

Dokumentacja projektowa

**Inżynieria Biomedyczna w Praktyce**

**Bartosz Rolnik**

Kierunek: *Informatyka*

Wydział: *Matematyki Stosowanej*

Rok akademicki *2021/2022*

**Tytuł:** *Kalkulator masy ciała, masy mięśniowej, masy tłuszczu, procentu tłuszczu w organizmie oraz BMI na podstawie różnych obwodów ciała, wzrostu i płci*

1. **Wstęp**

Projekt zakłada utworzenie prostego graficznego kalkulatora danych inbody na podstawie danych antropometrycznych. Źródłem danych były dwa pliki opracowywane w przeciągu piętnastu zajęć laboratoryjnych. Całość zagadnienia została podzielona na dwie części:

- Dodatkowe edycje plików CSV oraz testowanie obliczeń matematycznych potrzebnych później. Te fragment był pisany przy pomocy interaktywnego środowiska obliczeniowego Jupyter Notebook. W skład edycji plików wchodziło usuwanie zbędnych kolumn, łączenie plików z danymi antropometrycznymi oraz inbody, a następnie dzielenie ich według płci.

- Właściwa część projektu pisana w języku python. Wprowadzając odpowiednie wartości danych antropometrycznych w prostym menu, takich jak: płeć, wzrost oraz obwody: szyi, ramion, klatki, bioder, uda i pasa wyświetli nam się przewidywana waga osobnika oraz średni przewidywany błąd bezwzględny (wyrażony w tej samej jednostce co wynik). W każdej chwili liczenie wagi można zmienić na inne wartości inbody takie jak: procent tłuszczu w organizmie, masę tłuszczu w organizmie, masę mięśni w organizmie, BMI. Wszystkie wyniki są liczone przy użyciu regresji liniowej wielu zmiennych na zbiorze treningowym. Średni błąd bezwzględny sprawdzany jest na zbiorze walidacyjnym.

1. **Analiza zagadnienia**

Regresja - metoda pozwalająca na modelowanie związku pomiędzy zmiennymi i wykorzystanie tej wiedzy do przewidywania nieznanych wartości na podstawie znajomości innych. W przypadku mojej pracy zdecydowałem się na użycie regresji liniowej wielu zmiennych. Opis funkcji liniowej dla takiej regresji prezentuje się w następujący sposób:

+ e

Obsługa plików CSV była możliwa dzięki bibliotece pandas. Pierwszym krokiem w sprawie przygotowywania bazy danych było odpowiednie sklejenie plików „inbody” i „antropo”, a następnie usuwanie kolejnych kolumn, które zostały uznane za zbędne. W danych antropometrycznych zdecydowano się zostawić jedynie płeć, wzrost oraz obwody: szyi, ramion, klatki, bioder, uda i pasa. Reszta statystyk została uznana za niezwiązaną z ideą projektu. Następny w kolejności był plik danych inbody, z którego usunięto wszystkie kolumny poza tymi zaznaczonymi w temacie, czyli: masa ciała, masa mięśniowa, BMI, masa tłuszczu i jego procentowy udział w organizmie. W kolejnym kroku usunięto wszelkie linie, w których pojawiły się braki danych. Nie zdecydowano się na zastąpienie poszczególnych komórek przy pomocy jakiejś konwencji, aby być pewnym, że ewentualne błędy obliczeniowe nie wynikają z wybranego sposobu uzupełniania danych.

Komunikacja między plikiem edytującym bazę oraz właściwą aplikacją odbywała się dzięki dwóm plikom zawierającym dane nastolatków. Zostały one podzielone według płci, jednak dane inbody i antropometryczne zostały sklejone dla danej płci do jednej paczki danych.

Posiadając oba pliki, można było rozpocząć budowanie modelu regresji liniowej wielu zmiennych przy pomocy biblioteki sklearn w pythonie. Zanim model został zbudowany, pliki zostały podzielone na zbiór treningowy i walidacyjny w proporcjach 70:30. Po zbudowaniu regresji można było przetestować o ile nasz model się myli względem realnych danych, które posiadano i dla każdego rodzaju zmiennej został wyliczony średni błąd bezwzględny sprawdzany na zbiorze walidacyjnym.[[1]](#endnote-1)

Graficzne GUI zostało przygotowane dzięki bibliotece tkinter, która pokazały prosto, czytelnie i mile dla oka przygotowane grafiki dla programu. Pozwoliła ona na zbudowanie ośmiu wejść dla danych antropometrycznych, kilku przycisków oraz dwóch wyjść z wynikami, które są wyświetlane na podstawie naszego modelu. Każdorazowe włączenie programu spowoduje wylosowanie się zbiorów treningowych i walidacyjnych, więc każdorazowo wyniki mogą się nieznacznie różnić.

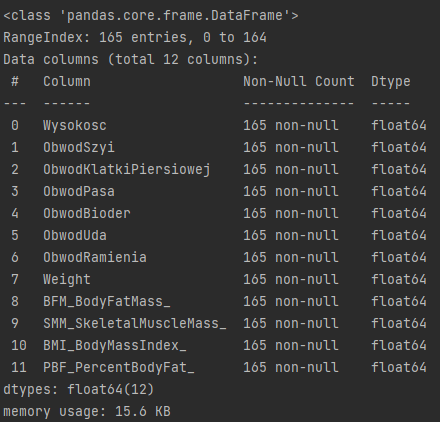
1. **Specyfikacja wewnętrzna**

**Plik Jupyter Notebook:**

Plik głównie do czyszczenia bazy danych i ewentualnych testów. Zawiera zaledwie jedną klasę o nazwie „Project”. Jest w niej funkcja do usuwania kolumn w pandas.dataframe po otrzymaniu listy z nazwami kolumn. Reszta funkcji służy do testowania metod matematycznych potrzebnych do pliku z właściwą częścią projektu. Całość tego programu do pewnego momentu to usuwanie niepotrzebnych kolumn oraz czyszczenie i analiza danych, później jest kilka próbnych linijek użycia metod z biblioteki sklearn.

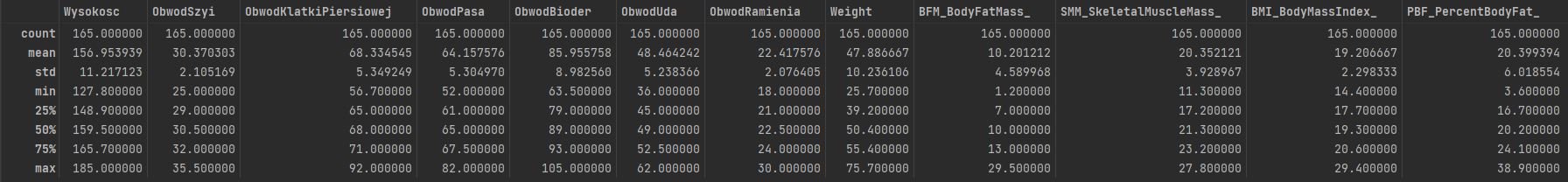
**Bazy danych:**

Podstawowe informacje dla bazy osób płci żeńskiej i męskiej (różnica występuje tylko w liczbie rekordów)



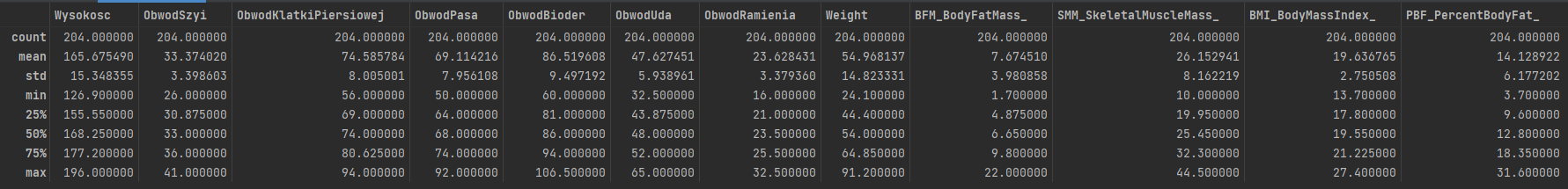
* Baza kobiet (164 rekordy):

Baza składa się z danych antropometrycznych: Wysokość, obwód szyi, obwód ramion, obwód klatki, obwód bioder, obwód uda, obwód pasa oraz danych inbody: masa ciała, masa mięśniowa, BMI, masa tłuszczu i procentowy udział tłuszczu w organizmie.



* Baza mężczyzn (203 rekordy):

Pod względem struktury bazy danych nie różni się ona niczym od bazy kobiet.



Różnice liczbowe:

* Porównywalne: obwód klatki piersiowej, obwód bioder, obwód uda, obwód ramienia, BMI
* Zdecydowane: wysokość, obwód szyi, masa, procent tłuszczu w organizmie, masa tłuszczu, masa mięśniowa

**Plik Python:**

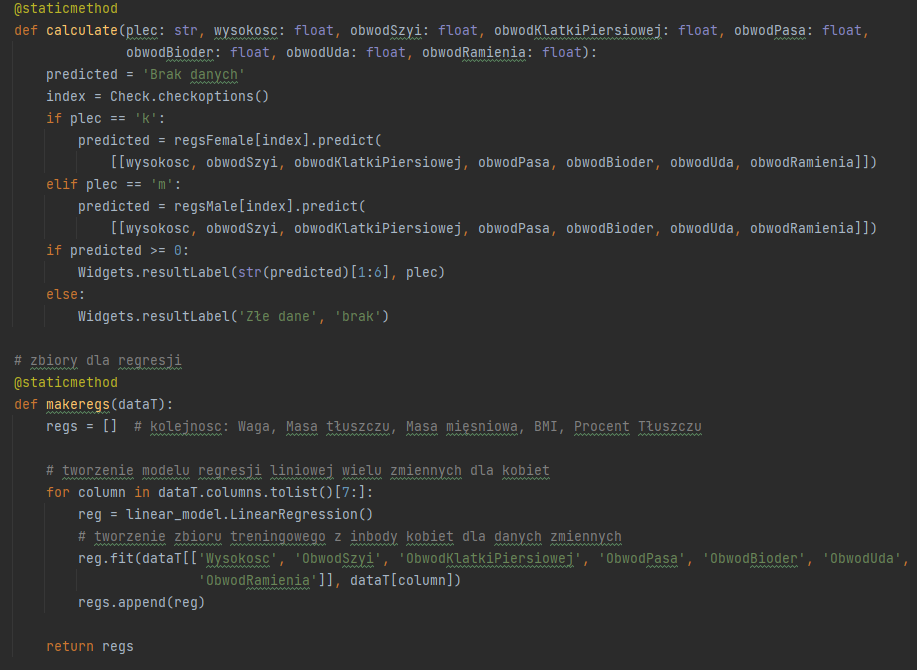
Program posiada cztery uporządkowane klasy: Check, Widgets, Predictor oraz ProcessingData.

**- Klasa Check**: posiada cztery metody, z czego dwie służą do poprawnej walidacji danych wpisywanych w wejścia:

* Walidacja danych liczbowych: nie pozwala na wpisywanie liczb ujemnych, liter oraz zbyt długich liczb,
* Walidacja płci: dopuszcza tylko wpisanie „k” lub „m”,
* Następna funkcja o nazwie „checkoptions” sprawdza jedynie, która z rubryk inbody została wybrana i zwraca nam jej index,
* Ostatnia metoda nazwana „isResultPossible” odpowiada za sprawdzenie, czy we wszystkich wejściach zostały wprowadzone dane. W przypadku braku danych program stosownie o tym poinformuje.

**- Klasa Widgets**: posiada pięć funkcji. Każda z nich odpowiada za inny typ widgetu, który jest wyświetlany na ekranie. Kolejno odpowiadają za zbudowanie: wejść na dane, etykiety dla wejść i wyjść, okno wyboru danych inbody do wyliczenia, potrzebne przyciski oraz etykiety wyjścia, które mają wstępne sprawdzenie wyniku.

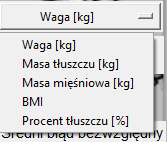
**- Klasa Predictor**: Najważniejsza klasa projektu. Posiada metody odpowiadające za liczenie dopasowanej funkcji liniowej dla obu płci, wyliczanie wartości dla zmiennej zależnej oraz liczenie średniego błędu bezwzględnego na zbiorach walidacyjnych.



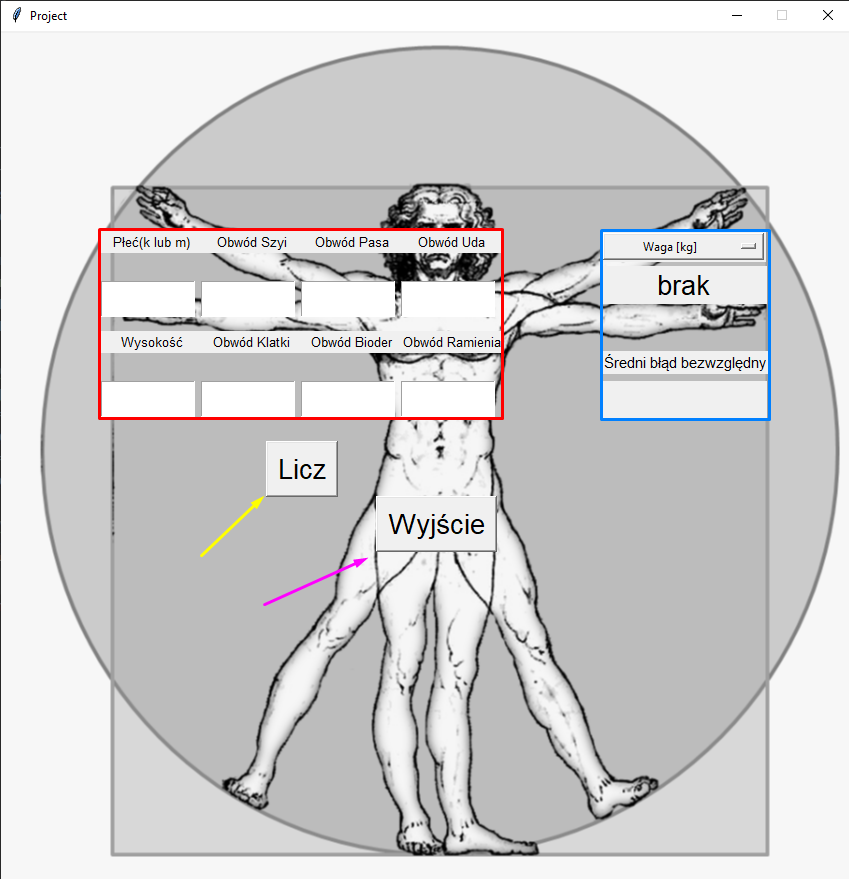
**- Inicjacja programu:** ostatni fragment kodu odpowiada za uruchomienie programu oraz odpowiednich metod startowych.

1. **Specyfikacja zewnętrzna**

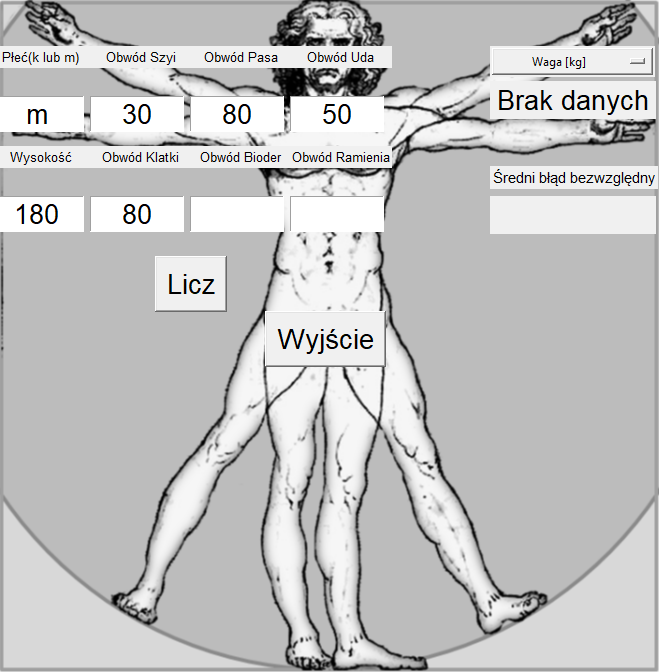
* Czerwony obszar: miejsce do wpisywania odpowiednich danych.
* Niebieski obszar: miejsce, w którym pojawia się wynik i średni błąd bezwzględny. W miejscu wyniku może pojawić się komunikat o jego braku (jak na ekranie poniżej), wprowadzeniu złych danych, niekompletnych danych lub właściwej liczby zmiennoprzecinkowej. Opcje do wyboru:



* Żółta strzałka: przycisk uruchamiający liczenie.
* Różowa strzałka: przycisk wyłączający aplikację.

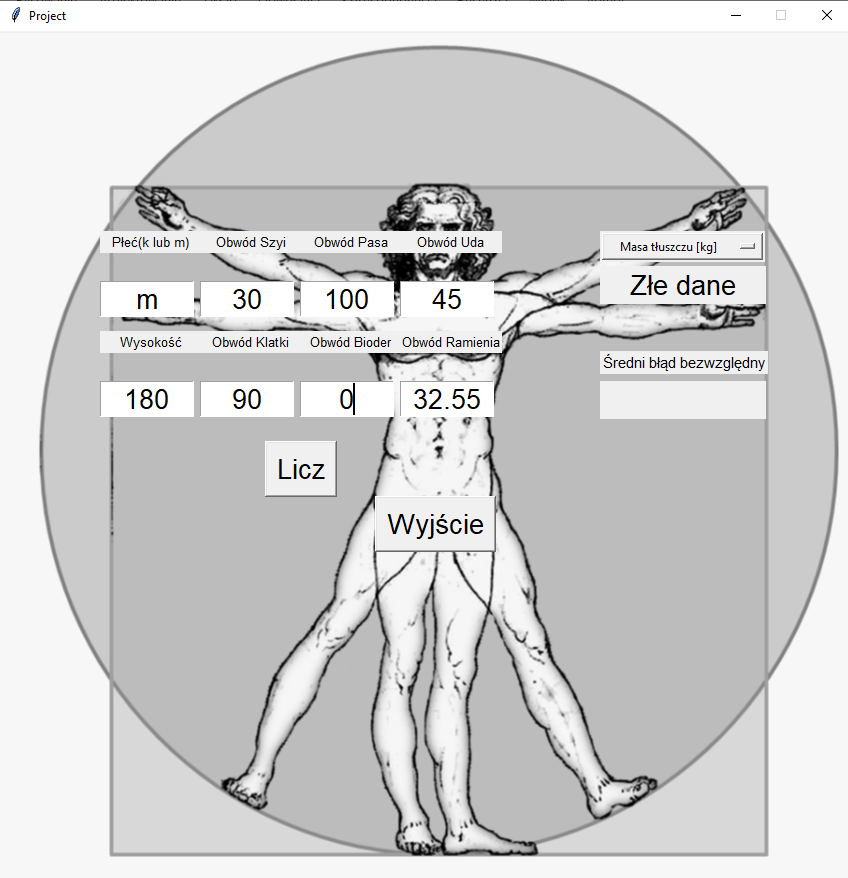


1. **Testowanie**
2. W przypadku nieuzupełnienia jednej lub większej ilości komórek, program w miejscu wyniku wyświetli komunikat „Brak danych”:

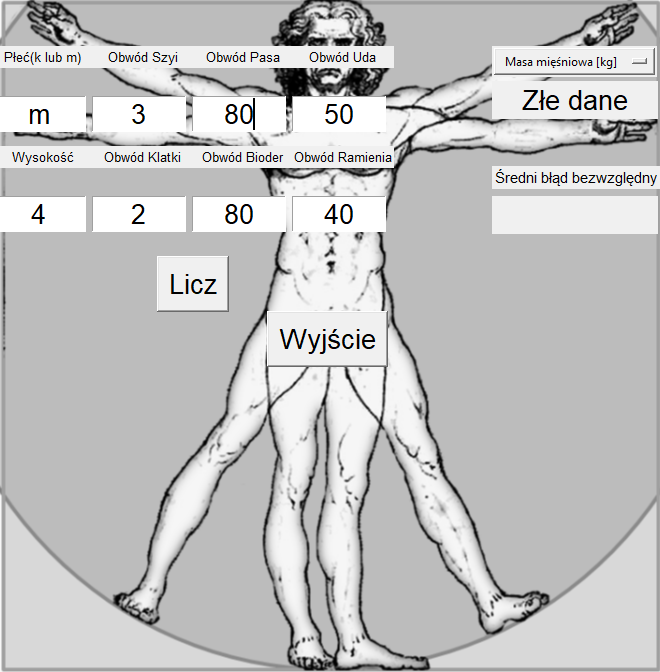


1. Program ma zaimplementowany prosty system służący do wychwytywania błędnych danych wejściowych:

* W przypadku wprowadzania prawidłowych danych, ale pozostawienia co najmniej jednej komórki z zerową wartością program wyświetli komunikat „Złe dane”

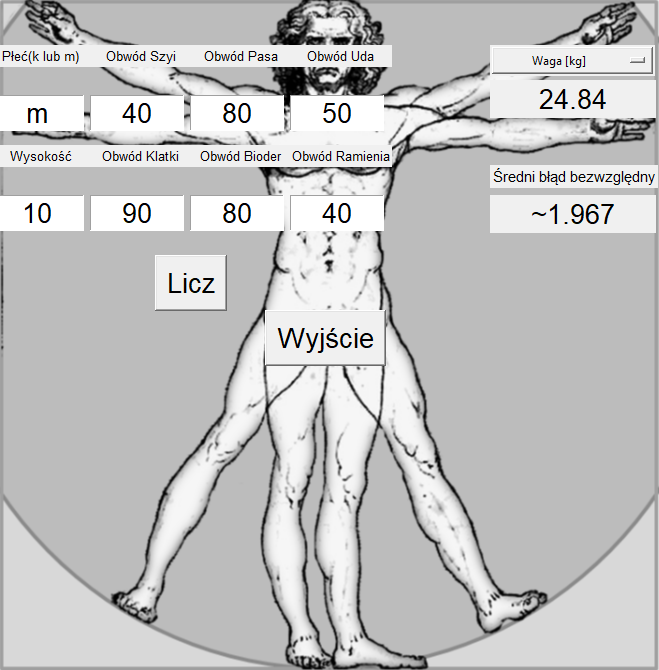


* W przypadku wprowadzenia nieprawdopodobnych liczb takich jak człowiek mający 4 cm wzrostu, obwód szyi mierzący 3 cm oraz obwód klatki mający 2 cm program również wyświetli komunikat „Złe dane”:

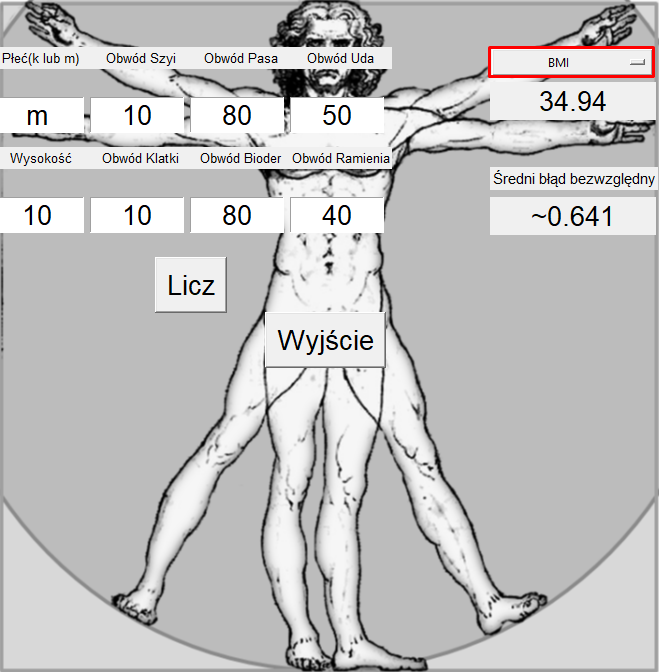
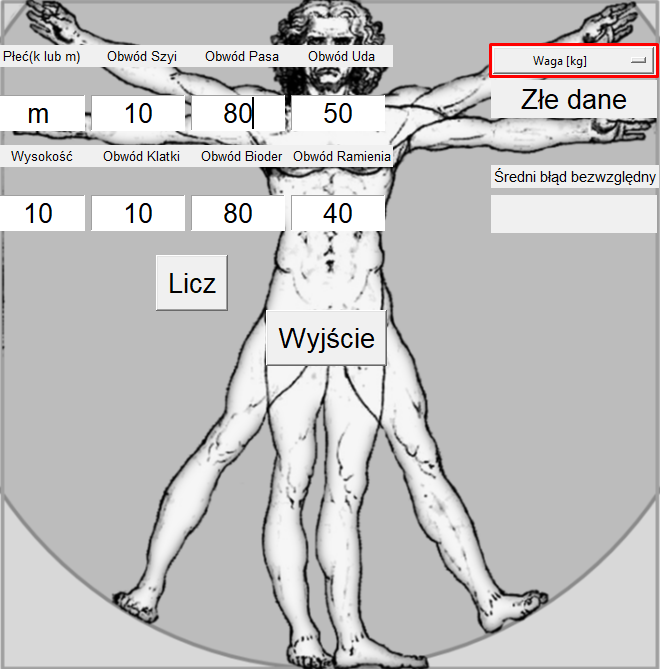


1. Podczas testowania zostały odkryte mankamenty tak prostego systemu wykrywania błędów:

* Często w przypadku wprowadzenia zaledwie jednej nieprawdopodobnej zmiennej program nie wychwytywał błędu:



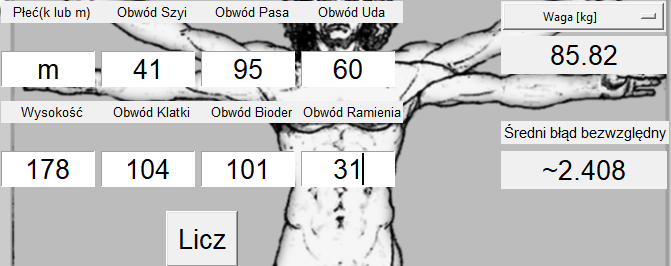
* Komunikat „Złe dane” odnosi się jedynie do szczególnego typu wyniku takiego jak „waga”. Dla innych danych typu inbody wynik może być odbiegający od normy:



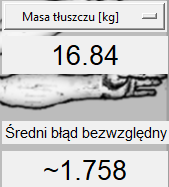
1. **Wyniki**

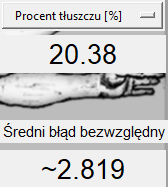
Wyliczenia w tym rozdziale zostały wykonane na podstawie rzeczywistego obiektu i porównane przy pomocy szczegółowej elektronicznej wagi.

Dane wprowadzone do aplikacji:



Pozostałe wyniki dla tych samych danych:





Realne wartości:

* Masa: 85.5 kg
* Masa mięśniowa: 38.1 kg
* Masa tłuszczu: 14.6 kg
* BMI: 26.99
* % tłuszczu w organizmie: 17.1 %

Błędy:

* Masa: 85.82 - 85.5 = 0.32kg, co stanowi ~0.38% realnej wartości

(znajduje się w granicach wyświetlanego błędu)

* Masa mięśniowa: 39.25 - 38.1 = 1.15 kg co stanowi ~3% realnej wartości (znajduje się w granicach wyświetlanego błędu)
* Masa tłuszczu: 16.84 - 14.6 = 2.24 kg, co stanowi ~15,3% realnej wartości

(nie znajduje się w granicach wyświetlanego błędu)

* BMI: 26.99 – 26.91 = 0.08, co stanowi ~0.29% realnej wartości

(znajduje się w granicach wyświetlanego błędu)

* % tłuszczu w organizmie: 20.38 - 17.1 = 3.28, co stanowi 19.2% realnej wartości

(nie znajduje się w granicach wyświetlanego błędu)

1. **Wnioski**

Analizując wyniki przedstawione w poprzednim rozdziale kalkulator radzi sobie różnie:

* Masa: w tym przypadku kalkulator wyliczył prawie perfekcyjnie masę obiektu myląc się o zaledwie 0.32 kg. Jak można zaobserwować po błędach wyświetlanych w programie jest to drugi najdokładniejszy pomiar.
* Masa mięśniowa: tutaj kalkulator napotkał lekkie problemy, ale wciąż były one zadowalające. Pomylił się o 1.15 kg, co stanowiło zaledwie około 3% realnej masy i mieściło się wciąż w granicach błędu.
* BMI: najłatwiejsza do wyliczenia szukana, ponieważ spora część danej jest już podana przy pomocy wzrostu, a dodatkowo jak można dostrzec w punkcie powyżej, masa jest również wyliczana bardzo dobrze, więc fenomenalny wynik nie dziwi.
* Masa tłuszczu: wynik zdecydowanie poniżej oczekiwań. Pomyłka zdecydowanie większa niż w poprzednich punktach, mimo mniejszej skali liczbowej.
* Procent tłuszczu w organizmie. Najgorzej liczona statystka, co można zaobserwować już przy wyświetlanych błędach. Nie dość, że korzysta z drugiej najgorzej liczonej szukanej, to dodatkowo wkrada się tu błąd związany z wagą.

Elementy wpływające na błąd pomiarowy:

* niedokładność sprzętu pomiarowego (grupy badanej i obiektu)
* obiekt znacznie odbiegał wiekiem od grupy badanej, a co z tym się wiąże miał inną budowę i proporcje ciała
* rząd wielkości wszystkich danych był zdecydowanie mniejszy w przypadku grupy badanej i błąd wyświetlany w aplikacji mógłby być inny przy bardziej zróżnicowanej grupie,
* mały zbiór treningowy (dla mężczyzn wynosiło to zaledwie 70% z 203 rekordów)

1. **Podsumowanie**

Celem było wykonanie kalkulatora, który na podstawie grupy wyselekcjonowanych obwodów części ciała liczył wybrane cechy inbody. Dodatkowym zamiarem było stworzenie intuicyjnego interfejsu graficznego. Rezultatem jest aplikacja spełniająca wszystkie wyżej wymienione zamierzenia, która jednak ma problemy z dokładnością niektórych wyników.

Program można rozwijać w wielu kierunkach, w tym między innymi:

* Zwiększenie grupy badanej u ludzi w bardziej zróżnicowanym wieku i różnej budowie ciała, co spowoduje większą dokładność,
* Rozbudowanie algorytmu o inne cechy inbody takie jak: masa układu kostnego, masa poszczególnych części ciała oraz inne cechy inbody dla nich,
* Poprawienie strony graficznej programu,
* Wyeliminowanie danych abstrakcyjnych takich jak np. obwód szyi równy 10 cm,
* Dodanie różnych wykresów i tabel,
* Poprawa algorytmów liczenia błędów i rezultatów.

**Źródła**

* <https://matematyka.poznan.pl/artykul/regresja-liniowa-czyli-o-zastosowaniu-funkcji-liniowej-w-analizie-statystycznej/>
* <https://www.investopedia.com/terms/m/mlr.asp>
* <https://pl.m.wikipedia.org/wiki/Regresja_liniowa>

1. [↑](#endnote-ref-1)