

Drzewo decyzyjne dla wyboru obiadu

Jakub Augustynek, Jakub Ignatik, Artur Karamon, Jarosław Kmak

15 grudnia 2018

Wstęp

Istnieją decyzje, które podejmować należy codziennie. Do takich decyzji należy również wybór miejsca, gdzie zjemy dzisiejszy obiad. Ograniczeni jesteśmy takimi czynnikami jak czas, pieniądze, czy też dostępność składników w lodówce. Nasze drzewko decyzyjne ma za zadanie pomóc przy rozwianiu takich właśnie wątpliwości, biorąc pod uwagę to, jak bardzo bogaci jesteśmy w zasoby pozwalające na spożycie upragnionego posiłku. Zgodnie z naszym programem, obiad możemy spożyć w trzech głównych formach: zrobiony własnoręcznie, zjedzony w lokalu oraz zamówiony na dowóz. Pod uwagę bierzemy następujące czynniki:

Czas przygotowania: Jakim czasem dysponujemy? Zmienna podzielona jest na 3 kategorie (do 15 minut, od 15 do 30 minut, powyżej 30 minut)

Pieniądze: Jaki mamy budżet na dzisiejszy obiad? Tutaj również podział na 3 kategorie (do 10 złotych, od 10 do 20 złotych, powyżej 20 złotych)

Dostępność składników: Jak bardzo bogata w składniki jest nasza lodówka? Występuje tu podział na 4 kategorie (brak składników, mała dostępność, średnia dostępność, duża dostępność)

Poziom głodu: Jak bardzo jesteśmy głodni? Możliwości są dwie: trochę albo bardzo głodni.

Odległość od lokalu: Ile minut drogi mamy do najbliższego punktu z jedzeniem? Podział czasowy na 3 kategorie (do 2 minut, od 2 do 10 minut, więcej niż 10 minut)

Metodologia: W naszym projekcie wykorzystujemy podział na gałęzie wg entropii. Udało nam się zautomatyzować proces dzielenia się drzewa, dzięki czemu po zastosowaniu funkcji w odpowiedzi uzyskamy drzewo podzielone poziomowo. Po odpowiedzi na pytanie z danego poziomu należy przejść na poziom następny, aż do uzyskania jednej z trzech możliwych decyzji. Do pracy dołączamy również graficznie przedstawione drzewo decyzyjne. W załącznikach znajdzie się też plik z danymi, do których własnoręcznie przypisywaliśmy decyzje.

Drzewo decyzyjne

Poniżej przedstawiony jest kod tworzący drzewo decyzyjne, wraz z wyżej wymienionym poziomowaniem w odpowiedzi oraz z komentarzem wyjaśniającym działaniem kodu.

```
#wczytanie danych
data <- read.csv("C:/Users/Jan/Desktop/ESI_bin.csv", sep = ";", header = FALSE)

#przekształcenia ramki z danymi (tak aby nazwy wierszow oznaczały
#poszczególne przesłanki, natomiast kolumn kolejne kombinacje)
kategorie <- paste0(data[,1],": ", data[,2])
data <- data[,-c(1,2)]
colnames(data) <- c(1:ncol(data))
rownames(data) <- kategorie
#ustawienie zmiennej oznaczającej liczbę możliwości konkluzji
#(w przypadku tego drzewa jest to 3)
ldec <- 3
ldec2 <- ldec - 1

#funkcja rysująca drzewo
drzewo_decyzyjne<-function(data){
```

```

#podfunkcja wykonujaca dzielenie tabeli na 2, wg entropii
dziel<-function(data){

  #obliczenie entropii dla konkluzji
  ISUM <- 0
  for (i in (nrow(data)-2):(nrow(data))) {
    I = -(sum(data[i,]==1)/ncol(data))*log2(sum(data[i,]==1)/ncol(data))
    if((sum(data[i,]==1)/ncol(data)) == 0){
      I <- 0
    }
    ISUM <- ISUM + I
  }

  #obliczenie entropii czastokowych dla wszystkich przeslanek i
  #zapisanie ich w wektorze entropii (wektor_e)
  Ic <- 0
  wektor_e <- c()
  for(i in 1:(nrow(data)-ldec)){
    Ic_p_sum <- 0
    Ic_n_sum <- 0
    for (j in (nrow(data)-ldec2):nrow(data)){
      Ic_lp <- sum((data[i,] == data[j,]) & (data[i,] != 0))
      Ic_mp <- sum(data[i,] == 1)
      Ic_p <- -(Ic_lp/Ic_mp)*log2(Ic_lp/Ic_mp)
      Ic_ln <- sum((data[j,]==1) & (data[i,] == 0))
      Ic_mn <- sum(data[i,] == 0)
      Ic_n <- -(Ic_ln/Ic_mn)*log2(Ic_ln/Ic_mn)
      if(Ic_lp == 0){
        Ic_p <- 0
      }
      if(Ic_ln == 0){
        Ic_n <- 0
      }
      Ic_p_sum <- Ic_p_sum + Ic_p
      Ic_n_sum <- Ic_n_sum + Ic_n
    }
    E <- ((Ic_mp/ncol(data))*Ic_p_sum) + ((Ic_mn/ncol(data))*Ic_n_sum)
    wektor_e[i] <- ISUM-E
  }

  #ustalenie wg. ktorej przeslanki nastepuje dzielenie (zapisanie jej nazwy)
  #i podzial tabeli na dwie (pierwsza, ktora przyjmuje wartosci PRAWDA(1)
  #oraz druga, ktora przyjmuje wartosci FALSZ(0))
  podzial <- min(which(wektor_e == max(wektor_e)))
  data_P <- as.data.frame(data[,which(data[podzial,] == 1)])
  data_F <- as.data.frame(data[,which(data[podzial,] == 0)])
  przeslanka_dziel <- rownames(data)[podzial]

  #stworzenie warunkow pozwalajacych na przypisanie w miejsce jednej z
  #podzielonych tabel wartosci konkluzji, jesli wszystkie wartosci
  #wierszy danej konkluzji wskazuja jej pojawienie sie (maja wart. 1).
  for (l in 0:ldec2){

```

```

warunek_p <- (mean(as.numeric(data_P[nrow(data_P)-1,])) == 1)
if (warunek_p==TRUE) {
  data_P<-rownames(data[nrow(data_P)-1,])
  break
}
}

for (l in 0:ldec2){
  warunek_f <- (mean(as.numeric(data_F[nrow(data_F)-1,])) == 1)
  if (warunek_f==TRUE) {
    data_F<-rownames(data[nrow(data_F)-1,])
    break
  }
}

#zwrocenie listy zawierajacej 2 wyniki podzialu tabeli oraz
#przeslankę wg ktorej nastapil podzial
return(list(data_P,data_F,przeslanka_dziel))
}

#przeprowadzenie kolejnych podzialow za pomoca podfunkcji dziel(),
#budujac drzewo decyzyjne, oraz ustawienie kilku zmiennych na wartosci startowe
war_wyj<-FALSE
i_while<-0
#przeprowadzenie pierwszego podzialu startowego
lista_poziom<-dziel(data)[c(1,2)]
wektor_nazw<-c(unlist(dziel(data)[3]))
wektor_i_for<-c(1)
#przeprowadzanie podzialow utworzonych z poprzednich podzialow tabel
#i wypisywanie wynikow, az do momentu, gdy wszystkie wyniki
#na danym poziomie beda konkluzjami
while(war_wyj==FALSE){
  i_while<-i_while+1
  print(paste0("==== PODZIAŁ ",(i_while)," ====="))
  lista_poziom2<-list()
  wektor_i_for2<-c()
  wektor_nazw2<-c()
  i_if<-0

  #petla odtwarzajaca wyniki(tabele/konkluzje) uzyskane z poprzedniego podzialu
  for (i_for in 1:length(lista_poziom)) {
    #print(i_for)
    if (i_for%%2==1) {
      odp <- "Tak"
      i_if<-i_if+1
      cat("\n" )
    }
    else{ odp <- "Nie"}

    if (class(lista_poziom[[i_for]])!="character") {
      lista_poziom_iter<-(dziel(lista_poziom[[i_for]]))[c(1,2)]
      wektor_nazw2<-c(wektor_nazw2,unlist((dziel(lista_poziom[[i_for]]))[3]))
      wektor_i_for2<-c(wektor_i_for2,i_for)
    }
  }
}

```

```

    print(paste0(wektor_i_for[i_if], " --> ",odp," --> ",i_for))
    print(paste0((wektor_nazw[i_if]), " = ",odp," -----> ",unlist((dziel(lista_poziom[[i_for]]))
  }
  else{
    lista_poziom_iter<-list()
    print(paste0(wektor_i_for[i_if], " --> ",odp," --> ",i_for))
    print(paste0((wektor_nazw[i_if]), " = ",odp," -----> ",unlist(lista_poziom[[i_for]])))
  }

  lista_poziom2<-c(lista_poziom2,lista_poziom_iter)

}
wektor_nazw<-wektor_nazw2
wektor_i_for<-wektor_i_for2
lista_poziom<-lista_poziom2
cat("\n" )
cat("\n" )
cat("\n" )

#ustawienie warunku wyjścia
if(length(lista_poziom)==0){
  war_wyj<-TRUE
}
}
}

#wykorzystanie funkcji na danych
drzewo_decyzyjne(data)

## [1] "===== PODZIAŁ 1 ====="
##
## [1] "1 --> Tak --> 1"
## [1] "Odleglosc od lokalu: >10 = Tak -----> Pieniadze: <10"
## [1] "1 --> Nie --> 2"
## [1] "Odleglosc od lokalu: >10 = Nie -----> Dostepnosc skladnikow: brak"
##
##
##
## [1] "===== PODZIAŁ 2 ====="
##
## [1] "1 --> Tak --> 1"
## [1] "Pieniadze: <10 = Tak -----> Dostepnosc skladnikow: brak"
## [1] "1 --> Nie --> 2"
## [1] "Pieniadze: <10 = Nie -----> Dostepnosc skladnikow: duza"
##
## [1] "2 --> Tak --> 3"
## [1] "Dostepnosc skladnikow: brak = Tak -----> Decyzja: Jem na miescie"
## [1] "2 --> Nie --> 4"
## [1] "Dostepnosc skladnikow: brak = Nie -----> Pieniadze: <10"
##
##
##
## [1] "===== PODZIAŁ 3 ====="
##

```

```

## [1] "1 --> Tak --> 1"
## [1] "Dostepnosc skladnikow: brak = Tak -----> Decyzja: Jem na miescie"
## [1] "1 --> Nie --> 2"
## [1] "Dostepnosc skladnikow: brak = Nie -----> Dostepnosc skladnikow: mala"
##
## [1] "2 --> Tak --> 3"
## [1] "Dostepnosc skladnikow: duza = Tak -----> Pieniadze: <10,20>"
## [1] "2 --> Nie --> 4"
## [1] "Dostepnosc skladnikow: duza = Nie -----> Czas na przygotowanie: <30"
##
## [1] "4 --> Tak --> 5"
## [1] "Pieniadze: <10 = Tak -----> Poziom glodu: troche glodny"
## [1] "4 --> Nie --> 6"
## [1] "Pieniadze: <10 = Nie -----> Dostepnosc skladnikow: mala"
##
##
##
## [1] "===== PODZIAŁ 4 ====="
##
## [1] "2 --> Tak --> 1"
## [1] "Dostepnosc skladnikow: mala = Tak -----> Poziom glodu: troche glodny"
## [1] "2 --> Nie --> 2"
## [1] "Dostepnosc skladnikow: mala = Nie -----> Decyzja: Robie sam"
##
## [1] "3 --> Tak --> 3"
## [1] "Pieniadze: <10,20> = Tak -----> Decyzja: Robie sam"
## [1] "3 --> Nie --> 4"
## [1] "Pieniadze: <10,20> = Nie -----> Czas na przygotowanie: <15"
##
## [1] "4 --> Tak --> 5"
## [1] "Czas na przygotowanie: <30 = Tak -----> Decyzja: Zamawiam"
## [1] "4 --> Nie --> 6"
## [1] "Czas na przygotowanie: <30 = Nie -----> Dostepnosc skladnikow: brak"
##
## [1] "5 --> Tak --> 7"
## [1] "Poziom glodu: troche glodny = Tak -----> Czas na przygotowanie: <15"
## [1] "5 --> Nie --> 8"
## [1] "Poziom glodu: troche glodny = Nie -----> Dostepnosc skladnikow: duza"
##
## [1] "6 --> Tak --> 9"
## [1] "Dostepnosc skladnikow: mala = Tak -----> Decyzja: Jem na miescie"
## [1] "6 --> Nie --> 10"
## [1] "Dostepnosc skladnikow: mala = Nie -----> Odleglosc od lokalu: <2"
##
##
##
## [1] "===== PODZIAŁ 5 ====="
##
## [1] "1 --> Tak --> 1"
## [1] "Poziom glodu: troche glodny = Tak -----> Decyzja: Robie sam"
## [1] "1 --> Nie --> 2"
## [1] "Poziom glodu: troche glodny = Nie -----> Czas na przygotowanie: <15"
##
## [1] "4 --> Tak --> 3"

```

```

## [1] "Czas na przygotowanie: <15 = Tak -----> Decyzja: Robie sam"
## [1] "4 --> Nie --> 4"
## [1] "Czas na przygotowanie: <15 = Nie -----> Czas na przygotowanie: <15,30>"
##
## [1] "6 --> Tak --> 5"
## [1] "Dostepnosc skladnikow: brak = Tak -----> Decyzja: Zamawiam"
## [1] "6 --> Nie --> 6"
## [1] "Dostepnosc skladnikow: brak = Nie -----> Pieniadze: <10,20>"
##
## [1] "7 --> Tak --> 7"
## [1] "Czas na przygotowanie: <15 = Tak -----> Odleglosc od lokalu: <2"
## [1] "7 --> Nie --> 8"
## [1] "Czas na przygotowanie: <15 = Nie -----> Decyzja: Robie sam"
##
## [1] "8 --> Tak --> 9"
## [1] "Dostepnosc skladnikow: duza = Tak -----> Odleglosc od lokalu: <2"
## [1] "8 --> Nie --> 10"
## [1] "Dostepnosc skladnikow: duza = Nie -----> Czas na przygotowanie: <15,30>"
##
## [1] "10 --> Tak --> 11"
## [1] "Odleglosc od lokalu: <2 = Tak -----> Pieniadze: <10,20>"
## [1] "10 --> Nie --> 12"
## [1] "Odleglosc od lokalu: <2 = Nie -----> Czas na przygotowanie: <15"
##
##
##
## [1] "===== PODZIAŁ 6 ====="
##
## [1] "2 --> Tak --> 1"
## [1] "Czas na przygotowanie: <15 = Tak -----> Decyzja: Robie sam"
## [1] "2 --> Nie --> 2"
## [1] "Czas na przygotowanie: <15 = Nie -----> Decyzja: Jem na miescie"
##
## [1] "4 --> Tak --> 3"
## [1] "Czas na przygotowanie: <15,30> = Tak -----> Decyzja: Zamawiam"
## [1] "4 --> Nie --> 4"
## [1] "Czas na przygotowanie: <15,30> = Nie -----> Poziom glodu: troche glodny"
##
## [1] "6 --> Tak --> 5"
## [1] "Pieniadze: <10,20> = Tak -----> Dostepnosc skladnikow: mala"
## [1] "6 --> Nie --> 6"
## [1] "Pieniadze: <10,20> = Nie -----> Czas na przygotowanie: <15"
##
## [1] "7 --> Tak --> 7"
## [1] "Odleglosc od lokalu: <2 = Tak -----> Decyzja: Jem na miescie"
## [1] "7 --> Nie --> 8"
## [1] "Odleglosc od lokalu: <2 = Nie -----> Dostepnosc skladnikow: mala"
##
## [1] "9 --> Tak --> 9"
## [1] "Odleglosc od lokalu: <2 = Tak -----> Czas na przygotowanie: <30"
## [1] "9 --> Nie --> 10"
## [1] "Odleglosc od lokalu: <2 = Nie -----> Decyzja: Robie sam"
##
## [1] "10 --> Tak --> 11"

```

```

## [1] "Czas na przygotowanie: <15,30> = Tak -----> Dostepnosc skladnikow: mala"
## [1] "10 --> Nie --> 12"
## [1] "Czas na przygotowanie: <15,30> = Nie -----> Decyzja: Jem na miescie"
##
## [1] "11 --> Tak --> 13"
## [1] "Pieniadze: <10,20> = Tak -----> Dostepnosc skladnikow: srednia"
## [1] "11 --> Nie --> 14"
## [1] "Pieniadze: <10,20> = Nie -----> Decyzja: Jem na miescie"
##
## [1] "12 --> Tak --> 15"
## [1] "Czas na przygotowanie: <15 = Tak -----> Pieniadze: <10,20>"
## [1] "12 --> Nie --> 16"
## [1] "Czas na przygotowanie: <15 = Nie -----> Pieniadze: <10,20>"
##
##
##
## [1] "===== PODZIAŁ 7 ====="
##
## [1] "4 --> Tak --> 1"
## [1] "Poziom glodu: troche glodny = Tak -----> Decyzja: Robie sam"
## [1] "4 --> Nie --> 2"
## [1] "Poziom glodu: troche glodny = Nie -----> Decyzja: Zamawiam"
##
## [1] "5 --> Tak --> 3"
## [1] "Dostepnosc skladnikow: mala = Tak -----> Decyzja: Zamawiam"
## [1] "5 --> Nie --> 4"
## [1] "Dostepnosc skladnikow: mala = Nie -----> Czas na przygotowanie: <15"
##
## [1] "6 --> Tak --> 5"
## [1] "Czas na przygotowanie: <15 = Tak -----> Decyzja: Zamawiam"
## [1] "6 --> Nie --> 6"
## [1] "Czas na przygotowanie: <15 = Nie -----> Dostepnosc skladnikow: mala"
##
## [1] "8 --> Tak --> 7"
## [1] "Dostepnosc skladnikow: mala = Tak -----> Decyzja: Jem na miescie"
## [1] "8 --> Nie --> 8"
## [1] "Dostepnosc skladnikow: mala = Nie -----> Decyzja: Robie sam"
##
## [1] "9 --> Tak --> 9"
## [1] "Czas na przygotowanie: <30 = Tak -----> Decyzja: Robie sam"
## [1] "9 --> Nie --> 10"
## [1] "Czas na przygotowanie: <30 = Nie -----> Decyzja: Jem na miescie"
##
## [1] "11 --> Tak --> 11"
## [1] "Dostepnosc skladnikow: mala = Tak -----> Decyzja: Jem na miescie"
## [1] "11 --> Nie --> 12"
## [1] "Dostepnosc skladnikow: mala = Nie -----> Odleglosc od lokalu: <2"
##
## [1] "13 --> Tak --> 13"
## [1] "Dostepnosc skladnikow: srednia = Tak -----> Decyzja: Jem na miescie"
## [1] "13 --> Nie --> 14"
## [1] "Dostepnosc skladnikow: srednia = Nie -----> Poziom glodu: troche glodny"
##
## [1] "15 --> Tak --> 15"

```

```

## [1] "Pieniadze: <10,20> = Tak -----> Dostepnosc skladnikow: srednia"
## [1] "15 --> Nie --> 16"
## [1] "Pieniadze: <10,20> = Nie -----> Decyzja: Jem na miescie"
##
## [1] "16 --> Tak --> 17"
## [1] "Pieniadze: <10,20> = Tak -----> Poziom glodu: troche glodny"
## [1] "16 --> Nie --> 18"
## [1] "Pieniadze: <10,20> = Nie -----> Dostepnosc skladnikow: srednia"
##
##
##
## [1] "===== PODZIAŁ 8 ====="
##
## [1] "4 --> Tak --> 1"
## [1] "Czas na przygotowanie: <15 = Tak -----> Decyzja: Robie sam"
## [1] "4 --> Nie --> 2"
## [1] "Czas na przygotowanie: <15 = Nie -----> Poziom glodu: troche glodny"
##
## [1] "6 --> Tak --> 3"
## [1] "Dostepnosc skladnikow: mala = Tak -----> Poziom glodu: troche glodny"
## [1] "6 --> Nie --> 4"
## [1] "Dostepnosc skladnikow: mala = Nie -----> Decyzja: Zamawiam"
##
## [1] "12 --> Tak --> 5"
## [1] "Odleglosc od lokalu: <2 = Tak -----> Decyzja: Jem na miescie"
## [1] "12 --> Nie --> 6"
## [1] "Odleglosc od lokalu: <2 = Nie -----> Decyzja: Robie sam"
##
## [1] "14 --> Tak --> 7"
## [1] "Poziom glodu: troche glodny = Tak -----> Czas na przygotowanie: <15"
## [1] "14 --> Nie --> 8"
## [1] "Poziom glodu: troche glodny = Nie -----> Decyzja: Jem na miescie"
##
## [1] "15 --> Tak --> 9"
## [1] "Dostepnosc skladnikow: srednia = Tak -----> Decyzja: Jem na miescie"
## [1] "15 --> Nie --> 10"
## [1] "Dostepnosc skladnikow: srednia = Nie -----> Poziom glodu: troche glodny"
##
## [1] "17 --> Tak --> 11"
## [1] "Poziom glodu: troche glodny = Tak -----> Decyzja: Robie sam"
## [1] "17 --> Nie --> 12"
## [1] "Poziom glodu: troche glodny = Nie -----> Czas na przygotowanie: <15,30>"
##
## [1] "18 --> Tak --> 13"
## [1] "Dostepnosc skladnikow: srednia = Tak -----> Czas na przygotowanie: <15,30>"
## [1] "18 --> Nie --> 14"
## [1] "Dostepnosc skladnikow: srednia = Nie -----> Poziom glodu: troche glodny"
##
##
##
## [1] "===== PODZIAŁ 9 ====="
##
## [1] "2 --> Tak --> 1"
## [1] "Poziom glodu: troche glodny = Tak -----> Decyzja: Robie sam"

```



```

## [1] "2 --> Nie --> 2"
## [1] "Poziom glodu: troche glodny = Nie -----> Decyzja: Zamawiam"
##
## [1] "3 --> Tak --> 3"
## [1] "Poziom glodu: troche glodny = Tak -----> Decyzja: Robie sam"
## [1] "3 --> Nie --> 4"
## [1] "Poziom glodu: troche glodny = Nie -----> Decyzja: Zamawiam"
##
## [1] "7 --> Tak --> 5"
## [1] "Czas na przygotowanie: <15 = Tak -----> Decyzja: Jem na miescie"
## [1] "7 --> Nie --> 6"
## [1] "Czas na przygotowanie: <15 = Nie -----> Decyzja: Robie sam"
##
## [1] "10 --> Tak --> 7"
## [1] "Poziom glodu: troche glodny = Tak -----> Decyzja: Robie sam"
## [1] "10 --> Nie --> 8"
## [1] "Poziom glodu: troche glodny = Nie -----> Decyzja: Jem na miescie"
##
## [1] "12 --> Tak --> 9"
## [1] "Czas na przygotowanie: <15,30> = Tak -----> Dostepnosc skladnikow: srednia"
## [1] "12 --> Nie --> 10"
## [1] "Czas na przygotowanie: <15,30> = Nie -----> Dostepnosc skladnikow: srednia"
##
## [1] "13 --> Tak --> 11"
## [1] "Czas na przygotowanie: <15,30> = Tak -----> Decyzja: Jem na miescie"
## [1] "13 --> Nie --> 12"
## [1] "Czas na przygotowanie: <15,30> = Nie -----> Poziom glodu: troche glodny"
##
## [1] "14 --> Tak --> 13"
## [1] "Poziom glodu: troche glodny = Tak -----> Czas na przygotowanie: <15,30>"
## [1] "14 --> Nie --> 14"
## [1] "Poziom glodu: troche glodny = Nie -----> Decyzja: Zamawiam"
##
##
##
## [1] "===== PODZIAŁ 10 ====="
##
## [1] "9 --> Tak --> 1"
## [1] "Dostepnosc skladnikow: srednia = Tak -----> Decyzja: Zamawiam"
## [1] "9 --> Nie --> 2"
## [1] "Dostepnosc skladnikow: srednia = Nie -----> Decyzja: Jem na miescie"
##
## [1] "10 --> Tak --> 3"
## [1] "Dostepnosc skladnikow: srednia = Tak -----> Decyzja: Jem na miescie"
## [1] "10 --> Nie --> 4"
## [1] "Dostepnosc skladnikow: srednia = Nie -----> Decyzja: Robie sam"
##
## [1] "12 --> Tak --> 5"
## [1] "Poziom glodu: troche glodny = Tak -----> Decyzja: Zamawiam"
## [1] "12 --> Nie --> 6"
## [1] "Poziom glodu: troche glodny = Nie -----> Decyzja: Jem na miescie"
##
## [1] "13 --> Tak --> 7"
## [1] "Czas na przygotowanie: <15,30> = Tak -----> Decyzja: Robie sam"

```

```
## [1] "13 --> Nie --> 8"  
## [1] "Czas na przygotowanie: <15,30> = Nie -----> Decyzja: Jem na miescie"
```

Graficzne przedstawienie drzewa

```
knitr::include_graphics("drzewko.png")
```



Podsumowanie

Można zauważyć, że drzewo jest dość skomplikowane, sięgając 10 poziomów. Pozwala to jednak na określenie z pełną skutecznością, gdzie udać się dzisiejszego dnia na obiad. Zaskoczeniem może być fakt, że drzewo zaczyna się od określenia odległości od lokalu, a nie dostępności czasu czy też pieniędzy. Tak jednak uzyskana struktura jest jak najbardziej prawidłowa, gdyż duża odległość zniechęca do udania się do lokalu oraz zachęca do zamówienia czegoś na dowóz bądź też przygotowania posiłku własnymi siłami. Jak już wspomniane było we wstępie, całe drzewo do przeanalizowania graficznie, w czytelniejszej formie, dostępne jest w załączniku.