

# Przedstawienie wyników projektu #1 z przedmiotu Zaawansowane Metody Uczenia Maszynowego

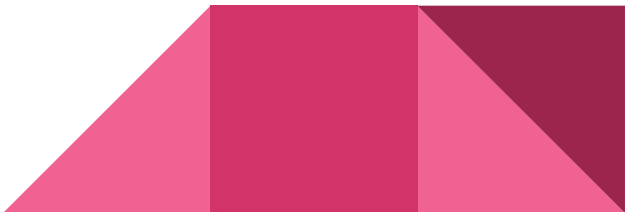
Piotr Podbielski

# Opis postępowania

1. Załadowanie bibliotek,
2. wczytanie danych,
3. podział danych treningowych na porcje w celu użycia kros-walidacji,
4. transformacja cech,
5. wytrenowanie modeli,
6. wyliczenie prawdopodobieństw dla zbioru z pliku testx.txt.



# 1. Załadowanie bibliotek

- `scikit-learn` – algorytmu dla uczenia maszynowego,
  - `category_encoders` – różne sposoby enkodowania zmiennych kategorycznych,
  - `matplotlib` – wizualizacja danych,
  - `numpy` – obliczenia na macierzach,
  - `pandas` – manipulacja ramkami danych,
  - `seaborn` – wysoko-poziomowy interfejs dla `matplotlib`, który pozwala w prosty sposób wizualizować różne ciekawe informacje,
  - `tqdm` – pasek postępu,
  - `xgboost` – gradient boosting na drzewach.
- 

## 2. wczytanie danych

Z plików `txt` załadowano zbiór zarówno treningowy, jak i ten, na którym ma zostać dokonana predykcja.



### 3. podział danych treningowych na porcje w celu użycia kros-walidacji

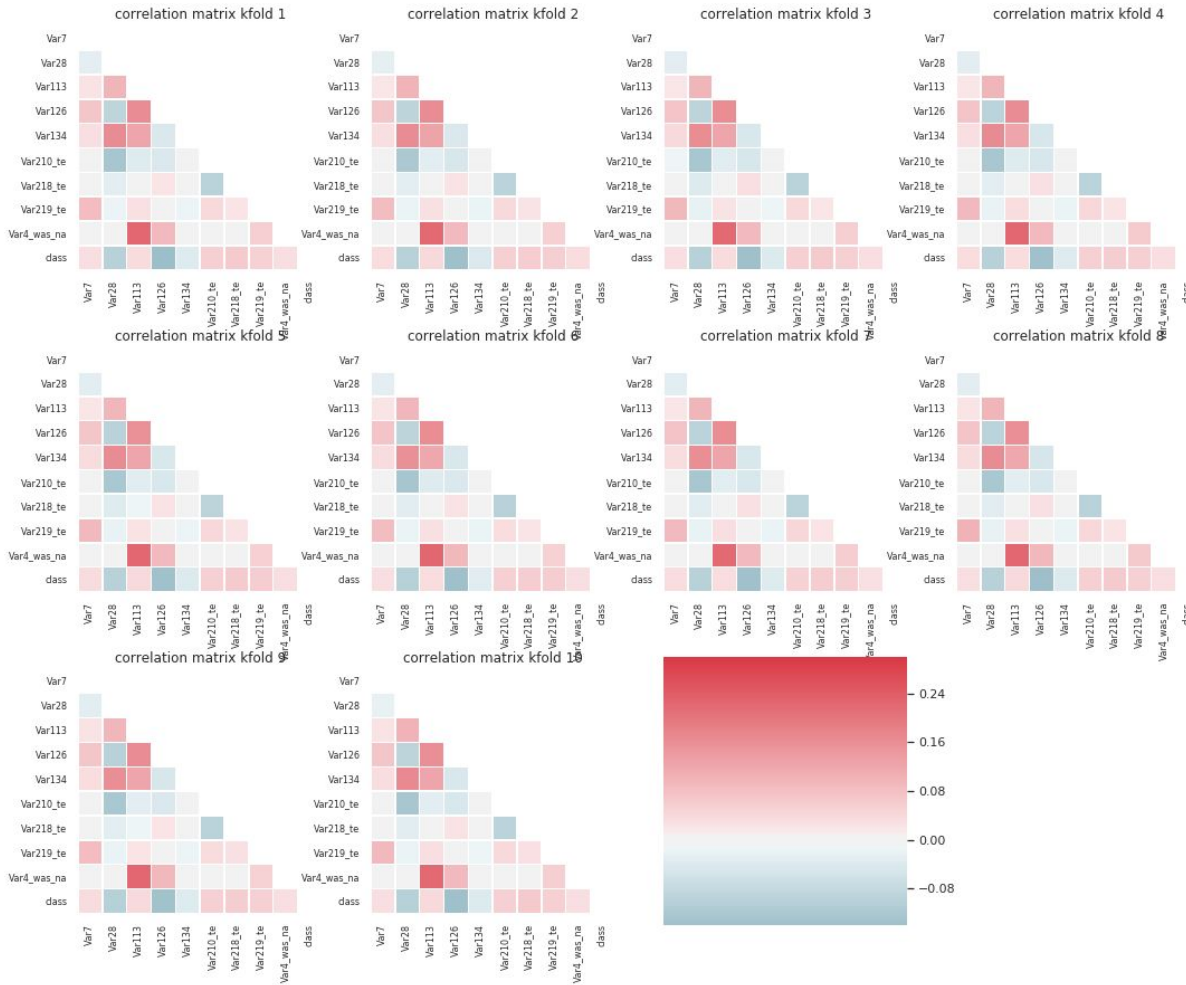
W celu późniejszego porównywania modeli zbiór treningowo podzielono na 10 części. Trenowanie i ewaluacja modelu odbywa się w przypadku kros-walidacji 10-krotnie. W każdym kroku pewne 9 części tworzy zbiór treningowe a pozostała część zbiór walidacyjny. Wyniki z wszystkich iteracji zostają uśrednione.



## 4. transformacja cech 1/2

1. Usunięcie cech ze wszystkimi wartościami będącymi brakami (NA),
2. usunięcie cech ze zmiennością równą 0 – cechy z tylko jedną wartością zostaną usunięte,
3. usunięcie cech kategorycznych (będącymi hashami), które mają ponad 30 wartości,
4. zakodowanie pozostałych cech kategorycznych sposobem `TargetEncoding`. Cechy są zastępowane mieszanką prawdopodobieństwa aposteriori zmiennej celu pod warunkiem wartości danej cechy i prawdopodobieństwa apriori zmiennej celu względem wszystkich danych treningowych,
5. zamiana cech z frakcją braków powyżej 0 na flagę czy dana wartość cechy miała brak (1) czy nie (0),
6. normalizacja danych (odjęcie średniej i podzielenie przez odchylenie standardowe),
7. usunięcie cech, których korelacja ze zmienną celu wynosi mniej niż 0.03,
8. usunięcie z pary skorelowanych cech jednej cechy, której współczynnik korelacji z drugą wynosi ponad 0.25.

# 4. transformacja cech 2/2



## 5. wytrenowanie modeli

- `LogisticRegression` – `solver` - `saga`, `C` (parametr regularyzacji) - `0.01`, `max_iter` - `1000000`, `penalty` (typ regularyzacji) - `l2`; zbiór treningowy - `0.262`, zbiór walidacyjny - **0.26**,
- `MLPClassifier` – `early_stopping` - `True`, `hidden_layer_size` - `(50)`, `activation` - `relu`, `max_iter` - `200`; zbiór treningowy - `0.294`, zbiór walidacyjny - **0.290**,
- `RandomForestClassifier` – `n_estimators` - `1000`, `min_samples_split` (minimalna liczba elementów, aby dokonać kolejnego podziału) - `200`, `max_depth` (maksymalna głębokość drzewa) - `3`; zbiór treningowy - `0.406`, zbiór walidacyjny - **0.401**,
- `XGBClassifier` – `objective` - `binary:logistic`, `n_estimators` - `50`; zbiór treningowy - `0.400`, zbiór walidacyjny - **0.396**.



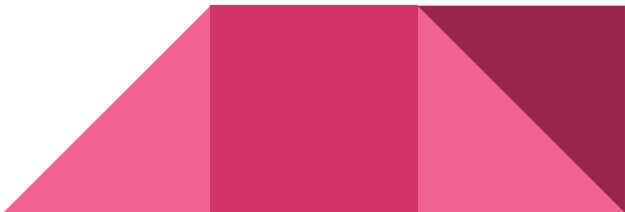
## 6. wyliczenie prawdopodobieństw dla zbioru z pliku testx.txt

Po wytrenowaniu 10 modeli (metoda kros-walidacji) na zbiorze treningowym dokonano predykcji dla danych udostępnionych przez prowadzącego (dla owych 10 modeli) i uśredniono prawdopodobieństwa klasy 1 dla każdego z modelu. Prawdopodobieństwa te zapisano do pliku `PIOPOD.txt`.



# Wnioski

Podczas rozwiązywania zadania natknęto się na kilka problemów, których rozwiązanie było kluczowe dla jego rozwiązania:

- poradzenie sobie z cechami, które mają dużo braków,
  - zakodowanie zmiennych kategorycznych,
  - poprawne zaimplementowanie metryki LIFT<sub>10</sub>,
  - niezbalansowanie zbioru treningowego,
  - zapewnienie niezależności zbiorów treningowych i walidacyjnych.
- 



Dziękuję!

*Czy są jakieś pytania?*