

Zastosowanie metod uczenia maszynowego

Klasyfikacja diabetyków

Aniela Brodziak

Informatyka Stosowana

1 WSTĘP

Projekt obejmuje stworzenie klasyfikacji diabetyków na podstawie danych ze strony Kaggle.

<https://www.kaggle.com/datasets/akshaydattatraykhare/diabetes-dataset>

Ten zestaw danych pochodzi pierwotnie z National Institute of Diabetes and Digestive and Kidney Diseases¹. Celem zestawu danych jest diagnostyczne przewidywanie, czy pacjent ma cukrzycę, na podstawie pewnych pomiarów diagnostycznych zawartych w zestawie danych. Wszyscy pacjenci tutaj są kobietami i mają co najmniej 21 lat.

2 OCENA PRZYDATNOŚCI DANYCH

2.1 RODZAJE ZMIENNYCH

W zbiorze danych wszystkie zmienne są typu numerycznego, z wyjątkiem zmiennej wyjściowej Outcome (kategoryczna).

Nazwa	Opis	Typ
pregnancies	number of pregnancies (NOPs),	numeryczna
glucose	plasma glucose concentration over 2-hours in an oral glucose tolerance test (Glucose),	numeryczna
bloodpressure	diastolic blood pressure (DBP) (mm Hg),	numeryczna
skinthickness	triceps skin fold thickness (TSFT) (mm),	numeryczna
insulin	2-hours serum insulin (μ U/ml) (Insulin),	numeryczna
bmi	and body mass index (BMI).	numeryczna
diabetespedigree	diabetes pedigree function (DPF),	numeryczna
age	age (in years),	numeryczna
outcome	study unit type (SUT) (0=non- diabetic, 1=diabetic),	kategoryczna

Po określeniu typów zmiennych przystąpiono do zaimportowania danych w portalu Azure, by przeprowadzić głębszą analizę danych.

¹ <https://www.niddk.nih.gov/>

2.2 ROZKŁAD CZĘSTOŚCI ZMIENNYCH

W zbiorze danych nie występują stałe, wartości niepowtarzalne ani zmienne monotoniczne.

2.3 STATYSTYKI OPISOWE I GRAFICZNE

	min	maks	średnia	odchylenie	wariancja	skośność	kurtoza
pregnancies	0	17	3,85	3,37	11,35	0,9	0,14
glucose	0	199	120,89	31,97	1,022	0,17	0,62
bloodpressure	0	122	69,11	19,36	374,65	-1,84	5,12
skinthickness	0	99	20,54	15,95	254,47	0,11	-0,53
insulin	0	846	79,8	115,24	13,281	2,26	7,13
bmi	0	67,1	31,99	7,88	62,16	-0,43	3,24
diabetespedigree	0,08	2,42	0,47	0,33	0,11	1,91	5,53
age	21	81	33,24	11,76	138,3	1,13	0,62
outcome	0	1	0,35	0,48	0,23	0,63	-1,6

Pregnancies

- **Tendencja centralna:** Średnia wynosi 3.85, co sugeruje, że większość kobiet w zbiorze danych miała około 4 ciążę.
- **Rozkład:** Skos wynosi 0.9, co wskazuje na prawoskośność. Większość wartości jest mniejsza, a wartości skrajnie większe (np. 17) ciągną rozkład w prawo.
- **Rozrzut:** Wariancja wynosi 11.35, co oznacza dość dużą zmienność liczby ciąż.

Glucose

- **Tendencja centralna:** Średnia to 120.89. Z uwagi na niewielką dodatnią skośność (0.17), rozkład jest zbliżony do symetrycznego.
- **Rozrzut:** Odchylenie standardowe to 31.97, co sugeruje znaczną zmienność poziomu glukozy w populacji.
- **Kurtosis (kurtosis):** Wartość 0.62 wskazuje na łagodną koncentrację danych wokół średniej.

Blood Pressure

- **Tendencja centralna:** Średnia wynosi 69.11, a skośność -1.84 wskazuje na lewoskośność, co oznacza, że więcej osób ma ciśnienie wyższe niż 69.
- **Rozrzut:** Duża wariancja (374.65) sugeruje rozproszenie wyników ciśnienia.
- **Symetria:** Lewoskośność wskazuje, że większość wyników jest większa od średniej.

Skin Thickness

- **Tendencja centralna:** Średnia 20.54, a skośność 0.11 wskazuje na niemal symetryczny rozkład.
- **Rozrzut:** Wariancja wynosi 254.47, co sugeruje duże zróżnicowanie.
- **Symetria:** Rozkład bliski normalnemu.

Insulin

- **Tendencja centralna:** Średnia wynosi 79.8, a skośność 2.26 wskazuje na silną prawoskośność. Większość osób ma niski poziom insuliny, z nielicznymi wartościami skrajnie wysokimi.
- **Rozrzut:** Duże odchylenie standardowe (115.24) i wysoka kurtoza (7.13) wskazują na obecność ekstremalnych wartości.

BMI

- **Tendencja centralna:** Średnia 31.99, a skośność -0.43 wskazuje na delikatną lewoskośność.
- **Rozrzut:** Odchylenie standardowe wynosi 7.88, co sugeruje umiarkowaną zmienność.
- **Symetria:** Rozkład jest niemal symetryczny.

Diabetes Pedigree

- **Tendencja centralna:** Średnia wynosi 0.47, a skośność 1.91 wskazuje na prawoskośność. Wyższe wartości (rzadsze) tworzą długi ogon.
- **Rozrzut:** Niewielkie odchylenie standardowe (0.33), co oznacza względnie małą zmienność.
- **Symetria:** Rozkład jest asymetryczny.

Age










- **Tendencja centralna:** Średnia 33.24, skośność 1.13 wskazuje na prawoskośność.
- **Rozrzut:** Wariancja 138.3 pokazuje znaczne zróżnicowanie wieku.
- **Symetria:** Rozkład jest umiarkowanie asymetryczny.

Outcome

- **Tendencja centralna:** Średnia wynosi 0.35. Jest to zmienna binarna (0 lub 1), gdzie 1 wskazuje na obecność cukrzycy, a 0 na jej brak.
- **Rozkład:** Rozkład bliski symetrycznemu.
- **Rozrzut:** Odchylenie standardowe 0.48 wskazuje, że zmienna jest dobrze rozproszona.

Podsumowując:

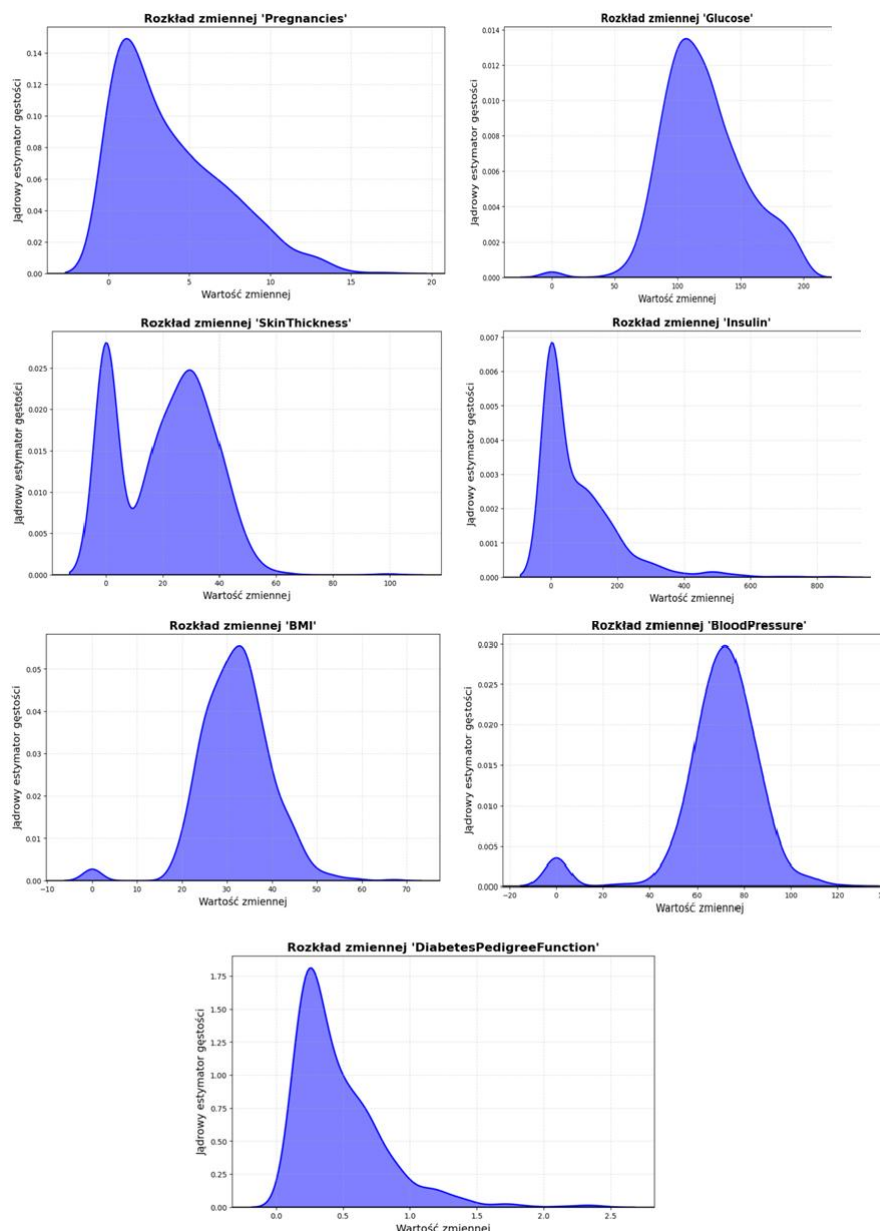
- **Symetria:** Większość zmiennych (np. glucose, skinthickness, bmi) jest zbliżona do symetrycznego rozkładu, ale insulin i diabetespedigree wykazują silne prawoskośności.
- **Rozrzut:** Zmienność poziomu insuliny i ciśnienia krwi jest wyraźnie wysoka.
- **Tendencja centralna:** W przypadku zmiennych skośnych (np. insulin, age) lepszym miernikiem niż średnia będzie mediana.

Feature	Count	Unique Value Count	Missing Value Count	Min	Max	Mean	Mean Deviation	1st quantile
								
Pregnancies	768	17	0	0	17	3.845052	2.77162	1
Glucose	768	136	0	0	199	120.894531	25.181793	99
BloodPressure	768	47	0	0	122	69.105469	12.639425	62
SkinThickness	768	51	0	0	99	20.536458	13.659627	0
Insulin	768	186	0	0	846	79.799479	84.505079	0
BMI	768	248	0	0.0	67.1	31.992578	5.84227	27.3
DiabetesPedigreeFunction	768	517	0	0.078	2.42	0.471876	0.247309	0.24375
Age	768	52	0	21	81	33.240885	9.586405	24
Outcome	768	2	0	0	1	0.348958	0.454373	0

Największą różnorodność wykazują zmienne **BMI**, **DiabetesPedigreeFunction**, oraz **Insulin**, co wskazuje na ich potencjalną zmienność i znaczenie w analizie.

Zmienna **Outcome** jest binarna i dobrze przygotowana do problemów klasyfikacji (np. analiza występowania cukrzycy).

Na pierwszy rzut oka żadne dane nie wymagają uzupełnienia, ale potencjalne transformacje (np. normalizacja, standaryzacja) mogą być potrzebne dla zmiennych o wysokiej różnorodności lub dużym zakresie wartości.



2.4 BRAKUJĄCE WARTOŚCI LUB NIEPRAWIDŁOWE DANE

Po zastosowaniu modułu Summarize Data w Azure Designer, wydaje się że dataset nie zawiera wartości pustych NULL, jednak pojawiają się wartości równe 0, które w niektórych zmiennych mogą sugerować brak danych lub nieprawidłowy pomiar.

Kolumnami w których 0 nie oznacza braku danych będą kolumny [pregnancies], [outcome], [diabetespedigree]

Kolumny w których 0 musi oznaczać brak danych to wszystkie pozostałe, gdyż wartość 0 w przypadku glukozy, ciśnienia, grubości skóry, insuliny, BMI i wieku nie może być równa 0.

Do zidentyfikowania braków w danych użyto skryptu Python i zamieniono wartości równe 0 w podanych kolumnach na NaN.

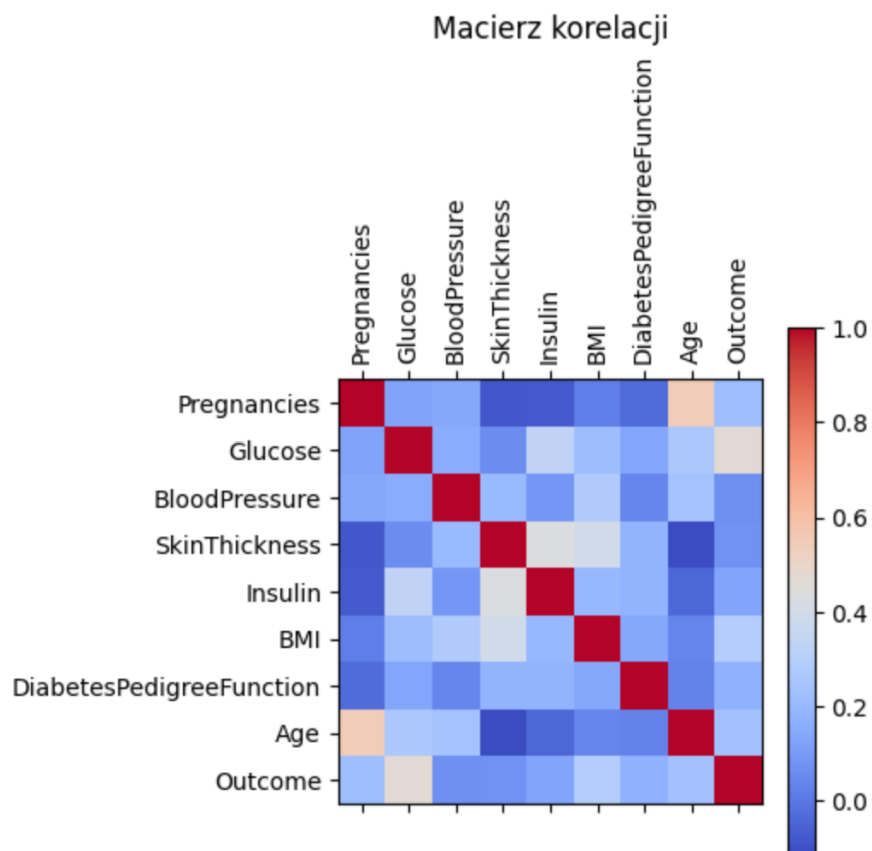
768 obserwacji	brakujące wartości	brakujące wartości (%)	sugerowana metoda zastąpienia pustych wartości
pregnancies	-	-	-
glucose	5	1%	mediana
bloodpressure	35	5%	mediana
skinthickness	227	30%	KNN/MICE
insulin	374	49%	KNN/MICE
bmi	247	32%	KNN/MICE
diabetespedigree	-	-	-
age	-	-	-
outcome	-	-	-

Największe braki wartości występują w kolumnie insulin bo jest to aż 49%, skinthickness 30%, bmi (32%).

Zdecydowano się na zastąpienie pustych wartości w tych kolumnach, zamiast usuwania obserwacji, aby uniknąć utraty danych, a także z uwagi na już i tak mały zbiór obserwacji w zbiorze danych (768).

2.5 KORELACJE

W Jupyter zaimplementowano skrypt, który wygenerował macierz korelacji dla zmiennych.



- **BloodPressure** (0.065068): Niska korelacja — może być mniej istotna.
- **SkinThickness** (0.074752): Niska korelacja — podobnie jak BloodPressure, może mieć mniejsze znaczenie.

Z matrycy korelacji między zmiennymi można zauważyć:

- **Silną korelację między Insulin a SkinThickness (0.436783)** oraz BMI (0.392573).
- **Age i Pregnancies** są skorelowane (0.544341), co może sugerować powiązanie, ale niekoniecznie wymaga usunięcia.

Jeśli zmienne są silnie współliniowe, jedna z nich może być usunięta, aby uprościć model.

2.6 OUTLIERY

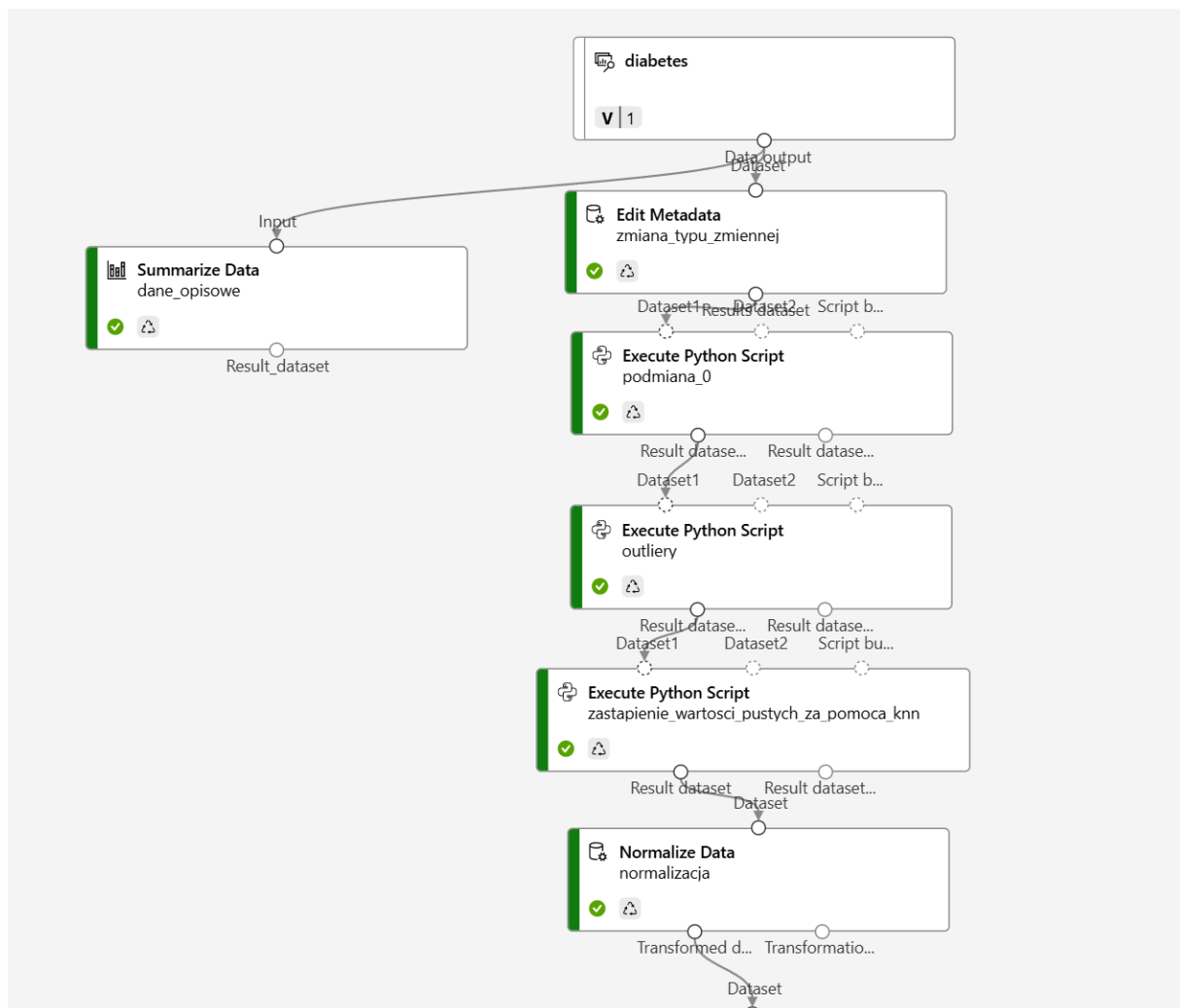
Przy pomocy Jupyter, za pomocą skryptu wygenerowano wyniki możliwych outlierów w poszczególnych kolumnach, by zweryfikować przyjęte granice w ZScore i IQR w późniejszym etapie przekształcania outlierów.

Liczba outlierów w każdej kolumnie (Z-Score > 3):

Glucose	5
BloodPressure	35
SkinThickness	1
Insulin	18
BMI	14
Age	5

3 WDROŻENIE MODELU

3.1 PRZYGOTOWANIE DANYCH



- Edit metadata – ustawienie Outcome jako kategorię
- Execute Python Script (podmiana_0) – zamiana 0 na NaN za pomocą skryptu Python
- Execute Python Script (outliery)

Zastąpiono outliery następującymi metodami:

Z-Score: Stosowana dla zmiennych z symetrycznym rozkładem, np. Glucose, BloodPressure, BMI, gdzie outliery są wartościami skrajnie oddalonymi od średniej.

Zastępuje wartości odstające odpowiednio:

Gdy wartość przekracza granicę górną, zastępuje ją $\text{mean} + \text{threshold} \times \text{std}$

Gdy wartość przekracza granicę dolną, zastępuje ją $\text{mean} - \text{threshold} \times \text{std}$

IQR: Stosowana dla zmiennych z rozkładem asymetrycznym, np. Pregnancies, Insulin, gdzie metoda oparta na kwantylach lepiej wychwytywa outliery.

Oblicza pierwsze (Q1) i trzecie (Q3) kwartyle oraz IQR ($\text{IQR} = Q3 - Q1$)

Wyznacza dolną i górną granicę dla wartości odstających:

Dolna granica: $Q1 - \text{multiplier} \times IQR$

Górna granica: $Q3 + \text{multiplier} \times IQR$

Zastępuje wartości odstające wartościami granicznymi:

Wartości mniejsze od dolnej granicy są zastępowane wartością dolnej granicy.

Wartości większe od górnej granicy są zastępowane wartością górnej granicy.

- Execute Python Script (zastąpienie_wartosci_pustych)

Zdecydowano, by do zastąpienia wartości NaN użyć KNN z uwagi na sporą ilość pustych obserwacji, przekraczającą 10%, więc zastąpienie wartości pustych medianą, średnią czy usunięciem obserwacji byłoby niemięarodajne. Algorytm wykorzystuje wartości sąsiednich próbek do imputacji brakujących danych.

Imputacja jest oparta na wzorcach w danych, co pozwala lepiej zachować ich spójność.

Użycie sąsiednich wartości redukuje ryzyko wprowadzania nieadekwatnych wartości w brakujących miejscach.

KNN Imputer został wybrany w przypadku małego zbioru danych, ponieważ:

Jest prostszy, szybszy i bardziej odpowiedni przy niewielkiej liczbie obserwacji.

Wykorzystuje lokalne wzorce w danych, co działa efektywnie w małych zbiorach.

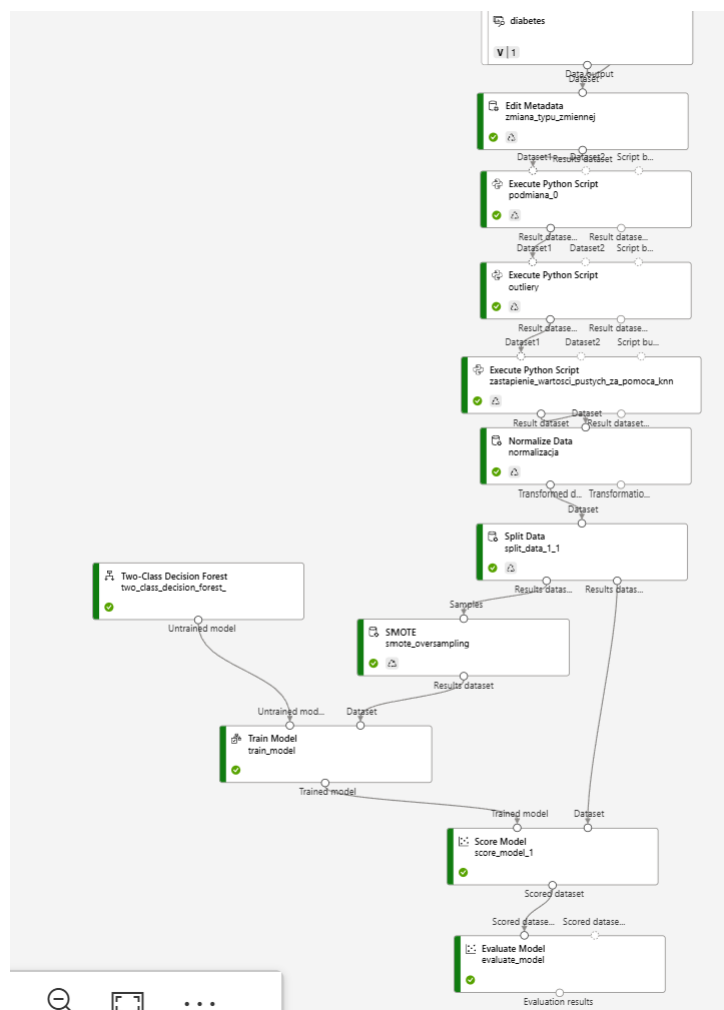
Nie wymaga budowy modeli predykcyjnych, co minimalizuje ryzyko przetrenowania.

Wyniki są łatwiejsze do interpretacji i weryfikacji w małych zbiorach danych.

Z kolei **MICE** jest bardziej zaawansowany i lepiej sprawdza się w większych zbiorach danych, gdzie zmienne są silnie skorelowane, a liczba obserwacji pozwala na skuteczne iteracyjne modelowanie. W przypadku małych zbiorów, MICE może być nieefektywny i przynieść niepewne rezultaty.

- Normalize Data – znormalizowano dane za pomocą MinMax

4 WYBÓR ALGORYTMÓW



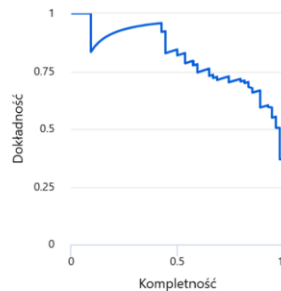
- Split data – podzielono dane na treningowe (80%) i testowe (20%)
- Smote – zastosowano oversampling z uwagi na małą ilość danych i nierównowagę w danych, podczas eksperymentowania zauważono, że Smote znacznie poprawiło wyniki modelu poprzez zwiększenie reprezentacji klasy mniejszościowej, usprawnienie procesu uczenia, poprawę balansu między klasami. Smote zastosowano tylko dla danych treningowych by uniknąć **wycieku danych** między zestawami treningowym a testowym i **zachować wiarygodność wyników**, pozwalając zmierzyć zdolność modelu do generalizacji na rzeczywiste, niezrównoważone dane.
- Two-Class Decision Forest – ostatecznie wybrano ten algorytm, gdyż osiągał najlepsze wyniki w trakcie eksperymentowania. Jest to algorytm dobrze radzący sobie z różnorodnymi danymi, oferuje wysoką dokładność i odporność na szum oraz brakujące dane. W połączeniu z SMOTE, świetnie nadaje się do radzenia sobie z niezrównoważonymi zbiorami danych.
- Przy pomocy Tune Model Hyperparameters automatycznie dobrano parametry dla algorytmu. Zastąpiono go modułem Train Model i ustawiono ręcznie najlepsze parametry dla algorytmu.

• Wynikowy zestaw danych (lewy port)

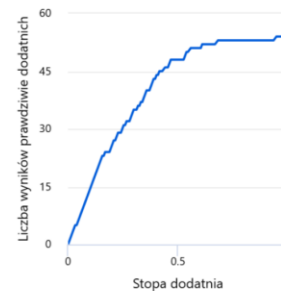
Krzywa ROC



Krzywa precyzji w funkcji czułości (precision-recall)



Krzywa przyrostu



Próg

Dokładność 0.792

Dokładność 0.657

Kompletność 0.852

Miara F1 0.742

Pole pod krzywą 0.885

	Rzeczywiste	
	1.0	0.0
Prognostowane 1.0	46	24
Prognostowane 0.0	8	76

- **Dokładność (Accuracy): 0.792**

- Model prawidłowo klasyfikuje około **79,2% wszystkich obserwacji**.
- Jest to wskaźnik ogólnej skuteczności modelu, ale nie mówi, jak model radzi sobie z poszczególnymi klasami.

- **Czułość (Recall/Kompletność): 0.852**

- Model wykrywa **85,2% rzeczywistych przypadków pozytywnych (Outcome = 1)**.
- Wysoka wartość tego wskaźnika oznacza, że model dobrze identyfikuje osoby chore.

- **Precyzja (Precision): ~0.657**

- Spośród obserwacji zaklasyfikowanych jako pozytywne, **65,7% jest rzeczywiście pozytywnych**.
- Jest to ważne, jeśli błędne diagnozy mają wysoką cenę (np. fałszywe alarmy).

- **Miara F1: 0.742**

- Miara F1 równoważy precyzję i czułość, sugerując, że model dobrze radzi sobie w balansowaniu między fałszywymi pozytywnymi a fałszywymi negatywnymi wynikami.

- **Pole pod krzywą ROC (AUC): 0.885**

- Wysoka wartość AUC wskazuje, że model dobrze rozdziela klasy (Outcome = 0 i Outcome = 1).

Model ma wysoki odsetek prawidłowych pozytywnych wyników (TP), ale generuje również znaczną liczbę fałszywych pozytywnych wyników (FP).

FN (przeoczone osoby chore) są relatywnie niewielkie, co sugeruje, że model dobrze identyfikuje osoby chore.

Obecny model ma podobną dokładność jak eksperyment przeprowadzony na surowych danych (0.799 vs. 0.792). Jednak wzrost dokładności wynika głównie z poprawy klasyfikacji negatywnych przypadków (TN).

Model po transformacjach znacznie lepiej identyfikuje osoby chore (0.852 vs. 0.519). Eksperyment na surowych danych przeoczył aż 26 chorych osób (FN), co jest dużym problemem w kontekście wykrywania chorób.

Model ma niższą precyzję (0.848 vs. 0.657). To oznacza, że przewidywania chorych osób są bardziej trafne (mniej fałszywych alarmów - FP).

Model lepiej balansuje precyzję i czułość ($F1 = 0.742$ vs. 0.644).

Końcowy model ma lepsze AUC (0.885 vs. 0.865), co oznacza, że lepiej rozdziela klasy (chory/zdrowy).

5 PODSUMOWANIE

Końcowy model wykazuje **lepszą równowagę między precyzją a czułością**:

- Choć surowe dane dały wyższą precyzję, model na przetworzonych danych miał lepszą **czułość**, co w kontekście medycznym jest bardziej istotne.
- Wyższa czułość oznacza, że mniej chorych osób zostało błędnie zaklasyfikowanych jako zdrowe (False Negatives).

Chociaż model na surowych danych osiąga wyższą precyzję, ma znacznie gorszą czułość:

- Wysoka liczba False Negatives (FN) oznacza, że wiele osób chorych jest błędnie klasyfikowanych jako zdrowe.
- W medycynie **czułość** jest często ważniejsza niż precyzja, ponieważ pominięcie chorego przypadku ma poważniejsze konsekwencje.
- Balans między precyzją a czułością: Model lepiej identyfikuje osoby chore (wyższa czułość), co jest kluczowe w medycynie.
- Lepsza jakość danych: Imputacja braków, eliminacja outlierów i normalizacja poprawiają generalizację modelu.
- Zrównoważenie klas: Dzięki SMOTE model radzi sobie lepiej z niebalansowanymi danymi.
- Powtarzalność: Pipeline preprocessingu jest bardziej uniwersalny i można go stosować w innych zbiorach danych.