

```

# Analysis of Admissions data
df <- read.csv("~/teach/data/berkelyAdmissions",skip=1,header=TRUE)
berkAdmits <- array(data.matrix(df[[2:5]]),dim=c(6,2,2),
  dimnames=list(Program=LETTERS[1:6],Admission=c("Y","N"),
    Sex=c("M","F")))
progAdmits <- apply(berkAdmits,1:2,sum)
print(progAdmits)
percents(progAdmits,1)
chisq.test(progAdmits)
pairwise.prop.test(progAdmits)

# Analysis of Job Satisfaction Data
df <- read.csv("~/teach/data/agrestiIncome",skip=1,header=TRUE,sep="\t")
incomeSat <- data.matrix(df[,-1])
dimnames(incomeSat)[[1]] <- as.character(df[,1])
percents(incomeSat,1)
chisq.test(incomeSat)
dimnames(incomeSat)[[2]] <- c("--","-","+","++")
names(dimnames(incomeSat)) <- c("income","satisfaction")

counts <- as.vector(incomeSat)
income <- factor(row(incomeSat), labels=dimnames(incomeSat)[[1]])
jobSat <- factor(col(incomeSat), labels=dimnames(incomeSat)[[2]])
indepFit <- glm(counts ~ income + jobSat , family=poisson)

# Residual examination
stdResidTab <- incomeSat
stdResidTab[] <- rstandard(indepFit)
print(round(stdResidTab,2))

# Ordinal model fits
satFit <- glm(counts ~ income * jobSat , family=poisson)
ord2Fit <- glm(counts ~ income + jobSat +
  as.integer(income):as.integer(jobSat), family=poisson)
ord1Fit <- glm(counts ~ income + jobSat +
  income:as.integer(jobSat), family=poisson)

anova(indepFit,ord2Fit,ord1Fit, satFit,test="Chi")

# Compare fitted and actual row percentages
fitMat <- matrix(fitted(ord2Fit), nrow=4,dimnames=dimnames(incomeSat))
actProp <- round(percents(incomeSat,1,pretty=FALSE),2)
fitProp <- round(percents(fitMat,1,pretty=FALSE),2)
print(actProp)
print(fitProp)

# Purum Marriage example - structural zero's, quasi-independence.
df <- read.csv("~/teach/data/purumMarriage",skip=1)
charData <- matrix(scan("~/teach/data/purumMarriage",
  what="",sep=" ",skip=1), byrow=TRUE,nrow=6)
freqMat <- matrix(as.integer(charData[-1,-1]),nrow=5,
  dimnames=list(wife=charData[-1,1],husband=charData[-1,1]))

wife <- factor(row(freqMat),labels=dimnames(freqMat)[["wife"]])
husband <- factor(col(freqMat),labels=dimnames(freqMat)[["husband"]])
counts <- as.integer(freqMat)
llfit <- glm( counts ~ wife + husband, family=poisson)
satFit <- glm( counts ~ factor(seq(along=counts)), family=poisson)
anova(llfit,satFit,test="Chi")

fMat <- freqMat
fMat[is.na(fMat)] <- 0
pctMat <- percents(fMat,2)
pctMat[is.na(freqMat)] <- "-"
print(pctMat)

```

```

residTab <- freqMat
residTab[] <- NA
residTab[!is.na(freqMat)] <- rstandard(llfit)
print(round(residTab,2))

# Analysis of Marijuana smokers
freqMat <- matrix(scan(),byrow=TRUE,ncol=2)
4 9
3 1

dimnames(freqMat) <- list(smoker=c("+","-"),control=c("+","-"))

mcnemar.test(freqMat)
binom.test(freqMat[2,1],freqMat[2,1]+freqMat[1,2])
prop.test(freqMat[2,1],freqMat[2,1]+freqMat[1,2],p=.5)

# Analysis of Social Class data - test for marginal homogeneity
freqMat <- matrix(scan(), nrow=3, byrow=TRUE)
588 395 159 349 714 447 114 320 411

dimnames(freqMat) <- list(father=c("Upper","Middle","Lower"),
                          son=c("Upper","Middle","Lower"))
print(freqMat)

print(percents(apply(freqMat,1,sum)))
print(percents(apply(freqMat,2,sum)))

# the concord library provides the stuart maxwell test.
library(concord)
stuart.maxwell.mh(freqMat)

```