Projekt predictora do klasyfikacji nowotworów piersi

Jakub Jasonek

2020

1 Wstep

2 Opis danych

Nowotwór piersi jest najczęstszym typem nowotworu kobiet na świecie. Co roku w Unii Europejskiej diagnozuje się 350tysięcy nowych przypadków oraz 100 zgonów związanych z tymi nowotworami. Jak w większości przypadków nowotworów, największe szanse leczenia daję wczesna diagnoza i możliwie najszybsze rozpoczęcie leczenia. W związku z tym konieczne jest opracowywanie skuteczniejszych metod diagnostycznych.

2.1 Źródło danych

Zbiór danych do analizy został pobrany ze strony: `https://www.kaggle.com/merishnasuwal/breast-cancer-prediction-dataset`
Oryginalnie jest to zbiór zebrany przez Uniwetrsytet w Wisconsin.

2.2 Opis zbioru

```
[5 rows x 6 columns]
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 6 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   mean_radius      569 non-null    float64
 1   mean_texture     569 non-null    float64
 2   mean_perimeter   569 non-null    float64
 3   mean_area        569 non-null    float64
 4   mean_smoothness  569 non-null    float64
 5   diagnosis        569 non-null    int64
dtypes: float64(5), int64(1)
memory usage: 26.8 KB
None
```

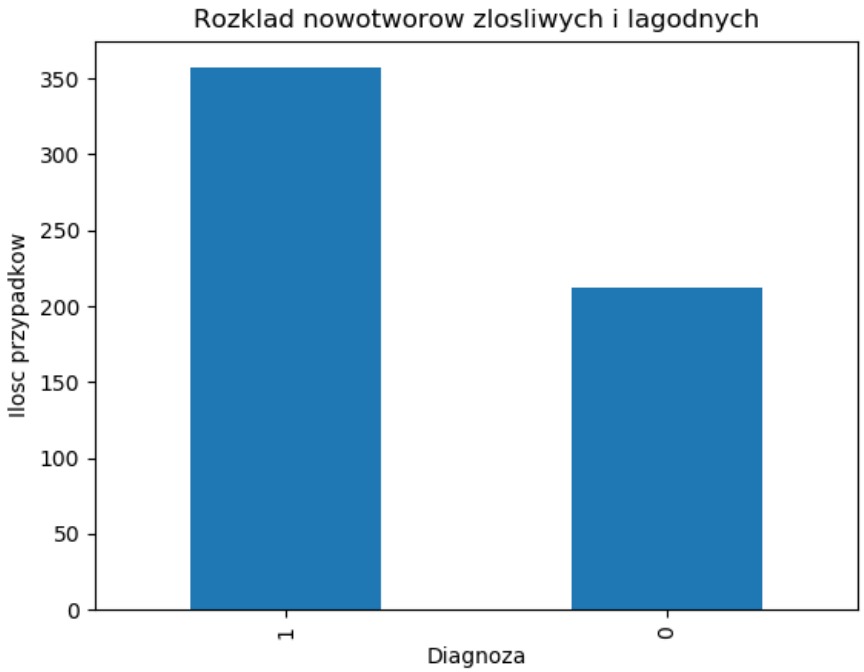
W zbiorze danych umieszczono następujące parametry:

- Diagnoza - określa jak guz został zidentyfikowany (1 - złośliwy, 0 - łagodny)
- Promień guza - średnia odległość od środka guza do jego obwodu
- Tekstura - odchylenie standardowe koloru w skali szarości
- Obwód - określa rozmiar guza
- Powierzchnia - określa wielkość guza
- Gładkość - średnia odchyleń od długości promienia guza

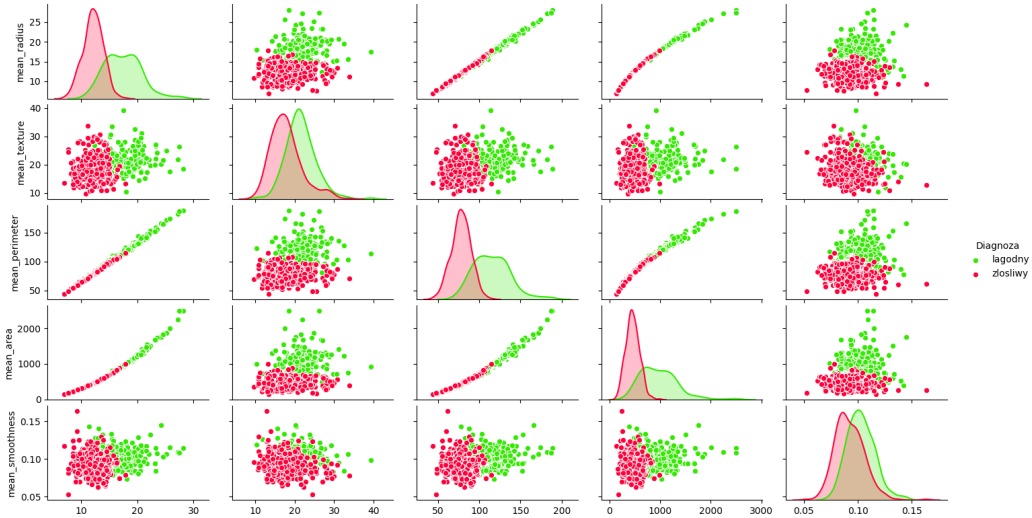
Zbiór danych zawiera 569 pozycji.

3 Analiza danych

Jako zmienną celu przyjęto parametr "diagnosis"dzielący zbiór danych na nowotwory złośliwe (diagnosis = 1) oraz łagodne (diagnosis = 0). Zbiór zawiera 62% przypadków nowotworów złośliwych (357) oraz 38% nowotworów łagodnych, więc jest wystarczająco zbalansowany.



Podczas analizy wykazano, że część parametrów, tzn. promień nowotworu, jego powierzchnia i obwód, jest ze sobą skorelowana zależnościami (co zrozumiałe, ponieważ to są parametry geometryczne powiązane znanymi zależnościami). W projekcie zdecydowano zastosować dwa modele: z wykorzystaniem wszystkich parametrów i z pominięciem parametrów skorelowanych, w celu zweryfikowania który model pozwala na lepsze dopasowanie.



Wektor parametrów został znormalizowany z wykorzystaniem funkcji z biblioteki sklearn.

```
# normalizacja wektorow
# L1
X_norm_l1_train = sklearn.preprocessing.normalize(train, norm="l1")
X_norm_l1_test = sklearn.preprocessing.normalize(test, norm="l1")
X_norm_l1 = sklearn.preprocessing.normalize(feature, norm="l1")

# L2
X_norm_l2_train = sklearn.preprocessing.normalize(train, norm="l2")
X_norm_l2_test = sklearn.preprocessing.normalize(test, norm="l2")
X_norm_l2 = sklearn.preprocessing.normalize(feature, norm="l2")
```

4 Modelowanie danych

4.1 Opis metody

W projekcie wykorzystano metodę K najbliższych sąsiadów (KNN). Jest to metoda klasyfikacyjna, w której na podstawie zbioru uczącego znanych, poprawnie sklasyfikowanych obiektów tworzy się klasyfikator do prognozowania wartości obiektów tej samej klasy. Zbiór uczący składa się ze zbioru znanych obiektów do których przypisane są wektory zmiennych opisujących i zmienną celu. Algorytm mapuje obiekty na wielowymiarową przestrzeń zmiennych opisujących, następnie przypisuje badanemu obiektowy wartość zmiennej celu na podstawie najbliżych obiektów ze zbioru uczącego. Odległość między obiektami oznacza wykorzystanie pewnej metryki np. metryki euklidesowej.

W ocenie klasyfikatora wykorzystano funkcję **fit_classifier()** udostępnioną na wykładzie.

Kolejno zbadano:

- klasyfikator z nienormalizowanym wektorem danych
- klasyfikator z nienormalizowanym, zredukowanym wektorem danych
- klasyfikator z normalizowanym L1 wektorem danych
- klasyfikator z normalizowanym l2, zredukowanym wektorem danych
- klasyfikator z normalizowanym L1 wektorem danych
- klasyfikator z normalizowanym l2, zredukowanym wektorem danych

Wyniki przedstawiono w rozdziale 5.

4.2 Optymalizacja Modelu KNN

Klasyfikator KNN dostępny w bibliotece sklearn przyjmuje parametry zmieniające działanie algorytmu. Parametry te to:

- **n_neighbors** - określający ilość sąsiadów wykorzystywanych przy klasyfikacji

- **weight** - określający w jakiej proporcji odległość najbliższych sąsiadów wpływa na klasyfikację. Dostępne opcje to: **uniform** - wszystkie punkt mają taką samą wagę oraz **distance** - waga sąsiądów jest proporcjonalna do ich odległości od klasyfikowanego punktu.
- **algorithm** - dobiera domyślny algorytm podawania danych
- **leafsize** - określa sposób budowania i przeszukiwana zbioru.
- **p** - określa jaka metryka jest wykorzystana do liczenia odległości między punktami. Dostępne opcje to: **2** - metryka euklidesowa, **1** - metryka manhattan.
- **metric** - określa wskaźnik odległości używany dla drzewa.
- **metric_params** - wartość domyślna None.
- **n_jobs** - liczba równoległych wątków procesora wykorzystana do budowania modelu.

W procesie optymalizacji modelu wykorzystano trzy parametry: n_neightbors, p oraz weights. W tym celu skorzystano z klasy **GridSearchCV()**.

```
def model_optymalization(x, y):
    search_grid = [
        {
            'weights': ['uniform', 'distance'],
            'p': [1, 2],
            'n_neighbors': [5, 10, 15, 20]
        }
    ]

    scorer = {'acc': 'accuracy', 'f1': 'f1', 'prec': 'precision', 'rec': 'recall'}

    search_func = GridSearchCV(estimator=sklearn.neighbors.KNeighborsClassifier(),
                               param_grid=search_grid,
                               scoring=scorer,
                               n_jobs=-1, iid=False, refit='acc', cv=5)
    search_func.fit(x, y)
    print(search_func.best_estimator_)
    print(search_func.best_params_)
    print(search_func.best_score_)
    results = pd.DataFrame(search_func.cv_results_)

    return results
```

4.3 Badanie istotności parametrów metodą lasów losowych

W celu ustalenia istotności parametrów w modelowaniu zastosowano metodę lasów losowych.

```
forest = sklearn.ensemble.RandomForestClassifier(random_state=4021)
forest.fit(X_train, y_train)
pd.Series(forest.feature_importances_,
          index=X_train.columns[0:5]).sort_values(ascending=False).to_csv("relevance.csv")
forest.fit(X_r_train, y_r_train)
pd.Series(forest.feature_importances_,
          index=X_r_train.columns[0:3]).sort_values(ascending=False).to_csv("relevance_red.csv")
```

5 Rezultaty, wnioski i ich dyskusja

Resultaty dla badanych modeli:

	ACC_test	ACC_ucz	F1_test	F1_ucz	P_test	P_ucz	R_test	R_ucz
knn_05_full	0.89473684	0.91208791	0.9230769	0.9305556	0.98630137	0.943662	0.86747	0.917808
knn_05_red	0.90350877	0.90989011	0.9271523	0.9289428	0.95890411	0.943662	0.897436	0.914676
knn_05_full_norm_l1	0.9122807	0.88791209	0.9358974	0.9131175	1	0.943662	0.879518	0.884488
knn_05_red_norm_l1	0.56140351	0.77142857	0.6710526	0.8272425	0.69863014	0.876761	0.64557	0.783019
knn_05_full_norm_l2	0.92982456	0.89010989	0.9473684	0.914966	0.98630137	0.947183	0.911392	0.884868
knn_05_red_norm_l2	0.55263158	0.77802198	0.6709677	0.8313856	0.71232877	0.876761	0.634146	0.790476
knn_10_2_dist	0.89473684	1	0.9230769	1	0.98630137	1	0.86747	1
knn_15_red	0.92105263	0.89010989	0.9419355	0.9143836	1	0.940141	0.890244	0.89
knn_10_norm_l1	0.92105263	0.88131868	0.9403974	0.9065744	0.97260274	0.922535	0.910256	0.891156
knn_15_norm_l2	0.88596491	0.87912088	0.918239	0.9072513	1	0.947183	0.848837	0.87055
knn_20_dist_red_norm_l1	0.59649123	1	0.7160494	1	0.79452055	1	0.651685	1
knn_20_dist_red_norm_l2	0.59649123	1	0.7125	1	0.78082192	1	0.655172	1

Istotność dla parametrów niezredukowanych:

mean_area	0.315977373
mean_perimeter	0.289653853
mean_radius	0.170909367
mean_texture	0.118789819
mean_smoothness	0.104669588

Istotność dla parametrów zredukowanych:

mean_radius	0.56610179
mean_texture	0.24095817
mean_smoothness	0.19294004

Analizując wyniki danych można zaobserwować kilka zależności:

1. Zredukowanie liczby parametrów modelu powoduje pogorszenie zdolności klasyfikacyjnych modelu (nawet poniżej 60

2. Najlepiej klasyfikującym modelem jest "knn_05_full_norm_l2- wykorzystujący 5 najbliższych sąsiadów i normalizację L2.

3. Najwyższą istotność zarówno dla modeli o prametrach zredukowanych i pełnych mają parametry geometryczne guza (dla modelu zredukowanego - promień, dla pełnego - powierzchnia guza). Można z tego wysnuć wniosek, że złośliwość guza zależy przede wszystkim od jego wielkości.