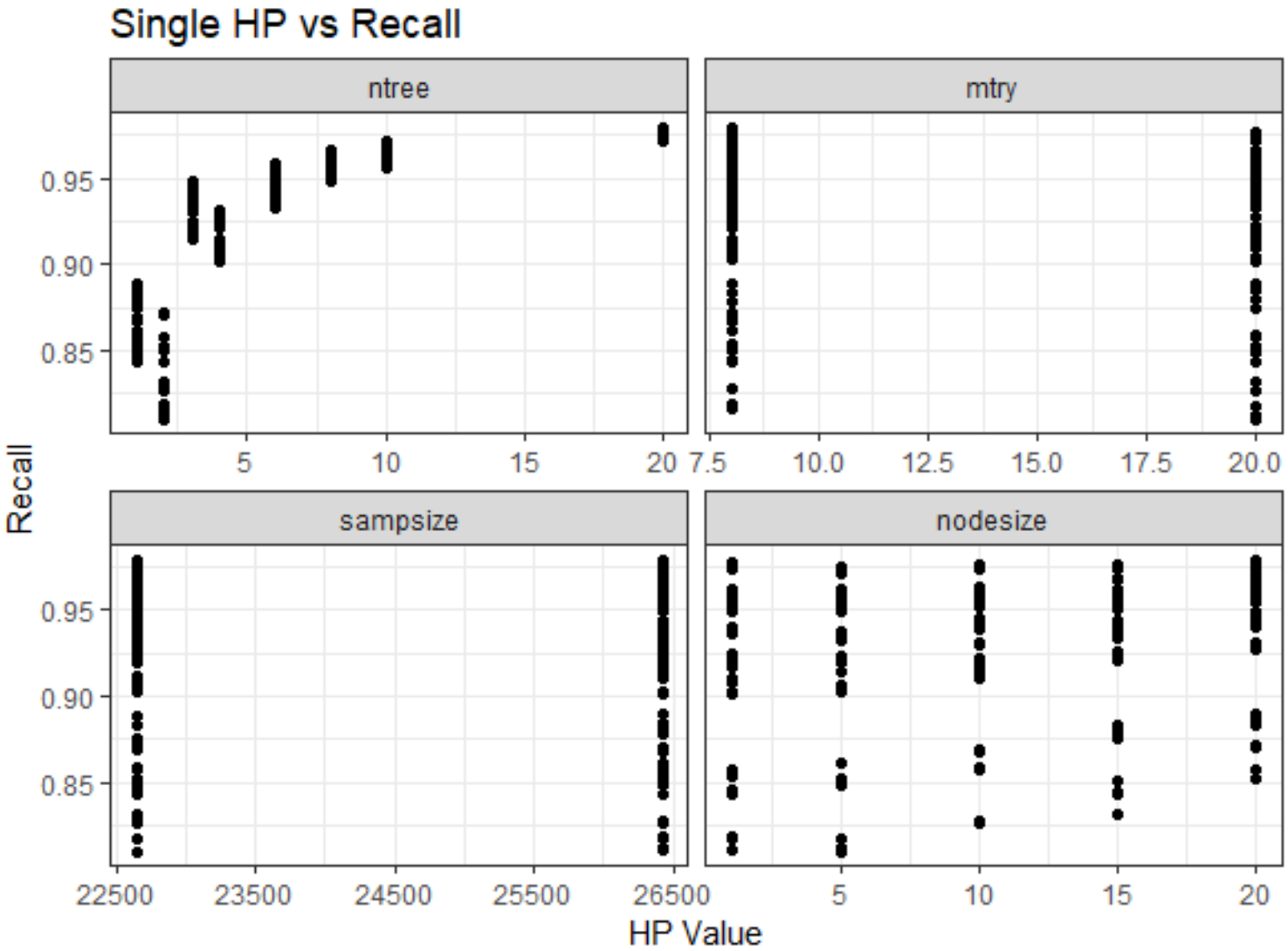


# Random Forest

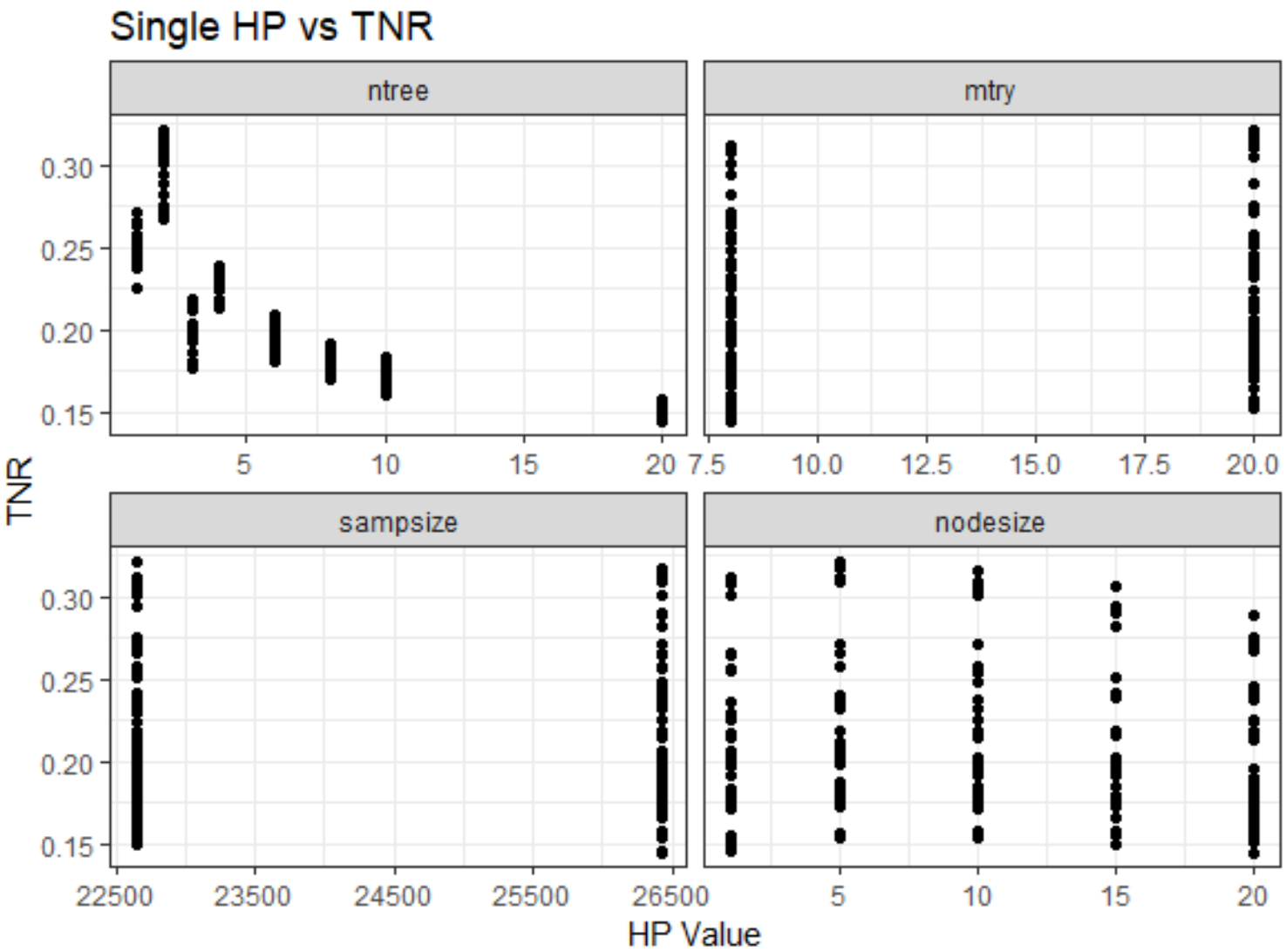
- Budowa modelu: Podział na równoliczne zbiory treningowy, walidacyjny, testowy.
- Wyniki: Tabela w R dla zbioru walidacyjnego.



# Random Forest

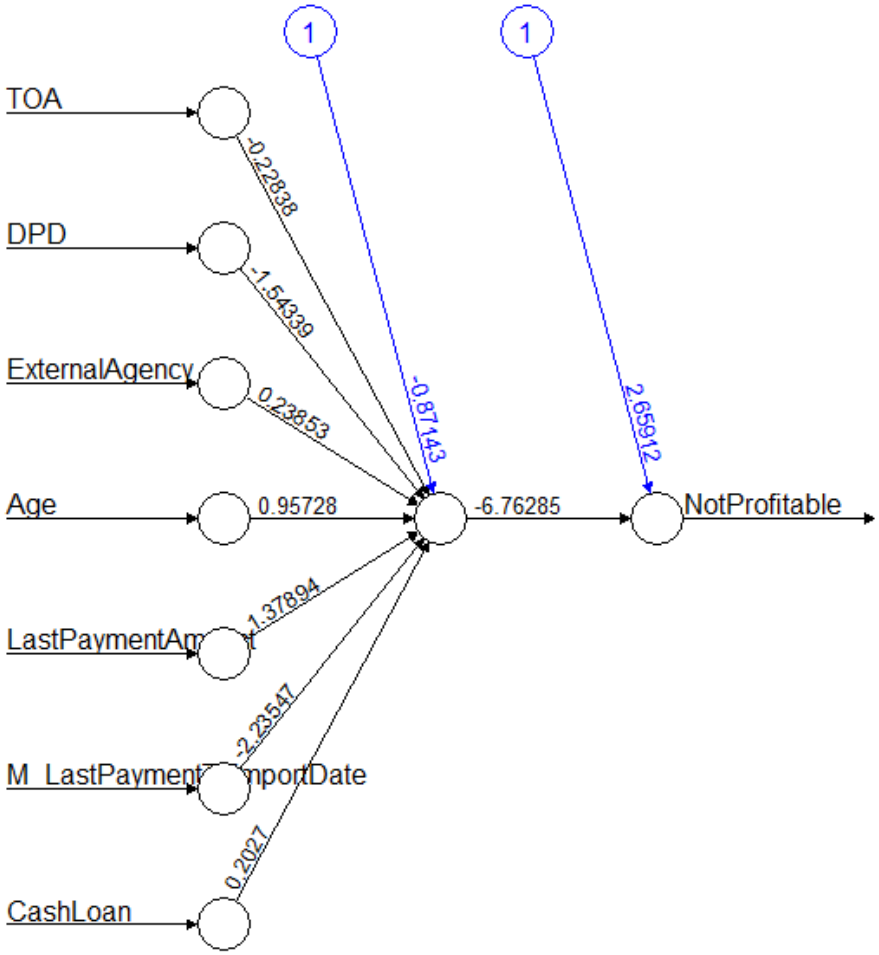


# Random Forest



# Neural network

- Budowa modelu: Podział na równoliczne zbiory treningowy i testowy; struktura lejka.
- Wyniki: Tabela w R dla zbioru walidacyjnego.




Error: 3346.81136 Steps: 2844

# Koniec