Teildatensatz flow cytometer Daten als Beispiel für Sören

FS

2023-09-26

Ich habe aus deinen Daten einen vereinfachten Teildatensatz rausgenommen, und versucht, die zellbiologischen Details zu verblinden, so dass keine Information zu den konkreten Zelltypen, Medien, Markern gegeben wird. Damit kein Risiko besteht, dass wir (falls wir soweit kommen!) in einer stat. Publikation irgendwelche konkreten Ergebnisse/Themen eurer Forschung vorwegnehmen. Wenn Dir/Euch das noch nicht verblindet genug ist, kann ich das auch noch weiter reduzieren.

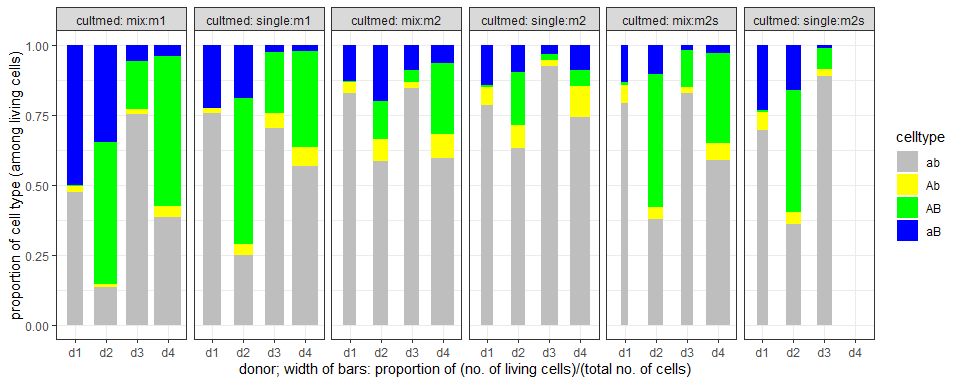
Wir würden im Zusammenhang mit diesen Teildatensatz gerne folgendes erklären:

* den grundlegenden Versuchsaufbau:
  + 4 Donoren von denen die Zellen kommen (donor: d1,d2,d3,d4)
  + und die an 4 getrennten Zeitpunkten prozessiert werden (?)
  + bei denen jeweils die Zellen auf 3x2 “Behandlungs”-gruppen aufgeteilt werden,
  + 3 Medien (med: ‘m2’, ‘m3’, ‘m3s’) ACHTUNG: Streng genommen kommen je Donor und cult die beiden Beobachtungen zu med m3 und med m3s aus dem gleichen Kulturgefäss (Probenahme von 2 verschiedene Fraktionen des gleichen Mediums), hier ist als eigentlcih noch eine weitere hierarchische struktur emnthalten, die aber bei m2 nicht vorkommt. Gegebenenfalls: med: m3s aus Analyse rauslassen.
  + 2 Kulturvarianten (cult: ‘single’, ‘mix’)
* die ‘Datenentstehung’:
  + je Donor und ‘Behandlungs’-gruppe sollen
  + 10000 Zellen im Flow cytometer klassifiziert werden:
  + vereinzelt reicht das Material nicht für 10000 Zellen
  + von denen ist ein Teil nicht klassifizierbar ist, weil geklumpt,
  + ein Teil der Zellen ist tot,
  + und der Rest (‘living’): wird anhand von 2 Markern mit je 2 Zuständen (aA, bB) in 4 Kategorien eingeteilt: celltype: ab, Ab, AB, aB
  + von Interesse sind Vergleiche der Häufigkeiten dieser 4 Kategorien zwischen den 6 Behandlungsgruppen

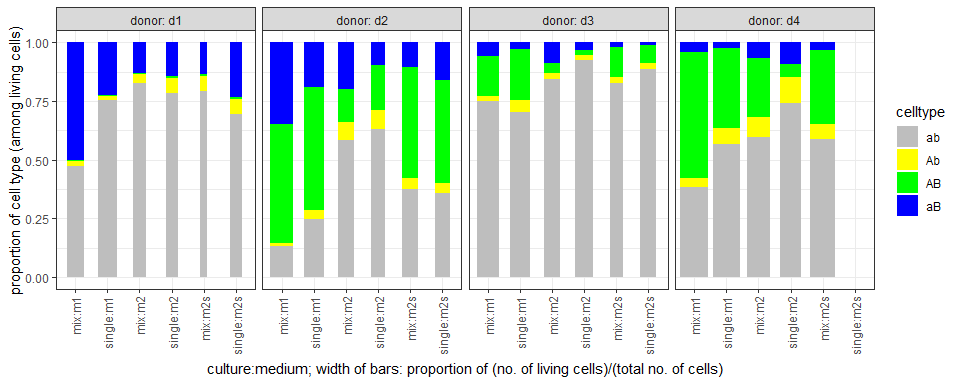
Ist das so für euch ok, wenn diese Beschreibung mit dem Datensatz verwendet würde?

load("dflowcytr.rda")

dflowcytr %>% ggplot(aes(y=count/living, x=donor))+  
 #geom\_col(aes(width=1), fill="white", color="black")+  
 geom\_col(aes(fill=celltype, group=donor:cult:med, width=living/nocells), stat="identity", position="stack") +  
 facet\_grid(~cultmed, labeller=label\_both) +   
 scale\_fill\_manual(values=c("grey", "yellow", "green", "blue"))+  
 theme\_bw() +  
 ylab("proportion of cell type (among living cells)")+  
 xlab("donor; width of bars: proportion of (no. of living cells)/(total no. of cells)")



dflowcytr %>% ggplot(aes(y=count/living, x=cultmed))+  
 #geom\_col(aes(width=1), fill="white", color="black")+  
 geom\_col(aes(fill=celltype, group=donor:cult:med, width=living/nocells), stat="identity", position="stack") +  
 facet\_grid(~donor, labeller=label\_both) +   
 scale\_fill\_manual(values=c("grey", "yellow", "green", "blue"))+  
 theme\_bw() + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))+  
 ylab("proportion of cell type (among living cells)")+  
 xlab("culture:medium; width of bars: proportion of (no. of living cells)/(total no. of cells)")



dflowcytrw <- pivot\_wider(dflowcytr, names\_from=celltype, values\_from=count)  
str(as.data.frame(dflowcytrw))

## 'data.frame': 23 obs. of 13 variables:  
## $ ID : int 2 1 5 3 6 4 11 8 19 14 ...  
## $ donor : Factor w/ 4 levels "d1","d2","d3",..: 1 1 1 1 1 1 2 2 2 2 ...  
## $ med : Factor w/ 3 levels "m1","m2","m2s": 1 1 2 2 3 3 1 1 2 2 ...  
## $ cult : Factor w/ 4 levels "mix","co pol",..: 1 4 1 4 1 4 1 4 1 4 ...  
## $ nocells : num 10000 10000 10000 10000 6782 ...  
## $ singlets: num 6785 7684 5590 4840 3081 ...  
## $ dead : num 1643 1935 1445 1069 1615 ...  
## $ living : num 5142 5749 4145 3771 1466 ...  
## $ cultmed : Factor w/ 6 levels "mix:m1","single:m1",..: 1 2 3 4 5 6 1 2 3 4 ...  
## $ ab : num 2434 4343 3431 2961 1162 ...  
## $ Ab : num 107 101 156 242 94 155 86 242 383 366 ...  
## $ AB : num 20 3 13 28 14 ...  
## $ aB : num 2581 1302 545 540 196 ...

write.xlsx(dflowcytrw, file="dflowcytrw.xlsx")

# Analysis by 4 separate quasibinomial models each celltype vs the rest

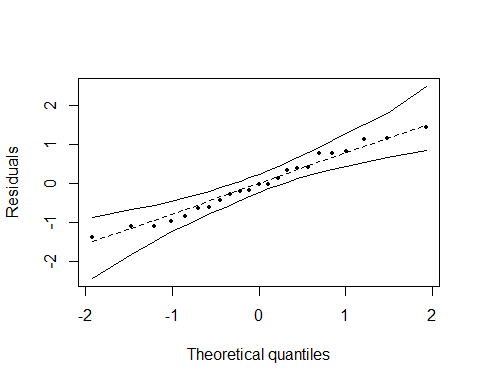
fab <- glm(cbind(ab,living-ab) ~ donor + cultmed, data=dflowcytrw, family=quasibinomial())  
fAb <- glm(cbind(Ab,living-Ab) ~ donor + cultmed, data=dflowcytrw, family=quasibinomial())  
fAB <- glm(cbind(AB,living-AB) ~ donor + cultmed, data=dflowcytrw, family=quasibinomial())  
faB <- glm(cbind(aB,living-aB) ~ donor + cultmed, data=dflowcytrw, family=quasibinomial())

## Check adequacy of quasibinomial assumption by category using hnp()

* In diesem Teildatensatz kein wiederspruch zur annahme quasibinomial je Endpunkt

library(hnp)  
  
hnp(fab, halfnormal = FALSE, paint.out=TRUE, sim=2000, pch=16)

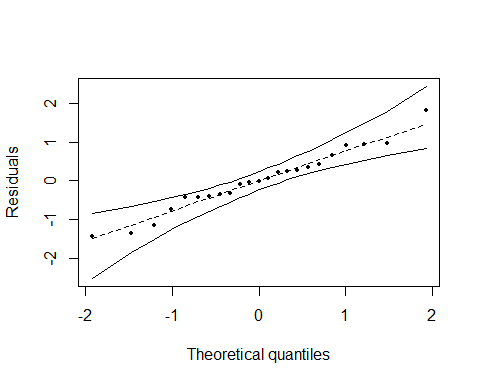
## Quasi-binomial model



## Total points: 23   
## Points out of envelope: 0 ( 0 %)

hnp(fAb, halfnormal = FALSE, paint.out=TRUE, sim=2000, pch=16)

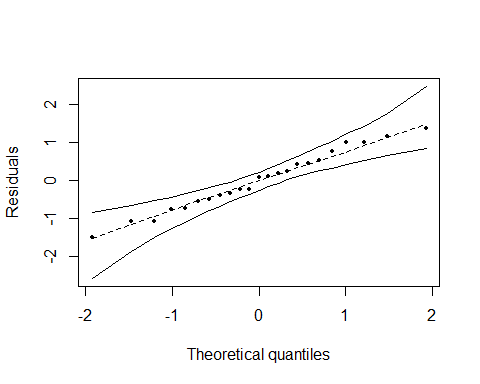
## Quasi-binomial model



## Total points: 23   
## Points out of envelope: 0 ( 0 %)

hnp(fAB, halfnormal = FALSE, paint.out=TRUE, sim=2000, pch=16)

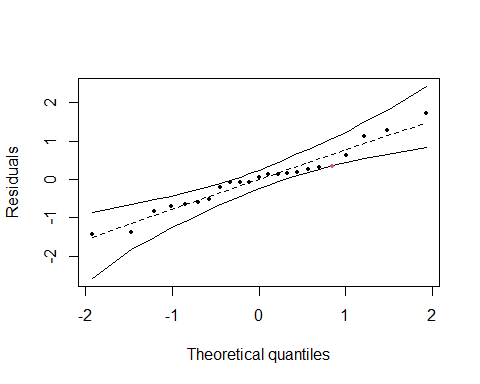
## Quasi-binomial model



## Total points: 23   
## Points out of envelope: 0 ( 0 %)

hnp(faB, halfnormal = FALSE, paint.out=TRUE, sim=2000, pch=16)

## Quasi-binomial model



## Total points: 23   
## Points out of envelope: 1 ( 4.35 %)

## Observed disperions by category

* Unterschiedliche Dispersionsschätzer, fraglich ob das bei der Fallzahl auch zufall sein kann.

summary(fab)$dispersion

## [1] 148.2511

summary(fAb)$dispersion

## [1] 36.25262

summary(fAB)$dispersion

## [1] 130.7617

summary(faB)$dispersion

## [1] 159.508

## Analysis of deviance

anova(fab, test="F")

## Analysis of Deviance Table  
##   
## Model: quasibinomial, link: logit  
##   
## Response: cbind(ab, living - ab)  
##   
## Terms added sequentially (first to last)  
##   
##   
## Df Deviance Resid. Df Resid. Dev F Pr(>F)   
## NULL 22 25648.6   
## donor 3 15748.9 19 9899.7 35.410 8.643e-07 \*\*\*  
## cultmed 5 7833.1 14 2066.7 10.567 0.0002326 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(fAb, test="F")

## Analysis of Deviance Table  
##   
## Model: quasibinomial, link: logit  
##   
## Response: cbind(Ab, living - Ab)  
##   
## Terms added sequentially (first to last)  
##   
##   
## Df Deviance Resid. Df Resid. Dev F Pr(>F)   
## NULL 22 1994.35   
## donor 3 793.15 19 1201.19 7.2928 0.00351 \*\*  
## cultmed 5 707.96 14 493.23 3.9057 0.01996 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(fAB, test="F")

## Analysis of Deviance Table  
##   
## Model: quasibinomial, link: logit  
##   
## Response: cbind(AB, living - AB)  
##   
## Terms added sequentially (first to last)  
##   
##   
## Df Deviance Resid. Df Resid. Dev F Pr(>F)   
## NULL 22 29343.0   
## donor 3 19446.4 19 9896.6 49.572 1.055e-07 \*\*\*  
## cultmed 5 8172.8 14 1723.8 12.500 9.422e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(faB, test="F")

## Analysis of Deviance Table  
##   
## Model: quasibinomial, link: logit  
##   
## Response: cbind(aB, living - aB)  
##   
## Terms added sequentially (first to last)  
##   
##   
## Df Deviance Resid. Df Resid. Dev F Pr(>F)   
## NULL 22 14244.2   
## donor 3 9417.5 19 4826.7 19.6803 2.724e-05 \*\*\*  
## cultmed 5 2766.8 14 2059.9 3.4692 0.03009 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## emmeans

library(emmeans)  
  
emmfab<-emmeans(fab, specs="cultmed", contr="pairwise", adj="mvt")  
emmfAb<-emmeans(fAb, specs="cultmed", contr="pairwise", adj="mvt")  
emmfAB<-emmeans(fAB, specs="cultmed", contr="pairwise", adj="mvt")  
emmfaB<-emmeans(faB, specs="cultmed", contr="pairwise", adj="mvt")  
  
emmfab$contrasts

## contrast estimate SE df z.ratio p.value  
## mix:m1 - single:m1 -0.6129 0.226 Inf -2.711 0.0714  
## mix:m1 - mix:m2 -1.2707 0.250 Inf -5.077 <.0001  
## mix:m1 - single:m2 -1.6785 0.269 Inf -6.247 <.0001  
## mix:m1 - mix:m2s -0.9264 0.255 Inf -3.627 0.0038  
## mix:m1 - single:m2s -0.9474 0.309 Inf -3.062 0.0259  
## single:m1 - mix:m2 -0.6578 0.251 Inf -2.619 0.0907  
## single:m1 - single:m2 -1.0656 0.269 Inf -3.954 0.0010  
## single:m1 - mix:m2s -0.3136 0.256 Inf -1.224 0.8221  
## single:m1 - single:m2s -0.3345 0.312 Inf -1.072 0.8903  
## mix:m2 - single:m2 -0.4078 0.288 Inf -1.416 0.7128  
## mix:m2 - mix:m2s 0.3443 0.275 Inf 1.251 0.8081  
## mix:m2 - single:m2s 0.3233 0.329 Inf 0.983 0.9220  
## single:m2 - mix:m2s 0.7520 0.292 Inf 2.579 0.1000  
## single:m2 - single:m2s 0.7311 0.342 Inf 2.139 0.2628  
## mix:m2s - single:m2s -0.0209 0.334 Inf -0.063 1.0000  
##   
## Results are averaged over the levels of: donor   
## Results are given on the log odds ratio (not the response) scale.   
## P value adjustment: mvt method for 15 tests

emmfAb$contrasts

## contrast estimate SE df z.ratio p.value  
## mix:m1 - single:m1 -0.7141 0.295 Inf -2.418 0.1457  
## mix:m1 - mix:m2 -0.9428 0.295 Inf -3.192 0.0171  
## mix:m1 - single:m2 -1.1581 0.290 Inf -3.997 0.0009  
## mix:m1 - mix:m2s -0.6597 0.317 Inf -2.078 0.2925  
## mix:m1 - single:m2s -0.8138 0.377 Inf -2.160 0.2509  
## single:m1 - mix:m2 -0.2287 0.248 Inf -0.923 0.9389  
## single:m1 - single:m2 -0.4440 0.241 Inf -1.842 0.4315  
## single:m1 - mix:m2s 0.0544 0.274 Inf 0.199 1.0000  
## single:m1 - single:m2s -0.0997 0.341 Inf -0.292 0.9997  
## mix:m2 - single:m2 -0.2152 0.241 Inf -0.893 0.9468  
## mix:m2 - mix:m2s 0.2832 0.274 Inf 1.035 0.9038  
## mix:m2 - single:m2s 0.1290 0.342 Inf 0.377 0.9990  
## single:m2 - mix:m2s 0.4984 0.268 Inf 1.860 0.4198  
## single:m2 - single:m2s 0.3442 0.336 Inf 1.025 0.9073  
## mix:m2s - single:m2s -0.1541 0.364 Inf -0.424 0.9982  
##   
## Results are averaged over the levels of: donor   
## Results are given on the log odds ratio (not the response) scale.   
## P value adjustment: mvt method for 15 tests

emmfAB$contrasts

## contrast estimate SE df z.ratio p.value  
## mix:m1 - single:m1 0.274 0.235 Inf 1.166 0.8472  
## mix:m1 - mix:m2 1.410 0.291 Inf 4.842 <.0001  
## mix:m1 - single:m2 2.127 0.363 Inf 5.864 <.0001  
## mix:m1 - mix:m2s 0.510 0.254 Inf 2.008 0.3270  
## mix:m1 - single:m2s 0.639 0.334 Inf 1.912 0.3831  
## single:m1 - mix:m2 1.135 0.297 Inf 3.820 0.0017  
## single:m1 - single:m2 1.852 0.368 Inf 5.038 <.0001  
## single:m1 - mix:m2s 0.236 0.261 Inf 0.904 0.9431  
## single:m1 - single:m2s 0.365 0.342 Inf 1.068 0.8894  
## mix:m2 - single:m2 0.717 0.405 Inf 1.772 0.4726  
## mix:m2 - mix:m2s -0.899 0.311 Inf -2.890 0.0422  
## mix:m2 - single:m2s -0.770 0.382 Inf -2.014 0.3238  
## single:m2 - mix:m2s -1.616 0.379 Inf -4.263 0.0003  
## single:m2 - single:m2s -1.487 0.438 Inf -3.392 0.0087  
## mix:m2s - single:m2s 0.129 0.356 Inf 0.363 0.9991  
##   
## Results are averaged over the levels of: donor   
## Results are given on the log odds ratio (not the response) scale.   
## P value adjustment: mvt method for 15 tests

emmfaB$contrasts

## contrast estimate SE df z.ratio p.value  
## mix:m1 - single:m1 0.8910 0.327 Inf 2.726 0.0671  
## mix:m1 - mix:m2 0.7612 0.344 Inf 2.214 0.2236  
## mix:m1 - single:m2 1.0652 0.379 Inf 2.809 0.0536  
## mix:m1 - mix:m2s 1.3950 0.459 Inf 3.041 0.0273  
## mix:m1 - single:m2s 0.9937 0.426 Inf 2.332 0.1749  
## single:m1 - mix:m2 -0.1299 0.382 Inf -0.340 0.9994  
## single:m1 - single:m2 0.1742 0.414 Inf 0.421 0.9982  
## single:m1 - mix:m2s 0.5040 0.490 Inf 1.028 0.9050  
## single:m1 - single:m2s 0.1026 0.458 Inf 0.224 0.9999  
## mix:m2 - single:m2 0.3040 0.428 Inf 0.710 0.9799  
## mix:m2 - mix:m2s 0.6339 0.501 Inf 1.265 0.7978  
## mix:m2 - single:m2s 0.2325 0.470 Inf 0.494 0.9962  
## single:m2 - mix:m2s 0.3298 0.526 Inf 0.627 0.9886  
## single:m2 - single:m2s -0.0716 0.496 Inf -0.144 1.0000  
## mix:m2s - single:m2s -0.4014 0.560 Inf -0.716 0.9791  
##   
## Results are averaged over the levels of: donor   
## Results are given on the log odds ratio (not the response) scale.   
## P value adjustment: mvt method for 15 tests

# Baseline logit model with multinomial assumption

library("VGAM")  
  
fit <- vglm(cbind(ab, Ab, AB, aB) ~ donor + cultmed, data=dflowcytrw, family=multinomial)  
summary(fit)  
class(fit)  
  
anova(fit)  
confint(fit)  
confint(fit, method="profile")  
  
# emmeans(fit, specs="cultmed")