STATYSTYCZNA ANALIZA DANYCH
INFORMATYKA I EKONOMETRIA
PROJEKT 2

BARTŁOMIEJ ORTYL

W moim projekcie, głównym celem będzie zastosowanie dwóch metod uczenia maszynowego na zbiorze danych dotyczących cukrzyków.

Projekt będzie składał się kolejno z : wstępu zawierającego cel pracy, opisu i wstępnej analizy danych, odpowiedniego przygotowania danych, podziału zbioru na uczący i testowy, zastosowania minimum 2 metod uczenia maszynowego, analizy uzyskanych wyników i podsumowania.

Metody uczenia maszynowego, których użyje to metoda k najbliższych sąsiadów oraz metoda Bayesa.

Sprawozdanie z mojego projektu, głównie będzie się skupiać na analizie danych i wyników.

OPIS I WSTEPNA ANALIZA DANYCH

	pregnant	glucose	pressure	triceps	insulin	mass	pedigree	age	diabetes
1	6	148	72	35	0	336	0,627	50	pos
2	1	85	66	29	0	266	0,351	31	neg
3	8	183	64	0	0	233	0,672	32	pos
4	1	89	66	23	94	281	0,167	21	neg
5	0	137	40	35	168	431	2,288	33	pos
6	5	116	74	0	0	256	0,201	30	neg
7	3	78	50	32	88	31	0,248	26	pos
8	10	115	0	0	0	353	0,134	29	neg
9	2	197	70	45	543	305	0,158	53	pos
10	8	125	96	0	0	0	0,232	54	pos
11	4	110	92	0	0	376	0,191	30	neg
12	10	168	74	0	0	38	0,537	34	pos
13	10	139	80	0	0	271	1,441	57	neg
14	1	189	60	23	846	301	0,398	59	pos
15	5	166	72	19	175	258	0,587	51	pos
16	7	100	0	0	0	30	0,484	32	pos
17	0	118	84	47	230	458	0,551	31	pos
18	7	107	74	0	0	296	0,254	31	pos
19	1	103	30	38	83	433	0,183	33	neg
20	1	115	70	30	96	346	0,529	32	pos
21	3	126	88	41	235	393	0,704	27	neg
22	8	99	84	0	0	354	0,388	50	neg
23	7	196	90	0	0	398	0,451	41	pos
24	9	119	80	35	0	29	0,263	29	pos
25	11	143	94	33	146	366	0,254	51	pos
26	10	125	70	26	115	311	0,205	41	pos
27	7	147	76	0	0	394	0,257	43	pos

Showing 1 to 28 of 768 entries, 9 total columns

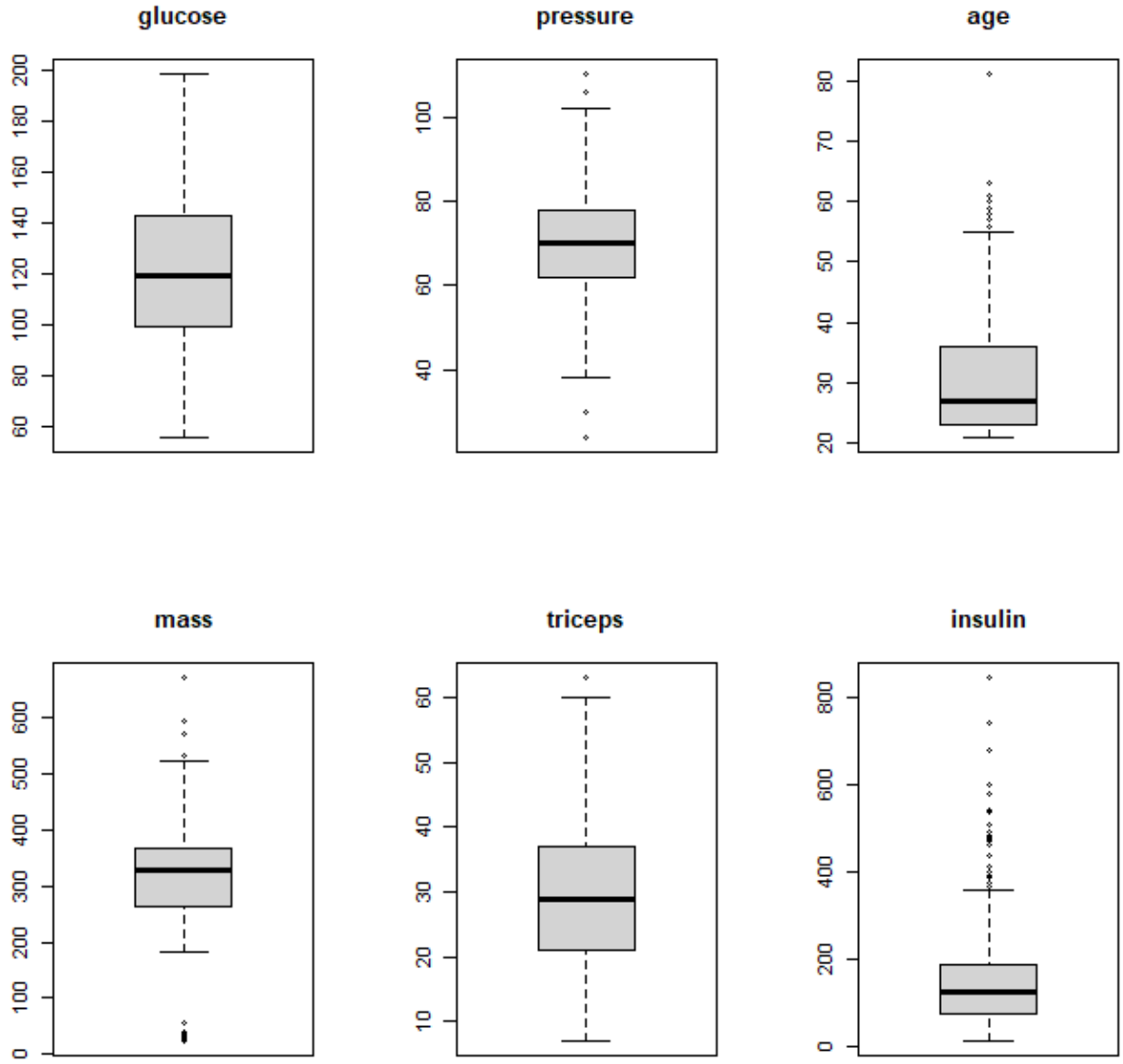
Dane dotyczące cukrzyków, których będę używał w moim projekcie przedstawiają 8 zmiennych objaśniających, i jedną zmienną objaśnianą, która mówi nam czy pacjent jest cukrzykiem, czy nie. Zmienne objaśniające przedstawiają najważniejsze parametry zdrowotne, potrzebne do zdiagnozowania cukrzycy u człowieka, do tych parametrów należą chociażby: grubość skóry na tricepsie, wiek pacjenta, stosunek wagi do wzrostu, ciśnienie krwi czy chociażby glukoza.

Podstawowe statystyki

```
summary(dane)
diabetes.glucose diabetes.pressure diabetes.mass    diabetes.age   diabetes.triceps diabetes.insulin
Min.   :  0.0    Min.   :  0.00    Min.   :  0.0   Min.   :21.00   Min.   : 0.00    Min.   :  0.0
1st Qu.: 99.0    1st Qu.: 62.00    1st Qu.:251.8   1st Qu.:24.00   1st Qu.: 0.00    1st Qu.:  0.0
Median :117.0    Median : 72.00    Median :309.0   Median :29.00   Median :23.00    Median : 30.5
Mean   :120.9    Mean   : 69.11    Mean   :289.8   Mean   :33.24   Mean   :20.54    Mean   : 79.8
3rd Qu.:140.2    3rd Qu.: 80.00    3rd Qu.:359.0   3rd Qu.:41.00   3rd Qu.:32.00    3rd Qu.:127.2
Max.   :199.0    Max.   :122.00    Max.   :671.0   Max.   :81.00   Max.   :99.00    Max.   :846.0
```

Na powyższym obrazku widzimy podstawowe statystyki opisowe wybranych przeze mnie zmiennych objaśniających, potrzebnym do mojego projektu, i przeprowadzenie dwóch metod maszynowych. Zdecydowałem się na te właśnie zmienne, ponieważ uważam że wpływają one najbardziej na osąd kto jest cukrzykiem, a kto nie. W statystykach nie widać nic nadzwyczajnego, ani niepokojącego, dlatego przejdę do następnej części projektu.

Wykresy pudełkowe zmiennych



Z wykresów można odczytać, że wartości odstające są bardzo rzadkie i pojedyncze, dlatego nie usuwam żadnych wartości.

Macierz korelacji

Zmienne nie są ze sobą skorelowane na doskonałym poziomie, jednakże postaram się przeprowadzić rzetelną i odpowiednią implementacje metod z ich wykorzystaniem.

Modyfikacja danych

Usuwam wiersze, w których występują zera, ponieważ zmienne o wartości zero można traktować jako wartości brakujące.

```
cukrzyca <- dane[!(dane$diabetes.glucose==0 | dane$diabetes.pressure== 0 |
                     dane$diabetes.mass == 0 | dane$diabetes.age == 0 | dane$diabetes.insulin == 0 | dane$diabetes.triceps == 0),]
```

PODZIAŁ NA ZBIÓR UCZĄCY I TESTOWY

```
set.seed(9)

index <- sample(392,300,replace = F)
uczacy.cukrzyca <- cukrzyca[index,]
testowy.cukrzyca <- cukrzyca[-index,]
```

Dzielimy zbiór na uczący i testowy, przyjmuje się że zbiór uczący może mieć od 70-90% danych, natomiast zbiór testowy od 10-30% danych.

Standaryzacja danych ze zbioru testowego i uczącego

```
# standaryzacja danych
uczacy.cukrzyca1 <- as.data.frame(scale(uczacy.cukrzyca[,-7]))
uczacy.cukrzyca1$diabetes.diabetes <- uczacy.cukrzyca$diabetes.diabetes
testowy.cukrzyca1 <- as.data.frame(scale(testowy.cukrzyca[,-7]))
testowy.cukrzyca1$diabetes.diabetes <- testowy.cukrzyca$diabetes.diabetes
```

Do przeprowadzenia metod uczenia maszynowego, dane poszczególnych zmiennych muszą być na podobnej skali, dlatego dokonuje standaryzacji danych przed dalszą analizą.

Standaryzuje dopiero po podziale na zbiór uczący i testowy, ponieważ model będę budował na zbiorze uczącym, więc chce żeby elementy ze zbioru testowego w żaden sposób nie wpływały na zbiór uczący.

Statystyki zbioru uczącego i testowego

```
> summary(uczacy.cukrzyca)
 diabetes.glucose diabetes.pressure diabetes.mass    diabetes.age   diabetes.triceps diabetes.insulin
 Min.   : 56.0    Min.   : 24.00    Min.   : 24.0   Min.   :21.00   Min.   : 7.00    Min.   : 14.0
 1st Qu.: 99.0    1st Qu.: 62.00    1st Qu.:268.8   1st Qu.:23.00   1st Qu.:20.75    1st Qu.: 75.0
 Median :118.5    Median : 70.00    Median :326.0   Median :27.00   Median :29.00    Median :125.5
 Mean   :122.3    Mean   : 70.56    Mean   :309.8   Mean   :30.72   Mean   :29.00    Mean   :158.7
 3rd Qu.:142.0    3rd Qu.: 78.50    3rd Qu.:367.2   3rd Qu.:35.00   3rd Qu.:36.00    3rd Qu.:193.2
 Max.   :198.0    Max.   :110.00    Max.   :671.0   Max.   :81.00   Max.   :63.00    Max.   :846.0
> summary(testowy.cukrzyca)
 diabetes.glucose diabetes.pressure diabetes.mass    diabetes.age   diabetes.triceps diabetes.insulin
 Min.   : 68.0    Min.   : 48.00    Min.   : 26.0   Min.   :21.00   Min.   :10.00    Min.   : 15.0
 1st Qu.:101.5    1st Qu.: 62.00    1st Qu.:252.0   1st Qu.:23.00   1st Qu.:22.00    1st Qu.: 86.5
 Median :121.5    Median : 70.00    Median :332.0   Median :27.00   Median :29.00    Median :127.5
 Mean   :123.8    Mean   : 71.01    Mean   :302.3   Mean   :31.34   Mean   :29.63    Mean   :147.3
 3rd Qu.:144.0    3rd Qu.: 78.00    3rd Qu.:369.5   3rd Qu.:37.25   3rd Qu.:37.00    3rd Qu.:180.0
 Max.   :195.0    Max.   :106.00    Max.   :532.0   Max.   :61.00   Max.   :56.00    Max.   :540.0
```

Statystyki powinny być do siebie zbliżone, i jak widać na załączonym wyżej obrazku są one do siebie bardzo podobne, co oznacza że wszystko jest tak jak powinno być.

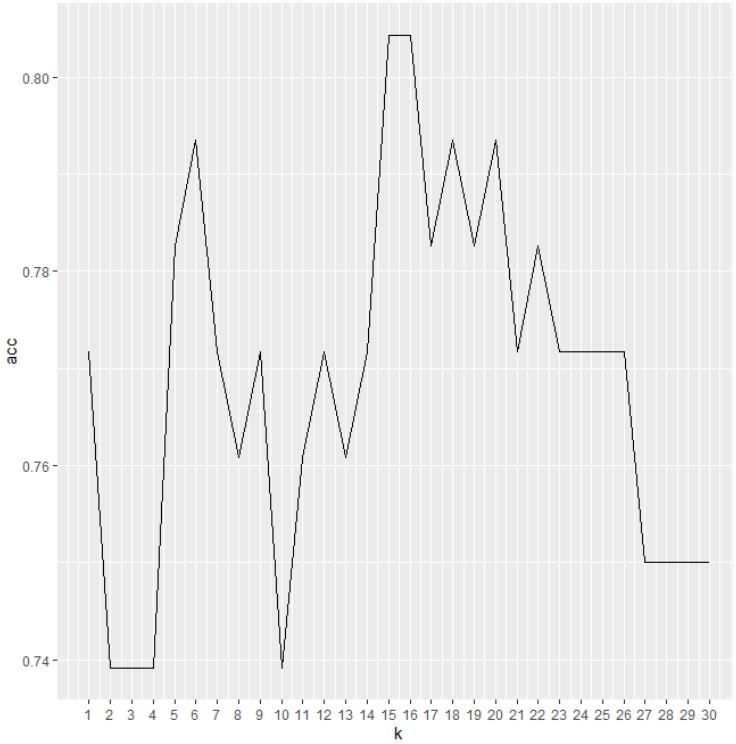
METODA K NAJBLIŻSZYCH SĄSIADÓW (KNN)

Wybór k dla KNN

Szukam takiego k, dla którego będę mógł zmaksymalizować dokładność na zbiorze testowym.

Tworze pętle i buduje model dla różnych k.

```
#wybór k dla knn

acc <- c()
for (i in 1:30){

  cukrzyca.knn <- knn(train = uczacy.cukrzyca1[,-7],          #wszystkie poza zmienna y
                      test = testowy.cukrzyca1[,-7],
                      cl = uczacy.cukrzyca1[, 7],          # okresla co prognozujemy, u mnie 7 kol
                      k = i )

  t = table(cukrzyca.knn, testowy.cukrzyca1[, 7])
  acc[i] = (t[1,1] + t[2,2])/sum(t)
}

library(ggplot2)

ggplot(data.frame(acc, k =1:30))+
  geom_line(aes(x = k, y = acc))+
  scale_x_continuous(breaks = 1:30)
acc[16]
```



Otrzymuje wykres, który przedstawia zmianę dokładności na zbiorze testowym dla różnej liczby sąsiadów.

Najlepsza dokładność wyszła dla liczby sąsiadów równej 15 lub 16.

Mimo wyników przedstawionych na wykresie, sprawdzając każdą liczbę k osobno ręcznie, okazało się że najlepsze dokładności wyszły dla k równego 3, dlatego zdecydowałem przeprowadzić metodę KNN najbliższych sąsiadów dla k = 3.

Funkcja knn zbiór uczący

W pakiecie ‘’class’’ mam dostępną funkcje ‘knn’, która przyjmuje 4 argumenty. Funkcja zwraca bezpośrednio wartości prognozy, czyli konkretnie mówi nam czy dany pacjent jest cukrzykiem, czy nie.

```
library(class)

knn.uczacy <- knn(train = uczacy.cukrzyca1[,-7],          #wszystkie poza zmienna y
                  test = uczacy.cukrzyca1[,-7],
                  cl = uczacy.cukrzyca1[, 7],           # okresla co prognozujemy, u mnie 7 kol
                  k = 3                               # liczba sasiadow ktora chce uwzglednic
                    )
```

Macierz błędów zbiór uczący

Korzystam z klasyfikacji, więc wyniki oceniam na podstawie macierzy błędów, które przedstawiane są w macierzach kwadratowych, lub wyższych w zależności od tego ile mamy zmiennych.

Ja stworze tą macierz za pomocą funkcji table.

```
#macierz bledu

tmb<- table(knn.uczacy, uczacy.cukrzyca1$diabetes.diabetes)
tmb
```

Uzyskuje tabele, w której ma zestawienie wartości prognozowanych w wierszach, oraz wartości rzeczywistych, które ustawione są w kolumnach.

```
> tmb

knn.uczacy neg pos
       neg 180  28
       pos  20  72
```

Dokładność na zbiorze uczącym

Dokładność opisuje jaka część obserwacji została poprawnie sklasyfikowana. Sumuje przekątne i dziele przez całość, aby obliczyć jaki procent obserwacji został dobrze sklasyfikowany.

```
> #dokladnosc na zbiorze uczacym
> (tmb[1,1] + tmb[2,2])/sum(tmb)
[1] 0.84
>
```

Model zachowuje się bardzo dobrze, ponieważ aż 84 % obserwacji zostało dobrze sklasyfikowanych.

Funkcja knn zbiór testowy

```
#zbiór testowy

knn.testowy <- knn(train = testowy.cukrzyca1[,-7],
                   test = testowy.cukrzyca1[,-7],
                   cl = testowy.cukrzyca1[, 7],
                   k = 3
)
knn.testowy
```

Macierz błędów zbiór testowy

```
> tmb2<- table(knn.testowy, testowy.cukrzyca1$diabetes.diabetes)
> tmb2

knn.testowy neg pos
        neg  58   6
        pos   4  24
```

Dokładność na zbiorze testowym

```
        pos   4  24
> #dokladnosc na zbiorze testowym
> (tmb2[1,1] + tmb2[2,2])/sum(tmb2)
[1] 0.8913043
```

Model zachowuje się bardzo dobrze, ponieważ aż 89 % obserwacji zostało dobrze sklasyfikowanych. Jest wyższa niż na zbiorze uczącym.

KLASYFIKATOR NAIWNY BAYESA

Metoda ta polega na przewidywaniu prawdopodobieństwa przynależności danego obiektu do danej klasy. Za pomocą konkretnych wzorów liczymy prawdopodobieństwo przynależności do konkretnej klasy i na podstawie tych wyników dokonywana jest ostateczna prognoza.

Zmiana zmiennych na kategoryczne

Na moich zmiennych ilościowych będę musiał zastosować zmianę charakteru zmiennej z ilościowych na kategoryczne.

```
bayes.cukrzyca <- as.data.frame(cukrzyca)

for (i in 1:nrow(bayes.cukrzyca))
{
  if(bayes.cukrzyca[i,1] < 140)
  {
    bayes.cukrzyca[i,1] = 1 # dobra glukoza
  }
  else
  {
    bayes.cukrzyca[i,1] = 2 #  nieprawidlowa glukoza
  }
}
```

Na załączonym wyżej obrazku przedstawiam zmianę zmiennych na kategoryczne dla tylko jednej zmiennej, robię tak dla wszystkich, których będę używał w tej metodzie. W tym konkretnym przypadku dziele zmienną na dwie grupy, czyli na grupę osób z odpowiednim poziomem glukozy we krwi i z nieodpowiednim poziomem.

Podział na zbiór uczący i testowy

```
# Podzial danych na zbior uczacy i testowy bayes

index.bayes <- sample(392, 300, replace = F)
uczacy.bayes <- bayes.cukrzyca[index.bayes,]
testowy.bayes <- bayes.cukrzyca[-index.bayes,]
```

Statystyki

```
> #statystyki
> summary(uczacy.bayes)
    Glukoza       Cisnienie       Insulina          BMI            Wiek       Klasyfikacja
 Min.   :1.00   Min.   :1.000   Min.   :1.000   Min.   :1.00   Min.   :1.000   Length:300
 1st Qu.:1.00   1st Qu.:1.000   1st Qu.:3.000   1st Qu.:1.00   1st Qu.:3.000   Class :character
 Median :1.00   Median :1.000   Median :3.000   Median :3.00   Median :3.000   Mode  :character
 Mean   :1.26   Mean   :1.093   Mean   :2.813   Mean   :2.28   Mean   :2.797
 3rd Qu.:2.00   3rd Qu.:1.000   3rd Qu.:3.000   3rd Qu.:3.00   3rd Qu.:3.000
 Max.   :2.00   Max.   :3.000   Max.   :3.000   Max.   :3.00   Max.   :3.000
> summary(testowy.bayes)
    Glukoza       Cisnienie       Insulina          BMI            Wiek       Klasyfikacja
 Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000   Length:92
 1st Qu.:1.000   1st Qu.:1.000   1st Qu.:3.000   1st Qu.:1.000   1st Qu.:3.000   Class :character
 Median :1.000   Median :1.000   Median :3.000   Median :3.000   Median :3.000   Mode  :character
 Mean   :1.326   Mean   :1.098   Mean   :2.815   Mean   :2.413   Mean   :2.707
 3rd Qu.:2.000   3rd Qu.:1.000   3rd Qu.:3.000   3rd Qu.:3.000   3rd Qu.:3.000
 Max.   :2.000   Max.   :3.000   Max.   :3.000   Max.   :3.000   Max.   :3.000
```

Klasyfikator naiwny Bayesa zbiór uczący

```
> cukrzyca.nb <- naiveBayes(uczacy.bayes$Klasyfikacja~., data = uczacy.bayes)
> cukrzyca.nb

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
      neg       pos
0.6833333 0.3166667

Conditional probabilities:
     Glukoza
Y         [,1]      [,2]
  neg 1.121951 0.3280305
  pos 1.557895 0.4992716

     Cisnienie
Y         [,1]      [,2]
  neg 1.087805 0.3861544
  pos 1.105263 0.3711307

     Insulina
Y         [,1]      [,2]
  neg 2.765854 0.5889546
  pos 2.915789 0.2791765

     BMI
Y         [,1]      [,2]
  neg 2.063415 1.0004303
  pos 2.747368 0.6679346

     Wiek
Y         [,1]      [,2]
  neg 2.712195 0.6860641
  pos 2.978947 0.1443214
```

Uzyskujemy prawdopodobieństwa, pierwsze prawdopodobieństwo jest obliczone na podstawie moich danych, z których wynika, że blisko 32 % osób z mojego zbioru danych ma pozytywny wynik na cukrzyce.

Poniżej natomiast, widać prawdopodobieństwa warunkowe dla każdej zmiennej z osobna.

Prognoza na zbiorze uczącym

Chcąc uzyskać końcowe prognozy korzystam z funkcji predict, dzięki której uzyskuje wyniki mojej klasyfikacji, czyli wynik pozytywny na cukrzyce lub negatywny.

```
> # prognoza na zbiorze u
> nb.prog.ucz <- predict(cukrzyca.nb, uczacy.bayes)
> nb.prog.ucz
  [1] neg neg pos neg neg pos pos pos neg pos neg neg neg neg neg neg pos pos neg pos neg neg neg pos neg neg pos
 [28] pos pos neg neg pos pos neg neg pos neg neg neg pos pos pos pos pos neg neg neg pos neg pos pos neg neg neg
 [55] pos pos neg pos neg neg pos neg pos pos pos neg pos pos pos neg pos neg neg pos pos pos pos neg pos pos neg
 [82] neg neg pos pos pos pos neg pos neg pos neg neg neg pos neg pos pos pos pos pos pos neg neg pos pos pos pos
[109] pos pos pos pos neg neg pos pos pos pos pos neg pos pos pos neg pos pos neg neg pos pos neg pos pos neg
[136] pos pos neg neg pos pos neg neg pos pos neg neg pos neg pos neg neg pos pos neg pos pos neg pos pos neg neg
[163] neg pos neg neg neg pos pos pos pos neg pos neg neg neg pos pos pos pos pos neg pos neg pos neg pos neg neg
[190] neg pos pos neg pos pos neg neg pos neg neg neg pos pos pos neg neg pos pos pos neg neg pos pos pos neg pos
[217] neg neg neg pos neg pos neg pos neg neg pos pos neg neg pos neg neg pos pos neg neg neg pos neg pos neg neg
[244] pos pos neg pos pos pos neg pos pos neg pos pos pos neg neg neg pos neg pos pos pos pos pos pos pos pos neg
[271] neg pos neg pos neg neg pos pos pos pos pos pos neg neg pos pos pos pos pos pos pos pos pos pos pos pos neg
[298] pos neg pos
```

Macierz błędów zbiór uczący

```
nb.prog.ucz neg pos
        neg 118  12
        pos  87  83
>
```

Dokładność zbiór uczący

```
> #dokladnosc
> sum(diag(tmb2))/sum(tmb2)
[1] 0.67
>
```

Dokładność nie jest doskonała, ale jest dobra i wynosi ona 67%.

Klasyfikator naiwny Bayesa zbiór testowy

```
> cukrzyca.nb2 <- naiveBayes(testowy.bayes$Klasyfikacja~., data = testowy.bayes)
> cukrzyca.nb2

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
      neg       pos
0.6195652 0.3804348

Conditional probabilities:
     Glukoza
Y         [,1]      [,2]
  neg 1.192982 0.3981473
  pos 1.542857 0.5054327

     Cisnienie
Y         [,1]      [,2]
  neg 1.105263 0.4505636
  pos 1.085714 0.3734914

     Insulina
Y         [,1]      [,2]
  neg 2.771930 0.5674990
  pos 2.885714 0.3228029

     BMI
Y         [,1]      [,2]
  neg 2.192982 0.9899242
  pos 2.771429 0.6456057

     Wiek
Y         [,1]      [,2]
  neg 2.596491 0.7759706
  pos 2.885714 0.4710082
```

W zbiorze testowym wyszło mi, że 38% pacjentów ma pozytywny wynik na cukrzyce co jest wynikiem większym o 6 punktów procentowych niż na zbiorze uczącym.

Prognoza na zbiorze testowym

```
> nb.prog.test <- predict(cukrzyca.nb2, testowy.bayes)
> nb.prog.test
 [1] neg neg pos pos pos pos pos neg neg neg neg neg pos neg pos neg neg pos pos pos pos neg pos neg neg pos neg
[28] pos pos neg neg pos pos pos neg pos pos pos neg pos pos neg pos neg pos neg pos pos pos neg pos pos pos neg
[55] pos pos pos neg pos neg pos pos pos pos pos neg neg pos pos pos neg neg neg neg neg neg pos pos pos neg neg
[82] pos neg pos pos pos neg pos pos neg pos neg
Levels: neg pos
```

Macierz błędów zbiór testowy

```
> #macierzbledow
> tmb2 <- table(nb.prog.test, testowy.bayes$Klasyfikacja)
> tmb2

nb.prog.test neg pos
         neg  32   7
         pos  25  28
```

Dokładność zbiór testowy

```
> #dokladnosc
> sum(diag(tmb2))/sum(tmb2)
[1] 0.6521739
> 
```

Dokładność w zbiorze testowym wyszła mniejsza niż w zbiorze uczącym o 2pkt procentowe.

Krzywa ROC i wartość AUC

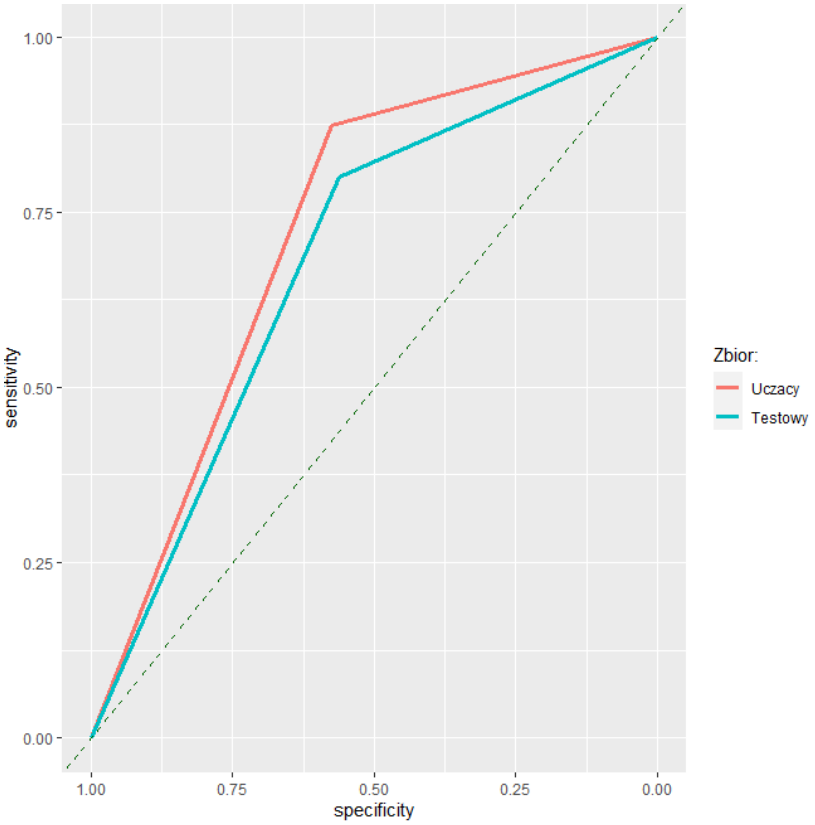
Krzywa ROC – tworzona jest na podstawie wartości czułości i specyficzności.

Wartość AUC – pole pod krzywą ROC (miara dobroci danego modelu)

Korzystając z funkcji roc, osobno dla zbioru uczącego i testowego liczymy wszystko.

```
library(pROC)

# AUC dla zbioru u

ROC.uczacy <- roc(uczacy.bayes$Klasyfikacja, as.numeric(nb.prog.ucz))
auc(ROC.uczacy)

# AUC dla zbioru t

ROC.testowy <- roc(testowy.bayes$Klasyfikacja ,as.numeric(nb.prog.test))
auc(ROC.testowy)

# Krzywa ROC (Bayes)

ggroc(list(Uczacy = ROC.uczacy, Testowy = ROC.testowy), size=1.3)+
  labs(colour="Zbior:")+
  geom_abline(intercept = 1, color="darkgreen", linetype="dashed")
```

Nakładając oba wykresy na siebie uzyskuje takie wartości.



Im wykres idzie bardziej na lewo, tym lepiej. Jeśli będzie się za bardzo zbliżał do przekątnej narysowanej przerywaną linią, będzie to oznaczało że klasyfikator jest losowy, natomiast jeśli będzie poniżej tej przekątnej to będzie oznaczać, że klasyfikator jest gorszy niż losowy.

```
> auc(ROC.testowy)
Area under the curve: 0.6807
> auc(ROC.uczacy)
Area under the curve: 0.7246
```

Stosując polecenie auc uzyskałem pola pod wykresami dla zbioru uczącego i testowego. Im wyższe wartości tym lepiej. W moim przypadku wartości te są bardzo dobre, co oznacza że klasyfikator jest dobry.

PODSUMOWANIE

Podsumowując wyniki działania modeli są satysfakcjonujące, przy metodzie k najbliższych sąsiadów dokładności poziomu 84% i 89% są zdecydowanie zadowalające. Nieco gorzej, ale również zadowalająco wypadła metoda Bayesa, pola pod wykresami dla zbiorów uczącego i testowego wyniosły odpowiednio 72% i 68% co jest wynikiem satysfakcjonującym, jednakże nie tak dobrym jak w pierwszej metodzie. Powodem tak zaistniałej sytuacji jest to, że dane utraciły część swoich informacji przy zamianie ich z ilościowych na kategoryczne.