

# K-NN

Mikołaj Malec

April 20, 2020

## Czym jest algorytm K Nearest Neighbor?

Jest to algorytm który klasyfikuje dany rekord(w naszym przypadku dane o kliencie banku) za pomocą obliczenia k najbliższych sąsiadów. Z tych k sąsiadów przyporządkowuje go do najliczniejszego typu wśród tej k-licznej grupy.

## Wczytanie danych i bibliotek

```
library( class)
```

```
## Warning: package 'class' was built under R version 3.6.3
```

```
x_test <- read.csv( "x_test.csv")[-1]
x_train <- read.csv( "x_train.csv")[-1]
y_test <- read.csv( "y_test.csv")[-1]
y_train <- read.csv( "y_train.csv")[-1]
```

## Normalizacja

Do wyznaczenia sąsiadów będziemy używać zwykłej metryki euklidesowej. Jednakże dane trzeba znormalizować ponieważ w innym wypadku odległości pomiędzy danymi o wypłacie będą o wiele bardziej wpływowe niż np liczba dzieci. Taka różnica w wielkości zmiennych nieunormowanych spowodowała nieistotność zmiennej liczby dzieci co źle wpłynęło by na nasz model

```
#normaliacion to ~(0,1)
for( coli in 1:dim(x_train)[2]){
  #x and y have o be noemalized in the same way
  c_min <- min( c(x_train[,coli], x_test[,coli]))
  c_max <- max( c(x_train[,coli], x_test[,coli]))

  x_train[,coli] <- (x_train[,coli] - c_min) / (c_max - c_min)
  x_test[,coli] <- (x_test[,coli] - c_min) / (c_max - c_min)
}
```

## Metryka dokładności

Do pomocy w określeniu jakości naszego modelu napiszemy funkcję która wyliczy

```
accuracy <- function( table_in){  
  sum( diag( table_in)) / sum( table_in)  
}
```

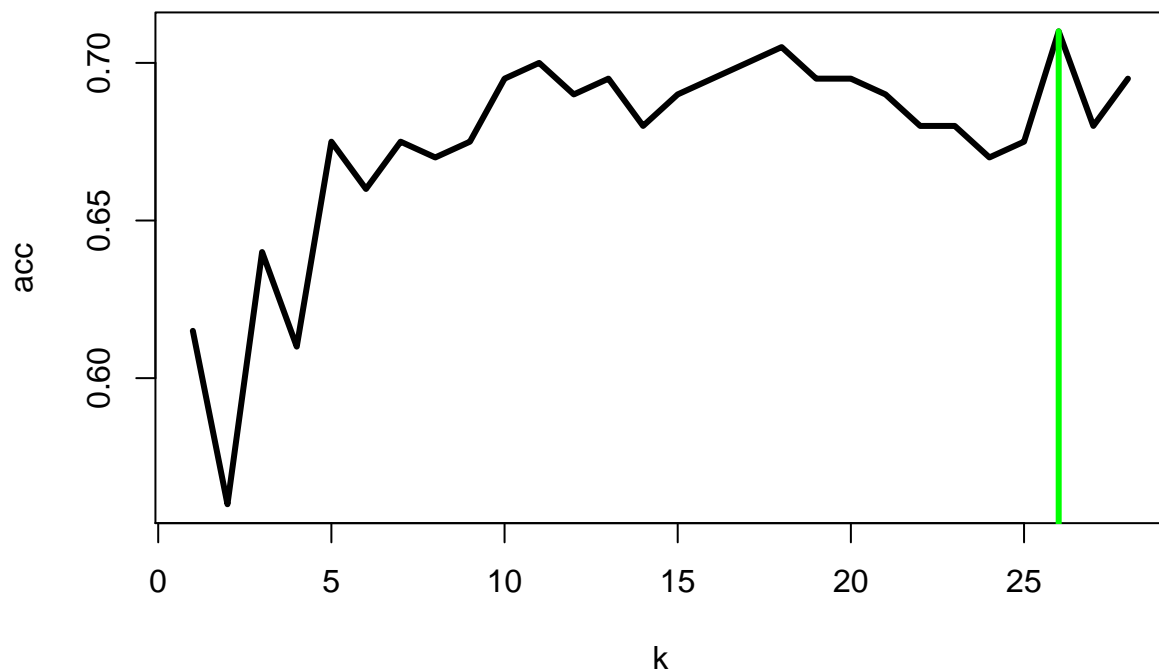
## Testowanie i Trenowanie naszego modelu dla różnej wartości parametru k

Spróbujmy teraz sprawdzić jak celny jest nasz model w zależności od parametru **k**. Do obliczenia jakości naszego modelu sprawdzimy jego działanie w z parametrem w przedziale od 1 do pierwiastka z wielkości danych trenningowych.

```
n <- sqrt( dim(x_train)[1])  
acc <- rep(0,n)  
for(k in 1:n){  
  test_pred <- knn( train = x_train[1:800,], test = x_test, cl = y_train[1:800,], k=k)  
  
  tab <- table( y_test[,1], test_pred)  
  acc[k] <- accuracy(tab)  
}  
k=1:n  
best_k <- which.max( acc)
```

Po przeiterowaniu przez wszystkie parametry zmiennej k można pokazać dane na wykresie.

```
plot( k, acc, type = "l", lwd = 3)  
lines(c(best_k,best_k),c(0,acc[best_k]),col = "green",lwd = 3)
```



Jak widać na wykresie najwyższą wartość celności, zaznaczoną na zielono, nasz model uzyskuje dla parametru  $k = 26$ . Sprawdźmy dokładniej jak nasz model przewiduje dla najlepszego parametru  $k$ .

```
test_pred <- knn( train = x_train[1:800,], test = x_test, cl = y_train[1:800,], k=best_k)
tab <- table( y_test[,1], test_pred)
knitr::kable(tab)
```

	0	1
0	11	53
1	7	129

Tak wygląda nasza tabela. Finalnie model uzyskuje 70% celności.