AKADEMIA GÓRNICZO-HUTNICZA

im. Stanisława Staszica w Krakowie

WYDZIAŁ ZARZĄDZANIA

STUDIA STACJONARNE

KIERUNEK: Informatyka i Ekonometria

PRACA DYPLOMOWA

licencjacka

Wiktoria Szczypka

Drzewa decyzyjne w problematyce medycznej

Application of decision trees in medical problems

Promotor: dr Łukasz Lach

Zatwierdzam do rejestracji i dopuszczam do obrony

.....

Data i podpis promotora

Kraków, 2019

Oświadczenie studenta

Uprzedzony(-a) o odpowiedzialności karnej na podstawie art. 115 ust. 1 i 2 ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2018 r. poz. 1191 z późn. zm.): „Kto przywłaszcza sobie autorstwo albo wprowadza w błąd co do autorstwa całości lub części cudzego utworu albo artystycznego wykonania, podlega grzywnie, karze ograniczenia wolności albo pozbawienia wolności do lat 3. Tej samej karze podlega, kto rozpowszechnia bez podania nazwiska lub pseudonimu twórcy cudzy utwór w wersji oryginalnej albo w postaci opracowania, artystyczne wykonanie albo publicznie zniekształca taki utwór, artystyczne wykonanie, fonogram, wideogram lub nadanie.", a także uprzedzony(-a) o odpowiedzialności dyscyplinarnej na podstawie art. 307 ust. 1 ustawy z dnia 20 lipca 2018 r. Prawo o szkolnictwie wyższym i nauce (Dz. U. z 2018 r. poz. 1668 z późn. zm.) „Student podlega odpowiedzialności dyscyplinarnej za naruszenie przepisów obowiązujących w uczelni oraz za czyn uchybiający godności studenta.", oświadczam, że niniejszą pracę dyplomową wykonałem(-am) osobiście i samodzielnie i nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.

Jednocześnie Uczelnia informuje, że zgodnie z art. 15a ww. ustawy o prawie autorskim i prawach pokrewnych Uczelni przysługuje pierwszeństwo w opublikowaniu pracy dyplomowej studenta. Jeżeli Uczelnia nie opublikowała pracy dyplomowej w terminie 6 miesięcy od dnia jej obrony, autor może ją opublikować, chyba że praca jest częścią utworu zbiorowego. Ponadto Uczelnia jako podmiot, o którym mowa w art. 7 ust. 1 pkt 1 ustawy z dnia 20 lipca 2018 r. – Prawo o szkolnictwie wyższym i nauce (Dz. U. z 2018 r. poz. 1668 z późn. zm.), może korzystać bez wynagrodzenia i bez konieczności uzyskania zgody autora z utworu stworzonego przez studenta w wyniku wykonywania obowiązków związanych z odbywaniem studiów, udostępniać utwór ministrowi właściwemu do spraw szkolnictwa wyższego i nauki oraz korzystać z utworów znajdujących się w prowadzonych przez niego bazach danych, w celu sprawdzania z wykorzystaniem systemu antyplagiatowego. Minister właściwy do spraw szkolnictwa wyższego i nauki może korzystać z prac dyplomowych znajdujących się w prowadzonych przez niego bazach danych w zakresie niezbędnym do zapewnienia prawidłowego utrzymania i rozwoju tych baz oraz współpracujących z nimi systemów informatycznych.

……

(czytelny podpis studenta)

Spis treści

1 Wstęp

1.1 Cel pracy, przedmiot badań

W niniejszej pracy licencjackiej mam dwa cele - jeden statystyczny, a drugi związany z problematyką medyczną. Po pierwsze chciałabym porównać metody tworzenia drzew decyzyjnych, od najprostszych do tych z dużą złożonością obliczeniową. Jest bardzo wiele metod, które zapewniają rozwiązanie tego problemu, pojawia się więc pytanie, którą z nich najlepiej używać. W tejże pracy chciałabym zbadać, która z metod jest najskuteczniejsza oraz czy metody bardziej skomplikowane, a co za tym idzie bardziej złożone obliczeniowo, sprawiają, że wyniki są znacząco lepsze. Po drugie w swojej pracy chciałabym zbadać jakie czynniki istotnie wpływają na śmiertelność noworodków. Wykorzystam do tego dane pochodzące z World Health Organization (WHO). Wybrałam kilka czynników, które moim zdaniem mogą mieć wpływ na ten medyczny problem, a za pomocą drzew decyzyjnych okaże się które z nich są ważne.

1.2 Motywacja do podjęcia tematu

Zgłębianie zagadnień związanych ze statystyką, ekonometrią i uczeniem maszynowym (ang. *Machine Learning*) to coś co dało mi najwięcej satysfakcji w trakcie studiów, z tym chciałabym też związać swoją przyszłość zawodową. Stąd też wzięły się tematyka oraz metody badawcze zawarte w niniejszej pracy licencjackiej. Drzewa decyzyjne to narzędzie, którego wynik zrozumie każdy, a nie tylko specjalista, można je bowiem prosto i atrakcyjnie wizualizować. Jednocześnie jest to bardzo ciekawe zagadnienie oraz jest wiele skomplikowanych metod powstałych w celu stworzenia jak najbardziej skutecznego drzewa decyzyjnego. Problematyka drzew decyzyjnych zainteresowała mnie na tyle, że postanowiłam w ramach pracy licencjackiej poszerzyć swoją wiedzę w tym zakresie.

Tematyka zdrowia, jakości życia człowieka oraz wpływu różnych czynników na te wartości zawsze mnie interesowała. Natomiast studia i zbudowany na ich bazie aparat ilościowy dał mi wreszcie możliwość spojrzenia na nie w obiektywny i usystematyzowany sposób. Przeglądając wskaźniki dostępne na stronach WHO, zaciekawił mnie ten dotyczący śmiertelności noworodków.

1.3 Wykorzystane metody badawcze

Jak już wspomniałam wcześniej, w niniejszej pracy licencjackiej będę korzystać z drzew decyzyjnych. Istnieje jednak wiele algorytmów za pomocą których można je konstruować.

Najpierw zbuduję najprostsze drzewo regresyjne (nie jest to skomplikowany algorytm). Zamieszczę je w celu porównania o ile lepsze są metody bardziej skomplikowane. Poza tym skonstruuję drzewa trzema metodami:

- bagging,
- lasy losowe (ang. *random forest*),
- boosting.

Część empiryczną swoich badań wykonam w języku R w środowisku RStudio. Jest to język programowania przeznaczony dla statystyków, który umożliwi mi sprawne skonstruowanie drzew decyzyjnych.

1.4 Streszczenie poszczególnych rozdziałów

W rozdziale pierwszym zaprezentuje swoje dane oraz ich źródła. Następnie opiszę zmienną objaśnianą i objaśniające oraz przedstawię ich podstawowe statystyki opisowe.

W drugim rozdziale opiszę wybrane przeze mnie metody badawcze. Przedstawię tutaj część teoretyczną mojego badania - niezbędne wzory i formuły, a także algorytm ich wykonania.

Trzeci rozdział będzie zawierał empiryczną część mojej pracy. Przedstawię w nim procedurę aplikacji każdej metody, a także ich wizualizacje. Zakładam, że za pomocą różnych metod otrzymam rożne drzewa oraz inną skuteczność prognozy. Będzie to najbardziej rozbudowana część mojej pracy.

Ostatni rozdział to podsumowanie - zawrę tutaj wnioski, które można wysnuć na podstawie przeprowadzonego badania. Opiszę, która z metod konstruowania drzew decyzyjnych okazała się najlepsza oraz które z wybranych przeze mnie czynników istotnie wpływają na śmiertelność noworodków.

1.5 Przegląd literatury

Szukając literatury dotyczącej badań na temat śmiertelności noworodków, najwięcej artykułów znalazłam na stronie NCBI (National Center for Biotechnology Information)[1]. Pośród wielu badań przedstawię te związane z wpływem różnych czynników na wskaźnik śmiertelności noworodków:

- wpływ edukacji i pochodzenia matki wśród chińskich amerykanów,

[1] https://www.ncbi.nlm.nih.gov (dostęp 16.03.2019)

- wpływ wykształcenia i warunków pracy położnych na Florydzie,
- wpływ edukacji matki i ojca, a także ich zawodów w Jordanii, Jemenie, Egipcie i Tunezji.

Badania te różnią się od tych, które zamierzam wykonać - nie są przeprowadzone za pomocą drzew decyzyjnych oraz nie uwzględniają większej liczby krajów. Czynniki poruszane w mojej pracy również są odmienne.

Szukając literatury, która bazuje na zastosowaniu drzew decyzyjnych, znalazłam wiele artykułów na stronach związanych z Data Science. Najobszerniejsze z nich znajdowały się na stronach DataCamp[2] oraz Towards Data Science[3]. Przedstawione są tam wady i zalety algorytmów konstruowania drzew. Natomiast nie została wybrana nigdzie najlepsza metoda.

Wszystko wskazuje więc, że niniejsza praca licencjacka będzie oryginalna i wniesie coś nowego w tematyce oceny skuteczności drzew decyzyjnych oraz badań nad przyczynami zgonu noworodków.

[2] https://www.datacamp.com (dostęp 29.03.2019)
[3] https://towardsdatascience.com (dostęp 29.03.2019)

2 Prezentacja zbioru danych

Dane, które będę używać do wykonania obliczeń, pochodzą ze strony WHO[4] z roku 2015. Wśród wielu przedstawionych wskaźników obrazujących sytuację zdrowotną na świecie wybrałam zmienną objaśnianą - śmiertelność noworodków oraz 9 zmiennych objaśniających. Utworzony przeze mnie zbiór danych zawiera 194 obserwacji - wskaźniki są przedstawione dla każdego kraju należącego do WHO w roku 2015.

2.1 Charakterystyka zmiennych

Opiszę teraz wszystkie zmienne znajdujące się w moim zbiorze danych.

Zmienna objaśniana:

- śmiertelność noworodków - prawdopodobieństwo śmierci na 1000 żywych urodzeń.

Zmienne objaśniające:

- wydatki - bieżące wydatki zdrowotne przypadające na jednego mieszkańca (per capita) w dolarach amerykańskich,
- BMI - średnia wartość wskaźnika masy ciała wśród dorosłych, standaryzowane pod względem wieku (kg/m^2),
- zabójstwa - szacowany wskaźnik zabójstw na 100 000 ludzi,
- zatrucia - wskaźnik śmiertelności spowodowanych niecelowymi zatruciami (ze względu na chemikalia) na 100 000 ludzi,
- higiena - procent społeczeństwa korzystających z przynajmniej podstawowych usług sanitarnych,
- paliwa ekologiczne - procent społeczeństwa, które w podstawowym stopniu opiera się na paliwach i technologiach ekologicznych; najmniejsza wartość to "<5", a największa ">95", do swoich badań zamieniam je na odpowiednio 5 i 95,
- samobójstwa - wskaźnik samobójstw na 100 000 ludzi, standaryzowany pod względem wieku,
- ciśnienie krwi - procent społeczeństwa z podniesionym ciśnieniem krwi - skurczowe ciśnienie krwi (SBP) $\geq$ 140 lub rozkurczowe (DBP) $\geq$ 90, wskaźnik surowy,
- region - ze względu na to, że każda obserwacja jest przypisana do danego kraju wprowadzam zmienną region. Do przypisywania krajów kieruje się podziałem

[4] https://www.who.int/en# (dostęp 1.04.2019)

obowiązującym w WHO z małymi zmianami. Dzielę też Europę na część wschodnią, zachodnią i Bałkany, a Amerykę na Amerykę Północną, Środkową i Południową.

- Ameryka Południowa: Argentyna, Boliwia, Brazylia, Chile, Kolumbia, Ekwador, Gujana, Paragwaj, Peru, Surinam, Trynidad i Tobago, Wenezuela,
- Ameryka Środkowa: Antigua i Barbuda, Bahamy, Barbados, Belize, Kostaryka, Kuba, Dominika, Dominikana, Salvador, Grenada, Gwatemala, Haiti, Honduras, Jamajka, Panama, Saint Kitts and Nevis, Saint Lucia, Saint Vincent i Grenadyny,
- Ameryka Północna: Kanada, Meksyk, Stany Zjednoczone,
- Afryka: Algieria, Angola, Benin, Botswana, Burkina Faso, Burundi, Republika Zielonego Przylądka, Kamerun, Republika Środkowoafrykańska, Czad, Komory, Kongo, Wybrzeże Kości Słoniowej, Demokratyczna Republika Konga, Erytrea, Eswatini, Etiopia, Gabon, Gambia, Ghana, Gwinea, Gwinea Bissau, Gwinea Równikowa, Kenia, Lesotho, Madagaskar, Malawi, Mali, Mauretania, Mauritius, Mozambik, Namibia, Niger, Nigeria, Rwanda, Wyspy Świętego Tomasza i Książęca, Senegal, Seszele, Sierra Leone, Południowa Afryka, Sudan Południowy, Sudan, Togo, Uganda, Tanzania, Zambia, Zimbabwe,
- Azja Południowa: Bangladesz, Bhutan, ludzka demokracja, Indie, Indonezja, Malediwy, Myanmar, Nepal, Sri Lanka, Tajlandia, Timor Wschodni,
- Wschodnia część Morza Śródziemnego: Afganistan, Bahrajn, Dżibuti, Egipt, Iran, Irak, Israel, Jordan, Kuwejt, Liban, Libia, Maroko, Oman, Pakistan, Katar, Arabia Saudyjska, Somalia, Syria, Tunezja, Zjednoczone Emiraty Arabskie, Jemen,
- Zachodni Pacyfik: Australia, Brunei, Kambodża, Chiny, Wyspy Cooka, Fidżi, Japonia, Kiribati, Laos, Malaysia, Wyspy Marshalla, Mikronezja, Mongolia, Nauru, Nowa Zelandia, Niue, Palau, Papua Nowa Gwinea, Filipiny, Korea, Korea Północna, Samoa, Singapur, Wyspy Salomona, Tonga, Tuvalu, Vanuatu, Wietnam,
- Europa Wschodnia, Azja Zachodnia: Armenia, Azerbejdżan, Białoruś, Cypr, Estonia, Gruzja, Kazachstan, Kirgistan, Litwa, Łotwa, Mołdawia, Rosja, Tadżykistan, Turcja, Turkmenistan, Ukraina, Uzbekistan,
- Bałkany: Albania, Bośnia i Hercegowina, Bułgaria, Grecja, Czarnogóra, Macedonia, Rumunia, Serbia, Słowenia,

- Europa Zachodnia: Andora, Austria, Belgia, Czechy, Dania, Finlandia, Francja, Niemcy, Węgry, Islandia, Irlandia, Włochy, Luksemburg, Malta, Monako, Holandia, Norwegia, Polska, Portugalia, San Marino, Słowacja, Hiszpania, Szwecja, Szwajcaria, Wielka Brytania.

2.2 Podstawowe statystyki opisowe

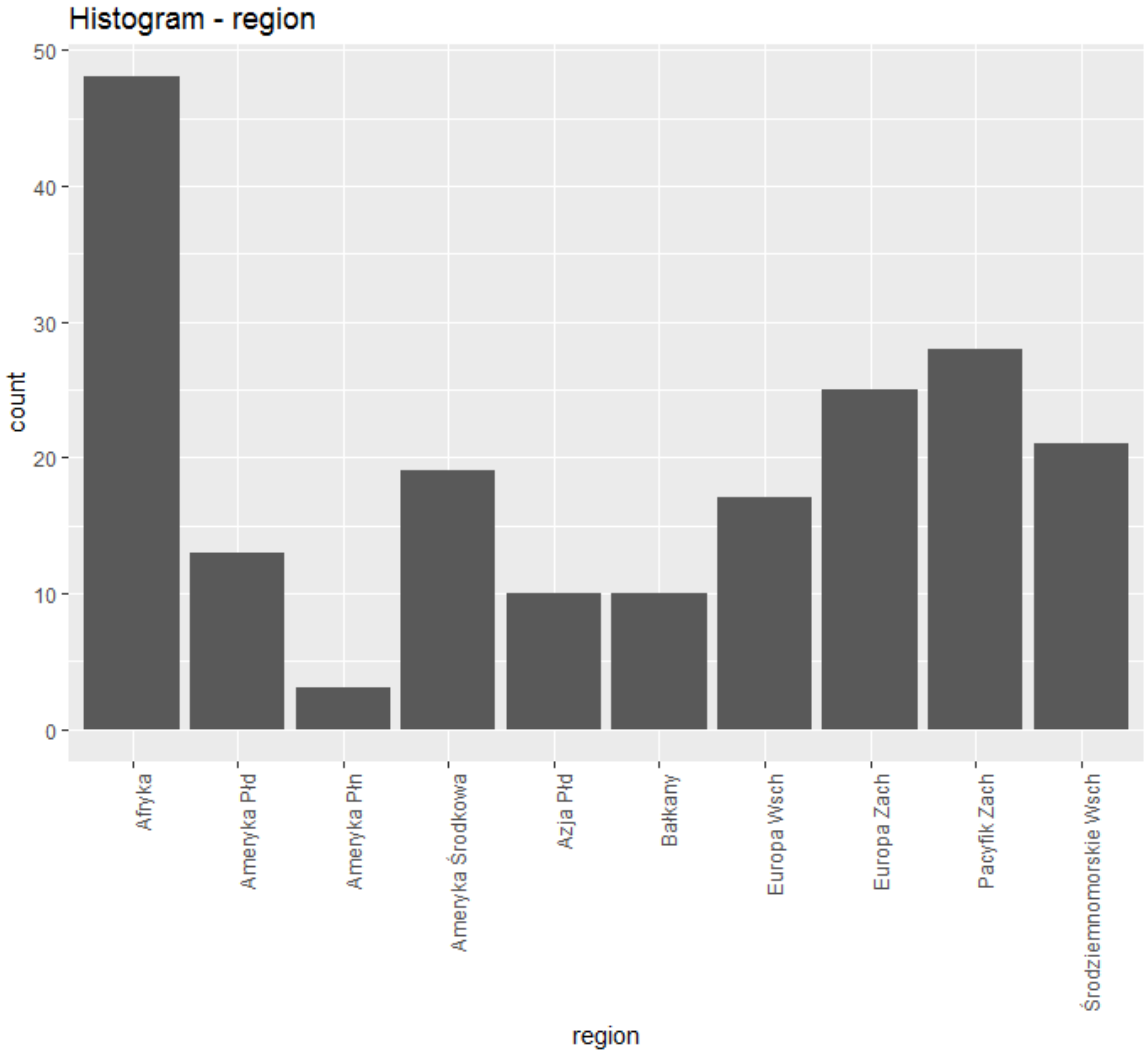
Poniżej przedstawię podstawowe statystyki opisowe dla zmiennych numerycznych oraz histogram dla zmiennej *region*.

zmienna	min	1 kwartyl	mediana	średnia	3 kwartyl	max	NA
śmiertelność	1,70	6,55	15,20	23,32	35,05	91,60	0
wydatki	16,60	91,35	366,00	999,28	965,73	9818,00	4
BMI	20,50	23,90	26,25	25,76	27,10	32,90	4
samobójstwa	0,40	5,70	9,10	9,77	12,30	30,30	11
ciśnienie krwi	12,60	20,02	22,25	23,19	24,98	41,00	4
zatrucia	0,00	0,30	0,55	1,15	1,80	5,30	11
paliwa ekologiczne	5,00	29,25	85,00	64,38	95,00	95,00	3
zabójstwa	0,2	1,925	5,25	8,221	9,9	58,3	11
higiena	0	47,25	88	73,13	98	100	0

Tabela 1. Statystyki opisowe [źródło: opracowanie własne]

Można zauważyć, że występują wartości odstające - w największym stopniu są one widoczne w przypadku zmiennej *wydatki*, gdzie wartość maksymalna jest około 10 razy większa od górnego kwartyla. Zmienna ta dobrze pokazuje dysproporcje wśród krajów - wydatki na zdrowie wahają się od 16 do 9819 dolarów. *Śmiertelność noworodków* kształtuje się od 1,7 do aż 91,6. Mediana *BMI* wynosi 26,30 (prawidłowe wynosi między 18,5, a 25). Sugeruje to, że społeczeństwa ponad połowy krajów cierpią na nadwagę. Kraje można podzielić również na te mniej depresyjne (zmienna *samobójstwa* wynosi tylko 0,4) oraz bardziej (30,3). Zaniepokojenie może również budzić fakt, że w każdym kraju średnio 23,21% społeczeństwa ma podwyższone ciśnienie. Zatrucia chemikaliami nie są powszechnym zjawiskiem. Ze statystyk można też wywnioskować, że większość państw otwiera się na ekologiczne rozwiązania. Dysproporcje pomiędzy krajami obrazuje też dobrze zmienna *higiena* - są miejsca, gdzie nikt nie ma dostępu do podstawowych usług sanitarnych.

Jak widać występują też braki w danych (NA). Nie jest ich jednak dużo, a tym problemem zajmę się w dalszej części pracy.



Wykres 1. Wykres częstości zmiennej region [źródło: opracowanie własne]

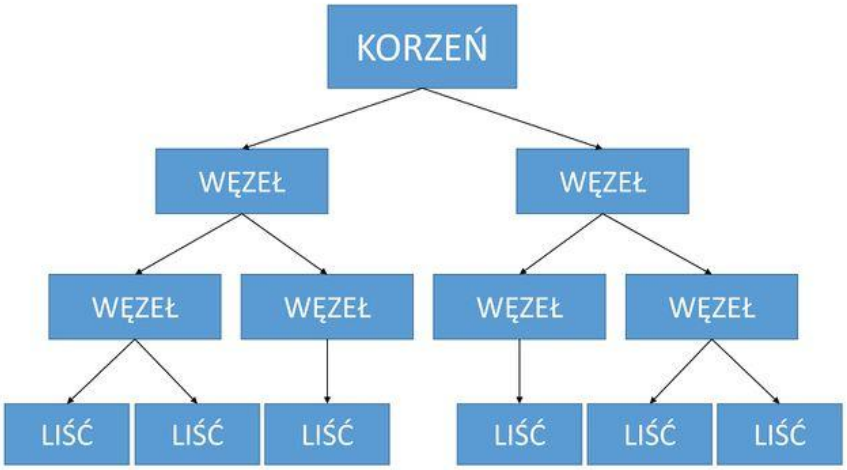
Już na pierwszy rzut oka widać, że najwięcej obserwacji pochodzi z Afryki (prawie 50), a najmniej z Ameryki Północnej (3). W pozostałych regionach występuje od 10 do 30 obserwacji.

3 Opis metod badawczych[5]

Rozdział ten będzie teoretyczną częścią pracy. Opiszę w nim algorytm konstruowania drzewa decyzyjnego za pomocą poszczególnych metod badawczych.

Zanim przejdę do dokładnego opisu metod przybliżę pojęcie drzewa decyzyjnego. Najprościej rzecz ujmując, jest to algorytm nadzorowanego uczenia, który może być używany zarówno do problemów klasyfikacyjnych jak i regresyjnych. Zaletą drzew jest to, że działają zarówno dla zmiennych wejściowych i wyjściowych ciągłych i kategorycznych.[6]

Poniżej przedstawiam strukturę drzewa.



Wykres 2. Struktura drzewa [źródło: https://mfiles.pl/pl/index.php/Drzewo_decyzyjne]

Korzeń reprezentuje całą populację lub próbkę. Potem następuje jego podział na więcej podzbiorów. Węzły posiadają pojedynczą cechę lub kombinacje cech z próbek treningowych. Liście to takie węzły, które już się nie dzielą - ostatecznie uszeregowane dane.[7]

Zalety drzew decyzyjnych:

- Łatwa interpretacja graficzna, bardzo intuicyjna. Nie trzeba być ekspertem, aby ją zrozumieć.
- Wykrywają zależności zarówno liniowe jak i nieliniowe.

[5] G. James, D. Witten, T. Hastie, R. Tibshirani *An Introduction to Statistical Learning* Springer Science + Business Media New York 2013.
[6] https://www.datacamp.com/community/tutorials/decision-trees-R (dostęp 10.04.2019).
[7] http://www.is.umk.pl/~duch/Wyklady/CIS/Prace%20zalicz/08-Bujak.pdf (dostęp 10.04.2019).

- Braki w danych nie stanowią problemu.
- Mała wrażliwość na obserwacje odstające.
- Można używać zarówno zmiennych ciągłych jak i kategorycznych.
- Nie wymaga normalizacji i skalowania danych.

Wady drzew decyzyjnych:

- W najprostszej wersji nie dają najlepszych wyników. Poprawa skuteczności następuje wraz ze wzrostem skomplikowania algorytmów.
- Algorytm drzew decyzyjnych nie daje stabilnych wyników. Niewielkie zaburzenia w danych mogą całkowicie zmienić wyniki.
- Bardzo łatwo jest o przeuczenie (ang. *overfitting*) drzewa. Oznacza to, że model jest za bardzo dopasowany do danych treningowych. Nauczył się on każdego szczegółu w tym zbiorze zamiast znaleźć ogólne zależności. Powoduje to, że jego skuteczność na zbiorze uczącym wynosi prawie 100%, lecz nie ma zdolności do generalizacji. Sprawia to, że skuteczność na zbiorze testowym będzie znacząco niższa niż na treningowym.

Konstruując drzewo, dane dzieli się na zbiór treningowy oraz testowy. Drzewo tworzy się na zbiorze treningowym, a następnie bada się jego skuteczność na testowym.

3.1 Drzewa regresyjne

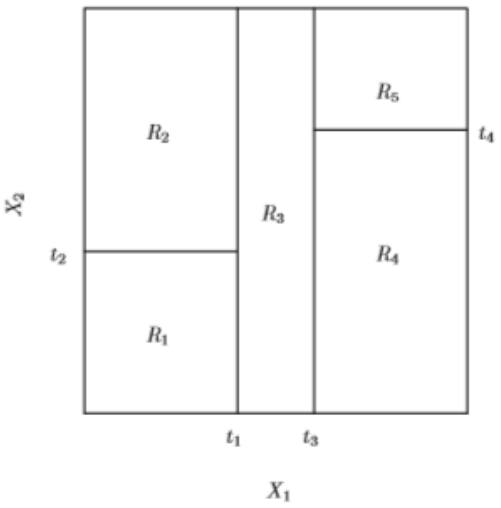
Drzewa regresyjna to takie, gdy przewidywany wynik jest rzeczywistą liczbą. W przypadku mojej pracy wynikiem będzie wskaźnik śmiertelności noworodków.

Proces budowy drzewa regresyjnego sprowadza się do dwóch kroków.

1. Dzieli się predykcyjną przestrzeń (zbiór możliwych wartości dla X_1, X_2, ..., X_p) na J odrębnych i nienakładających się regionów R_1, R_2, ..., R_J.
2. Dla każdej obserwacji należącej do regionu $R_J$przewiduje się taką samą wartość zmiennej responsywnej, co w praktyce oznacza, że rezultatem jest średnia wartość wyników dla obserwacji treningowych znajdujących się w R_J.

Głównym pytaniem dotyczącym procesu budowy drzewa jest to w jaki sposób stworzyć regiony R_1, R_2, ..., R_J.Teoretycznie regiony mogłoby mieć dowolny kształt. W praktyce dzieli

się predykcyjną przestrzeń na prostokąty. Podział ten powoduje, że interpretacja wyników jest bardzo łatwa.



Rysunek 1. Podział zbioru na regiony [źródło: G. James, D. Witten, T. Hastie, R. Tibshirani *An Introduction to Statistical Learning* Springer Science + Business Media New York 2013]

Celem jest znalezienie takich prostokątów, które minimalizują resztową sumę kwadratów (RSS) przedstawioną poniższym wzorem:

$$\sum_{j=1}^{J} \sum_{i \epsilon Rj} \left(y_i - \hat{y}_{Rj}\right)^2,$$

gdzie $\hat{y}_{Rj}$ jest średnim wynikiem dla obserwacji w zbiorze treningowym w j-tym regionie, zaś y_i jest rzeczywistą wartością obserwacji. Niestety jest to obliczeniowo niemożliwe, aby rozważyć każdą możliwą partycję przestrzeni obiektów w J prostokątach. Z tego powodu w praktyce używa się rekurencyjnego podziału binarnego. Algorytm ten rozpoczyna się od korzenia, a potem dzieli przestrzeń na regiony. W każdym kroku wykonuje podział najlepszy na tym konkretnie kroku, nie patrzy się wprzód, aby wykonać skuteczniejszy podział w przyszłych krokach.

W celu wykonania takiego podziału najpierw wybiera się predykator X_j oraz punkt odcięcia s taki, aby podział przestrzeni na regiony $\{X|X_j < s\}$ oraz $\{X|X_j \geq s\}$ prowadził do zminimalizowania RSS. Rozważa się więc wszystkie możliwe predykatory X_1, X_2, ..., X_p i możliwe punkty odcięcia s dla każdego z nich, a następnie wybiera się te wartości, przy których RSS jest najmniejsze. Kroki te następnie powtarzamy dla powstałych regionów.

Proces ten trwa do momentu, gdy zostanie spełniony warunek końcowy, na przykład można kontynuować proces tworzenia regionów, aż w każdym będzie nie więcej niż 5 obserwacji.

Po stworzeniu regionów R_1, R_2, ..., R_J przewiduje się wynik w każdym z nich na postawie średniego wyniku obserwacji treningowych należącego do tego regionu.

Powyższy proces spowoduje, że skuteczność dla zbioru treningowego będzie bardzo dobra, lecz model będzie przeuczony co prowadzi do słabych wyników na zbiorze testowym. Drzewo z mniejszą liczbą regionów może prowadzić do mniejszej wariancji, a dzięki temu do lepszych wyników na zbiorze testowym. W praktyce konstruuje się duże drzewo, a następnie je przycina przez co tworzy się poddrzewo. Celem jest wybranie takiego poddrzewa, by miało jak najmniejszy wskaźnik błędu. Można go oszacować za pomocą walidacji krzyżowej. Jest to jedna z technik używana do testowania efektywności modelu, to także procedura wielo losowania używana, gdy ma się ograniczoną liczbę danych. Prosta walidacja dzieli jedynie próbę na dwa zbiory: testowy i uczący. Natomiast w k-krotnej walidacji próba dzielona jest na K podzbiorów. Następnie kolejno każdego z nich używa się jako zbioru testowego, a na podstawie pozostałych konstruuje się model. Algorytm ten jest wykonywany K razy - uzyskuje się więc K różnych rezultatów, które następnie są uśredniane (lub łączone w inny sposób np. mediana) w celu uzyskania jednego wyniku.[8]

Drzewo regresyjne to prosty algorytm tworzenia pojedynczego drzewa. W praktyce używa się metod, które łączą wiele drzew, a dzięki temu polepsza się ich skuteczność. W kolejnych punktach opiszę 3 metody, które to umożliwiają.

3.2 Bagging

Jak już wspomniałam wcześniej, zwykłe drzewo regresyjne ma bardzo dużą wariancję, czyli jeżeli podzieli się dane treningowe losowo na pół i stworzy się dla każdego podzbioru drzewo regresyjne to mogą się one okazać zupełnie różne. Celem baggingu (zwanego inaczej *bootstrap aggregation*), jest redukcja wariancji.

Algorytm baggingu:

1. Ze zbioru treningowego losuje się ze zwracaniem B różnych podzbiorów. Zazwyczaj liczebność podzbiorów jest taka sama jak zbioru treningowego.

[8] https://pl.wikipedia.org/wiki/Sprawdzian_krzy%C5%BCowy#Prosta_walidacja (dostęp 10.06.2019)

2. Dla każdego podzbioru tworzy się drzewo regresyjne i dokonuje się predykcji na jego podstawie.
3. Końcowa predykcja to uśrednienie predykatorów powstałych z wielu podzbiorów. Uśrednia się poprzez średnią arytmetyczną, modalną lub medianę.

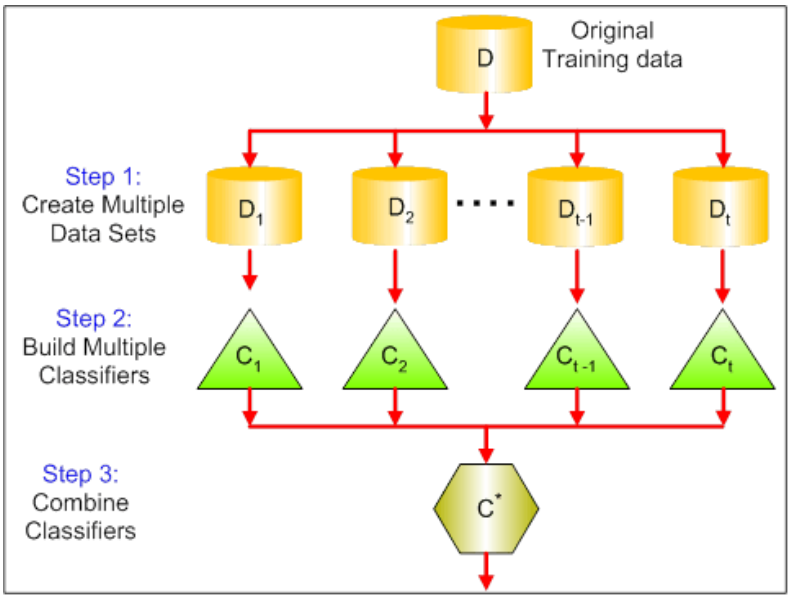
Poniżej równanie baggingu:

$$\hat{f}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}_b(x)\,,$$

gdzie B to liczba utworzonych podzbiorów, $\hat{f}_b(x)$ to predykcja dla b-tego podzbioru, zaś $\hat{f}_{bag}(x)$ to końcowa predykcja.

Jak już wspomniałam, pojedyncze drzewo ma wysoką wariancję, lecz niskie obciążenie, które sugeruje mało założeń o formie docelowej funkcji predykcyjnej. Uśrednienie B drzew powoduje zmniejszenie wariancji. Bagging powoduje wzrost dokładności predykcji uzyskiwanych za pomocą drzewa poprzez stworzenie jednego wyniku z kilkuset, a nawet tysięcy drzew.

Poniżej wizualne przedstawienie idei baggingu:



Rysunek 2. Bagging [źródło: https://www.datacamp.com/community/tutorials/decision-trees-R]

3.3 Lasy losowe

Lasy losowe, tak jak bagging, zmniejszają wariancję pojedynczych drzew, lecz dodatkowo dekorelują drzewa. W przypadku, gdy występuje jeden mocny predykator, prawdopodobnie

większość drzew w baggingu będzie wykorzystywać tą zmienną na samym początku. Spowoduje to, że drzewa będą do siebie bardzo podobne, więc prognozy z nich będą silnie skorelowane. W wyniku tego wariancja drzew nie zostanie tak bardzo zredukowana jak w przypadku nieskorelowanych drzew. Lasy losowe rozwiązują ten problem poprzez niewykorzystywanie wszystkich zmiennych objaśniających.

Algorytm lasów losowych :

1. Ze zbioru treningowego losuje się ze zwracaniem B różnych podzbiorów. Zazwyczaj liczebność podzbiorów jest taka sama jak zbioru treningowego.
2. Następnie losuje się m zmiennych objaśniających z p możliwych ($m < p$). Zazwyczaj $m \approx \sqrt{p}$.
3. Dla każdego podzbioru tworzy się drzewo regresyjne i dokonuje się predykcji na jego podstawie.
4. Końcowa predykcja to uśrednienie predykatorów powstałych z wielu podzbiorów.

Lasy losowe są bardzo skuteczne w przypadku, gdy brakuje dużej ilości danych. Potrafią też sobie radzić z dużymi i wielowymiarowymi zbiorami danych oraz równoważyć błędy w danych, gdzie klasy są niezrównoważone.

3.4 Boosting

Boosting to kolejne podejście w celu zwiększenia skuteczności drzew decyzyjnych. Działa podobnie do baggingu, lecz drzewa powstają sekwencyjnie - każde nowe drzewo korzysta z informacji z poprzednich. Boosting polega na udoskonalaniu pierwotnie stworzonych reguł wraz z iteracjami, a nie, jak w baggingu, tworzeniu nowych.

Algorytm boostingu:

1. Ustala się $\hat{f}(x) = 0$ oraz $r_i = y_i$ dla każdego i w zbiorze treningowym.
2. Dla b = 1,2, ... , B, powtarza się:
    a. Dopasowuje się drzewo $\hat{f}^b$ z d podziałami (d + 1 końcowych węzłów) do danych treningowych (X,r)
    b. Aktualizuje się $\hat{f}$ przez dodanie skurczonej wersji nowego drzewa:

$$\hat{f}(x) <- \hat{f}(x) + \lambda \hat{f}^b(x)$$

c. Aktualizuje się reszty

$$r_i <- r_i - \lambda \hat{f}^b(x_i)$$

3. Końcowy model

$$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x) ,$$

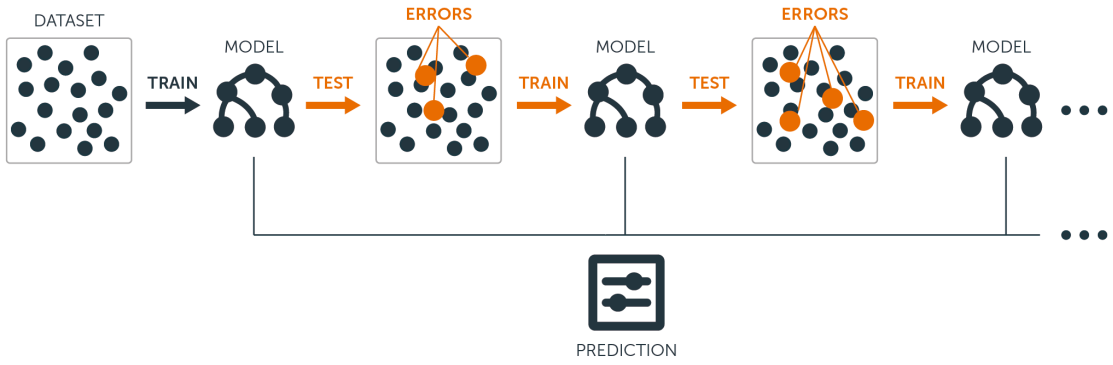
gdzie $\hat{f}(x)$ to końcowy model , r_i to reszta poszczególnej obserwacji, y_i to rzeczywista wartość obserwacji, B to liczba drzew, $\hat{f}^b$ to model drzewa, który jeszcze będzie aktualizowany , d to liczba węzłów w drzewie, zaś λ to parametr kurczenia się.

Metoda boosting uczy się relatywnie wolno. Biorąc obecny model, dopasowuje się drzewo do reszt z niego. W momencie tworzenia nowych drzew bierze się pod uwagę właśnie reszty, a nie zmienną objaśnianą Y. Za pomocą nowo utworzonego drzewa aktualizuje się reszty. Każde z tych drzew może być małe, definiuje to parametr d, który ustala się arbitralnie. W ten sposób powoli poprawia się dokładność $\hat{f}$ w regułach, gdzie były największe problemy. Parametr kurczenia się λ jeszcze bardziej spowalnia ten proces, aby więcej różnych drzew mogło wpłynąć na reszty. W odróżnieniu do baggingu konstrukcja każdego drzewa polega na wykorzystaniu tych, które już powstały.

Można zauważyć, że bardzo ważnym dla tej metody jest wybranie odpowiednich parametrów:

- Liczba drzew B - w przypadku, gdy jest za wysoka może dość do przeuczenia modelu. Używa się walidacji krzyżowej w celu wybrania odpowiedniej wartości.
- Parametr kurczenia się λ to mała dodatnie liczba, kontroluje tempo w którym model uczy się. Typowe wartości to 0.01 lub 0.001. Bardzo mała wartość może wymagać bardzo dużej wartości B, aby osiągnąć dobre wyniki.
- Liczba węzłów d w każdym drzewie kontroluje złożoność algorytmu boostingu. Często d=1 działa dobrze (wtedy drzewo jest pniem składającym się z jednego podziału). W takim przypadku tworzy się modele addytywne. Bardziej ogólnie, d odpowiada za głębokość interakcji i kontrolę kolejności interakcji modelu (skoro d węzłów może zawierać maksymalnie d zmiennych).

Poniżej ilustracja przedstawiająca algorytm boostingu:



Rysunek 3. Boosting [źródło: https://www.datacamp.com/community/tutorials/decision-trees-R]

3.5 Weryfikacja metod

W celu zbadania, który z modeli powstałych za pomocą wyżej wymienionych metod jest najlepszy użyję dwóch miar błędów. Umożliwią mi one porównanie stworzonych przeze mnie modeli. Poniżej przedstawię ich opis teoretyczny.

3.5.1 RMSE

RMSE, czyli pierwiastek błędu średniokwadratowego (ang. *Root Mean Squared Error*) to współczynnik oceny błędu prognozy ex-post. Prognozy ex-post opierają się na znanych wartościach zmiennych objaśnianych. [9]

Poniżej wzór na RMSE.

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - y_t^P)^2},$$

gdzie n to ilość prognozowanych zmiennych, y_t wartości rzeczywiste przewidywanych zmiennych, a y_t^P prognozy przewidywanych zmiennych.

Im mniejszy współczynnik RMSE tym lepiej. Oznacza to, że prognozy średnio różnią się mniej od wartości rzeczywistych.[10]

[9] http://www.prognozowanie.info/ex-ante-post/ (dostęp 04.05.2019)
[10] http://visualmonsters.cba.pl/index.php/prognozowanie/blad-sredniokwadratowy-mse-pierwiastek-bledu-sredniokwadratowego-rmse/ (dostęp 04.05.2019)

3.5.2 MAPE

MAPE, czyli średni bezwzględny błąd procentowy (ang. *Mean Absolute Percentage Error*) to również współczynnik oceny błędu prognozy ex-post. Informuje on o średniej wielkości błędów prognoz dla zbioru testowego, wyrażony jest w procentach. Pozwala on na porównanie dokładności prognoz różnych modeli.[11]

Poniżej wzór służący do obliczania MAPE:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n} \left| \frac{y_t - y_t^P}{y_t} \right| * 100\%,$$

gdzie n to liczba prognozowanych zmiennych, y_t wartości rzeczywiste przewidywanych zmiennych, a y_t^P prognozy przewidywanych zmiennych.

[11] http://visualmonsters.cba.pl/index.php/prognozowanie/blad-e-blad-procentowy-ep-sredni-blad-me-sredni-procentowy-blad-mpe-sredni-blad-bezwzgledny-mae-sredni-bezwzgledny-blad-procentowy-mape/ (dostęp 04.05.2019)

4 Przeprowadzenie badania

W tej części pracy przeprowadzę badanie empiryczne . Zajmę się najpierw przygotowaniem danych, a następnie zastosuję wcześniej wymienione metody badawcze.

4.1 Przygotowanie danych

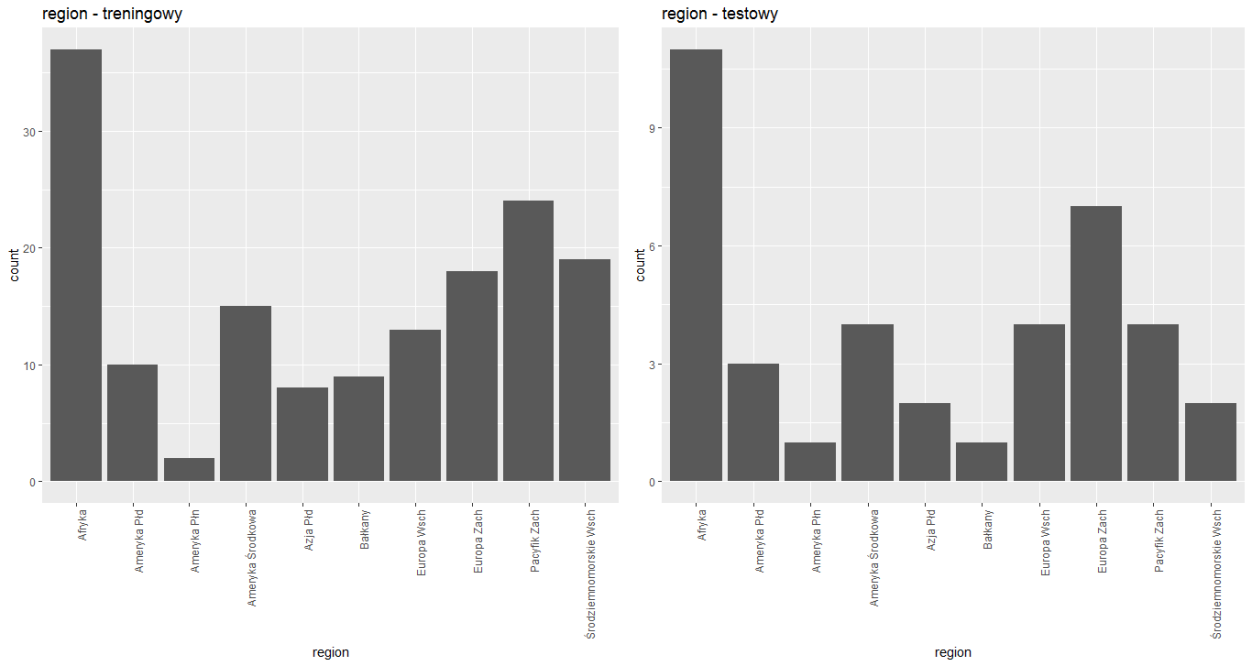
Jak już wcześniej wspomniałam zaletą drzew decyzyjnych jest fakt, że radzą sobie dobrze zarówno z wartościami brakującymi jak i odstającymi. Jest to ich ogromny plus, gdyż ułatwia i przyspiesza proces przygotowania danych.

W celu wykonania badania muszę podzielić zbiór moich danych na testowe i treningowe. Na danych treningowych będę uczyć drzewo, a na danych testowych sprawdzać dokładność predykcji. 80% moich danych przypiszę do zbioru treningowego, a resztę do testowego. Podział ten wykonam losowo, wybierając dane. Poniżej przedstawiam kod, który mi to umożliwia.

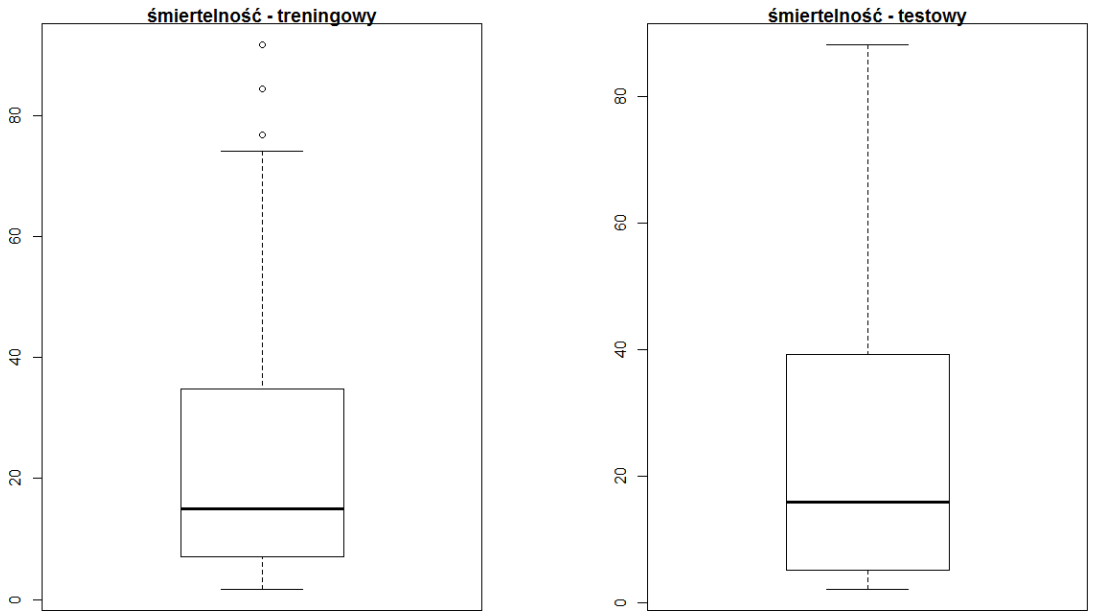
```
division <- sample(nrow(data), round(0.8*nrow(data)), replace = F)
train <- data[division,]
test <- data[-division,]
```

155 obserwacji zostało przypisane do zbioru treningowego, a 39 do testowego.

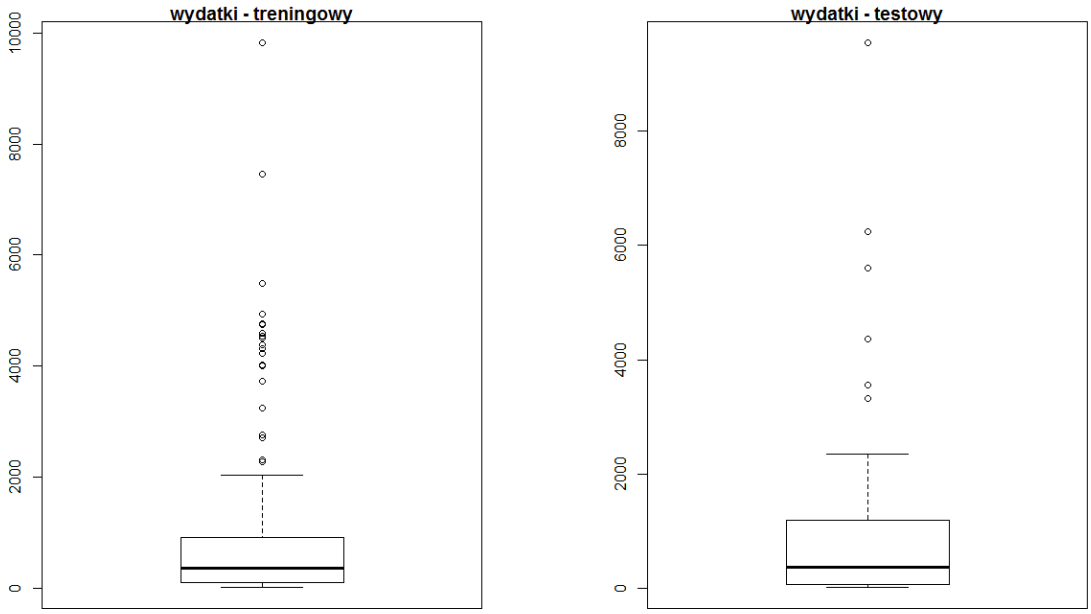
W celu sprawdzenia czy obserwacje w zbiorach są porównywalne przedstawię wykres częstości zmiennej *region* oraz wykresy pudełkowe pozostałych.



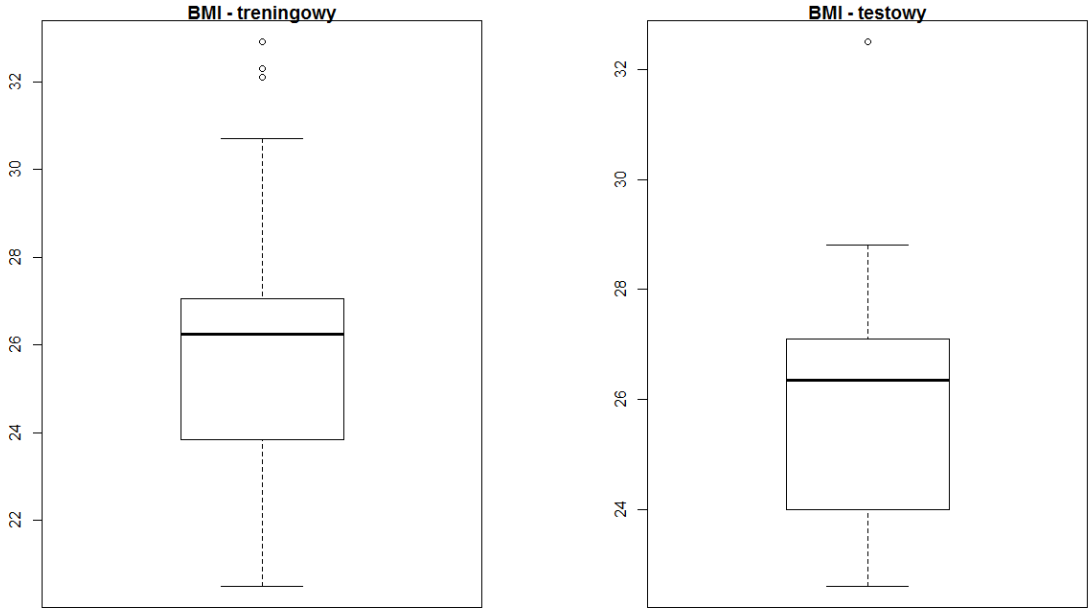
Wykres 3. Zmienna region - porównanie [źródło: opracowanie własne]



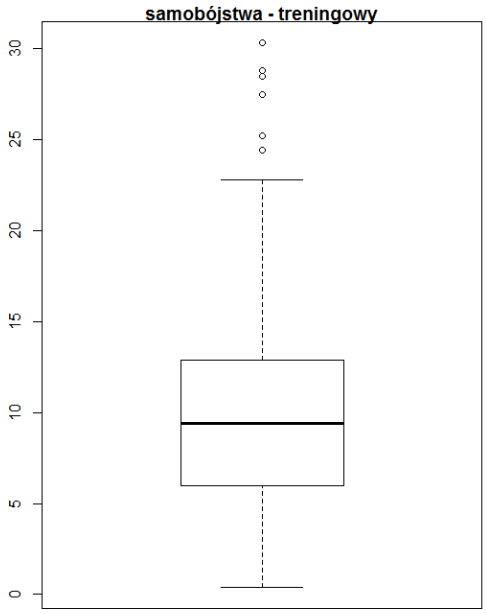
Wykres 4. Zmienna śmiertelność - porównanie [źródło: opracowanie własne]

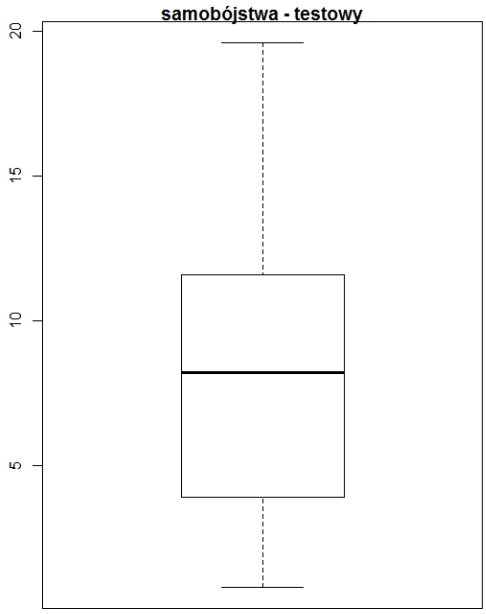


Wykres 5. Zmienna wydatki - porównanie [źródło: opracowanie własne]

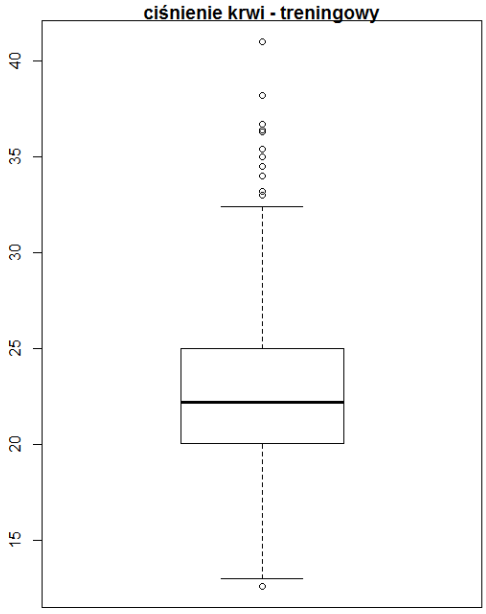


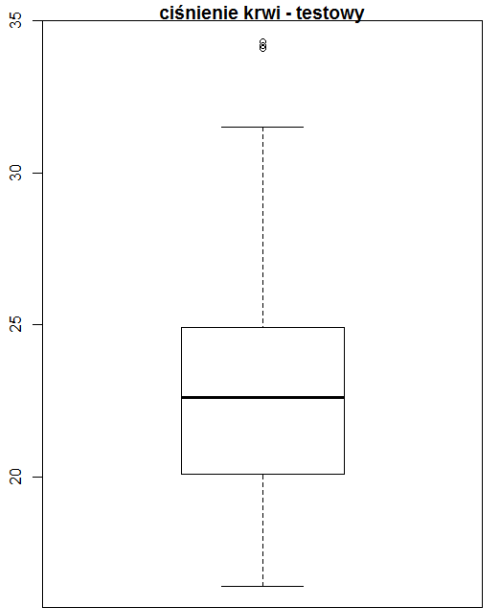
Wykres 6. Zmienna BMI - porównanie [źródło: opracowanie własne]



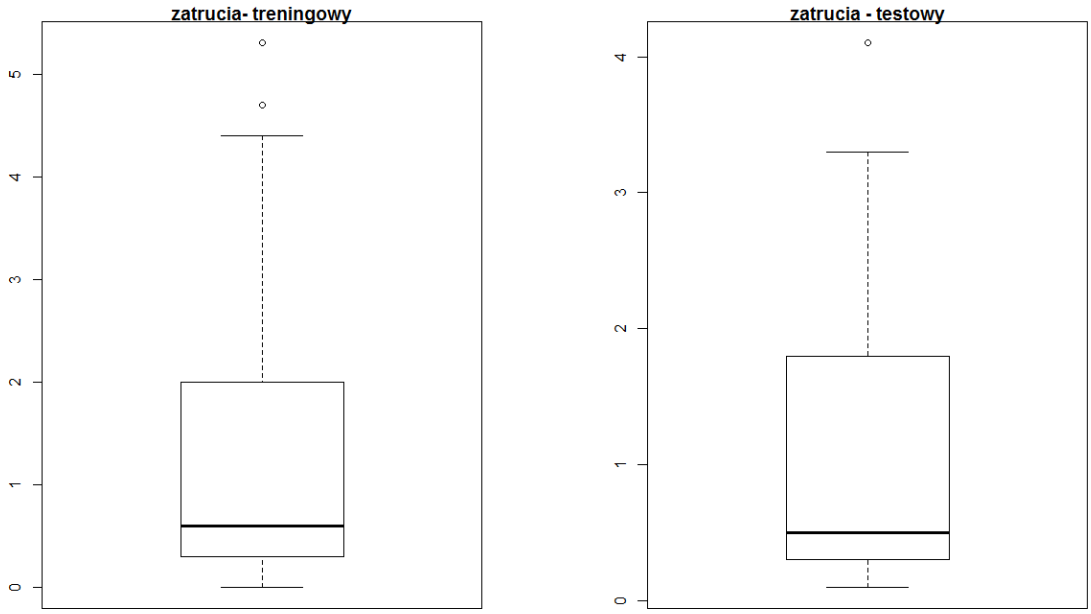


Wykres 7. Zmienna samobójstwa - porównanie [źródło: opracowanie własne]

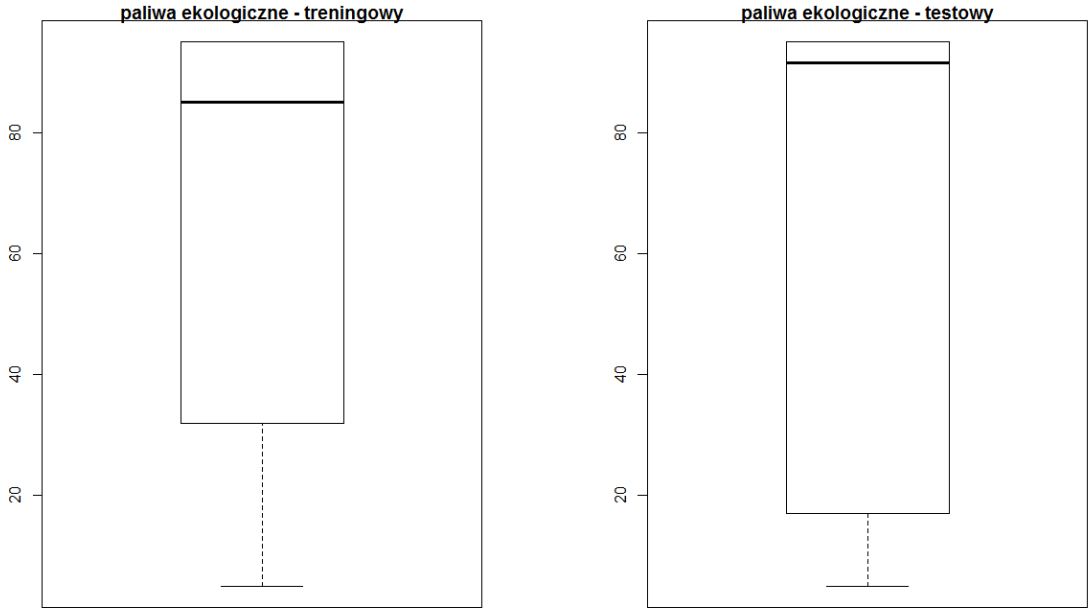




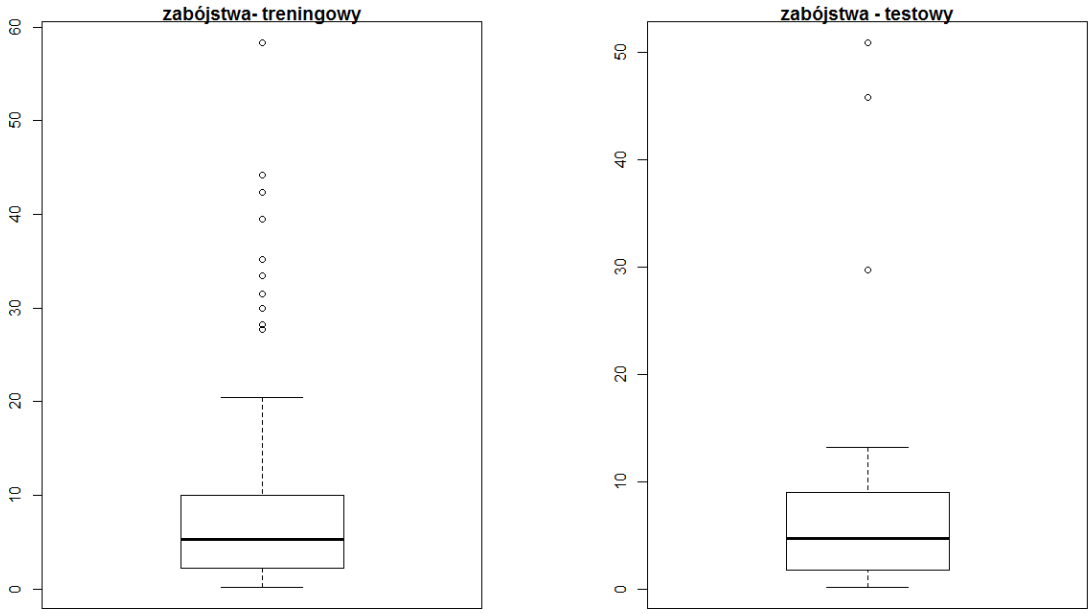
Wykres 8. Zmienna ciśnienie krwi - porównanie [źródło: opracowanie własne]



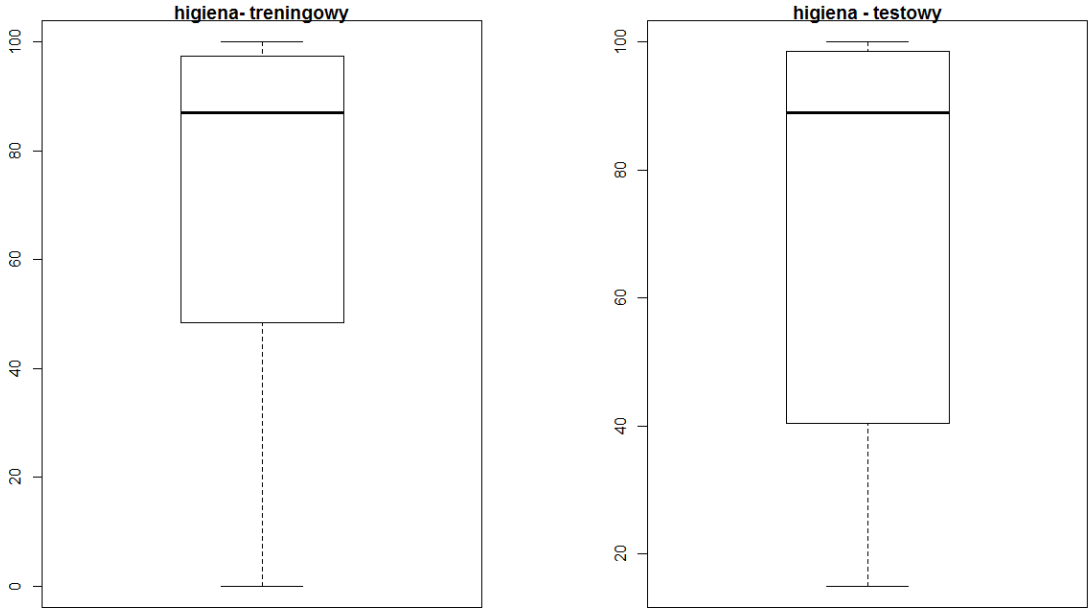
Wykres 9. Zmienna zatrucia - porównanie [źródło: opracowanie własne]



Wykres 10. Zmienna paliwa ekologiczne - porównanie [źródło: opracowanie własne]



Wykres 11. Zmienna zabójstwa - porównanie [źródło: opracowanie własne]



Wykres 12. Zmienna higiena - porównanie [źródło: opracowanie własne]

Oczywistym jest, iż nie jest możliwe stworzenie dwóch podzbiorów z takimi samymi statystykami oraz częstością, lecz można zauważyć na podstawie analizy powyższych wykresów, że zbiory są bardzo podobne. Uznaję więc, że zbiór danych został podzielony odpowiednio do wykonania właściwej części badania.

4.2 Drzewo regresyjne

Badanie zacznę od skonstruowania drzewa regresyjnego. Posłużę się do tego funkcją **rpart** z pakietu o tej samej nazwie. Poniżej przedstawiam strukturę funkcji wraz z parametrami, z których będę korzystać.

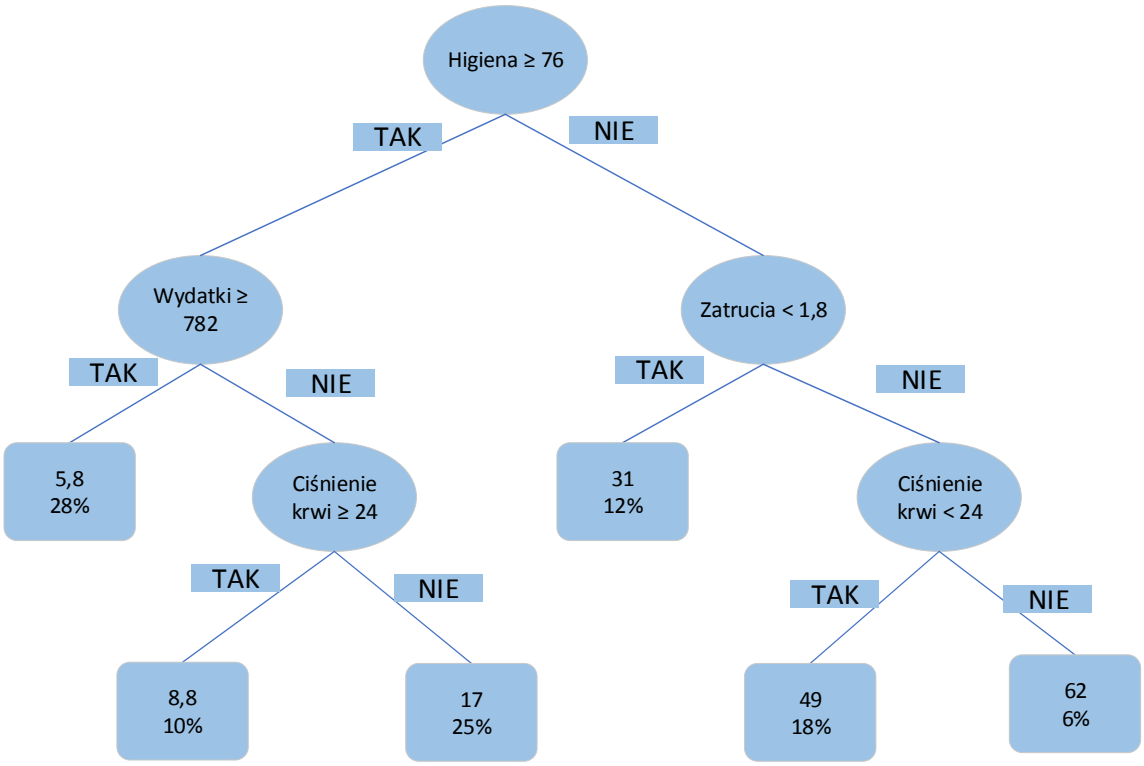
```
rpart(formula, data, method, control)
```

- *formula* - symboliczny opis modelu.
- *data* - macierz danych, która zawiera zmienne użyte w formule.
- *method* - metoda decydująca, w jaki sposób ma być tworzone drzewo. W swojej pracy jako metodę będę używać anovy.
- *control* - parametry, które pozwalają na kontrolowanie algorytmu powstawania drzewa. W swojej pracy będę używać poniższych parametrów:
    - *cp* - parametr złożoności, służy do kontroli wielkości drzewa (domyślnie 0,01).
    - *minsplit* - minimalna liczba obserwacji, która musi istnieć w węźle, aby nastąpił podział (domyślnie 20).
    - *minbucket* - minimalna liczba obserwacji w końcowym liściu (domyślnie *minbucket* =*minsplit* / 3).
    - *maxdepth* - maksymalna głębokość drzewa (domyślnie 30).

Najpierw przedstawię model drzewa, który został stworzony nie zmieniając wartości domyślnych parametrów. Model wykonuję na danych treningowych.

```
tree <- rpart(śmiertelność~.,data=train, method="anova")
```

Za pomocą funkcji **print** jestem w stanie uzyskać opis drzewa. Następnie w celu wizualizacji wyniku tej funkcji rysuje drzewo w programie Visio.



Wykres 13. Drzewo decyzyjne rpart [źródło: opracowanie własne]

Z łatwością można zauważyć, że dla drzewa zmiennymi decydującymi o podziale okazała się być *higiena*, *wydatki*, *zatrucia* oraz *ciśnienie krwi*. Końcowy wynik to średnia wartość wskaźnika śmiertelności noworodków oraz procent obserwacji zbioru treningowego należącego do tego regionu. W przypadku, gdy wskaźnik *higieny* w kraju jest większy bądź równy 76, a *wydatki na medycynę* wynoszą przynajmniej 782$ to wskaźnik śmiertelności noworodków wynosi 5,8. Analogicznie można by opisać pozostałe ścieżki przedstawione na wizualizacji.

Przedstawię teraz przy pomocy funkcji **varImp** z pakietu **caret** wpływ zmiennych na model.

```
imp <- varImp(tree)
imp <- data.frame(zmienna = rownames(imp),wartość = imp$Overall)
imp %>% arrange(desc(wartość))
```

zmienna	wartość
region	1.2976901
wydatki	1.1955063
higiena	1.1077628
zatrucia	0.9566928
paliwa ekologiczne	0.8873903
ciśnienie krwi	0.5961259
zabójstwa	0.4173112
BMI	0.2015833
samobójstwa	0

Tabela 2. Wpływ zmiennych - rpart [źródło: opracowanie własne]

Jak widać mimo tego, iż zmienna *region* nie pojawiła się w drzewie decyzyjnym, ma ona duży wpływ. Jest to spowodowane tym, iż funkcja **varImp** dla takiego modelu oblicza wpływ zmiennych poprzez redukcję funkcji straty (np. średni błąd kwadratowy) przypisanej do każdej zmiennej przy każdym podziale w drzewie. Są one przedstawiane w tabeli i sumowane. Ze względu na to mogą występować ważne zmienne, które nie są używane podczas podziału. Tak jak można było się spodziewać na podstawie modelu, zmienne *wydatki* i *higiena* mają duży wpływ. Zmienna *samobójstwa* nie ma żadnego wpływu, a *BMI* bardzo niewielki.

Oczywistym pytaniem które teraz się nasuwa jest to jak skuteczny będzie powyższy model dla zbioru testowego. Predykcję wykonam za pomocą funkcji **predict**. Natomiast dokładność tych wyników sprawdzę obliczając pierwiastek błędu średniokwadratowego (RMSE) używając funkcji **rmse** z pakietu **ModelMetrics** oraz średni bezwzględny błąd procentowy (MAPE) używając funkcji **MAPE** z pakietu **MLmetrics**.

```
predict <- predict(tree, newdata = test)
rmse(actual = test$śmiertelność, predicted = predict)
```

```
9.144288
```

```
MAPE(y_pred = predict, y_true = test$śmiertelność)
```

```
0.4968637
```

RMSE wyniosło 9,14, czyli prognozy różnią się od rzeczywistych wartości średnio o 9,14 - w rzeczywistości umiera więcej lub mniej 914 noworodków na 100 000 żywych urodzeń. Natomiast MAPE wyniosło około 0,5- oznacza to, że średnia wielkość błędu to 50%. Nie jest

to najlepszy model. Spróbuje teraz poprawić jego wydajność zmieniając parametry funkcji **rpart**.

Za pomocą pętli będę sprawdzać, która kombinacja parametrów sprawia, że RMSE przyjmuje najmniejszą wartość. Parametr *cp* będzie przyjmować wartości: 0; 0,001; 0,01; 0,1. *Minsplit* od 1 do 20, a *maxdepth* od 1 do 30. Poniżej kod, który umożliwia mi wybranie najlepszych parametrów.

```
hyper_grid <- expand.grid(
        cp = c(0,.001,0.01,0.1),
        minsplit=seq(1, 20, 1),
        maxdepth=seq(1, 30, 1))

rmse_err <- c()

for (i in 1:nrow(hyper_grid)) {
        model <- rpart(formula = śmiertelność ~ .,
                data = train,
                method="anova",
                cp = hyper_grid$cp[i],
                minsplit = hyper_grid$minsplit[i],
                maxdepth = hyper_grid$maxdepth[i])

pred <- predict(object = model, newdata = test)

rmse_err[i] = rmse(actual=test$śmiertelność, predicted = pred)
}

opt_i <- which.min(rmse_err)
print(hyper_grid[opt_i,])
```

cp	minsplit	maxdepth
0	12	4

Tabela 3. Optymalne parametry - rpart [źródło: opracowanie własne]

Końcowo parametry przy których model ma najmniejszy błąd RMSE to: *cp* = 0, *minsplit* = 12 oraz *maxdepth* = 4. Tworzę teraz nowy model z wybranymi przeze mnie parametrami oraz obliczam błędy.

```
model <- rpart(formula = śmiertelność ~ .,
        data = train,
        method="anova",
        cp = 0,
        minsplit = 12,
        maxdepth = 4)
pred <- predict(object = model, newdata = test)

rmse(actual=test$śmiertelność, predicted = pred)
```

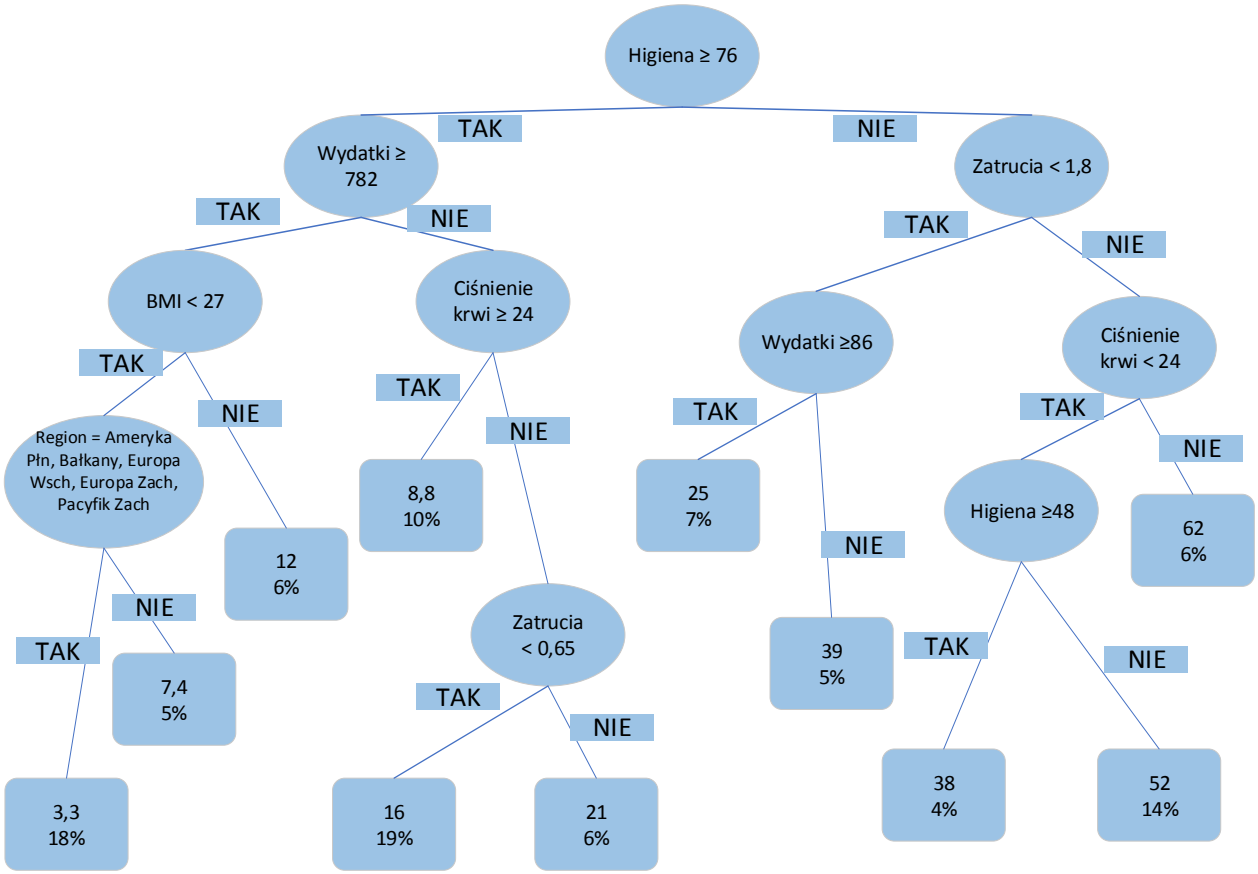
```
8.8321
```

```
MAPE(y_pred = pred, y_true = test$śmiertelność)
```

```
0.401964
```

Można zauważyć, że wartość RMSE i MAPE jest mniejsza niż z parametrami domyślnymi. Dalej jednak nie jest to najlepszy model, gdyż MAPE wynosi 40%.

W celu lepszej wizualizacji opisu modelu uzyskanego za pomocą funkcji **print** wykonam drzewo decyzyjne w programie Visio.



Wykres 14. Drzewo poprawione rpart [źródło: opracowanie własne]

Można zauważyć, że w porównaniu do poprzedniego drzewa wpływ na podział okazują się mieć także zmienne *BMI* oraz *region*. Po wykonaniu tego modelu widać, że zmienna *samobójstwa*, *zabójstwa* oraz *paliwa ekologiczne* nie wpływają na powstawanie drzewa.

Poniżej przedstawię wpływ zmiennych objaśniających na model.

```
imp <- varImp(model)
imp <- data.frame(zmienna = rownames(imp),wartość = imp$Overall)
imp %>% arrange(desc(wartość))
```

zmienna	wartość
wydatki	2.4747922
region	2.4109647
higiena	1.7743534
zabójstwa	1.5492193
zatrucia	1.2209968
BMI	1.1907341
paliwa ekologiczne	1.0842364
ciśnienie krwi	0.7819680
samobójstwa	0.3244071

Tabela 4. Wpływ zmiennych - rpart końcowy [źródło: opracowanie własne]

Ponownie widać, że zmienne *wydatki*, *region* i *higiena* mają największy wpływ. Zmienna *BMI* jest już ważniejsza niż w początkowym modelu, natomiast *samobójstwa* dalej są nieistotne.

4.3 Bagging

Drugą omawianą przeze mnie metodą jest bagging. Do wykonania go posłużę się funkcją o tej samej nazwie z pakietu **ipred**. Poniżej struktura funkcji.

```
bagging(formula, data, na.action=na.rpart, nbagg=25, ...)
```

Przyjmuje ona także parametry *control* analogicznie jak w przypadku funkcji **rpart.** Parametr *nbagg* oznacza liczbę drzew (wykonywanych powtórzeń).

Tak jak w poprzedniej metodzie najpierw zbuduję model z parametrami domyślnymi, a następnie skonstruuje możliwie najlepszy model. W przypadku tej metody będę używać funkcji **set.seed()**, która umożliwi odtworzenie wyników (inaczej za każdym razem drzewa oparte o algorytmy randomizujące byłyby inne).

```
set.seed(222)

bag_tree <- bagging(śmiertelność~., data=train, coob=TRUE)
pred <- predict(object = bag_tree, newdata = test)
rmse(actual=test$śmiertelność, predicted = pred)
```

```
7.569953
```

```
MAPE(y_pred = pred, y_true = test$śmiertelność)
```

```
0.4269637
```

Prognozy od wartości rzeczywistych różnią się średnio o 7,6, średni błąd oszacowania wynosi 43%. Teraz wykonam pętlę, która umożliwi mi wybranie najlepszych parametrów modelu. Będę zmieniać *nbagg* od 25 do 100 z krokiem 25, *maxdepth* od 1 do 30 z krokiem 3, a *minsplit* od 1 do 20 z krokiem 2. Wartości *cp* będą wynosić 0; 0,001; 0,01; 0,1. Kod, który umożliwia mi wybranie najlepszych parametrów przedstawiam poniżej.

```
hyper_grid <- expand.grid(
        cp = c(0,.001,0.01,0.1),
        minsplit=seq(1, 20, 2),
        maxdepth=seq(1, 30, 3),
        nbagg=seq(25,100, 25))

rmse_err <- c()

for (i in 1:nrow(hyper_grid)) {
        set.seed(222)
        model <- bagging(formula = śmiertelność ~ .,
                data = train,
                cp = hyper_grid$cp[i],
                minsplit = hyper_grid$minsplit[i],
                maxdepth = hyper_grid$maxdepth[i],
                nbagg = hyper_grid$nbagg[i])
pred <- predict(object = model, newdata = test)
rmse_err[i] = rmse(actual=test$śmiertelność, predicted = pred)
}
```

Poniżej wartości wybranych przeze mnie parametrów.

```
opt_i <- which.min(rmse_err)
print(hyper_grid[opt_i,])
```

cp	minsplit	maxdepth	nbagg
0	1	1	25

Tabela 5. Optymalne parametry - bagging [źródło: opracowanie własne]

Teraz stworzę model z powyższymi parametrami oraz przedstawię wartości błędów.

```
set.seed(222)

model_bag <- bagging(formula = śmiertelność ~ .,
        data = train,
        cp = 0,
        minsplit = 1,
        maxdepth = 1,
        nbagg = 25)
pred_bag <- predict(object = model_bag, newdata = test)
rmse(actual=test$śmiertelność, predicted = pred_bag)
```

```
7.569953
```

```
MAPE(y_pred = pred_bag, y_true = test$śmiertelność)
```

```
0.4269637
```

Jak widać niestety nie udało się istotnie poprawić modelu. Błędy prognozy nie pomniejszyły się.

W celu zbadania, które zmienne w modelu mają największy wpływ ponownie użyję funkcji **varImp** z pakietu **caret**.

```
imp <- varImp(model_bag)
imp <- data.frame(zmienna = rownames(imp),wartość = imp$Overall)
imp %>% arrange(desc(wartość))
```

zmienna	wartość
wydatki	1.7848170
higiena	1.6699035
region	1.4542129
zatrucia	1.2003859
zabójstwa	1.1169387
paliwa ekologiczne	0.9818312
ciśnienie krwi	0.8043539
BMI	0.7398713
samobójstwa	0.5699946

Tabela 6. Wpływ zmiennych - bagging [źródło: opracowanie własne]

Im wyższa wartość tym zmienna jest bardziej istotna dla modelu – widać więc, że zmienna *wydatki* i *higiena* są najważniejsze. Natomiast zmienna *samobójstwa* oraz *BMI* są mało istotne.

4.4 Lasy losowe

Następną omawianą przeze mnie metodą są lasy losowe. Do wykonania tego modelu użyję funkcji **randomForest** z pakietu o tej samej nazwie. Poniżej struktura funkcji.

```
randomForest(formula, data, mtry, nodesize, ntree)
```

- *na.action* - co należy zrobić w przypadku wartości NA, domyślnie pojawia się błąd
- *mtry* - liczba zmiennych objaśniających brana pod uwagę przy tworzeniu drzew (domyślnie pierwiastek z liczby zmiennych objaśniających)
- *nodesize* - minimalny rozmiar końcowego liścia (domyślnie 5)
- ntree - liczba drzew (domyślnie 500)

W pakiecie **randomForest** znajduje się funkcja, która umożliwia uzupełnienie wartości NA przy pomocy lasów losowych. Użyję jej teraz w celu dokonania dokładniejszej predykcji.

```
set.seed(111)
data_imputed <- rfImpute(śmiertelność~., data)
train_imputed <- data_imputed[division,]
test_imputed <- data_imputed[-division,]
```

Najpierw wykonam model z parametrami domyślnymi. Podobnie jak w przypadku poprzedniej metody użyje funkcji **set.seed()**.

```
set.seed(222)
rforest <- randomForest(śmiertelność ~ ., data=train_imputed)
```

Dokonam teraz predykcji na zbiorze testowym, a następnie obliczę błąd RMSE oraz MAPE.

```
pred <- predict(object = rforest, newdata = test_imputed)
rmse(actual=test_imputed$śmiertelność, predicted = pred)
```

```
7.363881
```

```
MAPE(y_pred = pred, y_true = test_imputed$śmiertelność)
```

```
0.3896514
```

Średnia wielkość błędu predykcji wynosi około 39%. Jest to do tej pory najlepszy wynik. Spróbuję teraz poprawić dokładność prognozy poprzez zmianę domyślnych parametrów modelu. Będę zmieniać wartość *mtry* od 1 do 9 (maksymalnie 9, gdyż taka jest liczba wszystkich zmiennych objaśniających). Wartość *nodesize* będzie przyjmować wartości od 2 do 8, a *ntree* od 100 do 1000 z krokiem 100. Poniżej zamieszczam kod umożliwiający mi wybranie najlepszych parametrów.

```
hyper_grid <- expand.grid(
        mtry=seq(1, 9, 1),
        nodesize=seq(2, 8, 1),
        ntree=seq(100, 1000, 100))
rmse_err <- c()
for (i in 1:nrow(hyper_grid)) {

        set.seed(222)

        model <- randomForest(formula = śmiertelność ~ .,
                data = data_imputed[division,],
                mtry = hyper_grid$mtry[i],
                nodesize = hyper_grid$nodesize[i],
                ntree = hyper_grid$ntree[i])

pred <- predict(object = model, newdata = data_imputed[-division,])
```

```
rmse_err[i] = rmse(actual=data_imputed[-division,1], predicted = pred)
}
```

Poniżej przedstawiam parametry, przy których model ma najmniejszą wartość RMSE.

```
opt_i <- which.min(rmse_err)
print(hyper_grid[opt_i,])
```

mtry	nodesize	ntree
3	4	300

Tabela 7. Optymalne parametry - lasy losowe [źródło: opracowanie własne]

Tworzę teraz model z parametrami *mtree* = 3, *nodesize* = 4 oraz *ntree* = 300.

```
set.seed(222)
model_rf <- randomForest(formula = śmiertelność ~ .,
        data = train_imputed,
        mtry = 3,
        nodesize = 4,
        ntree=300)
```

Na jego podstawię dokonuje predykcji na zbiorze testowym, a następnie obliczam wartość błędu RMSE oraz MAPE.

```
pred_rf <- predict(object = model_rf, newdata = test_imputed)
rmse(actual=test_imputed$śmiertelność, predicted = pred_rf)
```

```
6.9467
```

```
MAPE(y_pred = pred_rf, y_true = test_imputed$śmiertelność)
```

```
0.3719984
```

Wartość błędów niewiele się pomniejszyła. Teraz średni błąd predykcji wynosi 37%.

Na podstawie końcowego modelu (*model_rf*) przedstawię jak kształtuje się wpływ zmiennych na model. Użyję do tego funkcji **importance** z pakietu **randomForest**.

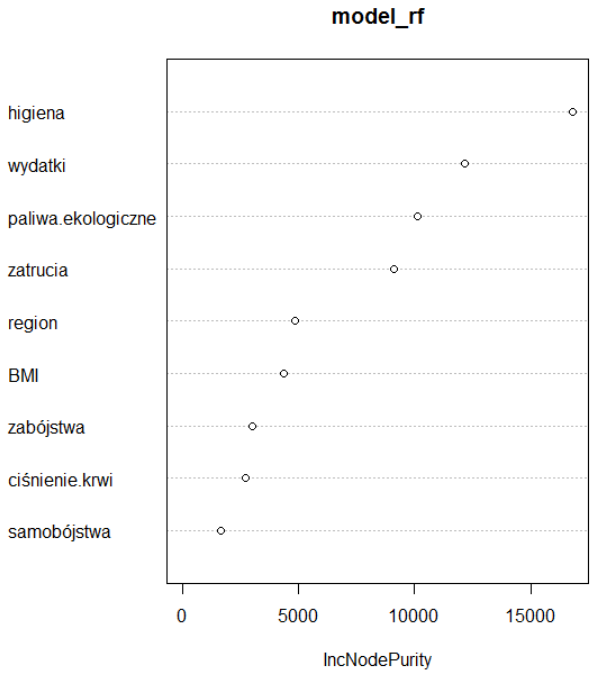
```
importance(model_rf)
```

zmienna	wartość
wydatki	12180.943
BMI	4348.365
samobójstwa	1650.069
ciśnienie krwi	2682.994
zatrucia	9116.859
paliwa ekologiczne	10127.405
zabójstwa	3007.945
higiena	16829.906
region	4812.800

Tabela 8. Wpływ zmiennych - lasy losowe [źródło: opracowanie własne]

W celu lepszej wizualizacji użyję funkcji **varImpPlot** z pakietu **randomForest**.

```
varImpPlot(model_rf)
```



Wykres 15. Wpływ zmiennych lasy losowe [źródło: opracowanie własne]

Widać teraz, że zmienna *higiena* ma największy wpływ na kształtowanie modelu. Zmienne *wydatki, paliwa ekologiczne* oraz *zatrucia* również mają duży wpływ. Pozostałe zmienne są już mniej istotne.

4.5 Boosting

Ostatnią omawianą przeze mnie metodą jest boosting. Do wykonania jej użyję funkcji **gbm** z pakietu o tej samej nazwie. Poniżej struktura funkcji.

```
gbm(formula, distribution, data, n.trees = 100,interaction.depth = 1,
        n.minobsinnode = 10, shrinkage = 0.1, cv.folds = 0)
```

- *distribution* - nazwa wykorzystywanego rozkładu prawdopodobieństwa, będę używać wartości "gaussian"
- *n.trees* - liczba drzew (domyślnie 100)
- *interaction.depth* - maksymalna głębokość drzewa (domyślnie 1)
- *n.minobsinnode* -minimalna liczba obserwacji w końcowym węźle drzewa (domyślnie 10)
- *shrinkage* - parametr kurczenia (domyślnie 0.1)
- *cv.folds* – parametr złożoności walidacji krzyżowej (domyślnie 0)

Zbuduję teraz model z domyślnymi wartościami funkcji. Tak samo jak w poprzednich dwóch metodach będę używać funkcji **set.seed()**.

```
set.seed(222)
boost <- gbm(śmiertelność~., data=train)
```

Dokonam teraz predykcji, a następnie obliczę RMSE oraz MAPE. Wywołanie funkcji **predict** różni się nieco od wcześniejszych metod. Przyjmuje ona jeszcze jeden parametr - *n.trees* - należy podać liczbę drzew, którą użyto do stworzenia modelu.

```
pred<- predict(object = boost,newdata = test, n.trees = 100)

rmse(actual=test$śmiertelność, predicted = pred)
```

```
7.367695
```

```
MAPE(y_pred = pred, y_true = test$śmiertelność)
```

```
0.3260951
```

Błąd RMSE wynosi 7,37, natomiast wartość MAPE jest równa 33% - jest to stosunkowo dobry wynik. Spróbuję teraz dostosować parametry modelu tak by błąd był jak najmniejszy. Parametr *shrinkage* będzie przyjmować wartości 0,001; 0,01; 0,1 oraz 0,3. *Interaction.depth* będzie równy 1, 3 oraz 5, *cv.folds* 0, 2 oraz 4, *n.minobsinnode* 5, 10 oraz 15, a liczba drzew (*n.trees*) będzie przyjmować wartości od 100 do 1000 z krokiem 100. Poniżej kod umożliwiający wybranie parametrów, które minimalizują błąd RMSE.

```
hyper_grid <- expand.grid(
       shrinkage = c(.001,.01, .1, .3),
       interaction.depth = c(1, 3, 5),
```

```
        cv.folds = c(0,2,4),
        n.minobsinnode=c(5,10,15),
        n.trees = seq(100, 1000, 100))
rmse_err <- c()
for (i in 1:nrow(hyper_grid)) {

        set.seed(222)
        model <- gbm(formula = śmiertelność ~ .,
                data = train,
                distribution = "gaussian",
                n.trees=hyper_grid$n.trees[i],
                interaction.depth = hyper_grid$interaction.depth[i],
                shrinkage = hyper_grid$shrinkage[i],
                cv.folds = hyper_grid$cv.folds[i])

pred    <-    predict(object    =    model,newdata    =    test,    n.trees    =
hyper_grid$n.trees[i])

rmse_err[i] = rmse(actual=test$śmiertelność, predicted = pred)
}
```

Poniżej parametry przy których RMSE jest najmniejsze.

```
opt_i <- which.min(rmse_err)
print(hyper_grid[opt_i,])
```

shrinkage	interaction.depth	cv folds	n.minobsinnode	n.trees
0.01	5	2	5	1000

Tabela 9. Optymalne parametry - boosting [źródło: opracowanie własne]

Na podstawię parametrów powyżej tworzę nowy model, a następnie obliczam błędy RMSE oraz MAPE.

```
set.seed(222)

model_boost <- gbm(formula = śmiertelność ~ .,
      data = train,
      distribution = "gaussian",
      n.trees=1000,
      interaction.depth = 5,
      shrinkage = .01,
      cv.folds = 2,
      n.minobsinnode = 5)

pred_boost <- predict(object = model_boost, newdata = test, n.trees=1000)
rmse(actual=test$śmiertelność, predicted =pred_boost)
```

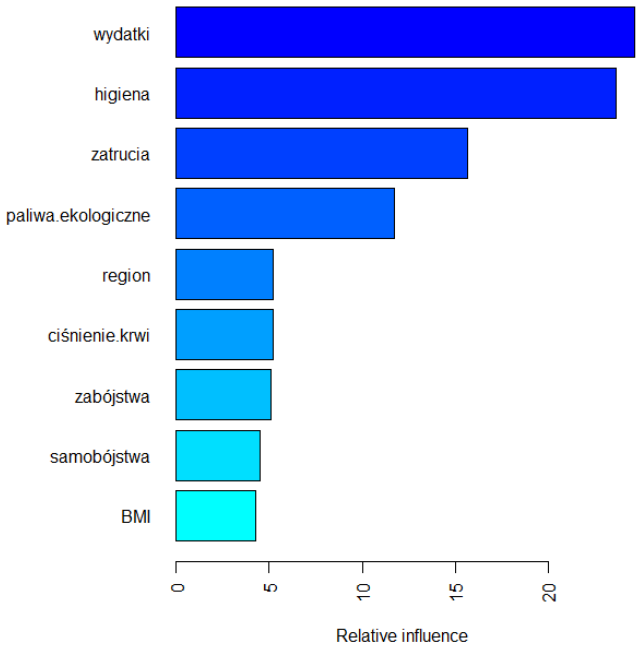
5.865869

```
MAPE(y_pred = pred_boost, y_true = test$śmiertelność)
```

0.3302084

Można zauważyć, że wartość RMSE jest mniejsza o 2 względem domyślnego modelu, natomiast MAPE nie pomniejszyło się.

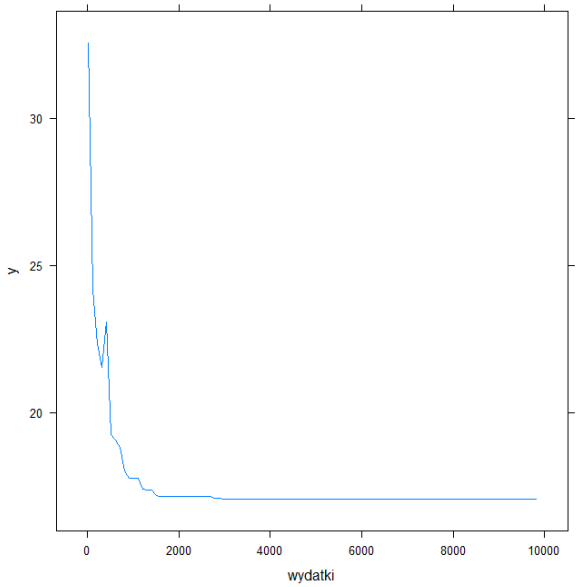
Poniżej wykres wpływu poszczególnych zmiennych na model wykonany za pomocą funkcji **summary**.



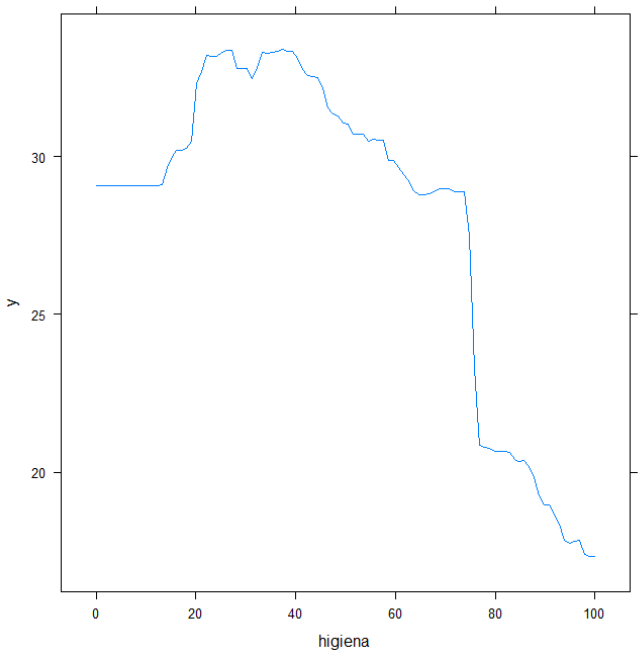
Wykres 16. Wpływ zmiennych boosting [źródło: opracowanie własne]

Można zauważyć, że największy wpływ na śmiertelność noworodków po raz kolejny ma zmienna *wydatki* oraz *higiena*, natomiast najmniejszy zmienna *BMI* i *samobójstwa*.

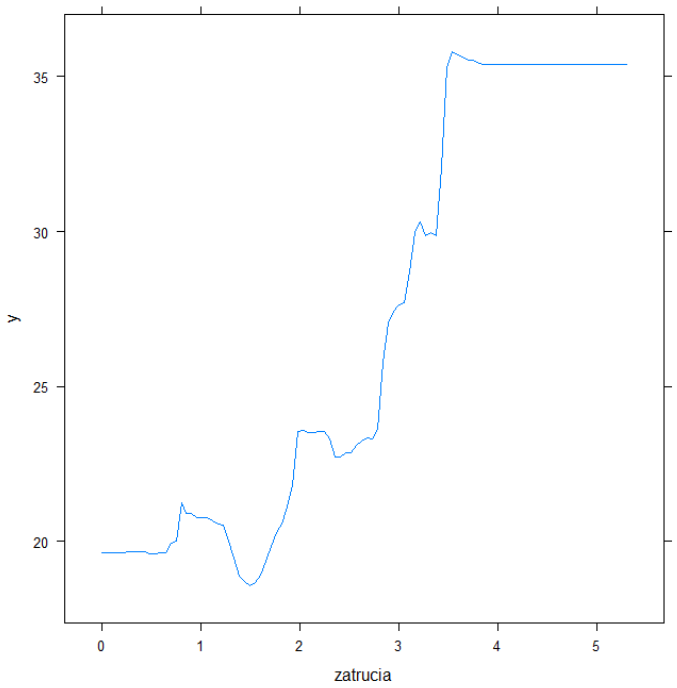
Poniżej wykresy, które przedstawiają wpływ zmiany zmiennych objaśniających na objaśnianą. Wykresy zostały wykonane za pomocą funkcji **plot.gbm**. Funkcja ta tworzy wykres marginalnego efektu wybranej zmiennej poprzez połączenie pozostałych zmiennych. W celu stworzenia tego połączenia funkcja wybiera siatkę punktów i używa ważonej metody przechodzenia drzewa (ang. *tree traversal*).



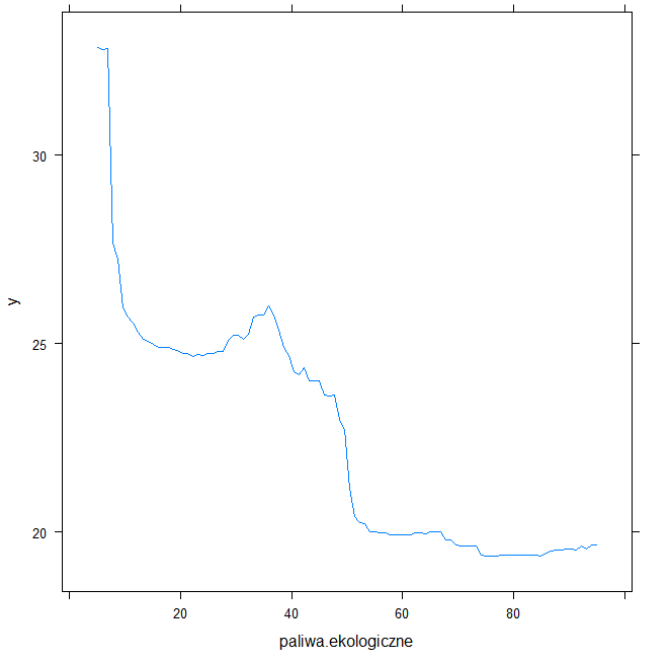
Wykres 17. Wydatki, a śmiertelność [źródło: opracowanie własne]



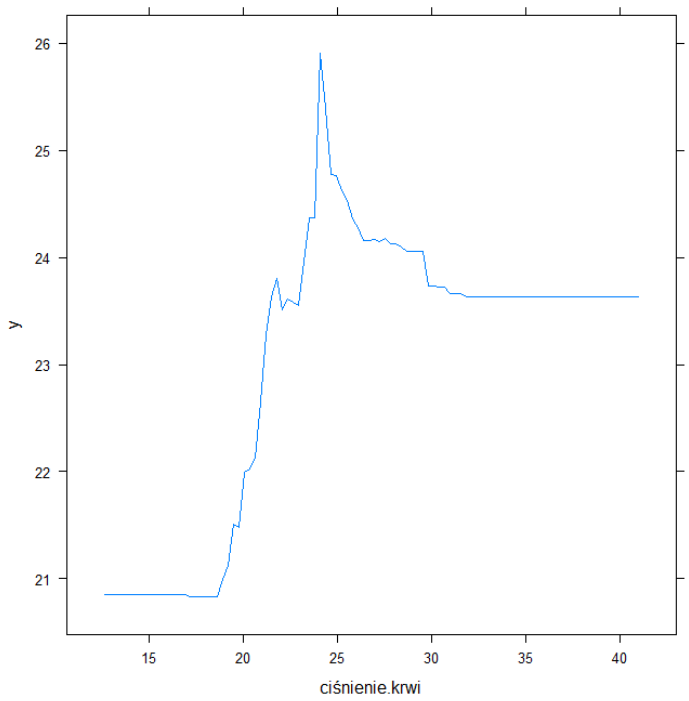
Wykres 18. Higiena, a śmiertelność [źródło: opracowanie własne]



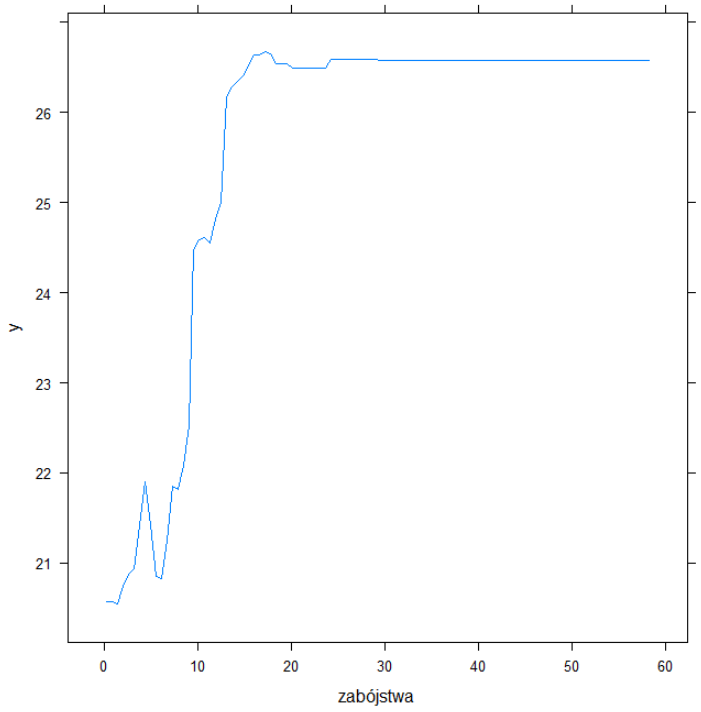
Wykres 19. Zatrucia, a śmiertelność [źródło: opracowanie własne]



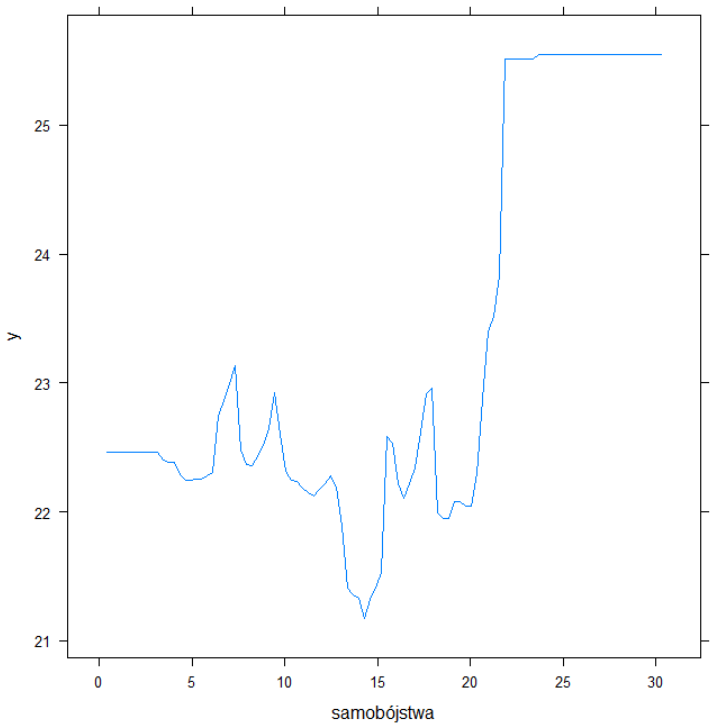
Wykres 20. Paliwa ekologiczne, a śmiertelność [źródło: opracowanie własne]



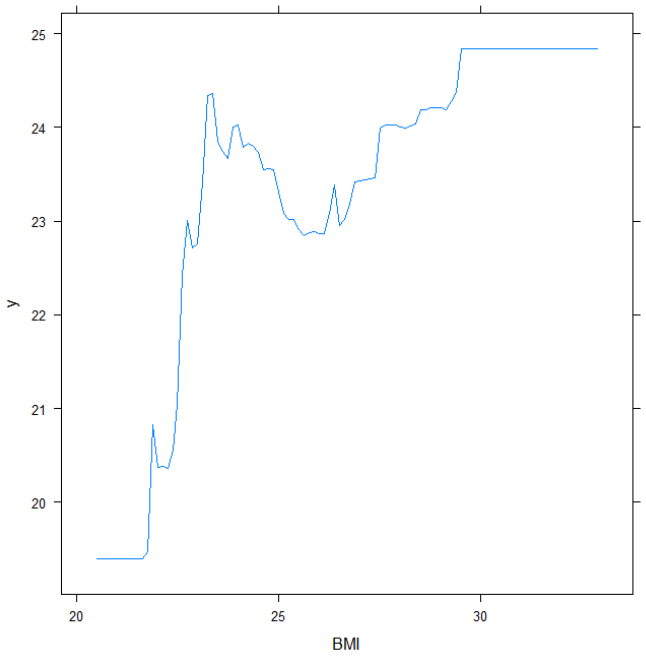
Wykres 21. Ciśnienie krwi, a śmiertelność [źródło: opracowanie własne]



Wykres 22. Zabójstwa, a śmiertelność [źródło: opracowanie własne]



Wykres 23. Samobójstwa, a śmiertelność [źródło: opracowanie własne]



Wykres 24. BMI, a śmiertelność [źródło: opracowanie własne]

Wykresy są przedstawione w kolejności od zmiennych, które mają największy wpływ, do tych z najmniejszym. Widać na pierwszych wykresach, że łatwo znaleźć zależności między

zmienną objaśniającą a objaśnianą. Natomiast na późniejszych wykresach zależności te są bardzo nieregularne. Widać na przykład, że w przypadku zmiennej *samobójstwa* nie da się znaleźć sensownych przedziałów do tworzenia drzewa.

5 Podsumowanie

Jak już wspomniałam we wstępie, w niniejszej pracy licencjackiej miałam dwa cele - zbadać jaka metoda konstruowania drzew decyzyjnych jest najlepsza oraz które czynniki wpływają istotnie na śmiertelność noworodków. Poniżej przedstawiam tabelę ze wskaźnikami oceny błędu RMSE oraz MAPE dla każdej z omawianych metod.

	drzewo regresyjne	**bagging**	**lasy losowe**	**boosting**
RMSE	8,97	7,57	6,95	5,87
MAPE	0,40	0,43	0,37	0,33

Tabela 10. Porównanie błędów [źródło: opracowanie własne]

Widać, że metoda boosting okazała się być najlepszą. Zarówno RMSE jak i MAPE są najniższe ze wszystkich metod. Należy się jednak zastanowić czy otrzymany wynik jest satysfakcjonujący. Mimo, że to jest najlepsza metoda to średnia wielkość błędu to 33%. Może być to spowodowane faktem, iż śmiertelność ma w sobie składnik losowy, którego nie można lepiej przewidzieć w oparciu o zebrane dane. Zaraz za metodą boosting najlepsze są lasy losowe. Najgorsze okazało się być drzewo regresyjne oraz bagging. Zaskakujące jest to, że mimo większej złożoności obliczeniowej od zwykłego drzewa bagging nie dał lepszych wyników. Podsumowując, najlepszą metodą okazała się być metoda boosting, lecz mimo to można się było spodziewać dokładniejszych wyników.

Chciałabym też odpowiedzieć na pytanie, które czynniki są istotne w przypadku śmiertelności noworodków. W wyniku każdej metody otrzymałam wpływ zmiennych objaśniających na objaśnianą.

Wspólną zależnością dla wszystkich modeli jest największy wpływ zmiennych *wydatki* i *higiena*. W dwóch pierwszych modelach (tabela 4 oraz 6) ważną okazuje się być zmienna *region*, jednak w dwóch późniejszych (tabela 8 oraz wykres 16) nie jest ona, aż tak ważna. Można również stwierdzić, że zmienne *BMI, samobójstwa* i *ciśnienie krwi* nie wpływają istotnie na śmiertelność noworodków. Pozostałe zmienne - *paliwa ekologiczne, zabójstwa* i *zatrucia* - w zależności od modelu okazują się mieć większy lub mniejszy wpływ. Analizując wszystkie modele można przypisać zmienną *zabójstwa* do tych mniej istotnych, natomiast wpływ *paliw ekologicznych* oraz *zatruć* do bardziej istotnych.

Dość oczywistym wydaje się być fakt dużego wpływu na śmiertelność noworodków zmiennych *wydatki* i *higiena*, gdyż przedstawiają one w dużym stopniu stopień zaawansowania medycyny danego państwa. Mimo przydzielenia krajów do regionów są one dość różnorodne, dlatego dwie wcześniej wymienione zmienne dosyć wyraźnie wpływają na zmienną objaśnianą. Ciekawą zależnością wydaje się być dość duży wpływ *paliw ekologicznych* oraz *zatruć* na *śmiertelność*. Może być to spowodowane tym, że przedstawiają one stopień zanieczyszczenia środowiska, który jak widać ma zauważalny wpływ również na śmiertelność noworodków.

6 Bibliografia

1. M. Walesiak, E. Gantar *Statystyczna analiza danych z wykorzystaniem programu R* Wydawnictwo Naukowe PWN Warszawa 2009
2. G. James, D. Witten, T. Hastie, R. Tibshirani *An Introduction to Statistical Learning* Springer Science + Business Media New York 2013
3. P. Biecek *Przewodnik po pakiecie R* Wydawnictwo GIS 2017
4. http://visualmonsters.cba.pl/index.php/prognozowanie (dostęp 04.05.2019)
5. http://www.prognozowanie.info/ex-ante-post/ (dostęp 04.05.2019)
6. https://www.datacamp.com/community/tutorials/decision-trees-R (dostęp 10.04.2019)
7. https://pl.wikipedia.org/wiki/Sprawdzian_krzy%C5%BCowy#Prosta_walidacja (dostęp 10.06.2019)

7 Spis tabel, wykresów i ilustracji

7.1 Spis tabel

7.2 Spis wykresów

7.3 Spis ilustracji