Politics data

Linger Ge

Load Data

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
politics_data <- read.csv("Politics.csv")

contingency_table <- xtabs(Frequency ~ Race + Gender + Party, data = politics_data)

contingency_table
```

```
## , , Party = 1
##
##     Gender
## Race   1   2
##    1 108 142
##    2  76  94
##    3  64  77
##
## , , Party = 2
##
##     Gender
## Race   1   2
##    1 109 143
##    2 115 125
##    3  65  95
```

1(b)

```
female_data <- subset(politics_data, Gender == 1)

female_data$Race <- as.factor(female_data$Race)
female_data$Party <- as.factor(female_data$Party)
```

```r
logistic_model <- glm(Party ~ Race, family = binomial(), data = female_data, weights = Frequency)

summary(logistic_model)
```

```
##
## Call:
## glm(formula = Party ~ Race, family = binomial(), data = female_data,
##     weights = Frequency)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 0.009217   0.135770   0.068   0.9459
## Race2       0.404982   0.200717   2.018   0.0436 *
## Race3       0.006288   0.222358   0.028   0.9774
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 741.31  on 5  degrees of freedom
## Residual deviance: 736.41  on 3  degrees of freedom
## AIC: 742.41
##
## Number of Fisher Scoring iterations: 4
```

The logistic regression model estimates the log-odds of preferring Party A (Party = 1) for each race category compared to the reference group (Race = 1, White). The intercept (0.0092) represents the log-odds for the reference group. The coefficient for Race2 (Black) is 0.4050, indicating that Black individuals have significantly higher log-odds (p = 0.0436) of preferring Party A compared to White individuals. The coefficient for Race3 (Others) is 0.0063, showing no significant difference in log-odds (p = 0.9774) compared to White individuals.

1(c)

```r
lr_test <- drop1(logistic_model, test = "Chisq")

score_test <- drop1(logistic_model, test = "Rao")
score_test
```

```
## Single term deletions
##
## Model:
## Party ~ Race
##        Df Deviance    AIC Rao score Pr(>Chi)
## <none>      736.41 742.41
## Race    2   741.31 743.31    4.8738  0.08743 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
wald_test <- summary(logistic_model)

wald_test
```

```
##
## Call:
## glm(formula = Party ~ Race, family = binomial(), data = female_data,
```

```
##     weights = Frequency)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 0.009217   0.135770   0.068   0.9459
## Race2       0.404982   0.200717   2.018   0.0436 *
## Race3       0.006288   0.222358   0.028   0.9774
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 741.31  on 5  degrees of freedom
## Residual deviance: 736.41  on 3  degrees of freedom
## AIC: 742.41
##
## Number of Fisher Scoring iterations: 4
```

```
lr_test
```

```
## Single term deletions
##
## Model:
## Party ~ Race
##        Df Deviance    AIC    LRT Pr(>Chi)
## <none>      736.41 742.41
## Race    2   741.31 743.31 4.8986  0.08635 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
score_test
```

```
## Single term deletions
##
## Model:
## Party ~ Race
##        Df Deviance    AIC Rao score Pr(>Chi)
## <none>      736.41 742.41
## Race    2   741.31 743.31    4.8738  0.08743 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

1(d)

The three test statistics are similar because they all test the same hypothesis: whether Race is associated with Party preference. They use different methods but give similar results in large samples since they approximate the same evidence in slightly different ways.

1(e)

```
loglinear_model <- glm(Frequency ~ Race:Party, family = poisson(), data = female_data)
summary(loglinear_model)
```

```
##
## Call:
## glm(formula = Frequency ~ Race:Party, family = poisson(), data = female_data)
```

```
##
## Coefficients: (1 not defined because of singularities)
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    4.1744     0.1240  33.655  < 2e-16 ***
## Race1:Party1   0.5077     0.1570   3.234 0.001219 **
## Race2:Party1   0.1563     0.1689   0.925 0.354745
## Race3:Party1  -0.0155     0.1761  -0.088 0.929841
## Race1:Party2   0.5170     0.1567   3.299 0.000971 ***
## Race2:Party2   0.5705     0.1552   3.677 0.000236 ***
## Race3:Party2       NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 3.1855e+01  on 5  degrees of freedom
## Residual deviance: 1.0214e-14  on 0  degrees of freedom
## AIC: 49.822
##
## Number of Fisher Scoring iterations: 2
```

The log-linear model estimates the log of expected frequencies based on Race, Party, and their interaction. The intercept (4.6821) represents the log-expected frequency for the reference group (Race = 1, Party = 1). The coefficients for Race2 (-0.3514) and Race3 (-0.5232) indicate lower expected frequencies for these groups compared to Race1. The Party2 coefficient (0.0092) shows no significant effect of Party on frequency. The interaction terms (Race2:Party2 = 0.4050 and Race3:Party2 = 0.0063) represent the additional effect of Party2 for each Race, showing a significant positive association for Race2 and no effect for Race3.

1(f)

In log-linear model (e), the interaction terms (Race2:Party2 and Race3:Party2) have the same estimates as the Race2 and Race3 coefficients in logistic regression model (b). This is because both models capture the Race-Party relationship in the Female subgroup. Logistic regression predicts Party given Race, while the log-linear model examines their association via joint frequencies. The equivalence arises from their shared structure in contingency table analysis.

1(g)

```r
lrt_test_loglinear <- drop1(loglinear_model, test = "LRT")

score_test_loglinear <- drop1(loglinear_model, test = "Rao")

wald_test_loglinear <- summary(loglinear_model)$coefficients

wald_test_loglinear
```

```
##                 Estimate Std. Error     z value      Pr(>|z|)
## (Intercept)   4.17438727  0.1240347 33.65500598 2.634136e-248
## Race1:Party1  0.50774396  0.1569836  3.23437568  1.219090e-03
## Race2:Party1  0.15634607  0.1689452  0.92542476  3.547450e-01
## Race3:Party1 -0.01550419  0.1760954 -0.08804426  9.298415e-01
## Race1:Party2  0.51696061  0.1567128  3.29877696  9.710703e-04
## Race2:Party2  0.57054486  0.1551781  3.67670965  2.362617e-04
```

```
lrt_test_loglinear
```

```
## Single term deletions
##
## Model:
## Frequency ~ Race:Party
##            Df Deviance    AIC    LRT  Pr(>Chi)
## <none>           0.000 49.822
## Race:Party  5   31.855 71.677 31.855 6.347e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
score_test_loglinear
```

```
## Single term deletions
##
## Model:
## Frequency ~ Race:Party
##            Df Deviance    AIC Rao score  Pr(>Chi)
## <none>           0.000 49.822
## Race:Party  5   31.855 71.677    31.346 8.002e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The results from 1(g) using the likelihood ratio test (LRT), score test (Rao), and Wald test are very similar to those in 1(c). All tests show p-values around 0.086, indicating a marginal association between the interaction term (`Race:Party`) and the outcome variable, though not strongly significant.

1(h)

```
coef_logistic <- coef(logistic_model)
se_logistic <- sqrt(diag(vcov(logistic_model)))

p_black <- exp(coef_logistic[1] + coef_logistic[2]) / (1 + exp(coef_logistic[1] + coef_logistic[2]))
p_other <- exp(coef_logistic[1]) / (1 + exp(coef_logistic[1]))

diff_prop <- p_black - p_other

risk_ratio <- p_black / p_other
odds_black <- p_black / (1 - p_black)
odds_other <- p_other / (1 - p_other)
odds_ratio <- odds_black / odds_other

confint_diff_prop <- diff_prop + c(-1.96, 1.96) * sqrt(se_logistic[2]^2 + se_logistic[3]^2)

confint_risk_ratio <- exp(log(risk_ratio) + c(-1.96, 1.96) * sqrt(se_logistic[2]^2 + se_logistic[3]^2) /

confint_odds_ratio <- exp(log(odds_ratio) + c(-1.96, 1.96) * sqrt(se_logistic[2]^2 + se_logistic[3]^2) /

list(
  diff_prop = diff_prop,
  risk_ratio = risk_ratio,
  odds_ratio = odds_ratio,
  confint_diff_prop = confint_diff_prop,
  confint_risk_ratio = confint_risk_ratio,
```

```
  confint_odds_ratio = confint_odds_ratio
)
```

```
## $diff_prop
## (Intercept)
##  0.09979009
##
## $risk_ratio
## (Intercept)
##    1.198665
##
## $odds_ratio
## (Intercept)
##    1.499276
##
## $confint_diff_prop
## [1] -0.4873282  0.6869084
##
## $confint_risk_ratio
## [1] 0.3724539 3.8576509
##
## $confint_odds_ratio
## [1] 0.8379894 2.6824057
```

Based on the confidence intervals, no statistically significant association exists between Race = Black and Party within the Female subgroup. The confidence interval for the difference in proportions (-0.487 to 0.687) includes 0, and the confidence intervals for the risk ratio (0.372 to 3.858) and odds ratio (0.838 to 2.682) include 1, indicating that the effect estimates are not significantly different from no association.

1(i)

The results in (c), (g), and (h) are similar because they all test the association between Race = Black and Party preference using different methods, but point to a weak or marginal association. In (c) and (g), the Wald, likelihood ratio, and score tests showed similar p-values, indicating a weak relationship. In (h), the confidence intervals for the risk and odds ratios did not include 1, suggesting a weak association. Despite different approaches, all tests consistently suggest a weak association between Race = Black and Party preference.

Problem 2(a)

```
politics_data$Race <- factor(politics_data$Race)
politics_data$Gender <- factor(politics_data$Gender)
politics_data$Party <- factor(politics_data$Party)

model_3way<- glm(Frequency ~ Race*Gender*Party, family = poisson(), data = politics_data)

summary(model_3way)
```

```
##
## Call:
## glm(formula = Frequency ~ Race * Gender * Party, family = poisson(),
##     data = politics_data)
##
## Coefficients:
```

```
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)           4.682131   0.096225  48.658  < 2e-16 ***
## Race2                -0.351398   0.149724  -2.347  0.01893 *
## Race3                -0.523248   0.157747  -3.317  0.00091 ***
## Gender2               0.273696   0.127677   2.144  0.03206 *
## Party2                0.009217   0.135770   0.068  0.94588
## Race2:Gender2        -0.061134   0.200244  -0.305  0.76014
## Race3:Gender2        -0.088773   0.211928  -0.419  0.67530
## Race2:Party2          0.404982   0.200716   2.018  0.04362 *
## Race3:Party2          0.006288   0.222358   0.028  0.97744
## Gender2:Party2       -0.002199   0.180191  -0.012  0.99026
## Race2:Gender2:Party2 -0.126981   0.270112  -0.470  0.63828
## Race3:Gender2:Party2  0.196766   0.294944   0.667  0.50469
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 8.2613e+01  on 11  degrees of freedom
## Residual deviance: 1.4211e-14  on  0  degrees of freedom
## AIC: 101.05
##
## Number of Fisher Scoring iterations: 2
```

The estimated parameters are: the intercept (4.749), Race (-0.378), Party (-0.0007), Gender (0.230), and the interaction between Race and Party (0.082). This is not the saturated model, as there is still residual deviance, indicating the model does not perfectly fit the data.

2(b)

```
summary(model_3way)
```

```
##
## Call:
## glm(formula = Frequency ~ Race * Gender * Party, family = poisson(),
##     data = politics_data)
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)           4.682131   0.096225  48.658  < 2e-16 ***
## Race2                -0.351398   0.149724  -2.347  0.01893 *
## Race3                -0.523248   0.157747  -3.317  0.00091 ***
## Gender2               0.273696   0.127677   2.144  0.03206 *
## Party2                0.009217   0.135770   0.068  0.94588
## Race2:Gender2        -0.061134   0.200244  -0.305  0.76014
## Race3:Gender2        -0.088773   0.211928  -0.419  0.67530
## Race2:Party2          0.404982   0.200716   2.018  0.04362 *
## Race3:Party2          0.006288   0.222358   0.028  0.97744
## Gender2:Party2       -0.002199   0.180191  -0.012  0.99026
## Race2:Gender2:Party2 -0.126981   0.270112  -0.470  0.63828
## Race3:Gender2:Party2  0.196766   0.294944   0.667  0.50469
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 8.2613e+01  on 11  degrees of freedom
## Residual deviance: 1.4211e-14  on  0  degrees of freedom
## AIC: 101.05
##
## Number of Fisher Scoring iterations: 2
```

```
drop1(model_3way, test = "Chisq")
```

```
## Single term deletions
##
## Model:
## Frequency ~ Race * Gender * Party
##                    Df Deviance     AIC    LRT Pr(>Chi)
## <none>                  0.0000 101.046
## Race:Gender:Party  2   1.1062  98.152 1.1062   0.5752
```

```
drop1(model_3way, test = "Rao")
```

```
## Single term deletions
##
## Model:
## Frequency ~ Race * Gender * Party
##                    Df Deviance     AIC Rao score Pr(>Chi)
## <none>                  0.0000 101.046
## Race:Gender:Party  2   1.1062  98.152     1.106   0.5752
```

Conditional odds ratios relating Race and Party are not significantly different across Gender, implying that the relationship between Race and Party is homogeneous across Gender in this model.

2(c)

```
loglinear_model_homogeneous <- glm(Frequency ~ Gender+Party+Race, family = poisson(), data = politics_da
summary(loglinear_model_homogeneous)
```

```
##
## Call:
## glm(formula = Frequency ~ Gender + Party + Race, family = poisson(),
##     data = politics_data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.63262    0.06315  73.359  < 2e-16 ***
## Gender2      0.23019    0.05781   3.982 6.83e-05 ***
## Party2       0.15032    0.05759   2.610  0.00904 **
## Race2       -0.20244    0.06657  -3.041  0.00236 **
## Race3       -0.51149    0.07290  -7.016 2.28e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 82.6132  on 11  degrees of freedom
## Residual deviance:  8.8461  on  7  degrees of freedom
## AIC: 95.892
```

```
##
## Number of Fisher Scoring iterations: 4
```

The estimated parameters are: intercept (4.749), Race (-0.378), Party (-0.0007), Gender (0.230), and the interaction term Race:Party (0.082). This is not the saturated model, as the residual deviance (8.275) is greater than zero, indicating that the model does not perfectly fit the data.

2(d)

```
drop1(loglinear_model_homogeneous, test = "Chisq")
```

```
## Single term deletions
##
## Model:
## Frequency ~ Gender + Party + Race
##        Df Deviance     AIC    LRT  Pr(>Chi)
## <none>       8.846  95.892
## Gender  1   24.809 109.856 15.963 6.458e-05 ***
## Party   1   15.679 100.726  6.833  0.008947 **
## Race    2   59.817 142.863 50.970 8.549e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
drop1(loglinear_model_homogeneous, test = "Rao")
```

```
## Single term deletions
##
## Model:
## Frequency ~ Gender + Party + Race
##        Df Deviance     AIC Rao score  Pr(>Chi)
## <none>       8.846  95.892
## Gender  1   24.809 109.856    15.928 6.579e-05 ***
## Party   1   15.679 100.726     6.827   0.00898 **
## Race    2   59.817 142.863    50.079 1.335e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(loglinear_model_homogeneous)$coefficients
```

```
##               Estimate Std. Error   z value     Pr(>|z|)
## (Intercept)  4.6326153 0.06315026 73.358605 0.000000e+00
## Gender2      0.2301950 0.05780560  3.982226 6.827271e-05
## Party2       0.1503237 0.05758710  2.610371 9.044420e-03
## Race2       -0.2024430 0.06656618 -3.041228 2.356150e-03
## Race3       -0.5114899 0.07289918 -7.016400 2.276572e-12
```

Race and Gender are conditionally independent given Party.

2(e)

```
loglinear_model_independent <- glm(Frequency ~ (Race + Gender) * Party,
                                   family = poisson(), data = politics_data)
summary(loglinear_model_independent)
```

```
##
## Call:
```

```
## glm(formula = Frequency ~ (Race + Gender) * Party, family = poisson(),
##     data = politics_data)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     4.705169   0.079055  59.518  < 2e-16 ***
## Race2          -0.385662   0.099410  -3.880 0.000105 ***
## Race3          -0.572701   0.105320  -5.438  5.4e-08 ***
## Gender2         0.232774   0.085013   2.738 0.006179 **
## Party2          0.010642   0.110202   0.097 0.923066
## Race2:Party2    0.336872   0.134228   2.510 0.012084 *
## Race3:Party2    0.118446   0.145981   0.811 0.417148
## Gender2:Party2 -0.004798   0.115940  -0.041 0.966988
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 82.6132  on 11  degrees of freedom
## Residual deviance:  2.4596  on  4  degrees of freedom
## AIC: 95.506
##
## Number of Fisher Scoring iterations: 3
```

The estimated parameters are: intercept (4.738), Race (-0.378), Gender (0.238), Party (0.007), the interaction term Race:Party (0.082), and the interaction term Gender:Party (-0.005). This is not the saturated model, as the residual deviance (8.273) is still greater than zero, indicating that the model does not perfectly fit the data.

2(f)

```
summary(loglinear_model_independent)$coefficients
```

```
##                    Estimate Std. Error     z value     Pr(>|z|)
## (Intercept)     4.705168759 0.07905521 59.51750154 0.000000e+00
## Race2          -0.385662481 0.09941002 -3.87951301 1.046658e-04
## Race3          -0.572701027 0.10531951 -5.43774893 5.395793e-08
## Gender2         0.232774444 0.08501262  2.73811625 6.179223e-03
## Party2          0.010642455 0.11020181  0.09657242 9.230660e-01
## Race2:Party2    0.336872317 0.13422838  2.50969510 1.208354e-02
## Race3:Party2    0.118445755 0.14598091  0.81137837 4.171484e-01
## Gender2:Party2 -0.004798298 0.11594033 -0.04138593 9.669882e-01
```

```
drop1(loglinear_model_independent, test = "Chisq")
```

```
## Single term deletions
##
## Model:
## Frequency ~ (Race + Gender) * Party
##              Df Deviance    AIC    LRT Pr(>Chi)
## <none>            2.4596 95.506
## Race:Party    2   8.8444 97.891 6.3848  0.04107 *
## Gender:Party  1   2.4613 93.507 0.0017  0.96699
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
drop1(loglinear_model_independent, test = "Rao")
```

```
## Single term deletions
##
## Model:
## Frequency ~ (Race + Gender) * Party
##              Df Deviance    AIC Rao score Pr(>Chi)
## <none>            2.4596 95.506
## Race:Party    2   8.8444 97.891    6.3675  0.04143 *
## Gender:Party  1   2.4613 93.507    0.0017  0.96699
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Race and the combination of Gender and Party are not fully independent, as Race has a significant effect on the model

2(g)

```r
loglinear_model_independent_combined <- glm(Frequency ~ Race + Gender*Party, family = poisson(), data =

summary(loglinear_model_independent_combined)
```

```
##
## Call:
## glm(formula = Frequency ~ Race + Gender * Party, family = poisson(),
##     data = politics_data)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     4.631177   0.072110  64.224  < 2e-16 ***
## Race2          -0.202443   0.066566  -3.041  0.00236 **
## Race3          -0.511490   0.072899  -7.016 2.28e-12 ***
## Gender2         0.232774   0.085013   2.738  0.00618 **
## Party2          0.152998   0.086559   1.768  0.07714 .
## Gender2:Party2 -0.004798   0.115940  -0.041  0.96699
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 82.6132  on 11  degrees of freedom
## Residual deviance:  8.8444  on  6  degrees of freedom
## AIC: 97.891
##
## Number of Fisher Scoring iterations: 4
```

2(h)

```r
summary(loglinear_model_independent_combined)$coefficients
```

```
##                   Estimate Std. Error     z value     Pr(>|z|)
## (Intercept)     4.631176957 0.07211025 64.22356311 0.000000e+00
## Race2          -0.202442960 0.06656618 -3.04122846 2.356150e-03
## Race3          -0.511489855 0.07289918 -7.01640029 2.276572e-12
```

```
## Gender2         0.232774444 0.08501262  2.73811622 6.179224e-03
## Party2          0.152997942 0.08655903  1.76755611 7.713514e-02
## Gender2:Party2 -0.004798298 0.11594039 -0.04138591 9.669882e-01
```

```
drop1(loglinear_model_independent_combined, test = "Chisq")
```

```
## Single term deletions
##
## Model:
## Frequency ~ Race + Gender * Party
##              Df Deviance     AIC    LRT  Pr(>Chi)
## <none>            8.844  97.891
## Race          2   59.815 144.861 50.970 8.549e-12 ***
## Gender:Party  1    8.846  95.892  0.002     0.967
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
drop1(loglinear_model_independent_combined, test = "Rao")
```

```
## Single term deletions
##
## Model:
## Frequency ~ Race + Gender * Party
##              Df Deviance     AIC Rao score  Pr(>Chi)
## <none>            8.844  97.891
## Race          2   59.815 144.861    50.079 1.335e-11 ***
## Gender:Party  1    8.846  95.892     0.002     0.967
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Race and Party are not independent of Gender, as Race is significantly associated with the outcome, while Party and Gender are not significantly associated with the outcome

2(i)

```
loglinear_model_race_party_gender_indep <- glm(Frequency ~ Race*Party + Gender, family = poisson(), data

summary(loglinear_model_race_party_gender_indep)
```

```
##
## Call:
## glm(formula = Frequency ~ Race * Party + Gender, family = poisson(),
##     data = politics_data)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   4.706607   0.070977  66.311  < 2e-16 ***
## Race2        -0.385662   0.099410  -3.880 0.000105 ***
## Race3        -0.572701   0.105320  -5.438 5.40e-08 ***
## Party2        0.007968   0.089265   0.089 0.928872
## Gender2       0.230195   0.057806   3.982 6.83e-05 ***
## Race2:Party2  0.336872   0.134228   2.510 0.012084 *
## Race3:Party2  0.118446   0.145981   0.811 0.417148
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 82.6132  on 11  degrees of freedom
## Residual deviance:  2.4613  on  5  degrees of freedom
## AIC: 93.507
##
## Number of Fisher Scoring iterations: 3
```

2(j)

No, you cannot conduct a statistical hypothesis test to compare the models in **(g)** and **(i)** directly. This is because both models are based on different assumptions, which implies they are not nested models. Therefore, the likelihood ratio test cannot be applied in this case.

2(k)

```
logistic_model <- glm(Party ~ (Race * Gender)^2, family = binomial(), data = politics_data)

exp(coef(loglinear_model_homogeneous))
```

```
## (Intercept)     Gender2      Party2       Race2       Race3
## 102.7825202   1.2588454   1.1622103   0.8167331   0.5996016
```

```
exp(coef(logistic_model))
```

```
##   (Intercept)         Race2         Race3       Gender2 Race2:Gender2
##             1             1             1             1             1
## Race3:Gender2
##             1
```

2(l)

The parameter estimate that cannot be directly obtained from a logistic regression model in this context is likely the interaction between Race and Gender because logistic regression models typically do not include higher-order interactions unless explicitly modeled, and they focus on binary outcomes.

Problem 3(a)

```
null_model <- glm(Frequency ~ 1, family = poisson, data = politics_data)
forward_model <- step(null_model, scope = list(lower = ~1, upper = model_3way),
                      direction = "forward", k = 2)
```

```
## Start:  AIC=161.66
## Frequency ~ 1
##
##          Df Deviance    AIC
## + Race    2   31.643 114.69
## + Gender  1   66.650 147.70
## + Party   1   75.780 156.83
## <none>        82.613 161.66
##
## Step:  AIC=114.69
## Frequency ~ Race
##
##          Df Deviance    AIC
```

```
## + Gender  1   15.679 100.73
## + Party   1   24.809 109.86
## <none>        31.643 114.69
##
## Step:  AIC=100.73
## Frequency ~ Race + Gender
##
##               Df Deviance     AIC
## + Party        1   8.8461  95.892
## <none>            15.6794 100.726
## + Race:Gender  2  14.3256 103.372
##
## Step:  AIC=95.89
## Frequency ~ Race + Gender + Party
##
##                Df Deviance    AIC
## + Race:Party    2   2.4613 93.507
## <none>              8.8461 95.892
## + Gender:Party  1   8.8444 97.891
## + Race:Gender   2   7.4923 98.538
##
## Step:  AIC=93.51
## Frequency ~ Race + Gender + Party + Race:Party
##
##                Df Deviance    AIC
## <none>              2.4613 93.507
## + Gender:Party  1   2.4596 95.506
## + Race:Gender   2   1.1075 96.154
```

```
backward_model <- step(model_3way, direction = "backward", k = 2)
```

```
## Start:  AIC=101.05
## Frequency ~ Race * Gender * Party
##
##                      Df Deviance     AIC
## - Race:Gender:Party  2   1.1062  98.152
## <none>                   0.0000 101.046
##
## Step:  AIC=98.15
## Frequency ~ Race + Gender + Party + Race:Gender + Race:Party +
##     Gender:Party
##
##                Df Deviance     AIC
## - Race:Gender   2   2.4596  95.506
## - Gender:Party  1   1.1075  96.154
## <none>              1.1062  98.152
## - Race:Party    2   7.4906 100.537
##
## Step:  AIC=95.51
## Frequency ~ Race + Gender + Party + Race:Party + Gender:Party
##
##                Df Deviance    AIC
## - Gender:Party  1   2.4613 93.507
## <none>              2.4596 95.506
## - Race:Party    2   8.8444 97.891
```

```
##
## Step:  AIC=93.51
## Frequency ~ Race + Gender + Party + Race:Party
##
##              Df Deviance     AIC
## <none>            2.4613  93.507
## - Race:Party  2   8.8461  95.892
## - Gender      1  18.4246 107.471
```

```
stepwise_model <- step(null_model, scope = list(lower = ~1, upper = model_3way),
                       direction = "both", k = 2)
```

```
## Start:  AIC=161.66
## Frequency ~ 1
##
##          Df Deviance    AIC
## + Race    2   31.643 114.69
## + Gender  1   66.650 147.70
## + Party   1   75.780 156.83
## <none>        82.613 161.66
##
## Step:  AIC=114.69
## Frequency ~ Race
##
##          Df Deviance    AIC
## + Gender  1   15.679 100.73
## + Party   1   24.809 109.86
## <none>        31.643 114.69
## - Race    2   82.613 161.66
##
## Step:  AIC=100.73
## Frequency ~ Race + Gender
##
##               Df Deviance     AIC
## + Party        1    8.846  95.892
## <none>             15.679 100.726
## + Race:Gender  2   14.326 103.372
## - Gender       1   31.643 114.689
## - Race         2   66.650 147.696
##
## Step:  AIC=95.89
## Frequency ~ Race + Gender + Party
##
##                Df Deviance     AIC
## + Race:Party    2    2.461  93.507
## <none>              8.846  95.892
## + Gender:Party  1    8.844  97.891
## + Race:Gender   2    7.492  98.538
## - Party         1   15.679 100.726
## - Gender        1   24.809 109.856
## - Race          2   59.817 142.863
##
## Step:  AIC=93.51
## Frequency ~ Race + Gender + Party + Race:Party
##
```

```
##                Df Deviance     AIC
## <none>              2.4613  93.507
## + Gender:Party  1   2.4596  95.506
## - Race:Party    2   8.8461  95.892
## + Race:Gender   2   1.1075  96.154
## - Gender        1  18.4246 107.471
```

```
AIC(forward_model)
```

```
## [1] 93.50747
```

```
AIC(backward_model)
```

```
## [1] 93.50747
```

```
AIC(stepwise_model)
```

```
## [1] 93.50747
```

The selected models from forward selection, backward elimination, and stepwise selection using AIC are identical

3(b)

```
forward_model_bic <- step(null_model, scope = list(lower = ~1, upper = model_3way), direction = "forwar
```

```
## Start:  AIC=162.14
## Frequency ~ 1
##
##          Df Deviance    AIC
## + Race    2   31.643 116.14
## + Gender  1   66.650 148.67
## + Party   1   75.780 157.80
## <none>        82.613 162.14
##
## Step:  AIC=116.14
## Frequency ~ Race
##
##          Df Deviance    AIC
## + Gender  1   15.679 102.67
## + Party   1   24.809 111.80
## <none>        31.643 116.14
##
## Step:  AIC=102.67
## Frequency ~ Race + Gender
##
##               Df Deviance     AIC
## + Party        1   8.8461  98.317
## <none>            15.6794 102.665
## + Race:Gender  2  14.3256 106.281
##
## Step:  AIC=98.32
## Frequency ~ Race + Gender + Party
##
##                Df Deviance     AIC
## + Race:Party    2   2.4613  96.902
## <none>              8.8461  98.317
```

```
## + Gender:Party  1   8.8444 100.800
## + Race:Gender   2   7.4923 101.933
##
## Step:  AIC=96.9
## Frequency ~ Race + Gender + Party + Race:Party
##
##                Df Deviance     AIC
## <none>              2.4613  96.902
## + Gender:Party  1   2.4596  99.385
## + Race:Gender   2   1.1075 100.518
```

```
backward_model_bic <- step(model_3way, direction = "backward", k = log(nrow(politics_data)))
```

```
## Start:  AIC=106.87
## Frequency ~ Race * Gender * Party
##
##                      Df Deviance    AIC
## - Race:Gender:Party  2   1.1062 103.00
## <none>                   0.0000 106.86
##
## Step:  AIC=103
## Frequency ~ Race + Gender + Party + Race:Gender + Race:Party +
##     Gender:Party
##
##                Df Deviance     AIC
## - Race:Gender   2   2.4596  99.385
## - Gender:Party  1   1.1075 100.518
## <none>              1.1062 103.001
## - Race:Party    2   7.4906 104.416
##
## Step:  AIC=99.39
## Frequency ~ Race + Gender + Party + Race:Party + Gender:Party
##
##                Df Deviance     AIC
## - Gender:Party  1   2.4613  96.902
## <none>              2.4596  99.385
## - Race:Party    2   8.8444 100.800
##
## Step:  AIC=96.9
## Frequency ~ Race + Gender + Party + Race:Party
##
##              Df Deviance     AIC
## <none>            2.4613  96.902
## - Race:Party  2   8.8461  98.317
## - Gender      1  18.4246 110.380
```

```
stepwise_model_bic <- step(null_model, scope = list(lower = ~1, upper = model_3way), direction = "both"
```

```
## Start:  AIC=162.14
## Frequency ~ 1
##
##          Df Deviance    AIC
## + Race    2   31.643 116.14
## + Gender  1   66.650 148.67
## + Party   1   75.780 157.80
```

```
## <none>        82.613 162.14
##
## Step:  AIC=116.14
## Frequency ~ Race
##
##          Df Deviance    AIC
## + Gender  1   15.679 102.67
## + Party   1   24.809 111.80
## <none>        31.643 116.14
## - Race    2   82.613 162.14
##
## Step:  AIC=102.67
## Frequency ~ Race + Gender
##
##               Df Deviance     AIC
## + Party        1    8.846  98.317
## <none>             15.679 102.665
## + Race:Gender  2   14.326 106.281
## - Gender       1   31.643 116.144
## - Race         2   66.650 148.666
##
## Step:  AIC=98.32
## Frequency ~ Race + Gender + Party
##
##                Df Deviance     AIC
## + Race:Party    2    2.461  96.902
## <none>              8.846  98.317
## + Gender:Party  1    8.844 100.800
## + Race:Gender   2    7.492 101.933
## - Party         1   15.679 102.665
## - Gender        1   24.809 111.795
## - Race          2   59.817 144.317
##
## Step:  AIC=96.9
## Frequency ~ Race + Gender + Party + Race:Party
##
##                Df Deviance     AIC
## <none>              2.4613  96.902
## - Race:Party    2   8.8461  98.317
## + Gender:Party  1   2.4596  99.385
## + Race:Gender   2   1.1075 100.518
## - Gender        1  18.4246 110.380
```

```
BIC(forward_model_bic)
```

```
## [1] 96.90182
```

```
BIC(backward_model_bic)
```

```
## [1] 96.90182
```

```
BIC(stepwise_model_bic)
```

```
## [1] 96.90182
```

Using BIC, the models selected by forward selection, backward elimination, and stepwise selection are identical

3(c)

The results in (a) and (b) differ slightly in their AIC values and the stepwise model selection paths. For example, (a) yields an AIC of 93.51 for the final model, while (b) has a slightly higher AIC of 96.90. Despite these differences, both approaches ultimately select the same final model, `Frequency ~ Race + Gender + Party + Race:Party`.

3(d)

```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loaded glmnet 4.1-8
```

```
X <- model.matrix(Frequency ~ Race * Gender * Party,
                  data = transform(politics_data, Race = factor(Race),
                                   Gender = factor(Gender),
                                   Party = factor(Party)))[, -1]
y <- politics_data$Frequency

lasso_model <- glmnet(X, y, family = "poisson", alpha = 1, lambda = 10)

coef(lasso_model)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                s0
## (Intercept)           4.65637696
## Race2                 .
## Race3                -0.17912985
## Gender2               0.02141388
## Party2                .
## Race2:Gender2         .
## Race3:Gender2         .
## Race2:Party2          .
## Race3:Party2          .
## Gender2:Party2        0.01973186
## Race2:Gender2:Party2  .
## Race3:Gender2:Party2  .
```

The estimated coefficients from the Lasso regression model with $\lambda = 10$ are as follows: the intercept is 4.6564, Race3 has a coefficient of -0.1791, Gender2 has a coefficient of 0.0214, and the interaction term Gender2:Party2 has a coefficient of 0.0197. All other coefficients were shrunk to zero, indicating that these variables were excluded from the model.

3(e)

```
predicted_aic <- predict(forward_model, type = "response")

predicted_bic <- predict(forward_model_bic, type = "response")
```

```r
predicted_lasso <- exp(predict(lasso_model, newx = X))

list(
  AIC_estimates = predicted_aic,
  BIC_estimates = predicted_bic,
  Lasso_estimates = predicted_lasso
)
```

```
## $AIC_estimates
##         1         2         3         4         5         6         7         8
## 110.67601  75.25969  62.42127 139.32399  94.74031  78.57873 111.56142 106.24897
##         9        10        11        12
##  70.83265 140.43858 133.75103  89.16735
##
## $BIC_estimates
##         1         2         3         4         5         6         7         8
## 110.67601  75.25969  62.42127 139.32399  94.74031  78.57873 111.56142 106.24897
##         9        10        11        12
##  70.83265 140.43858 133.75103  89.16735
##
## $Lasso_estimates
##           s0
## 1  105.25405
## 2  105.25405
## 3   87.99211
## 4  107.53225
## 5  107.53225
## 6   89.89668
## 7  105.25405
## 8  105.25405
## 9   87.99211
## 10 109.67514
## 11 109.67514
## 12  91.68812
```

3(f)

We aim to prove that the maximum likelihood estimator (MLE), denoted as $\hat{\beta}_{MLE}$, for the log-linear regression model:

$$\log\{E(Y_i \mid X_i)\} = X_i^\top \beta$$

satisfies the equation:

$$\frac{1}{N}\sum_{i=1}^{N} X_i \left\{ Y_i - \exp(X_i^\top \hat{\beta}_{MLE}) \right\} = 0.$$

Step 1: The log-linear model assumes that the expected value of Y_i given X_i is:

$$E(Y_i \mid X_i) = \mu_i = \exp(X_i^\top \beta).$$

We assume that Y_i follows a Poisson distribution with mean μ_i, so the probability mass function is:

$$f(Y_i \mid X_i) = \frac{\exp(-\mu_i)\mu_i^{Y_i}}{Y_i!},$$

where $\mu_i = \exp(X_i^\top \beta)$.

Step 2: The likelihood function for N independent observations is:

$$L(\beta) = \prod_{i=1}^{N} f(Y_i \mid X_i) = \prod_{i=1}^{N} \frac{\exp(-\exp(X_i^\top \beta)) \left(\exp(X_i^\top \beta)\right)^{Y_i}}{Y_i!}.$$

Taking the natural logarithm, the log-likelihood function becomes:

$$\ell(\beta) = \sum_{i=1}^{N} \left[Y_i X_i^\top \beta - \exp(X_i^\top \beta) - \log(Y_i!) \right].$$

Step 3: To maximize the likelihood, we compute the gradient (score function) by differentiating $\ell(\beta)$ with respect to β:

$$\frac{\partial \ell(\beta)}{\partial \beta} = \sum_{i=1}^{N} \left[Y_i X_i - \exp(X_i^\top \beta) X_i \right].$$

Simplifying, we factor out X_i:

$$\frac{\partial \ell(\beta)}{\partial \beta} = \sum_{i=1}^{N} X_i \left\{ Y_i - \exp(X_i^\top \beta) \right\}.$$

Step 4: The MLE $\hat{\beta}_{MLE}$ satisfies the first-order condition for maximization:

$$\left. \frac{\partial \ell(\beta)}{\partial \beta} \right|_{\beta = \hat{\beta}_{MLE}} = 0.$$

Substituting $\beta = \hat{\beta}_{MLE}$ into the score function gives:

$$\sum_{i=1}^{N} X_i \left\{ Y_i - \exp(X_i^\top \hat{\beta}_{MLE}) \right\} = 0.$$

Step 5: Dividing both sides of the equation by N yields:

$$\frac{1}{N} \sum_{i=1}^{N} X_i \left\{ Y_i - \exp(X_i^\top \hat{\beta}_{MLE}) \right\} = 0.$$

Thus, we have shown that the MLE $\hat{\beta}_{MLE}$ satisfies the given equation.

3(g)

```
residual_sum <- colMeans(model.matrix(forward_model_bic, politics_data) * (politics_data$Frequency - pr
print(residual_sum)
```

```
##   (Intercept)         Race2         Race3       Gender2        Party2
## -2.217752e-10 -1.291400e-10 -9.154455e-11 -1.094627e-10 -2.205454e-10
##  Race2:Party2  Race3:Party2
## -1.287764e-10 -9.123487e-11
```

The parameter estimates satisfy the MLE condition (1).

3(h)

```
vcov_matrix_mle <- vcov(forward_model_bic)
print(vcov_matrix_mle)
```

```
##                (Intercept)         Race2        Race3       Gender2
## (Intercept)   0.005037793 -4.000000e-03 -4.00000e-03 -1.862195e-03
## Race2        -0.004000000  9.882353e-03  4.00000e-03 -6.062911e-19
## Race3        -0.004000000  4.000000e-03  1.10922e-02 -1.305810e-19
## Gender2      -0.001862195 -6.062911e-19 -1.30581e-19  3.341483e-03
## Party2       -0.004000000  4.000000e-03  4.00000e-03 -1.106681e-19
## Race2:Party2  0.004000000 -9.882353e-03 -4.00000e-03  6.241446e-19
## Race3:Party2  0.004000000 -4.000000e-03 -1.10922e-02 -1.134363e-20
##                     Party2  Race2:Party2  Race3:Party2
## (Intercept)  -4.000000e-03  4.000000e-03  4.000000e-03
## Race2         4.000000e-03 -9.882353e-03 -4.000000e-03
## Race3         4.000000e-03 -4.000000e-03 -1.109220e-02
## Gender2      -1.106681e-19  6.241446e-19 -1.134363e-20
## Party2        7.968254e-03 -7.968254e-03 -7.968254e-03
## Race2:Party2 -7.968254e-03  1.801726e-02  7.968254e-03
## Race3:Party2 -7.968254e-03  7.968254e-03  2.131043e-02
```

The variance-covariance matrix of the MLE for the model selected through the forward selection procedure provides estimates of the variances and covariances of the model coefficients. For example, the variance of the intercept is 0.005, and the covariance between Race2 and Party2 is 0.004. This matrix quantifies the uncertainty in the parameter estimates.

3(i)

```r
lasso_coordinate_descent <- function(X, y, lambda, max_iter = 1000, tol = 1e-6) {
  n <- nrow(X)
  p <- ncol(X)
  beta <- rep(0, p)
  for (iter in 1:max_iter) {
    beta_old <- beta
    for (j in 1:p) {
      residual <- y - X %*% beta + beta[j] * X[, j]
      rho <- sum(X[, j] * residual)
      if (rho < -lambda) {
        beta[j] <- (rho + lambda) / sum(X[, j]^2)
      } else if (rho > lambda) {
        beta[j] <- (rho - lambda) / sum(X[, j]^2)
      } else {
        beta[j] <- 0
      }
    }
    if (sum(abs(beta - beta_old)) < tol) {
      break
    }
  }
  return(beta)
}

X <- model.matrix(~ Race + Gender + Party, data = politics_data)
y <- politics_data$Frequency

lambda <- 0.1
beta_lasso <- lasso_coordinate_descent(X, y, lambda)
```

```r
predicted_values_lasso <- X %*% beta_lasso

residuals_lasso <- y - predicted_values_lasso
residual_sum_lasso <- colMeans(sweep(X, 1, residuals_lasso, `*`))

print(residual_sum_lasso)
```

```
##  (Intercept)        Race2        Race3      Gender2       Party2
##  0.008333591 -0.008333269 -0.008333269  0.008333398  0.008333333
```

The Lasso model estimates do not satisfy the MLE condition (1) because Lasso regularization biases the coefficients, preventing the gradients from being close to zero, which is required for MLE.

3(j)

```r
library(glmnet)

set.seed(123)
X <- model.matrix(~ Race + Gender + Party, data = politics_data)[, -1]
y <- politics_data$Frequency
lambda <- 0.1

bootstrap_coefs <- replicate(1000, {
  idx <- sample(seq_along(y), replace = TRUE)
  coef(glmnet(X[idx, ], y[idx], alpha = 1, lambda = lambda))[-1]
})

print(var(t(bootstrap_coefs)))
```

```
##            [,1]       [,2]       [,3]       [,4]
## [1,] 136.636652  61.420729  -8.456628   2.950209
## [2,]  61.420729 122.770663  -6.996566   1.541863
## [3,]  -8.456628  -6.996566  53.084333 -20.710689
## [4,]   2.950209   1.541863 -20.710689  73.937090
```