Lab6: Klasyfikacja

Michał Ciach, Anna Macioszek

March 29, 2020

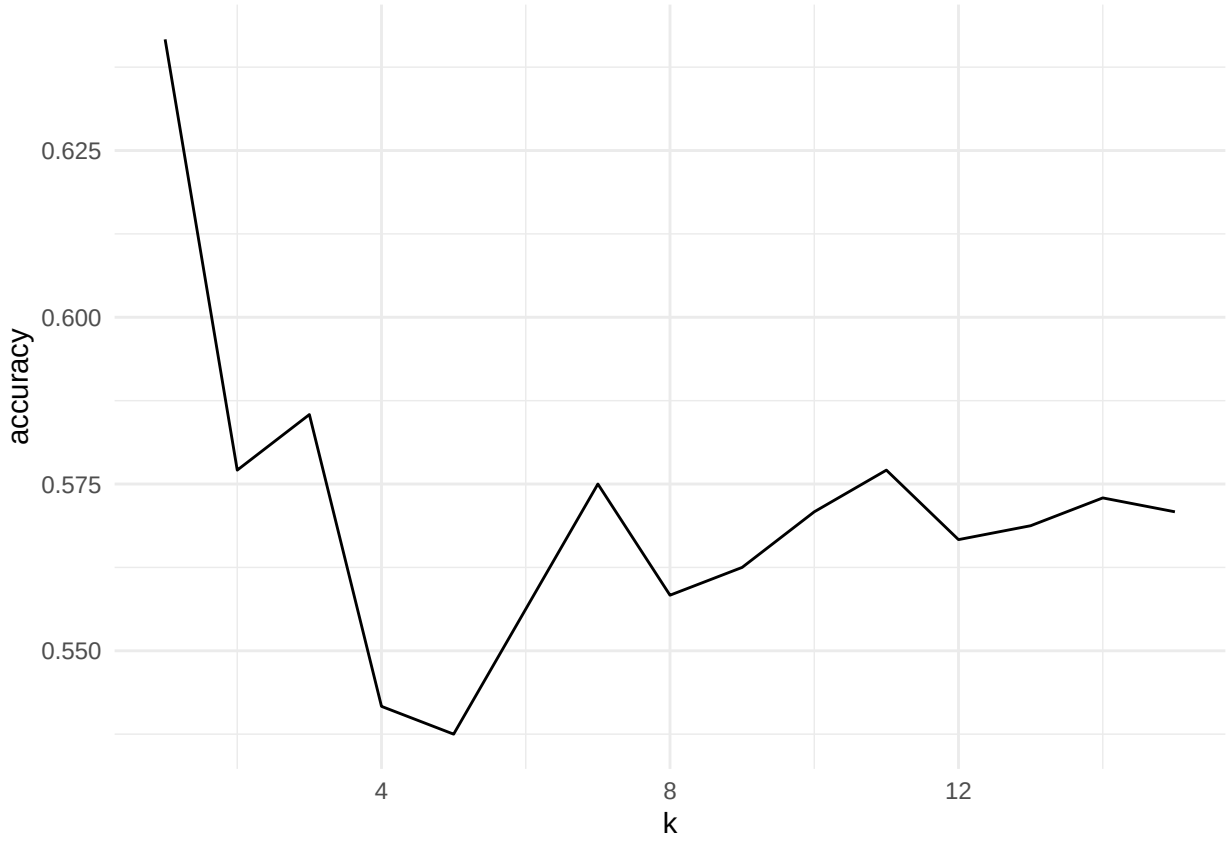
Zadanie 1. Przeskaluj wszystkie kolumny predyktorów z danych `wine`. Możesz w tym celu wykorzystać albo `apply()`, albo funkcję `scale()`. Następnie zrzutuj zmienną `Quality` na typ `factor` za pomocą funkcji `as.factor()` w następujący sposób: `wine$Quality <- as.factor(wine$Quality)`.

```
wine<-read.csv("wine.csv",sep="\t")
wine[,2:ncol(wine)] <- apply(wine[,-1],MARGIN = 2,function (x) (x-mean(x))/sd(x))
wine$Quality<-as.factor(wine$Quality)
indeksy_testowe <- sample(1:nrow(wine), 480, replace=F)
zbior_testowy <- wine[indeksy_testowe, ]
zbior_treningowy <- wine[-indeksy_testowe, ]
```

Klasyfikator kNN

Zadanie 3. Wykorzystując funkcję `sapply()` i bibliotekę `ggplot2`, przedstaw na wykresie punktowym zależność accuracy od wartości parametru k dla k od 1 do 15. Wykorzystaj w tym celu zbiór testowy i treningowy stworzony w powyższym przykładzie.
Wskazówka: Postaraj się, aby kod służący do otrzymania wektora zawierającego *accuracy* dla różnych wartości k zajął jedynie jedną linijkę.

```
acc<-function(x)
{
  wynik <- knn(zbior_treningowy[,-1], zbior_testowy[,-1], zbior_treningowy[,1], k=x)
  mean(wynik==zbior_testowy[,1])
}
x=1:15
y<-sapply(x,acc)
d<-data.frame("accuracy"=y,"k"=x)
ggplot(d,aes(y=accuracy,x=k))+geom_line()+theme_minimal()
```



Problem z accuracy

Zadanie 4. Rozpatrz klasyfikator, który przypisuje wszystkim obserwacjom z przykładu klasę A, oraz taki, który przypisuje wszystkim klasę B. Jakie wartości r_A, r_B, p_A, p_B będą miały oba klasyfikatory?

Jeżeli klasyfikator przypisuje wszystkim obiektom klasę A to jego precyzja będzie wynosiła procentowi prawdziwych obserwacji należących do klasy A natomiast recall będzie równy 1. Z drugiej strony dla obiektów klasy B mamy zerową precyzję oraz recall. Jeżeli wszystkie obiekty będą klasyfikowane jako B to role się zamieniają.

Zadanie 5. Utwórz macierz konfuzji dla wyników klasyfikatora kNN na zbiorze testowym z danych `wine`. Skorzystaj z funkcji `table`. Następnie oblicz *precision* oraz *recall* dla każdej klasy. Porównaj wyniki dla 3 wybranych wartości parametru k. Na podstawie otrzymanych wyników spróbuj oszacować odpowiedzi na następujące pytania:

- Jeśli klasyfikator twierdzi, że wino ma jakość 7, to jakie jest prawdopodobieństwo, że wino rzeczywiście ma taką jakość?
- Jeśli wino ma jakość 5, to jakie jest prawdopodobieństwo, że klasyfikator zaklasyfikuje je poprawnie?
- Czy klasyfikator lepiej klasyfikuje wina o rzadkich, czy o powszechnych jakościach? Która jakość wina jest poprawnie klasyfikowana najczęściej? Która klasa zwrócona przez klasyfikator jest najbardziej wiarygodna?
- Jaka jest szansa, że w rzeczywistości wino jest lepsze, niż twierdzi klasyfikator? A jaka, że gorsze?

Wskazówka. Do obliczenia *precision* oraz *recall* przydadzą się funkcje `diag()`, `rowSums()` oraz `colSums()`. Na przykład, mając macierz konfuzji w zmiennej `C`, do obliczenia *precision* dla każdej klasy wystarczy napisać `diag(C)/colSums(C)`.

```
matpar<-function(i,j,true,pred)
{
  sum(as.numeric(true)==i & as.numeric(pred)==j)
}
create_conf_matrix<-function(x)
{
  wynik <- knn(zbior_treningowy[,-1], zbior_testowy[,-1], zbior_treningowy[,1], k=x)
  table(zbior_testowy[,1],wynik)
}
prec<-function(confus)
{
  diag(confus)/colSums(confus)
}
reca<-function(confus)
{
  diag(confus)/rowSums(confus)
}
prec(create_conf_matrix(3))
```

```
##         3         4         5         6         7         8         9
##       NaN 0.3000000 0.5533333 0.6347032 0.5348837 0.4666667       NaN
```

```
reca(create_conf_matrix(3))
```

```
##         3         4         5         6         7         8         9
## 0.0000000 0.0625000 0.6335878 0.6194690 0.5000000 0.3888889       NaN
```

```
prec(create_conf_matrix(7))
```

```
##         3         4         5         6         7         8         9
##       NaN 0.4000000 0.5578231 0.6277056 0.4831461 0.0000000       NaN
```

```
reca(create_conf_matrix(7))
```

```
##          3          4          5          6          7          8          9
## 0.00000000 0.12500000 0.63358779 0.65044248 0.47727273 0.05555556        NaN
```

```
prec(create_conf_matrix(13))
```

```
##         3         4         5         6         7         8         9
##       NaN 0.6666667 0.5684932 0.6056911 0.5060241 0.0000000       NaN
```

```
reca(create_conf_matrix(13))
```

```
##         3         4         5         6         7         8         9
## 0.0000000 0.1250000 0.6259542 0.6637168 0.4772727 0.0000000       NaN
```

Bazując na wynikach funkcji precision dla różnych wartości k dochodzimy do wniosku, że klasa nr 7 jest dobrze oceniana w środnio połowie przypadków. Prawdopodobieństwo zaklasyfikowania klasy nr 5 poprawnie waha się od $0,55$ do $0,63$ w zależności od wartości k i największe wyniki uzyskuje dla najmnijszych wartości k. Klasyfikator najlepiej klasyfikuje wina jakości 5 oraz 6 (przy czym wino jakości 6 z lekko większą dokładnością). Najlepiej wychwytywaną klasą spośród wszystkich win jest także klasa 6 dla wiekszych wartości k oraz 5 dla $k = 3$.

```
a<-create_conf_matrix(3)
wynik <- knn(zbior_treningowy[,-1], zbior_testowy[,-1], zbior_treningowy[,1], k=3)
prob_less<-sum(sapply(1:6,function(x) sum(a[(x+1):7,x])))/length(wynik) #suma elementów nad diagonalą p
prob_more<-sum(sapply(1:6,function(x) sum(a[x,(x+1):7])))/length(wynik) #suma elementów pod diagonalą p
```

Widzimy, że prawdopodobieństwo zaklasyfikowania wina lepszym niż jest to 0.1958333 natomiast wina gorszym niż jest w rzeczywistości to 0.2354167.