Tabulární data, pozorované vs očekávané četnosti

Máme data o počtech např. samců a samic v populaci a zajímá nás, zda naše pozorované (observed) četnosti se liší od předpokládaného (expected). Příklad je tedy: celkem uloveno 47 zvířat, 29 samců a 18 samic; teoretický předpoklad je že zastoupení samců a samic je shodné, tedy 50:50. Nejprve vytvoříme vektor četností: zver<-c(29,18), který pak použijeme ve funkci chisq.test; chisq.test(zver). Tím, že jsme zadali pouze jeden parametr, fce předpokládá rovnoměrné zastoupení ve skupinách, tedy 50:50.
Výsledkem je p-value = 0.1086, což značí, že nezamítneme hypotézu H0. Tím tedy můžeme říci, že námi pozorované četnosti mohou pocházet z populace se stejným zastoupením samců a samic.

Dalším příkladem na srovnání pozorovaných a očekávaných četností jsou např. Mendelovy hrachy a štěpné poměry.
Data jsou následující:

	žluté, kulaté	žluté, svrasklé	zelené, kulaté	zelené, svrasklé
pozorované četnosti	315	101	108	32
očekávané štěpné poměry	9	3	3	1

Vytvoříme dva vektory pro pozorované a očekávané četnosti: hr.pozor<-c(315,101,108,32) a hr.ocek<-c(9,3,3,1) a následně použijeme fci chisq.test. V případě, že zadáváme očekávané četnosti tak, že součet vektoru není 1, je třeba u fce chisq.test použít parametr rescale.p=TRUE. Samotný test pak tedy vypadá následovně
chisq.test(hr.pozor,hr.ocek,rescale.p=TRUE)
p-value je v tomto případě vysoké, tedy opět nezamítáme hypotézu H0 a můžeme s klidným svědomím prohlásit, že pozorované četnosti odpovídají teoretickým štěpným poměrům.

Frekvenční analýza, čtyřpolní tabulky

V následujícím příkladě nás zajímá, zda sekání má pozitivní vliv na reprodukci studovaného druhu. V experimentu tedy máme dva druhy ošetření (sekané, nesekané) a pro každé máme patnáct ploch. V průběhu sezóny sledujeme, jestli druh vykvete. Nezajímá nás tedy kolik jedinců vykvete, ale zda se na ploše objeví alespoň jedna kvetoucí rostlina.

Výsledná tabulka sledování je zde:

X7	Faktor5	Faktor6
13	1	1
2	1	2
6	2	1
9	2	2

X7 - četnost pozorování, Faktor5 - typ ošetření (sekaná - 1, kontrola - 2), Faktor6 - kvetení (vykvetl - 1, nevykvetl - 2)

Experimenty s četnostmi nějakého jevu v závislosti na působení dvou či více faktorů se řeší pomocí metod frekvenční analýzy. V našem případě máme pouze dva faktory a oba mají pouze dvě hladiny, a tak se jedná o nejjednodušší typ frekvenčních analýz - tzv. čtyřpolní tabulku. Frekvenční analýzy se obvykle řeší pomocí testů Chí-kvadrát nebo Fisherovým

exaktním testem (ten pouze pro čtyřpolní tabulky). Předpokladem a nulovou hypotézou testu je nezávislost působení obou faktorů.

Pokud zadáme data do sloupců (sloupec četností, a kódování pro faktory), lze v S+ pouze vytvořit "Crosstabulation" z menu Statistics>Data Summarries>Crosstabulations...
V příkazovém řádku R fci xtabs.

```
      *** Crosstabulations ***
Call:
crosstabs(formula = X7 ~ Faktor5 + Faktor6, data = SDF1,
      na.action = na.exclude, drop.unused.levels = T)
30 cases in table
+----------+
|N         |
|N/RowTotal|
|N/ColTotal|
|N/Total   |
+----------+
Faktor5|Faktor6
       |1      |2      |RowTotl|
-------+-------+-------+-------+
1      |13     | 2     |15     |
       |0.87   |0.13   |0.5    |
       |0.68   |0.18   |       |
       |0.43   |0.067  |       |
-------+-------+-------+-------+
2      | 6     | 9     |15     |
       |0.4    |0.6    |0.5    |
       |0.32   |0.82   |       |
       |0.2    |0.3    |       |
-------+-------+-------+-------+
ColTotl|19     |11     |30     |
       |0.63   |0.37   |       |
-------+-------+-------+-------+
Test for independence of all factors
      Chi^2 = 7.033493 d.f.= 1 (p=0.007999917)
      Yates' correction not used
```

Tabulka spočte "observed" proporce pro jednotlivé faktory a to jak pro celková data (N/Total) tak i pro jednotlivé řádky a sloupce. Pokud chceme spočíst očekávané "expected" četnosti, musíme počítat v ruce: pro kombinaci 1&1 počítáme z marginálních četností (15*19)/30=9.5.

V menu pro vytvoření tabulky bohužel nelze nastavit podrobnější parametry Chí-kvadrát testu (Yatesova korekce). Pokud bychom chtěli spočíst Fisherův exaktní test či Chí-kvadrát s Yatesovou korekcí, je třeba data zadat buď přímo do kontingenční tabulky, nebo přímo pro každé sledování vlastní řádek (tj. v tomto případě: třináctkrát ve sloupcích Faktor5 a Faktor6 jedna, dvakrát ve sloupci Faktor5 jedna a Faktor6 dva....).
Z menu voláme Statistics>Compare Samples>Counts and Proportions a poté buď Fisherův exaktní test či Chí-kvadrát.
Výsledkem jsou tedy:

```
      Fisher's exact test

data:  Faktor5 and Faktor6 from data set frekv
p-value = 0.0209
alternative hypothesis:
  two.sided
```

```
        Pearson's chi-square test with Yates' continuity correction

data:  Faktor5 and Faktor6 from data set frekv
X-square = 5.1675, df = 1, p-value = 0.023
```

V našem případě oba testy zamítly nulovou hypotézu o nezávislosti působení obou jevů. Pokud chceme říci které kombinace faktorů jak ovlivňují kvetení, musíme ještě dopočítat očekávané četnosti a na základě porovnání se skutečnými správně interpretovat. Z výsledků v našem příkladu můžeme říci, že sekání louky pozitivně ovlivňuje kvetení sledovaného druhu rostliny. Yatesova korekce (dostupná pouze pro tabulky 2×2) je zde použita, protože frekvenční tabulky obsahují diskrétní data, avšak Chí-kvadrát je rozdělení spojité. Zejména by se měla použít, pokud se jedná o tabulky kdy pozorované četnosti jsou nízké (<5). Pro tyto tabulky je vhodnější použít Fisherův exaktní test.

	observed pozorované		expected očekávané
sekaná/vykvetl	13	>	(15*19)/30 = 9.5
sekaná/nevykvetl	2	<	(15*11)/30 = 5.5
kontrola/vykvetl	6	<	(15*19)/30 = 9.5
kontrola/nevykvetl	9	>	(15*11)/30 = 5.5

Alternativou pro analýzu dat kde vysvětlující proměnné jsou kategoriální a vysvětlovanou počty jsou kromě výše uvedených Chí-kvadrát testu a Fisherova exaktního testu i zobecněné lineární modely. Použijeme log-lineární model s poissonovským rozdělením chyb. V S+ pro tyto výpočty používáme funkci glm. Její parametry jsou obdobné jako u např. lm (lineární regrese). Jen je třeba doplnit typ rozložení chyb, v našem případě: poisson. Model uložíme pod jménem "pokus" a píšeme tedy:

```
pokus<-glm(x7~Faktor1+Faktor2, data=SDF1, poisson)
```

Výsledek vyvoláme a z celého výpisu nás zajímá pouze hodnota residuální deviance.

```
pokus
Call:
glm(formula = x7 ~ f1 + f2, family = poisson, data = SDF1)

Coefficients:
 (Intercept)             f1         f2
     1.97802 -2.279783e-008 -0.2732718

Degrees of Freedom: 4 Total; 1 Residual
Residual Deviance: 7.458882
```

Hodnotu residuální deviance srovnáme s hodnotou Chí-kvadrátu s jedním stupněm volnosti. Pro výpočet hladiny významnosti voláme:

```
1-pchisq(7.46,1)
[1] 0.006308501
```

Což ukazuje obdobný výsledek jako když jsme příklad počítali Fisherovým testem či přímo Chí-kvadrátem. Pozorované a očekávané hodnoty si vypíšeme pro určení, které kombinace faktorů mají pozitivní či negativní efekt.

```
> SDF1$x7
[1] 13  2  6  9
> fitted(model)
   1        2   3        4
 9.5 5.500001 9.5 5.500001
```

Logistická regrese

Pokud máme závislou proměnnou kategoriální a ještě k tomu nabývá pouze dvou hodnot (přežil/nepřežil, kvete/nekvete, nakažený/nenakažený) a zajímá nás které faktory a jak je ovlivňují.
V následujícím příkladu nás bude zajímat jak je pravděpodobnost přežití rostliny ovlivněna množstvím vytvořených semen v závislosti na její velikosti. K dispozici máme údaje o 59 jedincích, zda přežili do další sezóny (surv: 0 - nepřežil, 1 - přežil; flow - počet úborů, root - velikost kořene). Data jsou z příkladu GLEX26 programového balíku GLIM.
Výpočet je založen na závislosti $p = e^y/(1+e^y)$, kde hodnoty y získáme z regresních koeficientů (viz níže).

1. Výpočet

Vlastní analýzu budeme počítat pomocí logistické regrese z menu Statistics>Regression>Logistic... Menu, kde zadáváme vlastní analýzu se příliš neliší od klasické regrese. V tomto příkladě využijeme možnosti S+ a uložíme si model analýzy jako samostatný objekt se kterým budeme později pracovat. Pro uložení je potřeba jen zadat jméno analýzy do okna "Save Model Object".
Po zadání analýzy máme obecnou rovnici: `surv~flow+root`.
Ve výsledcích jsou opět uvedeny hodnoty pro rozdělení reziduí. Hlavní výsledek je v části "Coefficients" kde jsou uvedeny regresní koeficienty pro jednotlivé proměnné spolu s jejich standardními chybami a t-hodnotami. Pravděpodobnosti pro jednotlivé proměnné je třeba dopočítat ručně či je můžeme otestovat delečními testy (funce `drop1`). Z příkazového řádku voláme logistickou regresi:
`glex.1<-glm(surv~flow+root, data=GLEX26, binomial).`

```
      *** Generalized Linear Model ***

Call: glm(formula = surv ~ flow + root, family = binomial(link =
      logit), data = GLEX26, na.action = na.exclude, control
       = list(epsilon = 0.0001, maxit = 50, trace = F))
Deviance Residuals:
      Min         1Q     Median         3Q      Max
 -1.42963 -0.7899175 -0.1878944 0.7291494 2.359577

Coefficients:
                 Value Std. Error   t value
(Intercept)  0.9615073 0.61588600  1.561177
       flow -0.1064166 0.03331434 -3.194318
       root  6.6003244 2.09356086  3.152679

(Dispersion Parameter for Binomial family taken to be 1 )

    Null Deviance: 78.90332 on 58 degrees of freedom

Residual Deviance: 54.06811 on 56 degrees of freedom

Number of Fisher Scoring Iterations: 5
```

2. Test proměnných (funce drop1)

To zda jsou pro model významné můžeme otestovat kromě t-testu obdobným způsobem jako u "stepwise" analýzy, tedy postupně odebírat jednotlivé proměnné a sledovat výslednou statistiku Cp, která bere v potaz množství vysvětlené variability a počet proměnných zahrnutých v modelu.
Máme-li uložený model pod jménem "glex.1" vyvoláme analýzu postupného odebírání proměnných funkcí `drop1`. V okně "Commands" tedy voláme: `drop1 (glex.1)` a dostáváme tabulku.

```
> drop1(glex.1)
Single term deletions

Model:
surv ~ flow + root
           Df   Sum of Sq RSS         Cp
<none>                     52.46392   58.08506
flow       1    10.20366   62.66759   66.41501
root       1    9.93938    62.40331   66.15073
```

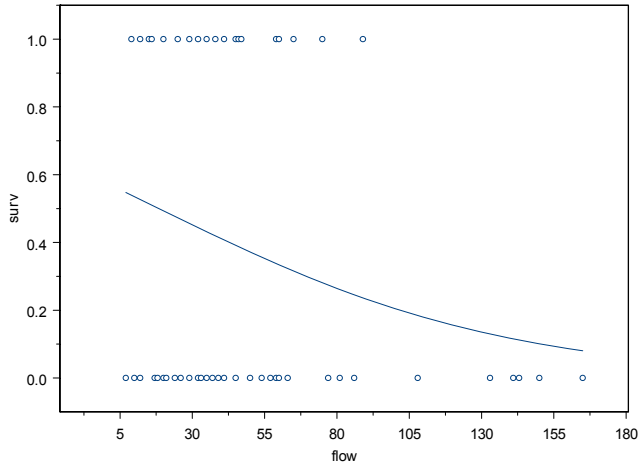
Testovací statistika Cp v žádném případě neklesla pod hodnotu kompletního modelu a tak tedy můžeme zahrnout obě proměnné do modelu. Získané závislosti tedy ukazují, že čím méně vytvořených květních úborů, tím vyšší pravděpodobnost přežití a zároveň s velikostí kořene také klesá pravděpodobnost smrti.

Získaná regresní rovnice by tedy vypadala:
$y = 0.9615 - 0.1064*flow + 6.6003*root$
a pravděpodobnosti přežití pro jedince s danými parametry bychom získali po zadání do výše zmíněné rovnice $p = e^y/(1+e^y)$.
Pokud chceme spočíst pravděpodobnost přežití pro jedince s námi známými parametry, pak můžeme využít další funkce `predict`. Pro jedince s počtem úborů 30 a velikostí kořene 0.52 voláme: `predict(glex.1,list(seeds=30,size=0.52),type="response")`.
Prvním parametrem funkce je jméno modelu, který je použit pro výpočet, dále následuje seznam hodnot pro všechny proměnné v modelu, v našem případě pro seeds a size. Poslední parametr zajišťuje převedení výsledku do intervalu 0 až 1. Výsledkem je tedy 77procentní pravděpodobnost přežití pro zadanou rostlinu.

3. Grafické znázornění

Získané závislosti pro jednotlivé proměnné si můžeme znázornit také v grafech.V S+ je to jednodušší provést pomocí příkazového řádku. Využijeme pro vytvoření grafu minulé funkce `predict`. Pro graf závislosti přežití na počtu vytvořených úborů musíme vytvořit odpovídající model (`glex.flow<-glm(surv~flow, data=GLEX26, binomial`. Dále si zjistíme rozsah hodnot, kterých může nabývat počet úborů (`min(flow)` a `max(flow)`. Pro graf tedy bude rozumné použít na ose x hodnoty od 0 do 180. Vytvoříme pomocnou proměnnou `rada<-seq(0,180, 10)`. Pro hodnoty této pomocné proměnné pak budeme počítat predikované hodnoty a těmi pak proložíme výslednou křivku. Nejdříve si vyneseme hodnoty počtu úborů oproti přežití a poté do grafu přidáme fitovanou křivku. Voláme tedy:

```
>plot(flow,surv)
>lines(rada,predict(glex.flow,list(flow=rada),type="response"))
```



4. Vliv interakce (funkce update)

Avšak může nás také zajímat, jaká je závislost mezi přežitím a oběma parametry navzájem, tedy interakce mezi velikostí a investicí do reprodukce. Přidáme tedy do našeho modelu hodnoty interakcí. Interakce přidáme jednoduše buď vytvořením nového modelu či pomocí funkce update. Změnu modelu zadáme pomocí "tečkové" konvence v S+. Model můžeme obecně vyjádřit jako závislost proměnné na levé straně na proměnných, které jsou uvedeny na straně pravé. Tedy v našem případě `surv ~ flow + root`, což je obecně . ~ ., kde tečka zastupuje veškeré proměnné na dané straně rovnice. Pokud tedy do modelu glex.1 chceme přidat interakce pak použijeme funkci update takto:
`update (glex.1, . ~ . +flow:root)`.
Obdobným způsobem můžeme proměnné i odebírat. Pro náš příklad tedy vytvoříme nový model (glex.full) na základě glex.1 s přidanými interakcemi. Voláme tedy:

```
glex.full<-update(glex.1, .~.+flow:root)
```

Po vyvolání modelu glex.full již S+ vypíše celý model i s interakcemi

```
> glex.full
Call:
glm(formula = surv ~ flow + root + flow:root, family =
     binomial(link = logit), data = GLEX26, na.action =
     na.exclude, control = list(epsilon = 0.0001, maxit =
     50, trace = F))

Coefficients:
 (Intercept)        flow     root  flow:root
   -2.960503 -0.07887645 25.15061 -0.2090799

Degrees of Freedom: 59 Total; 55 Residual
Residual Deviance: 37.12823
```

Nyní stejným způsobem jako pro model glex.1 bez interakcí otestujeme významnost jednotlivých proměnných. pomocí funkce drop1.

```
> drop1(glex.full)
Single term deletions
```

```
Model:
surv ~ flow + root + flow:root
          Df Sum of Sq      RSS       Cp
   <none>               60.63389 69.45336
flow:root  1   5.58286 66.21675 72.83135
```

A opět vidíme, že všechny proměnné použité v našem modelu mají své opodstatnění. Použité interakce v modelu (regresní koeficient -0.20908) můžeme interpretovat tak, že větší rostliny mají nižší pravděpodobnost přežití pokud vytvoří stejný počet květních úborů jako menší jedinci.

Srovnáme-li jednotlivé modely s interakcí a bez ní (obdobně jako např. u mnohonásobné regrese) vidíme, že modely se výrazně liší a tedy přidaná interakce výrazně ovlivnila vysvětlenou variabilitu modelu a měli bychom ji tedy zahrnout. Avšak pozor, zde používáme jako testovací statistiku rozdělení Chí-kvadrát!

```
> anova(model,model2, test="Chi")
Analysis of Deviance Table

Response: surv

                     Terms Resid. Df Resid. Dev        Test Df Deviance      Pr(Chi)
1             seeds + size        56   54.06811
2 seeds + size + seeds:size        55   37.12823 +seeds:size  1 16.93988 0.0000385824
>
```

Získaná (a finální) regresní rovnice tedy vypadá:
*y = -2.960503 - 0.07887645*flow + 25.15061*root - 0.2090799*flow:root*
a odhadnuté pravděpodobnosti přežití pro dané hodnoty vytvořených úborů a velikosti kořene získáme zadáním do $p = e^y/(1+e^y)$. Pokud budeme pravděpodobnosti přežití počítat v ruce, pak hodnoty pro interakce získáme prostým vynásobením hodnot pro kořen a květenství.

Pokud chceme data s interakcí zobrazit v grafu, lze použít postup, kdy např. pro určité hodnoty velikosti kořene vynášíme průběhy prsti přežití vs. počet úborů. Postup je obdobný jako u nakreslení jednoduché závislosti. Liší se pouze v nutnosti zadat hodnoty pro všechny vysvětlované proměnné.
Např tedy pokud budu chtít v grafu mít na ose x počet úborů a jednotlivé čáry budou vybrané průměry kořenu zjistím rozsahy hodnot kterých může nabývat počet úborů a velikost kořene. Pro počet úborů vytvořím pomocnou proměnnou funkcí seq (rada). Pro hodnoty této pomocné proměnné a pak budeme počítat predikované hodnoty a těmi pak proložíme výslednou křivku. Nejdříve si vyneseme hodnoty počtu úborů oproti přežití a poté do grafu přidáme fitovanou křivku. Voláme tedy:

```
>plot(flow,surv)
>lines(rada,predict(glex.flow,list(flow=rada,size=rep(0.1,length
(rada)),type="response"))
```

Předchozí příkaz tedy kreslí průběh pravděpodobnosti přežití pro celý rozsah počtu úborů, při konstantní velikosti kořene 0.1.
Stejný příkaz mohu opakovat pro různé velikosti kořene a tím získat představu o charakteru interakce a vlivu jednotlivých charakteristik pro přežití.

Použitá data z příkladu glex26 pro program GLIM:

surv	flow	root
0	165	1.57
0	41	0.2
0	33	0.2
0	141	0.91
0	150	2.36
0	21	0.2
0	35	0.2
0	57	0.66
0	108	1.57
1	38	0.66
1	41	0.44
0	37	0.2
0	86	0.66
0	32	0.25
1	45	0.55
0	26	0.2
0	143	1.57
0	39	0.25
1	20	0.2
0	50	0.44
0	29	0.1
1	75	1.18
0	60	0.2
1	47	0.2
0	10	0.05
0	81	0.55
1	16	0.29
0	133	0.88
0	54	0.25
0	57	0.04
0	17	0.2
1	20	0.55
0	77	0.15
0	20	0.1
0	41	0.2
1	32	0.30
0	63	0.66
1	9	0.2
0	7	0.02
1	46	0.55
1	15	0.25
0	24	0.15
0	12	0.01
0	18	0.15
1	35	0.25
1	29	0.39
0	21	0.25
1	12	0.25
0	45	0.2
0	21	0.25
0	59	0.44
1	35	0.98
1	60	0.66
1	35	0.44
1	59	0.66
1	16	0.15
1	65	1.57
1	25	0.44
1	89	1.9