Projekt 1

Anna Niżnik

10 października 2017

Opis zbioru danych treningowych

W zbiorze treningowym jest dostępnych 51 zmiennych, są to dane zarówno jakościowe jak i ilościowe.

Zmiennych ilościowych jest 30.

Są to: **d1, g1, h1, i1, j1, m1, n1, o1, q1, s1, u1, v1, w1, x1, a2, b2, e2, f2, g2, h2, k2, l2, m2, n2, s2, t2, u2, v2, w2, x2**. Mają one 4 poziomy, od A do D.

Zmiennych jakościowych jest 20.

Są to natomiast: **a1, b1, c1, e1, f1, k1, l1, p1, r1, t1, y1, z1, c2, d2, i2, j2, o2, p2, q2, r2**.

Klasy te nie są zrównoważone.

Przedstawienie elementów ze zbioru treningowego

```
##         y A1 B1 C1  D1 E1 F1   G1   H1  I1   J1 K1 L1   M1   N1  O1 P1  Q1
## 1 klasa -  A  B  A 7.2  B  B 17.3 96.8 8.9 62.7  A  A 98.2 48.1 5.3  C 2.3
##   R1 S1 T1   U1   V1   W1   X1 Y1 Z1   A2   B2 C2 D2   E2   F2 G2   H2 I2
## 1  B 17  B 37.6 61.9 20.9 54.3  B  C 43.8 18.7  C  D 84.3 33.9 86 92.8  B
##   J2   K2  L2   M2   N2 O2 P2 Q2 R2   S2   T2   U2 V2   W2   X2
## 1  B 37.3 8.4 97.8 70.2  B  C  A  B 81.4 53.9 60.9 51 42.1 39.1
```

W początkowej fazie analizy, można skorzystać z funkcji **summary**, aby spojrzeć na jego ogólne statystyki.

Analiza zmiennych ze zbioru treningowego

Podczas analizy zbioru treningowego warto przyjrzeć się osobno zmiennym ilościowym i jakościowym. Wówczas łatwiej będzie można wykryć zależności pomiędzy zmiennymi.

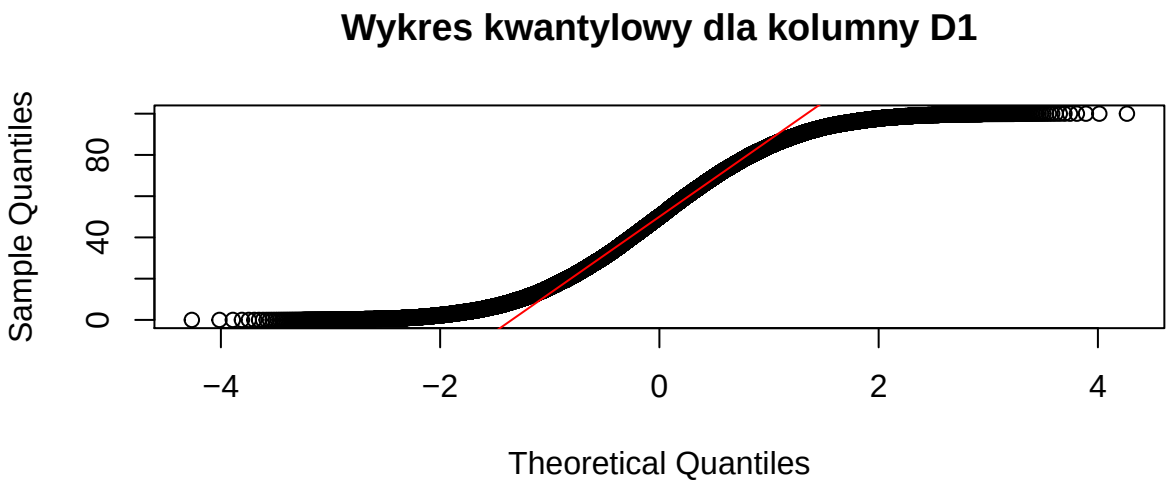
Analiza zmiennych ilościowych

Dla przeprowadzenia analizy, zmodyfikowano wejściową ramkę danych. Pobrano z niej tylko wartości numeryczne.

```
## (0, 100)
```

Warto zauważyć, że zmienne ilościowe mają wartości z zakresu [1-100]. Jest to zatem rozkład jednostajny.

Dla zmiennych dokonano także sprawdzenia, czy posiadają rozkład normalny. Badania przeprowadzono zarówno na całej ramce danych, jak i dla pierwszej kolumny, czyli dla D1.



Na podstawie powyższych wykresów można zauważyć, że zarówno dla całej ramki danych, jak i dla danych z kolumn, rozkład wartości nie jest normalny.

Analiza zmiennych jakościowych

Analogicznie jak powyżej, pobrano dane zawierające tylko zmienne jakościowe i dla nich przeprowadzono analizę.

```
levels(factor_train_vector)
```

```
## [1] "A" "B" "C" "D"
```

Jak wcześniej zauważono, są 4 poziomy zmiennych jakościowych.

Dla zmiennych jakościowych została także przeprowadzona analiza składowych głównych, która wykazała, że wszystkie zmienne są istotne dla tego modelu.

Liczba zmiennych w zbiorze danych jakościowych:

```
## [1] 950000
```

Liczba składowych głównych w zbiorze danych po zastosowaniu PCA:

```
## [1] 950000
```

Selekcja zmiennych

W zbiorze danych znajduje się zbyt wiele zmiennych. W rezultacie podczas analizy może to doprowadzić do szumu bądź zbyt wysokiej wariancji, które dadzą mało stabilne rozwiązanie.

Selekcję zmiennych można wykonać na dwa różne sposoby:

1) Korzystając z funkcji **chi-kwadrat**

Dla każdej zmiennej należy obliczyć p-value, a następnie zwrócić te wyniki, które mają wartość mniejszą niż 5%.

```
columns <- colnames(train)

chisq_result <- lapply(train, function(column){
  chisq_column_result <- chisq.test(train$y, column)
  if(chisq_column_result$p.value < 0.1){
    return(chisq_column_result$p.value)
  }
})
```

Otrzymane wyniki:

```
##             y            A1            B1            E1            F1
##  0.000000e+00  1.166366e-02  1.356103e-02 5.659211e-285  8.234509e-03
##            J1            L1            M1            Q1            W1
##  1.107462e-26  4.656045e-02  0.000000e+00  6.923376e-25  0.000000e+00
##            F2            L2            N2            P2            T2
##  2.572146e-28  4.665200e-02  2.149561e-03 4.603660e-260  3.478023e-21
##            U2
##  3.796093e-40
```

Na podstawie otrzymanych wyników można zauważyć, że zostało wybranych 15 zmiennych.

2) Korzystając z **regresji logistycznej**

Regresja logistyczna pozwala na włączenie do modelu zmiennych zarówno ilościowych (mierzonych na skali interwałowej), jak i ilościowych (mierzonych na skali nominalnej).

Na podstawie otrzymanego wyniku, można odrzucić zmienne, dla których pojawiła się wysoka wartość p-value. Świadczy ona o tym, że pomiędzy zmiennymi nie pojawia się zależność, więc należy je wykluczyć z modelu. Funkcja summary pomaga w wyborze tych nieistotnych zmiennych.

W wybranym modelu, pozostało 13 zmiennych.

Są to: **A1, B1, E1, F1, M1, P1, Q1, W1, Z1, O2, P2, T2, U2**.

Aby potwierdzić otrzymane wyniki, można wykorzystać funkcję step.

W celu ograniczenia liczby zmiennych, można wykorzystać kryterium AIC lub BIC, które biorąc pod uwagę wymiary, porównują modele. Mniejsze błędy można uzyskać stosując kryterium błędu BIC.

```
selekcja_cech <- step(glm_wynik, direction = "both", scope = glm_wynik, trace=FALSE)
```

Funkcja step ograniczyła liczbę zmiennych z 13 do 7. Po wykonaniu obliczeń wybrane zmienne to:

E1, M1, Q1, W1, P2, T2, U2. Na tych zmiennych będą wykonywane dalsze operacje.

Klasyfikacja na wybranych zmiennych

Aby budować model probabilistyczny określający, dla jakich wartości X bardziej prawdopodobne jest zaobserwowanie "klasy +", wykorzystano dwa klasyfikatory: **Glm** oraz **Random forest**. Pozwoli to na wybranie lepszego wyniku i w rezultacie zbudowanie lepszego modelu.

1) Klasyfikator **Glm**:

Aby przeprowadzić klasyfikację, należy użyć zmiennych wybranych podczas selekcji cech. Następnie warto podzielić zbiór wejściowy na testowy i treningowy. Dla wybranego zbioru testowego będzie przeprowadzana klasyfikacja.

Zbiór treningowy liczący **50 000** wyników został podzielony na testowy zawierający **15 000** oraz treningowy zawierający **35 000** rekordów.

```
glm_train <- glm_data[1:35000,]
glm_test <- glm_data[35001:50000,]
```

Klasyfikacja dla nowym zbiorze o zmodyfikowanej liczbie zmiennych:

```
glm_result <- glm(y ~ ., glm_train, family="binomial")
```

Predykcja przypisań do klas:

```
prediction <- as.vector(predict(glm_result, glm_test))
```

Wybór **1000** najlepszych wyników z otrzymanego wektora predykcji:

```
ordered_prediction <-order(prediction, decreasing=TRUE)

class_plus <- glm_test[ordered_prediction,c("y")] == "klasa +"
result <- class_plus[1:2000] #Wybór 2000 wyników z nadaną klasą +
probability <- mean(result == TRUE)
```

Prawdopodobieństwo poprawnego przypisania do klas:

```
## Prawdopodobieństwo: 0.752
```

2) Klasyfikator **Random Forest**:

Aby przeprowadzić klasyfikację za pomocą lasów losowych, należy użyć zmiennych wybranych podczas selekcji. Następnie warto podzielić zbiór wejściowy na testowy i treningowy. Dla wybranego zbioru testowego będzie przeprowadzana klasyfikacja.

Zbiór treningowy liczący 50 000 wyników został podzielony na testowy zawierający 15 000 oraz treningowy zawierający 35 000 rekordów. Zostało to wykonane analogicznie jak dla w przypadku regresji logistycznej. Następnie dokonano klasyfikacji metodą lasów losowych, predykcji i jak wcześniej policzono prawdopodobieństwo otrzymania **klasy +**.

```
## Prawdopodobieństwo: 0.698
```

Można zauważyć, że dla wybranych zmiennych klasyfikator **Random forest** uzyskuje gorsze wyniki.

> Wniosek: W zbudowaniu rankingu dla 50 000 obserwacji pochodzących ze zbioru testowego należy wykorzystać klasyfikator **Glm**.

Sprawdzenie istotnych zmiennych w wybranym modelu:

Do selekcji zmiennych można także wykorzystać **lasy losowe**. Aby sprawdzić zmienne istotne w modlu można użyć funkcji **varImp**.

```
##      Overall
## E1  945.4437
## M1 3749.4349
## Q1 2659.7992
## W1 3521.8129
## P2  898.1465
## T2 2656.7938
## U2 2701.8973
```

Na podstawie otrzymanego wyniku widać, że wszystkie wybrane zmienne okazały się istotne dla tego modelu.

Zbior testowy

Do zbudowania modelu i wyznaczenia rankingu wykorzystano klasyfikator **Glm** oparty na modelu składającym się z 7 wybranych wyżej zmiennych.

```
glm_data <- train[, c("y", "E1","M1","Q1","W1","P2","T2","U2")]
```

Na wybranym zbiorze analogicznie dokonano klasyfikacji, predykcji przypisań do klas. Predykcję wykonano w ten sposób, że dla wartości prawdopodobieństw wyższych niż 50% przypisywano klasę +, w przeciwnym przypadku klasę -.

Dodanie nowej kolumny **score** oraz zapisanie otrzymanych wyników do pliku **anna_niznik.txt**.