# WYKŁAD II: Klasyfikacja logistyczna

MiNI PW, semestr letni 2013/2014

Rozpatrywane dotąd metody klasyfikacji:

- LDA Fishera (liniowa reguła klasyfikacyjna);
- Reguła Bayesowska (jej wersja empiryczna dla rozkładów normalnych ze wspólną macierzą $\Sigma$ pokrywa się z LDA).

Inne metody liniowe ?
Klasyfikacja logistyczna oparta na modelu regresji logistycznej.
Jedno z zastosowań: reklamy pojawiające się na stronie są dobierane na podstawie modelu regresji logistycznej/probitowej gdzie zmiennymi objaśniajacymi są słowa kluczowe.

# Regresja logistyczna

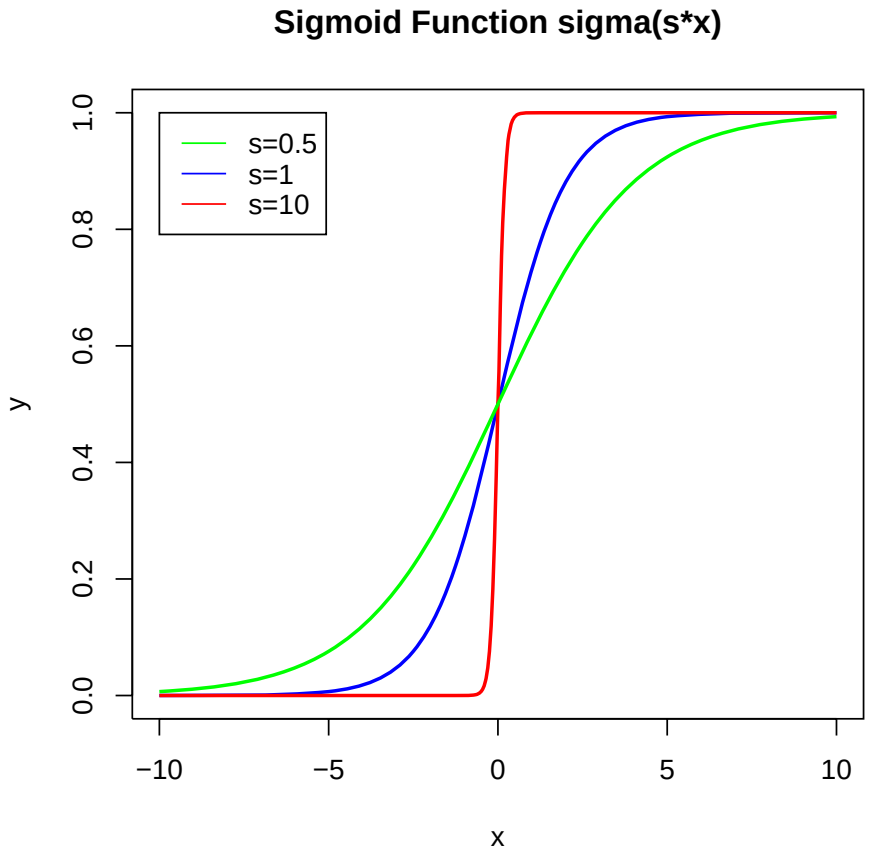
Bardzo częsta sytuacja: odpowiedź $Y$ jest zerojedynkowa, chcemy stwierdzić, jak zależy od wektora zmiennych objaśniających $\mathbf{x}$.
Najczęstsza sytuacja $Y$: 'sukces' (figuratywnie pojmowany) lub 'porażka'
Z reguły nie stosujemy bezpośrednio modelu regresji liniowej ($Y$ w modelu regresji liniowej jest cechą ilościową): problem estymacji prawd. aposteriori, maskowania się klas.

$$P(Y = 1|\mathbf{x}) = \pi(\mathbf{x}) = E(Y|\mathbf{x}) \quad \mathbf{x} \in R^p$$

$\pi(\mathbf{x})$ – modelujemy zależność $\pi$ od $\mathbf{x}$, a nie $Y$ od $\mathbf{x}$.
Regresja logistyczna

$$\pi(\mathbf{x}) = \frac{\exp(\beta'\mathbf{x})}{1 + \exp(\beta'\mathbf{x})} = \frac{1}{1 + \exp(-\beta'\mathbf{x})}$$

Niech $p = 1$ i $\beta = s$, popatrzmy na zachowanie się funkcji $1/(1 + \exp(-sx))$




Dla małych $s$ główna część krzywej w przybliżeniu liniowa, dla dużych $s$ - indykator zbioru $(0, \infty)$.

Dlaczego taka funkcja?
Dowolna dająca $0 \leqslant \pi(\mathbf{x}) \leqslant 1$ jest dobra. Ale ..

$$\text{logit}(\pi(\mathbf{x})) := \log(\frac{\pi(\mathbf{x})}{1-\pi(\mathbf{x})}) = \boldsymbol{\beta}'\mathbf{x},$$

logarytm szansy (szansa (odds): $\pi/(1-\pi)$) jest kombinacją liniową predyktorów.
Mamy zatem ( dla $\mathbf{x} = (1, x)'$)
$\text{logit}(\pi(x+1)) - \text{logit}(\pi(x)) = \beta_0 + \beta_1(x+1) - (\beta_0 + \beta_1 x) = \beta_1$

$$\exp(\beta_1) = \frac{\pi(x+1)}{1-\pi(x+1)}\Big(\frac{\pi(x)}{1-\pi(x)}\Big)^{-1}$$

$\exp(\beta_1)$ jest równa ilorazowi szans.

## Estymacja $\beta$

Założenia: $Y_1, Y_2, \ldots, Y_n$ – niezależne, $Y_i \sim Bin(1, \pi(\mathbf{x}_i))$

$$f_i(Y_i) = \pi_i^{Y_i}(1-\pi_i)^{1-Y_i}$$

$$\prod_{i=1}^{n} f_i(Y_i) = \prod_{i=1}^{n} \pi_i^{Y_i}(1-\pi_i)^{1-Y_i} =: L$$

$$\mathcal{L} = \log L = \sum Y_i \log(\frac{\pi_i}{1-\pi_i}) + \sum \log(1-\pi_i) =$$

$$\sum Y_i \beta' \mathbf{x}_i - \sum \log\{1 + \exp(\beta' \mathbf{x}_i)\}$$

Iteracyjne szukanie 0 pochodnej $\mathcal{L}$ metodą Raphsona–Newtona.

Inne możliwości: inaczej modelowana zależność $\pi(\mathbf{x})$ od $\mathbf{x}$. Regresja probitowa:

$$\Phi^{-1}(\pi(\mathbf{x})) = \beta'\mathbf{x}$$

$\Phi$ – dystrybuanta $N(0,1)$.

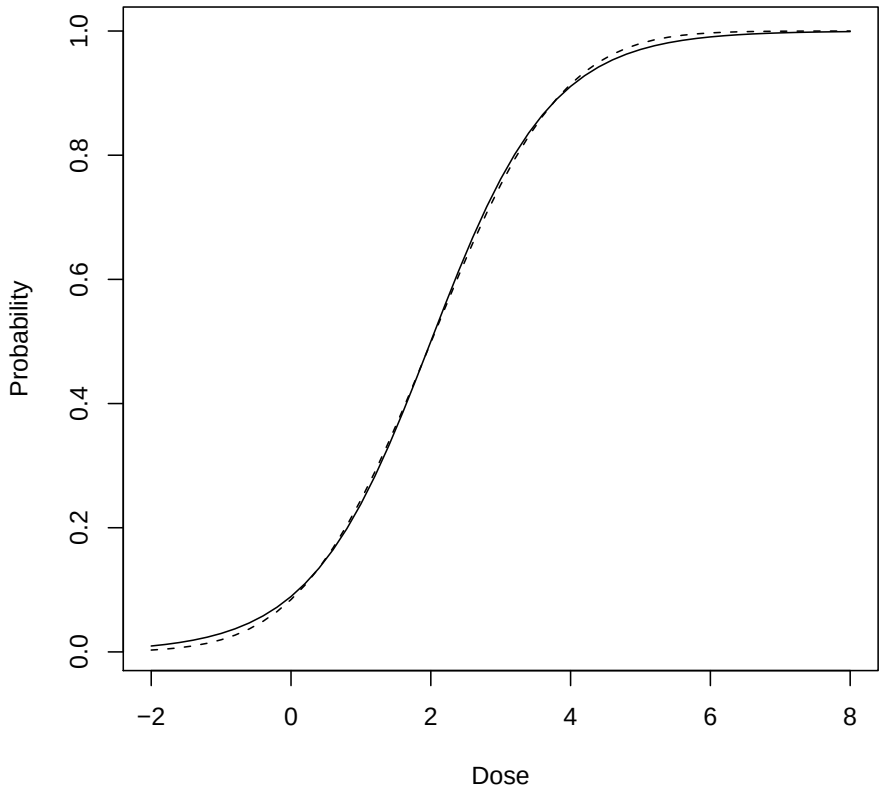
Zbior danych `bliss` dane dotyczący skuteczności środka owadobójczego. Dopasowanie modelu regresji logistycznej procedura **glm**, opcja family="binomial".

```
 dead alive conc
1    2    28    0
2    8    22    1
3   15    15    2
4   23     7    3
5   27     3    4

g <- glm(cbind(dead,alive) ~ conc, family=binomial, data=bliss)
gp <- glm(cbind(dead,alive) ~ conc, family=binomial(link=probit),
data=bliss)
pl= g$fit
   1          2          3          4          5
0.08917177 0.23832314 0.50000000 0.76167686 0.91082823

ilogit(g$coef[1]+g$coef[2]*bliss$conc)  # otrzymujemy to samo

pp=gp$fit
x <- seq(-2,8,0.2)
plot(x,pl,type="l",ylab="Probability",xlab="Dose")
lines(x,pp,lty=2)
```




Praktycznie bez różnicy dopasowania, poza ogonami.

# Odchylenie modelu od modelu, testy istotności współczynników

Niech $\omega$ będzie modelem regresji logistycznej o zmiennych $x_1, \ldots, x_q$.
$\omega \subset \Omega$, $\Omega$ - większy model zawierający dodatkowo zmienne $x_{q+1}, \ldots, x_p$.
Chcemy testować hipotezę, czy zmienne $x_{q+1}, \ldots, x_p$ wnoszą istotną wiedzę do modelu.
$H_0$ : $\omega$ (model $\omega$ jest adekwatny)
przeciwko
$H_1$ : $\Omega$ (model $\Omega$ jest adekwatny, a $\omega$ nie jest).
Testowanie hipotezy opiera się o statystykę odchylenia modelu $\Omega$ od $\omega$ wynoszącą

$$D_{\omega,\Omega} = 2 \ln \left\{ \frac{L(\hat{\beta}^{\Omega})}{(L(\hat{\beta}^{\omega})} \right\} \geqslant 0,$$

gdzie $L(\hat{\beta}^{\Omega})$ jest funkcją wiarogodności policzoną w estymatorze największej wiarogodności w modelu $\Omega$.

Fakt Przy spełnieniu hipotezy $H_0$ zmienna $D$ ma dla dużych liczności próby rozkład $\chi^2$ z $p - q$ stopniami swobody.

Typowe zastosowania:

- Istotność zestawu zmiennych: $\omega : y \sim 1$, $\Omega : y \sim x_1 + \ldots + x_p$.
- Istotność pojedynczej zmiennej dodanej do modelu: $\omega : y \sim x_1 + \ldots + x_q$, $\Omega : y \sim x_1 + \ldots + x_{q+1}$.

Sprawdźmy, czy zmienna `conc` istotnie wpływa na prawdopodobieństwo, że środek jest skuteczny. Wystarczy odwołać się do obiektu $g$ i wywołać jego statystyki zbiorcze (summary).

```
> summary(g)

Call:
glm(formula = cbind(dead, alive) ~ conc, family = binomial,
data = bliss)

.....................

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -2.3238     0.4179  -5.561 2.69e-08 ***
conc          1.1619     0.1814   6.405 1.51e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

    Null deviance: 64.76327  on 4  degrees of freedom
Residual deviance:  0.37875  on 3  degrees of freedom
AIC: 20.854
```

Odchylenie obliczamy jako różnicę między `null deviance` i `residual deviance`, $D = 64.7 - 0.4 = 64.3$, większe od kwantyla $q_{0.01}$ rozkładu $\chi^2$ z jednym stopniem swobody ($= 6.63$). Odrzucamy hipotezę o nieistotności zmiennej `conc`. Sprawdźmy jeszcze, czy do modelu warto dołączyć kwadrat tej zmiennej.

```
g2 <- glm(cbind(dead,alive) ~ conc +I(conc^2), family=binomial,
data=bliss)
> summary(g2)

...................

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.49589    0.59869  -4.169 3.06e-05 ***
conc         1.41018    0.61696   2.286   0.0223 *
I(conc^2)   -0.06117    0.14319  -0.427   0.6692

.....................

    Null deviance: 64.76327  on 4  degrees of freedom
Residual deviance:  0.19549  on 2  degrees of freedom
```

Odchylenie modelu z dwiema zmiennymi od modelu z jedną zmienną liczymy jako różnicę odchyleń resztowych ($D = 0.38 - 0.20 = 0.18$), wartość jest nieistotna przy porównaniu z kwantylem rozkładu chi kwadrat z jednym stopniem swobody. Oba wyniki potwierdzane przez statystykę Walda $t = \hat{\beta}/SE(\hat{\beta})$ (wartość z-value, trzecia kolumna zbioru wynikowego).
Uwaga Wartość `residual deviance` jest odchyleniem między rozpatrywanym modelem a tzw. modelem nasyconym, w którym liczba parametrów jest równa liczbie obserwacji. `Residual deviance` jest czasami wykorzystywana do testowania adekwatności modelu w schemacie $H_0 : \omega$ vs $H_1 : \Omega_{nasycony}$.

Statystyka $D$ ma w przybliżeniu rozkład chi kwadrat z $n - p$ stopniami swobody, ale tylko dla danych grupowanych, takich jak dane `bliss`, gdy liczba obserwacji dla ustalonej wartości zmiennych wynosi co najmniej 5.

Uogólnienie na $g \geqslant 2$.

Wbieramy populację referencyjną np. ostatnią (o numerze $g$)

$$\log \frac{p(1|\mathbf{x})}{p(g|\mathbf{x})} = \boldsymbol{\beta}_1' \mathbf{x}$$

$$\log \frac{p(2|\mathbf{x})}{p(g|\mathbf{x})} = \boldsymbol{\beta}_2' \mathbf{x}$$

$$\cdots\cdots\cdots\cdots$$

$$\log \frac{p(g-1|\mathbf{x})}{p(g|\mathbf{x})} = \boldsymbol{\beta}_{g-1}' \mathbf{x}$$

$\boldsymbol{\beta}_i' = (\beta_{i0}, \beta_{i1}, \ldots, \beta_{ip})'$
Nieznane parametry: $\boldsymbol{\beta}_1', \ldots, \boldsymbol{\beta}_{g-1}' \in R^{p+1}$, łącznie $(g-1)(p+1)$ parametrów jednowymiarowych.

**Uwaga** W pakiecie R jako populacja referencyjna wybierana jest ta, której nazwa jest pierwsza w porządku leksykograficznym.

Estymacja parametrów metodą NW: $\hat{\boldsymbol{\beta}}_1, \ldots, \hat{\boldsymbol{\beta}}_{g-1} \longrightarrow \hat{p}(1|\mathbf{x}), \ldots, \hat{p}(g|\mathbf{x})$

$$\hat{p}(k|\mathbf{x}) = \frac{\exp(\hat{\boldsymbol{\beta}}_k' \mathbf{x})}{1 + \sum_{i=1}^{g-1} \exp(\hat{\boldsymbol{\beta}}_i' \mathbf{x})} \quad k = 1, \ldots, g-1$$

$$\hat{p}(g|\mathbf{x}) = \frac{1}{1 + \sum_{i=1}^{g-1} \exp(\hat{\boldsymbol{\beta}}_i' \mathbf{x})}$$

Reguła dyskryminacyjna

Reguła bayesowska oparta na estymatorach otrzymanych w modelu logistycznym:

$$\text{Klasyfikuj do populacji } l \text{ gdzie } \; l = \underset{i}{\operatorname{argmax}}\, \hat{p}(i|\mathbf{x})$$

Zauważmy, że w modelu logistycznym w naturalny sposób otrzymujemy oszacowania interesujących prawdopodobieństw aposteriori i nie ma potrzeby oddzielnej estymacji $\pi_i$ i $p(x|i)$.

Nota bene

Jeśli $p(\mathbf{x}|i)$: gęstość rozkładu $N(\mathbf{m}_i, \Sigma)\, i = 1, \ldots, k$.
to

$$\log \frac{p(k|\mathbf{x})}{p(g|\mathbf{x})} = \frac{1}{2}(\mathbf{x}-\mathbf{m}_g)'\Sigma^{-1}(\mathbf{x}-\mathbf{m}_g) - \frac{1}{2}(\mathbf{x}-\mathbf{m}_k)'\Sigma^{-1}(\mathbf{x}-\mathbf{m}_g) + \log \frac{\pi_k}{\pi_g} =$$

$$\frac{1}{2}(\mathbf{m}_k - \mathbf{m}_g)'\Sigma^{-1}\mathbf{x} - \frac{1}{2}(\mathbf{m}_k + \mathbf{m}_g)'\Sigma^{-1}(\mathbf{m}_k - \mathbf{m}_g) + \log \frac{\pi_k}{\pi_g}$$

ma postać $\beta_k'\mathbf{x}$.
Ta zależność była również wykorzystywana w metodzie LDA.

Czym zatem różnią się te dwie metody?

Sposobem estymacji parametrów.

W regresji logistycznej maksymalizujemy ($g = 2$)

$$L = \prod_{i=1}^{n} P(Y = 1|X = \mathbf{x}_i)^{y_i}(1 - P(Y = 1|X = \mathbf{x}_i))^{1-y_i}$$

To jest warunkowa funkcja wiarogodności $p(y_1, \ldots, y_n|\mathbf{X} = \mathbf{x})$ wykorzystująca jedynie warunkowy rozkład $Y$ pod warunkiem $\mathbf{X}$. (brzegowy rozkład $\mathbf{X}$ nie odgrywa tu roli, nic o nim nie zakładamy!)

W przypadku LDA gęstość $p(\mathbf{X} = \mathbf{x}, Y = k)$ ma postać

$$p(\mathbf{x}, k) = \phi(\mathbf{x}, \boldsymbol{\mu}_k, \Sigma)\pi_k$$

Maksymalizacja pełnej funkcji wiarogodności o postaci

$$\tilde{L} = \prod_{i=1}^{n} p(\mathbf{x}_i, y_i)$$

prowadzi do rozpatrywanych poprzednio estymatorów

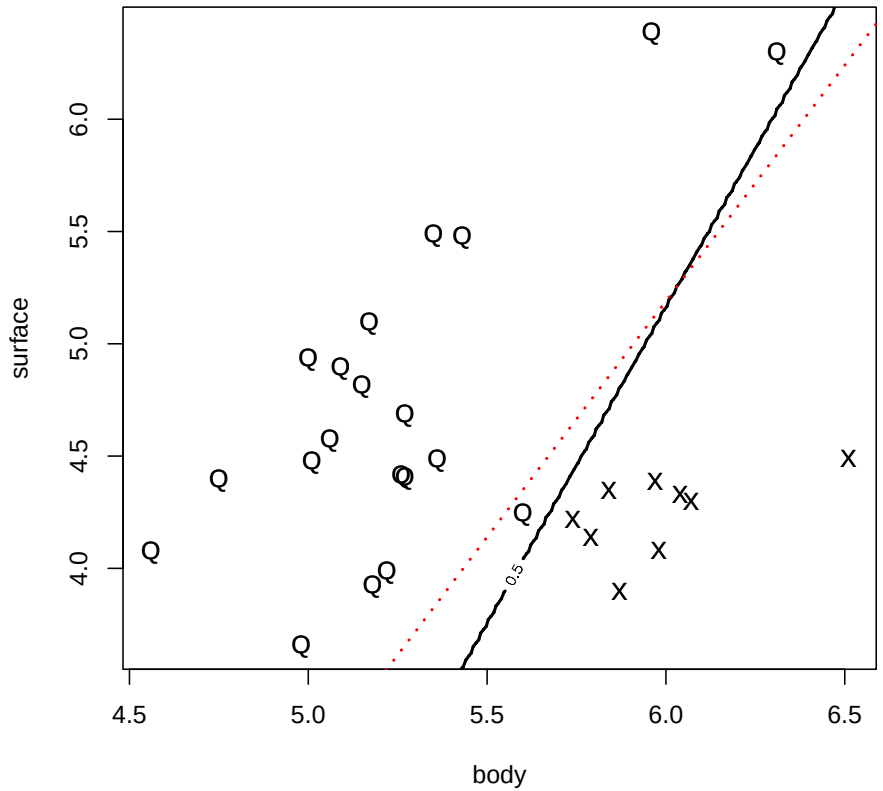
$$\hat{\boldsymbol{\mu}}_i = \bar{\mathbf{x}}_i$$

$$\hat{\Sigma} = \frac{1}{n-g} \sum_{k=1}^{g} (n_k - 1) S_k$$

$$\hat{\pi}_i = \frac{n_i}{n}$$

Można spodziewać się, że regresja logistyczna nie jest tak czuła na duże odstępstwa od normalności i równość macierzy kowariancji jak LDA.

Wykres rozproszenia danych `earthquake` z obszarami klasyfikacji wyznaczonymi przy użyciu klasyfikacji logistycznej (linia ciągła) i LDA (linia przerywana).



## Dane earthquake

Dopasujemy model logistyczny popn $\sim$ body + surface. Tworzymy nową ramkę danych z zero-jedynkową zmienną y zamiast popn. Dla dopasowania modelu logistycznego y $\sim$ body + surface wykorzystywana funkcja glm.
glm (skrót od *generalized linear model*) pozwala na dopasowanie modelu z klasy uogólnionych modeli liniowych.

Opcja family=binomial specyfikuje model logistyczny.

```
earthquake = read.table("earthquake.txt", header=TRUE)

equake = data.frame(y=ifelse(earthquake$popn=="equake", 0, 1),

body=earthquake$body, surface=earthquake$surface)

g2 = glm(y~ body + surface, data=equake, family=binomial)
```

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1151.09 1000874.08  -0.001    0.999
body           276.21  190711.00   0.001    0.999
surface        -98.01  118520.53  -0.001    0.999

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3.5924e+01  on 28  degrees of freedom
Residual deviance: 3.7043e-09  on 26  degrees of freedom
AIC: 6
```

Wartość odchylenia resztowego (residual deviance) jest bardzo mała i wskazuje na bardzo dobre dopasowanie, gdy jednocześnie wyniki testu $t$ mówią o nieistotności obu zmiennych. Taka paradoksalna sytuacja występuje często przy liniowej separowalności klas, gdy estymatory współczynników w modelu regresji logistycznej i ich błędy standardowe zachowują się niestabilnie ($p(1|\mathbf{x}_i) \approx 1, 0,\ i = 1, \ldots, n \Rightarrow ||\hat{\beta}||$-duża).

Tabelka i procent poprawnych reklasyfikacji.

```
Ypred =ifelse(g2$fitted.values < 0.5, 0, 1)

# klasyfikacja do klasy 1  dla prawd. aposteriori klasy 1 < 0,5.

print(kl =table(equake$y, Ypred))
print(procent= sum(diag(kl)) / sum(kl))
```

```
   Ypred
     0  1
  0 20  0
  1  0  9

[1] 1
```

Działanie klasyfikatora logistycznego różni się od klasyfikatora LDA: pierwszy z nich klasyfikuje bezbłędnie wszystkie elementy próby uczącej (sytuacja liniowo separowalnych klas). *Nie* należy wyciągać stąd wniosku, że klasyfikator logistyczny będzie działał lepiej dla nowych obserwacji.

Dane **urine**, wybór zmiennych w klasyfikacji.
Zmienna presence jest zmienną grupującą, pozostałe atrybuty: wartości pomiarów fizyko-chemicznych moczu. Model logistyczny presence $\sim$ sg+ph+mosm+mmho+urea+calcium

```
urine.glm=glm(presence ~ ., family = binomial, data = urine)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -355.33771  222.76696  -1.595  0.11069
sg           355.94379  222.11004   1.603  0.10903
ph            -0.49570    0.56976  -0.870  0.38429
mosm           0.01681    0.01782   0.944  0.34536
mmho          -0.43282    0.25123  -1.723  0.08493
urea          -0.03201    0.01612  -1.986  0.04703
calcium        0.78369    0.24216   3.236  0.00121

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 105.17  on 76  degrees of freedom
Residual deviance:  57.56  on 70  degrees of freedom
AIC: 71.56
```

$$Dev_{null,\omega} = Dev_{null} - Dev_{resid}.$$

Duża różnica odchylenia zerowego (null deviance) i resztowego (residual deviance) wskazuje na występowanie istotnych zmiennych w modelu, odpowiednia $p$-wartość, uzyskana na podstawie rozkładu chi kwadrat z 6 stopniami swobody wynosi pchisq(105.17- 57.56,7-1,lower=F).
i jest mniejsza od 0.001. Tabelka reklasyfikacji i procent poprawnej reklasyfikacji

```
     kl
       0  1
  no  40  4
  yes  8 25
```

```
[1] 0.8441558
```

Dokonajmy redukcji zmiennych w modelu metodą eliminacji wstecz, sprawdźmy, czy mniejszy model można uznać za adekwatny i jak wygląda reklasyfikacja.

urine.glm = glm(presence ∼ ., data=urine, family=binomial)
urine.step = step(urine.glm, direction="backward")
print(urine.step)

. . . . . .

```
Call:  glm(formula = presence ~ sg + mmho + urea + calcium, family = binomial,
           data = urine)

Coefficients:
(Intercept)           sg         mmho         urea      calcium
 -500.01090    497.12038     -0.20547     -0.01783      0.72232

Degrees of Freedom: 76 Total (i.e. Null);  72 Residual
Null Deviance:      105.2
Residual Deviance: 59.07        AIC: 69.07
```

Otrzymany podzbiór zmiennych objaśniających: calcium, mmho, sg, urea uzyskuje się również stosując metodę dołączania.
Przetestujmy teraz, czy model mniejszy jest adekwatny, przy użyciu statystyki równej różnicy odchyleń, która przy hipotezie $H_0$ (model mniejszy jest adekwatny) ma dla dużych liczności w przybliżeniu rozkład $\chi^2$ z $7 - 5 = 2$ stopniami swobody

print(anova(u2.glm, u.glm, test="Chi"))

```
Analysis of Deviance Table

Model 1: presence ~ sg + mmho + urea + calcium
Model 2: presence ~ sg + ph + mosm + mmho + urea + calcium
  Resid. Df Resid. Dev Df Deviance P(>|Chi|)
1        72     59.071
2        70     57.560  2    1.511     0.470
```

Porównując model mniejszy i większy nie odrzucamy hipotezy, że model mniejszy jest adekwatny. Dopasowujemy mniejszy model i przeprowadzamy reklasyfikację.

u2.glm = glm(presence $\sim$ sg + mmho + urea + calcium, data=urine, family=binomial)

```
     kl2
       0  1
  no  40  4
  yes  8 25

[1] 0.8441558
```

Otrzymaliśmy dokładnie takie same wyniki reklasyfikacji, jak dla większego zbioru atrybutów.

# Model proporcjonalnych szans

Przypuśćmy, że zmienna $Y$ jest zmienną nominalną o $g$ uporządkowanych kategoriach (np. kategorie wiekowe, kategorie klienta: spłaty terminowe, spłaty z opóźnieniem, brak spłat). Informacja o uporządkowaniu klas powinna być wykorzystana w modelu. Oznaczmy kategorie jako $1, 2, \ldots, g$. W modelu proporcjonalnych szans dla $j = 1, 2, \ldots, g-1$

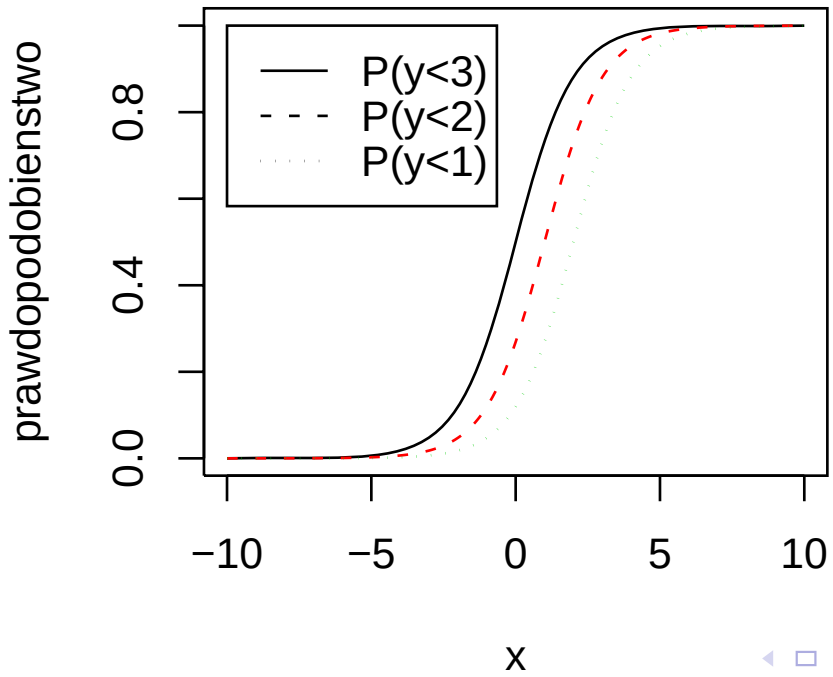
$$\log \frac{Pr(y \leqslant j|\mathbf{x})}{1 - Pr(y \leqslant j|\mathbf{x})} = \alpha_j - \beta'\mathbf{x}, \qquad (\star)$$

gdzie $\mathbf{x} = (x_1, \ldots, x_p)'$ jest wektorem predyktorów. Funkcja logitowa $\log(p/(1-p)) \uparrow$ dla $p \uparrow$ i $Pr(y \leqslant j|x) \uparrow$ gdy $j \uparrow \Rightarrow \alpha_1 \leqslant \alpha_2 \cdots \leqslant \alpha_{g-1}$.
Dla ustalonego $j$ model $(\star)$ jest modelem logistycznej regresji dla odpowiedzi binarnej 1 gdy $\{y \leqslant j\}$, i 0 gdy $\{y > j\}$.
Dla $g = 2$ otrzymujemy model regresji logistycznej.

Waz ze zmianą $j$ in $(\star)$ wyraz wolny $\alpha_j$ się zmienia, podczas gdy wektor $\beta$ pozostaje taki sam. Dla $\gamma_j(\mathbf{x}) = Pr(y \leqslant j|\mathbf{x})$ założenie modelowe oznacza, że funkcja $\gamma_j(\cdot)$ jest przesunięciem funkcji $\gamma_k(\cdot)$. Mianowicie, np. dla jednowymiarowego predyktora mamy

$$\gamma_k(x) = \frac{exp(\alpha_j - \beta(x - (\alpha_k - \alpha_j)/\beta)}{1 + exp(\alpha_j - \beta(x - (\alpha_k - \alpha_j)/\beta)} =$$

$$\gamma_j(x - (\alpha_k - \alpha_j)/\beta).$$



Nazwa 'model proporcjonalnych szans' ( proportional odds) związana z faktem, że założenie modelowe implikuje:

$$\frac{\gamma_i(\mathbf{x}_1)/(1-\gamma_i(\mathbf{x}_1))}{\gamma_i(\mathbf{x}_2)/(1-\gamma_i(\mathbf{x}_2))} = exp(-\beta'(\mathbf{x}_1-\mathbf{x}_2)).$$

Tak więc powyższy iloraz szans nie zależy od $i$. Konwencja znków $\beta$: dla $x_1 < x_2$ przy $\beta > 0$ chcemy, aby powyższy stosunek był $> 1$ ( Uwaga: procedura GENMOD (SAS) używa $\beta$ zamiast $-\beta$ w $(\star)$.
Parametry modelu estymowane przy użyciu metody największej wiarogodności.
Procedury: polr w R i Genmod w SAS.
Inne modele: model proporcjonalnych hazardów

$$\log(-\log(1-\gamma_j(\mathbf{x})) = \alpha_j + \beta'\mathbf{x}$$

Założenie implikuje, że $P(Y > j|\mathbf{x}_1) = P(Y > j|\mathbf{x}_2)^{\exp(\beta'(\mathbf{x}_1-\mathbf{x}_2))}$.

Inne metody dyskryminacji liniowej:
– perceptron Rosenblatta (sieci neuronowe);
– metoda oparta na regresji wielowymiarowej.

Druga metoda:
etykieta klasy kodowana jest jako wektor $g-$wymiarowy

$$y = (y^{(1)}, \ldots, y^{(g)})$$

dla klasy $k$, $y = (0, 0, \ldots, 0, 1, 0, \ldots, 0)$ (1 na $k-$tym miejscu)

$\mathbf{X}$ – macierz eksperymentu $n \times (p+1)$
$\mathbf{Y}$ – macierz odpowiedzi

$$\begin{pmatrix} y_1^{(1)} & \dots & y_1^{(g)} \\ . & & . \\ y_n^{(1)} & \dots & y_n^{(g)} \end{pmatrix}$$

Szukamy macierzy $\hat{\mathbf{B}}_{(p+1)\times g}$ minimalizującej

$$\sum_{i=1}^{n} ||y_i - [1, \mathbf{x}_i']\mathbf{B}||^2$$

– równoważne rozwiązaniu $g$ problemów regresji wielokrotnej oddzielnie. Macierz $\hat{\mathbf{B}}$ składa się z kolumn parametrów dla kolejnych problemów regresji.

Okazuje się, że prognoza $\hat{y}(\mathbf{x}) = [1, \mathbf{x}']\mathbf{B}$ ma własność

$$\sum_{i=1}^{n} \hat{y}^{(k)}(\mathbf{x}) = 1$$

Reguła klasyfikacyjna

$$\delta(\mathbf{x}) = \underset{k=1,2,\ldots,g}{\text{argmax}}\, \hat{y}^{(k)}(\mathbf{x})$$

Komentarz: $\delta(\cdot)$ dopuszcza uogólnienie nieliniowe
– dyskryminacja giętka (flexible discrimination)

## Kwestia skal pomiarowych atrybutów

Dotąd milcząco zakładaliśmy, że atrybuty przyjmują wartości rzeczywiste. Nie ma problemu dla zmiennych ilościowych dyskretnych ze stosowaniem LDA, dyskryminacji logistycznej, empirycznej metody bayesowskiej.

Wartości nominalne
$x$ – przyjmuje $r$ wartości
$i-$ta wartość $\longrightarrow (0, 0, \ldots, 0, 1, 0, \ldots, 0)'$ (1 na $i-$tym miejscu)
musimy mieć dane zawierające obserwacje dla każdego układu atrybutów, aby metoda była stabilna
wartości nominalne na skali porządkowej: metoda *ad hoc*
$i-$ta wartość $\longrightarrow (i-1)/n$

Inna metoda postępowania dla atrybutów nominalnych oparta na naiwnej metodzie bayesowskiej(zakładającej niezależność atrybutów)

$$\frac{p(2|\mathbf{x})}{p(1|\mathbf{x})} = \frac{\pi_2}{\pi_1}\frac{p(\mathbf{x}|2)}{p(\mathbf{x}|1)} = \frac{\pi_2}{\pi_1}\prod_{i=1}^{p}\frac{p(x^{(i)}|2)}{p(x^{(i)}|1)}$$

$\mathbf{x} = (x^{(1)}, \ldots, x^{(p)})'$

$$\log \frac{p(2|\mathbf{x})}{p(1|\mathbf{x})} = \log \frac{\pi_2}{\pi_1} + \sum_{i=1}^{p} \log \frac{p(x^{(i)}|2)}{p(x^{(i)}|1)}$$

atrybut $\mathbf{x}^{(i)}$ – poziomy $l = 1, \ldots, m_i$

$$\hat{P}(\mathbf{x}^{(i)} = l|k) = \frac{n_{ik}(l)}{n_k}$$

$n_{ik}(l)$ – # elementów klasy $k$, dla których $i-$ty atrybut jest równy $l$

$$\frac{p(\mathbf{x}^{(i)} = l|2)}{p(\mathbf{x}^{(i)} = l|1)} \quad \text{estymujemy przez} \quad \frac{n_{i2}(l)}{n_{i1}(l)} \cdot \frac{n_1}{n_2}$$

**Uwaga** Naiwna metoda bayesowska działa często dobrze nawet w przypadku, gdy atrybuty są zależne !