1. Incubation temperature can affect the sex of turtles. An experiment was conducted with three independent replicates for each temperature and the number of male and female turtles born was recorded. The data can be found in the turtle dataset in the faraway package.

Check for evidence of overdispersion in a binomial model for the sex of the turtle.

What problems can arise if you ignore overdispersion?

**Solution:** We fit a binomial regression and estimate the dispersion $\phi$.

```r
> library(faraway)
> data(turtle)
> bmod <- glm(cbind(male, female) ~ temp, data=turtle, family=binomial)
> summary(bmod)

Call:
glm(formula = cbind(male, female) ~ temp, family = binomial, 
data = turtle)

Deviance Residuals:
Min 1Q Median 3Q Max
-2.0721 -1.0292 -0.2714 0.8087 2.5550

Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept) -61.3183 12.0224 -5.100 3.39e-07 ***
temp 2.2110 0.4309 5.132 2.87e-07 ***

---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 74.508 on 14 degrees of freedom
Residual deviance: 24.942 on 13 degrees of freedom
AIC: 53.836

Number of Fisher Scoring iterations: 5

> (phihat <- sum( residuals(bmod, type="pearson")^2 )/13)
[1] 2.018641

> pchisq(24.942, 13, lower.tail=F)
[1] 0.02349208

$\hat{\phi}$ is a little bit larger than 1, indicating possible overdispersion. Also, the chi-squared test for model adequacy using the deviance gives a significant result, indicating that there is something left unexplained. This could be overdispersion, or a problem with the model. In this case it looks like a problem with the model, as if we add temp^2 to the model we can improve the fit, and the test for model adequacy is no longer significant at the 5% level (just).

```r
> bmod2 <- glm(cbind(male, female) ~ temp + I(temp^2), data=turtle, family=binomial)
> summary(bmod2)
```
Call:
\[
glm(formula = \text{cbind(male, female)} \sim \text{temp + I(temp}^2\text{)}, family = \text{binomial, data = turtle})
\]

Deviance Residuals:

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>1Q</th>
<th>Median</th>
<th>3Q</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-1.6703</td>
<td>-0.8875</td>
<td>-0.4194</td>
<td>0.9481</td>
<td>2.2198</td>
</tr>
</tbody>
</table>

Coefficients:

|                  | Estimate | Std. Error | z value | Pr(>|z|) |
|------------------|----------|------------|---------|----------|
| \( \text{Intercept} \) | -677.5950 | 268.7984   | -2.521  | 0.0117 * |
| \( \text{temp} \)       | 45.9173  | 18.9169    | 2.427   | 0.0152 * |
| \( \text{I(temp}^2\text{)} \) | -0.7745  | 0.3327     | -2.328  | 0.0199 * |

---

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 74.508 on 14 degrees of freedom
Residual deviance: 20.256 on 12 degrees of freedom
AIC: 51.15

Number of Fisher Scoring iterations: 4

\[
\phihat <- \text{sum(residuals(bmod2, type="pearson")}^2/12
\]

\[
\text{pchisq(20.256, 12, lower.tail=F)}
\]

The estimate of \( \phi \) has been reduced so overdispersion seems less likely. We can check the fit and residuals for this model

\[
> \text{with(turtle, plot(temp, male/(male+female)))
}\]
\[
> t <- \text{seq(27, 30, .1)
}\]
\[
> \text{lines(t, ilogit(-677.595 + 45.9173*t - 0.7745*t}^2\text{), col="red")
}\]
> par(mfrow=c(2,2))
> plot(predict(bmod), residuals(bmod))
> halfnorm(residuals(bmod), ylab="deviance resid")
> halfnorm(rstudent(bmod), ylab="jackknife resid")
> halfnorm(cooks.distance(bmod), ylab="cooks distance")
The fit still isn’t brilliant for the lower temperatures, but there is no evidence of outliers. Note however that the new model is not without problems of its own. In particular the relationship between temperature and the chance of being male is no longer monotonic (though it mostly is in the range of temperatures considered). This makes the model harder to interpret.

In general, if we do not account for overdispersion, then our tests for variable significance will be too sensitive. That is, they may indicate a variable is significant when it really isn’t. Similarly, if we do not account for overdispersion, confidence intervals for parameter estimates will be too small. See Question 3 for a good example of this.

2. In a binomial model we assume that any given observation is from a bin($m, p$) distribution, for some $m$ and $p$. That is, we count the number of successes from $m$ i.i.d. bernoulli($p$) trials. There are two main ways that overdispersion can arise: the trials are not identically distributed, or the trials are not independent.

A common way in which we can have heterogeneous trials is via clustering. Suppose that we have $m$ trials split into $h$ clusters of size $k = m/h$, and that the probability of success for a trial in the $i$-th cluster is $p_i$. Now suppose that $p_i$ is random, with $E(p_i) = p$ and $\text{Var}(p_i) = \tau^2 p(1 - p)$. Let the number of successes from cluster $i$ be $Z_i$ and let the total number of successes be $Y = Z_1 + \cdots + Z_h$.

Show that

(a) $E(Y) = mp$

(b) $\text{Var}(Y) = (1 + (k - 1)\tau^2)mp(1 - p)$

Hint: $\text{Var}(Y) = E(\text{Var}(Y|X)) + \text{Var}(E(Y|X))$. 

4
Thus \( Y \) is overdispersed, relative to a binomial.

**Solution:**

\[
\begin{align*}
EY &= \sum_{i=1}^{h} E(Z_i) = \sum_{i=1}^{h} E(Z_i|p_i) \\
&= h \sum_{i=1}^{h} Ekp_i = \sum_{i=1}^{h} kp = mp
\end{align*}
\]

By independence \( \text{Var} Y = \sum_i \text{Var} Z_i \), and

\[
\begin{align*}
\text{Var} Z_1 &= E \text{Var} (Z_1|p_1) + \text{Var} E(Z_1|p_1) \\
&= Ekp_1(1 - p_1) + \text{Var} kp_1 \\
&= kp - k(\tau^2 p(1 - p) + p^2) + k^2 \tau^2 p(1 - p) \\
&= k(1 - p)(1 + (k - 1)\tau^2).
\end{align*}
\]

Multiplying by \( h \) gives the result.

3. In Question 4 in last week’s lab we fitted a rate model (a type of Poisson regression) to data on the effect of gamma radiation on chromosomal abnormalities.

Show that these data are overdispersed compared to a Poisson distribution. Next test for an interaction between \textit{doserate} and \textit{doseamt}, firstly without allowing for overdispersion (fixing this dispersion \( \phi = 1 \)), and secondly allowing for overdispersion. Do you get different answers?

**Solution:** The fitted model from last week:

```r
> model <- glm(ca ~ offset(log(cells)) + doserate*doseamt, family=poisson, data=dicentric)
> summary(model)
```

```
Call: glm(formula = ca ~ offset(log(cells)) + doserate * doseamt, family = poisson, data = dicentric)

Deviance Residuals:
Min 1Q Median 3Q Max
-5.7308 -2.2842 -0.6264 3.3487 5.8272

Coefficients:

|                | Estimate | Std. Error | z value | Pr(>|z|) |
|----------------|----------|------------|---------|----------|
| (Intercept)    | -3.29994 | 0.06160    | -53.567 | < 2e-16  *** |
| doserate       | 0.06401  | 0.02922    | 2.191   | 0.028476 * |
| doseamt        | 0.61224  | 0.01707    | 35.862  | < 2e-16  *** |
| doserate:doseamt | 0.02715  | 0.00765    | 3.549   | 0.000387 *** |

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 4753.00 on 26 degrees of freedom
Residual deviance: 270.26 on 23 degrees of freedom
AIC: 453.67

Number of Fisher Scoring iterations: 4
```

The deviance of our fitted model is very high because of overdispersion. We estimate the dispersion parameter and get something much larger than 1. We then use our estimate to scale the variance of our estimates, and repeat our significance test for the interaction term (using an F test). We see that the interaction no longer appears significant.

Note that the command `drop1` does \textit{not} work for overdispersed models.

```r
> (phi <- sum(residuals(model, type="pearson")^2)/23)
```

5
> model1 <- glm(ca ~ offset(log(cells)) + doserate*doseamt, family=quasipoisson, data=dicentric)
> summary(model1) # same as summary(model, dispersion=phi)

Call:
glm(formula = ca ~ offset(log(cells)) + doserate * doseamt, family = quasipoisson, 
    data = dicentric)

Deviance Residuals:

Min 1Q Median 3Q Max
-5.7308 -2.2842 -0.6264 3.3487 5.8272

Coefficients:

              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3.29994   0.22188  -14.873 2.73e-13 ***
doserate     0.06401   0.10524    0.608   0.549

doseamt     0.61224   0.06149    9.957 8.29e-10 ***
doserate:doseamt  0.02715   0.02755    0.985   0.335

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 12.97228)

Null deviance: 4753.00 on 26 degrees of freedom
Residual deviance: 270.26 on 23 degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 4

> anova(model1, test="F") # same as anova(model, dispersion=phi, test="F")

Analysis of Deviance Table

Model: quasipoisson, link: log

Response: ca

Terms added sequentially (first to last)

      Df Deviance Resid. Df Resid. Dev F Pr(>F)
NULL     26 4753.00
doserate  1    231.3  25     4521.7 17.8319 0.0003232 ***
doseamt   1    4238.7  24    326.7535 4.33e-15 ***
doserate:doseamt  1     12.7  23    270.3098 0.9781 0.3329689

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

We refit omitting the interaction, to get our final model.

> model2 <- glm(ca ~ offset(log(cells)) + doserate + doseamt, family=poisson, data=dicentric)

> (phi2 <- sum(residuals(model2, type="pearson")^2)/24)

[1] 12.72343

> summary(model2, dispersion=phi2)

Call:
glm(formula = ca ~ offset(log(cells)) + doserate + doseamt, family = poisson, 
    data = dicentric)

Deviance Residuals:

Min 1Q Median 3Q Max

0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

We refit omitting the interaction, to get our final model.
Coefficients:

| Estimate | Std. Error | z value | Pr(>|z|) |
|----------|------------|---------|----------|
| (Intercept) -3.46115 | 0.15430 | -22.432 | <2e-16 *** |
| doserate 0.15501 | 0.04881 | 3.176 | 0.0015 ** |
| doseamt 0.66230 | 0.03456 | 19.163 | <2e-16 *** |

---

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 12.72343)

Null deviance: 4753.00 on 26 degrees of freedom
Residual deviance: 282.95 on 24 degrees of freedom
AIC: 464.35

Number of Fisher Scoring iterations: 4

In the `multinom` function from the `nnet` package, the response should be a factor with $K$ levels or a matrix with $K$ columns, which will be interpreted as counts for each of $K$ classes. The first case is a short hand for responses of the form multinomial(1, $\mathbf{p}$).

4 The `hsb` data from the `faraway` package was collected as a subset of the “High School and Beyond” study, conducted by the National Education Longitudinal Studies program of the U.K. National Center for Education Statistics. The variables are gender, race, socioeconomic status, school type, chosen high school program type, scores on reading, writing, math, science, and social studies. We want to determine which factors are related to the choice of the type of program—academic, vocational, or general—that the students pursue in high school. The response is multinomial with three levels.

(a) Fit a trinomial response model with the other relevant variables as predictors (untransformed).

Solution:

```r
> library(faraway)
> data(hsb)
> library(nnet)
> mmod <- multinom(prog ~ gender + race + ses + schtyp + read + write + math +
+                 science + socst, hsb, trace = FALSE)
> summary(mmod)

Call:
multinom(formula = prog ~ gender + race + ses + schtyp + read +
        write + math + science + socst, data = hsb, trace = FALSE)

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>GenderMale</th>
<th>RaceAsian</th>
<th>RaceHispanic</th>
<th>RaceWhite</th>
<th>SESLow</th>
<th>General</th>
<th>Vocation</th>
<th>SESMiddle</th>
<th>SCHTypPublic</th>
<th>Read</th>
<th>Write</th>
<th>Math</th>
<th>Science</th>
<th>Socst</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Intercept)</td>
<td>3.631901</td>
<td>-0.09264717</td>
<td>1.352739</td>
<td>-0.6322019</td>
<td>0.2965156</td>
<td>1.09864111</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vocation</td>
<td>7.481381</td>
<td>-0.32104341</td>
<td>-0.700070</td>
<td>-0.1993566</td>
<td>0.3358881</td>
<td>0.04747323</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SESMiddle</td>
<td>0.7029621</td>
<td>0.5845405</td>
<td>-0.04418353</td>
<td>-0.03627381</td>
<td>-0.1092888</td>
<td>0.10193746</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCHTypPublic</td>
<td>2.0553336</td>
<td>-0.03481202</td>
<td>-0.03166001</td>
<td>-0.1139877</td>
<td>0.05229938</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Socst</td>
<td>-0.01976995</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Std. Errors:

<table>
<thead>
<tr>
<th></th>
<th>GenderMale</th>
<th>RaceAsian</th>
<th>RaceHispanic</th>
<th>RaceWhite</th>
<th>SESLow</th>
<th>General</th>
<th>Vocation</th>
<th>SESMiddle</th>
<th>SCHTypPublic</th>
<th>Read</th>
<th>Write</th>
<th>Math</th>
<th>Science</th>
<th>Socst</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Intercept)</td>
<td>1.823452</td>
<td>0.4648778</td>
<td>1.058754</td>
<td>0.8935504</td>
<td>0.7354829</td>
<td>0.6066763</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vocation</td>
<td>2.104698</td>
<td>0.5021132</td>
<td>1.470176</td>
<td>0.8393676</td>
<td>0.7480573</td>
<td>0.7045772</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SESMiddle</td>
<td>0.5045938</td>
<td>0.5642925</td>
<td>0.03103707</td>
<td>0.03381324</td>
<td>0.03522441</td>
<td>0.03274038</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

7
(b) Use either backward elimination with $\chi^2$ tests (using the `anova` command), or the AIC (using `step`), to produce a parsimonious model. Give an interpretation of the resulting model.

**Solution:** I just used the AIC, as provided by `step`.

```r
> mmod2 <- step(mmod, scope='.', direction='backward', trace = FALSE)
```

```
trying - gender
trying - race
trying - ses
trying - schtyp
trying - read
trying - write
trying - math
trying - science
trying - socst
trying - gender
trying - ses
trying - schtyp
trying - read
trying - write
trying - math
trying - science
trying - socst
trying - ses
trying - schtyp
trying - read
trying - write
trying - math
trying - science
trying - socst
trying - ses
trying - schtyp
ttrying - read
ttrying - math
ttrying - science
ttrying - socst
ttrying - ses
ttrying - schtyp
ttrying - read
```

```r
> summary(mmod2)
```

```
Call:
  multinom(formula = prog ~ ses + schtyp + math + science + socst,
            data = hsb, trace = FALSE)

Coefficients:
(Intercept)  seslow  sesmiddle  schtyppublic  math  science
general    2.587029  0.87607389  0.6978995  0.6468812 -0.1212242  0.08209791
vocation   6.687272 -0.01569301  1.2065000  1.9955504 -0.1369641  0.03941237
socst
```
Compared to students from a high socioeconomic class, students from a low socioeconomic class are more likely to choose a general high school program, while students from a middle socioeconomic class are more likely to choose a general program but even more likely to choose a vocational program. It is interesting that students from a low socioeconomic class do not show more of an interest in vocational programs. Students from public schools are are more likely to choose a general program and much more likely to choose a vocational program, than students from private schools. High scores in maths and social sciences indicate a higher chance of choosing an academic program, while (curiously) high scores in science indicate a lower chance of choosing an academic program. If you wish to use a chi-squared test instead of the AIC, then you will have to separately fit all the candidate models, and then compare them using \texttt{anova}. For example:

```r
> mmodXgender <- multinom(prog ~ race + ses + schtyp + read + write + math + 
+  \hspace{1em} science + socst, hsb, trace = FALSE)
> anova(mmod, mmodXgender)
```

Clearly considering all possible variables to drop will take some time.

(c) For the student with id 99, compute the predicted probabilities of the three possible choices.

\textbf{Solution:}

```r
> hsb[hsb$id==99,]
```

\begin{verbatim}
id gender race ses schtyp prog read write math science socst
102 99 female white high public general 47 59 56 66 61
\end{verbatim}

```r
> predict(mmod2, newdata = hsb[hsb$id==99,], type="probs")
```

The \texttt{pneumo} data from the \texttt{faraway} package gives the number of coal miners classified by radiological examination into one of three categories of pneumoconiosis and by the number of years spent working at the coal face divided into eight categories.

(a) Treating the pneumoconiosis status as response variable as nominal, build a model for predicting the frequency of the three outcomes in terms of length of service and use it to predict the outcome for a miner with 25 years of service.

\textbf{Solution:} First we have a look at the data. Then the data needs to be reformatted before we can use the \texttt{multinom} function to fit a model. The fit looks quite good.
```r
# Load data
> data(pneumo)

# Compute counts
> counts <- xtabs(Freq ~ status + year, pneumo)

# Calculate proportions
> (props <- prop.table(counts, 2))

# Extract proportions for mild status
> props[1,]

# Extract proportions for normal status
> props[2,]

# Extract proportions for severe status
> props[3,]

# Define years vector
> years <- c(5.8, 15, 21.5, 27.5, 33.5, 39.5, 46, 51.5)

# Plot the proportions
> par(mfrow=c(1,1))
> plot(years, props[1,], col="red", ylim=c(0,1))
> points(years, props[2,], col="blue")
> points(years, props[3,], col="green")

# Fit multinomial model
> mmod <- multinom(t(counts) ~ years, trace=FALSE)

# Summary of the model
> summary(mmod)

Call: multinom(formula = t(counts) ~ years, trace = FALSE)

Coefficients:
   (Intercept)      years
normal  4.2916723   -0.08356506
severe  -0.7681706    0.02572027

Std. Errors:
   (Intercept)      years
normal  0.5214110   0.01528044
severe  0.7377192   0.01976662

Residual Deviance: 417.4496
AIC: 425.4496

# Predict probabilities
> fitted <- predict(mmod, newdata=list(year=years), type="probs")

# Plot predicted probabilities
> lines(years, fitted[,1], col="red")
> lines(years, fitted[,2], col="blue")
> lines(years, fitted[,3], col="green")
```
For a miner with 25 year down pit we have the following fitted probabilities

> predict(mmod, newdata=list(years=25), type="probs")

  mild  normal  severe
0.09148821 0.82778696 0.08072483

In the model above we had eight multinomial observations, with the number of trials equal to 98, 54, 43, 48, 51, 38, 28, 11. Each of these multinomials can be regarded as the sum of a number of independent multinomials each based on a single trial (just as a binomial is a sum of independent Bernoulli random variables). If we treat the data this way and fit a multinomial logistic regression, we get the same model, but what happens to the deviance degrees of freedom?

> pneumo2 <- data.frame(status = rep(pneumo$status, pneumo$Freq), + year = rep(pneumo$year, pneumo$Freq))
> mmod2 <- multinom(status ~ year, data = pneumo2, trace = FALSE)
> summary(mmod2)

Call:
  multinom(formula = status ~ year, data = pneumo2, trace = FALSE)

Coefficients:
  (Intercept)    year
normal  4.2916723 -0.08356506
severe -0.7681706  0.02572027

Std. Errors:
  (Intercept)    year
normal  0.5214110  0.02572027
(b) Repeat the analysis with the pneumonoconiosis status being treated as ordinal.

**Solution:**
First we convert status into an ordered factor (take care to get the order correct), then use the `polr` function. The fit looks good, and the AIC for this model is slightly smaller than that for the multinomial logistic regression model, so we prefer it.

```r
> pneumo2$status <- ordered(pneumo2$status, levels=c("normal", "mild", "severe"))
> library(MASS)
> omod <- polr(status ~ year, pneumo2)
> summary(omod)
```

**Coefficients:**

<table>
<thead>
<tr>
<th>Value</th>
<th>Std. Error</th>
<th>t value</th>
</tr>
</thead>
<tbody>
<tr>
<td>year</td>
<td>0.0959</td>
<td>0.01194</td>
</tr>
</tbody>
</table>

**Intercepts:**

<table>
<thead>
<tr>
<th>Value</th>
<th>Std. Error</th>
<th>t value</th>
</tr>
</thead>
<tbody>
<tr>
<td>normal</td>
<td>mild</td>
<td>3.9558</td>
</tr>
<tr>
<td>mild</td>
<td>severe</td>
<td>4.8690</td>
</tr>
</tbody>
</table>

Residual Deviance: 416.9188
AIC: 422.9188

```r
> plot(years, props[1,], col="red", ylim=c(0,1))
> points(years, props[2,], col="blue")
> points(years, props[3,], col="green")
> fitted <- predict(omod, newdata=list(year=years), type="probs")
> lines(years, fitted[,1], col="blue")
> lines(years, fitted[,2], col="red")
> lines(years, fitted[,3], col="green")
```
For a miner with 25 years exposure we have the following fitted probabilities

\[
\text{predict(omod, newdata=list(year=25), type="probs")}
\]

\[
\begin{array}{ccc}
\text{normal} & \text{mild} & \text{severe} \\
0.82610096 & 0.09601474 & 0.07788430 \\
\end{array}
\]

6 Suppose that \( X = (X_1, \ldots, X_k) \sim \text{multinomial}(n, \pi) \) where \( \pi = (\pi_1, \ldots, \pi_k) \). Since \( X_i \sim \text{bin}(n, \pi_i) \), we have \( \mathbb{E}X_i = n\pi_i \) and \( \text{Var} X_i = n\pi_i(1 - \pi_i) \). Show that for \( i \neq j \), \( \text{Cov}(X_i, X_j) = -n\pi_i\pi_j \).

Hint: just as for the binomial, we can write a multinomial\((n, \pi)\) as the sum of \( n \) independent multinomial\((1, \pi)\) random variables.

Alternative hint: \( \text{Var}(X + Y) = \text{Var} X + \text{Var} Y + 2\text{Cov}(X, Y) \).

**Solution:** If \( X \sim \text{multinomial}(1, \pi) \) then for \( i \neq j \) we have \( \mathbb{E}X_iX_j = 0 \) and thus \( \text{Cov}(X_i, X_j) = 0 - \mathbb{E}X_i\mathbb{E}X_j = -\pi_i\pi_j \). If \( X \sim \text{multinomial}(n, \pi) \) then it can be written as the sum of \( n \) independent multinomial\((1, \pi)\), whence we can multiply the covariances by \( n \) to get the result.

Alternatively, if we add \( X_i \) and \( X_j \) it is just as if we combined these two cases into a single case with probability \( \pi_i + \pi_j \). Thus

\[
\text{Cov}(X_i, X_j) = \frac{1}{2}(\text{Var}(X_i + X_j) - \text{Var} X_i - \text{Var} X_j) \\
= \frac{1}{2}(n(\pi_i + \pi_j)(1 - \pi_i - \pi_j) - n\pi_i(1 - \pi_i) - n\pi_j(1 - \pi_j)) \\
= -n\pi_i\pi_j
\]

7 Suppose that \( (X, Y, Z) \sim \text{multinomial}(n, (p_1, p_2, p_3)) \). Show that

\[
Y|\{X = x\} \sim \text{binomial}(n - x, p_2/(1 - p_1)).
\]

Hence obtain \( \mathbb{E}(Y|X = x) \).

13
Suppose that \( \gamma \) is the vector of differences between the log odds. Show that the relative odds for \( A \) compared to \( B \) is given by

\[
\frac{(n - x)!}{y!(n - x - y)!} \left( \frac{p_3}{1 - p_1} \right)^{n-x-y}
\]

But \( p_3/(1 - p_1) = 1 - p_2/(1 - p_1) \), so this is of the right form.

We get immediately that \( \mathbb{E}(Y|X = x) = (n - x) p_2/(1 - p_1) \). That is, given \( X = x \), we divvy up the remaining \( n - x \) trials between \( Y \) and \( Z \) proportionately to \( p_2 \) and \( p_3 \).

8 Proportional odds in ordinal regression. Suppose that \( Y \) takes values in the ordered set \( \{1, \ldots, J\} \). Using a logit link, our model for \( \gamma_{ij} = \mathbb{P}(Y_i \leq j) \) is

\[
\gamma_{ij} = \logit^{-1}(\theta_j - \mathbf{x}_i^T \beta).
\]

Thinking of \( \gamma_{ij} \) as a function of \( \mathbf{x}_i \), we can rewrite it as \( \gamma_j(\mathbf{x}_i) = \mathbb{P}(Y \leq j|\mathbf{x}_i) \).

Recall the odds for an event \( A \) are given by \( \mathbb{P}(A)/(1 - \mathbb{P}(A)) \). By relative odds we mean the ratio of two odds. Show that the relative odds for \( \{Y \leq j|\mathbf{x}_A\} \) and \( \{Y \leq j|\mathbf{x}_B\} \) do not depend on \( j \). For this reason, this model is often called the proportional odds model.

This independence of the odds ratio on \( j \) can be used to check the suitability of the model. Fit a proportional odds model to the \( \text{nes96} \) data, as in lectures. For \( j = 1 \) and \( j = 2 \), calculate the difference between the observed log odds at income level 1.5 and levels 4, 6, 8, 9.5, \ldots (the values in the vector \( \text{inca} \)). Do they look roughly the same?

Solution: The odds ratio is

\[
\frac{\mathbb{P}(Y \leq j|\mathbf{x}_A)}{1 - \mathbb{P}(Y \leq j|\mathbf{x}_A)} \div \frac{\mathbb{P}(Y \leq j|\mathbf{x}_B)}{1 - \mathbb{P}(Y \leq j|\mathbf{x}_B)} = \frac{\exp(\logit(\mathbb{P}(Y \leq j|\mathbf{x}_A)))}{\exp(\logit(\mathbb{P}(Y \leq j|\mathbf{x}_B)))} = \frac{\exp(\theta_j - \mathbf{x}_A^T \beta)}{\exp(\theta_j - \mathbf{x}_B^T \beta)} = \exp(-\mathbf{x}_A - \mathbf{x}_B)^T \beta
\]

which does not depend on \( j \), as required.

Note that the difference between the log odds is just \(-\mathbf{x}_A - \mathbf{x}_B)^T \beta\).

We check this for the \( \text{nes96} \) data. First we calculate \( \log \_\text{o1} \), the observed log odds of being Democrat, and \( \log \_\text{o2} \), the observed log odds of being Democrat or Independent. We do this for each income level.

\[
\text{data(nes96)}
\]
\[
\text{sPID <- nes96$SID}
\]
\[
\text{levels(sPID) <- c("Democrat","Democrat","Independent","Independent",}
+ \text{"Independent","Republican","Republican")}
\]
\[
\text{inca <- c(1.5,4,6,8,9.5,10.5,11.5,12.5,13.5,14.5,16,18.5,21,23.5,}
+ \text{27.5,32.5,37.5,42.5,47.5,55,67.5,82.5,97.5,115)}
\]
\[
\text{nicome <- inca[unclass(nes96$income)]}
\]
\[
\text{obs <- prop.table(table(nicome, sPID), 1)}
\]
\[
\text{p1 <- obs[,1]}
\]
\[
\text{p2 <- obs[,1] + obs[,2]}
\]
\[
\text{log_o1 <- log(p1/(1-p1))}
\]
\[
\text{log_o2 <- log(p2/(1-p2))}
\]

For the log odds of being Democrat, the difference between income level 1.5 and the rest is just given by \( \log \_\text{o1}[1] - \log \_\text{o1}[-1] \). This should look the same as \( \log \_\text{o2}[1] - \log \_\text{o2}[-1] \), which we check by taking differences. These differences are not particularly close to zero, but at least they don’t display any trend.
```r
> plot((log_o1[1] - log_o1[-1]) - (log_o2[1] - log_o2[-1]))
```