PHP2514_Basso_HW4_2021

December 12, 2021

1 PHP2514: Applied Generalized Linear Models

1.1 Homework 4

Antonella Basso

1.1.1 Question 1:

The "survey.csv" file contains data from a customer satisfaction survey comparing three different brands of the same product. Use this dataset to answer the following questions:

a) You want to test whether there is a strong association between product brand, level of contact with other costumers interested in the same product, and customer satisfaction.

- i. What GLM would you use to answer this question?
- ii. Using the GLM type you specified perform a model selection procedure to find the model that best fits the data.
- iii. Comment on the overall fit of the "best" model.
- iv. What is your conclusion regarding the primary research question based on the results from the "best" model?

b) Suppose that you want to determine how contact among customers interested in the same product may affect the level of satisfaction.

- i. What GLM would you use to answer this question?
- ii. Using the GLM type you specified perform a model selection procedure to find the model that best fits the data.
- iii. Comment on the overall fit of the "best" model.
- iv. Interpret the regression coefficients of the "best" model.

c) Suppose that you want to determine how contact among customers interested in the same product may affect the brand preference.

- i. What GLM would you use to answer this question?
- ii. Using the GLM type you specified perform a model selection procedure to find the model that best fits the data.
- iii. Comment on the overall fit of the "best" model.

- iv. Interpret the regression coefficients of the "best" model.

[1]:
```
#installing tidyverse packages
suppressMessages(install.packages("tidyverse"))
suppressMessages(library(tidyverse))
```

```
Warning message in system("timedatectl", intern = TRUE):
"running command 'timedatectl' had status 1"
```

[2]:
```
#installing ordinal and multinomial regression packages
suppressMessages(library(MASS))
suppressMessages(library(nnet))
suppressMessages(install.packages("VGAM"))
suppressMessages(library(VGAM))
suppressMessages(install.packages("brant"))
suppressMessages(library(brant))
```

[3]:
```
#installing survival analysis packages
suppressMessages(install.packages("survminer"))
suppressMessages(library(survminer))
suppressMessages(install.packages("survival"))
suppressMessages(library(survival))
suppressMessages(install.packages("flexsurv"))
suppressMessages(library(flexsurv))
```

[4]:
```
#importing "survey" data
survey <- read.csv("/home/jovyan/AGLM/HW4/survey.csv")
survey
```

A data.frame: 18 × 4

brand <int>	satisfaction <chr>	contact <chr>	frequency <int>
1	low	low	65
1	low	high	34
1	medium	low	54
1	medium	high	47
1	high	low	100
1	high	high	100
3	low	low	130
3	low	high	141
3	medium	low	76
3	medium	high	116
3	high	low	111
3	high	high	191
2	low	low	67
2	low	high	130
2	medium	low	48
2	medium	high	105
2	high	low	62
2	high	high	104

a) Variable Association Research Question: Is there a strong association between product brand, level of contact with other costumers interested in the same product, and customer satisfaction?

GLM: Log-linear Model for Contingency Tables

To answer this research question, it is best to employ a log-linear model to the data, since we are interested only in checking for association between variables.

"Best" Model: Homogeneous Assosiation (`ll_glm2`)

Based on the model selection procedure and model comparisons (via LRT and AIC scores), it is safe to assume that the variables in the data assume a homogeneous association. Thus, the homogeneous association model best describes the data and holds that every variable pair is associated.

Model Fit:

Obtaining a correlation coefficient of ≈ 0.9 indicates almost perfect linearity between our observed and fitted values and suggests that the homogeneous association model provides a good fit to the data. Moreover, the residual plots below indicate that residuals are roughly normally distributed, and that there are no outliers or influential points. Therefore, the model provides an adequate fit for the data.

Conclusion:

Given the fit of this model, there appears to be significant association between product brand, level of contact with other costumers interested in the same product, and customer satisfaction. Specifically, in the homogeneous association model providing the best posible fit to the data, we may deduce that this relationship between variables is such that each pair of variables is conditionally independent of the third. That is, all variables are strongly associated to one another, but not simultaneously.

[5]:
```r
#LOG-LINEAR MODEL - Model Selection (Backward Elimination)

#saturated model
ll_glm1 <- glm(frequency ~ brand + satisfaction + contact + #main effects
               brand*satisfaction + brand*contact + satisfaction*contact +␣
 ↪#two-way interactions
               brand*satisfaction*contact, #three-way interaction
               family=poisson, data=survey)

#homogeneous assosiation  <-  best model
ll_glm2 <- glm(frequency ~ brand + satisfaction + contact + #main effects
               brand*satisfaction + brand*contact + satisfaction*contact,␣
 ↪#two-way interactions
               family=poisson, data=survey)

#conditional independence (on brand)
ll_glm3 <- glm(frequency ~ brand + satisfaction + contact + #main effects
               brand*satisfaction + brand*contact, #two-way interactions
               family=poisson, data=survey)

#joint independence (between brand and contact)
ll_glm4 <- glm(frequency ~ brand + satisfaction + contact + #main effects
               brand*contact, #two-way interaction
               family=poisson, data=survey)

#mutual independence (additive model)
ll_glm5 <- glm(frequency ~ brand + satisfaction + contact, #main effects
               family=poisson, data=survey)
#null model
ll_glm0 <- glm(frequency ~ 1, family=poisson, data=survey)

summary(ll_glm2)
```

```
Call:
glm(formula = frequency ~ brand + satisfaction + contact + brand *
    satisfaction + brand * contact + satisfaction * contact,
    family = poisson, data = survey)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2002  -1.6819   0.6849   1.0680   3.8263

Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)                 4.19139    0.13141  31.895  < 2e-16 ***
brand                       0.32680    0.05483   5.960 2.52e-09 ***
```

```
satisfactionlow               -0.84198    0.18439  -4.566 4.96e-06 ***
satisfactionmedium            -0.56284    0.18924  -2.974 0.002938 **
contactlow                     0.12445    0.15430   0.807 0.419902
brand:satisfactionlow          0.25453    0.07283   3.495 0.000474 ***
brand:satisfactionmedium       0.07815    0.07662   1.020 0.307718
brand:contactlow              -0.23082    0.06230  -3.705 0.000211 ***
satisfactionlow:contactlow     0.25431    0.11623   2.188 0.028676 *
satisfactionmedium:contactlow -0.02816    0.12523  -0.225 0.822097
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 294.477  on 17  degrees of freedom
Residual deviance:  60.919  on  8  degrees of freedom
AIC: 194.02

Number of Fisher Scoring iterations: 4
```

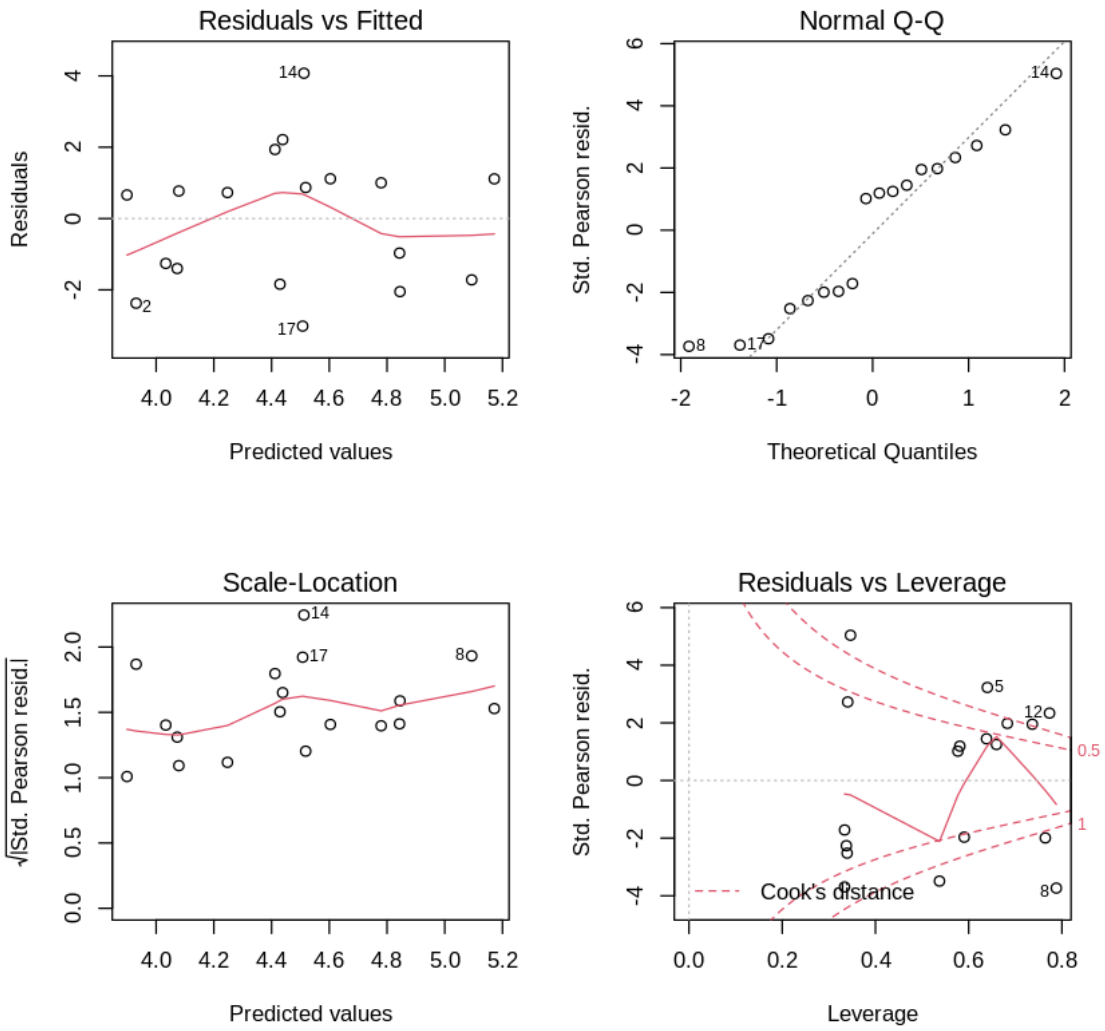
[6]:
```r
#LRT: comparing models

#mutual independence better than null (p<0.05)
#anova(ll_glm0, ll_glm5, test="LRT")
#joint independence better than mutual independence (p<0.05)
#anova(ll_glm5, ll_glm4, test="LRT")
#conditional independence better than joint independence (p<0.05)
#anova(ll_glm4, ll_glm3, test="LRT")
#homogeneous association better than conditional independence (p<0.05)
#anova(ll_glm3, ll_glm2, test="LRT")
#homogeneous association equal to saturated model (p>0.05)
anova(ll_glm2, ll_glm1, test="LRT")
#therefore, homogeneous association model fits the data best
```

A anova: 2 × 5

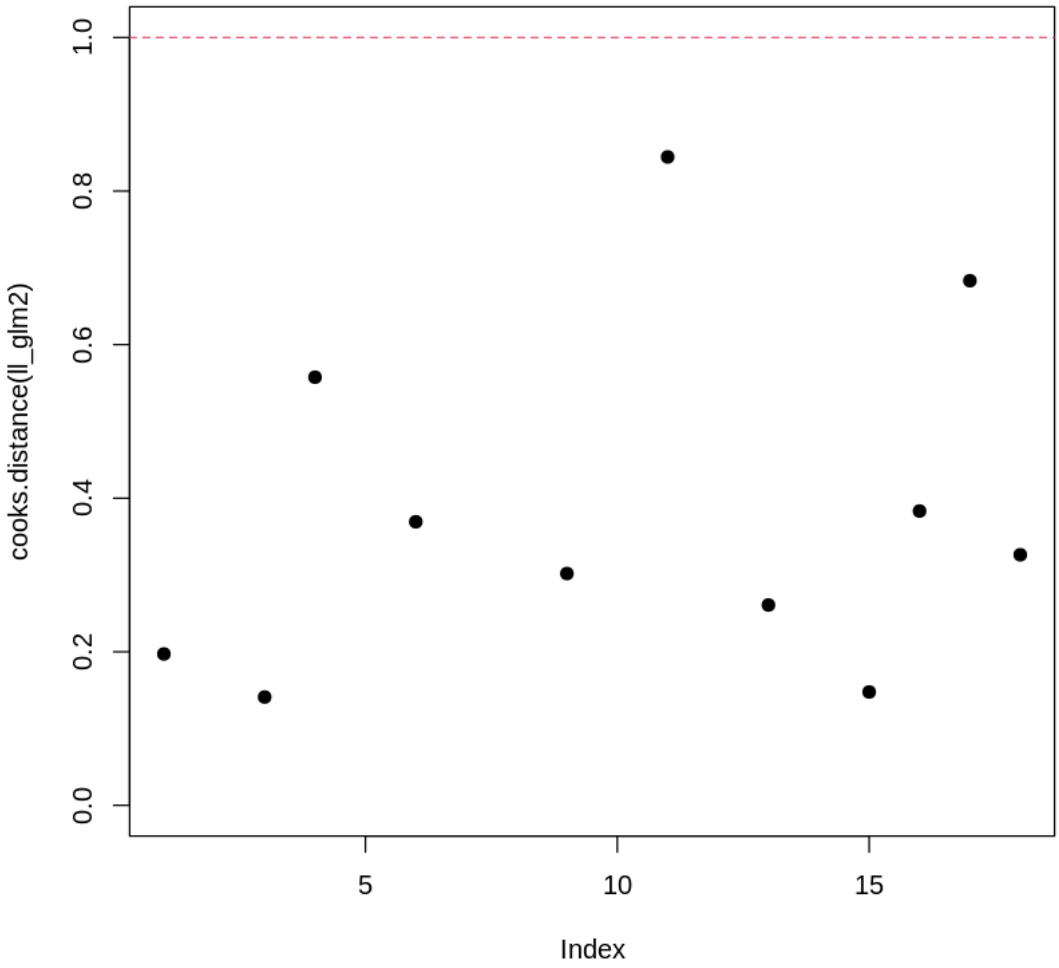
	Resid. Df <dbl>	Resid. Dev <dbl>	Df <dbl>	Deviance <dbl>	Pr(>Chi) <dbl>
1	8	60.91883	NA	NA	NA
2	6	60.19674	2	0.722091	0.6969473

[7]:
```r
#MODEL FIT:

#Correlation Coefficient
#strong positive linear relationship between observed and fitted values
cor(fitted(ll_glm2), survey$frequency)

#Residual Plots
par(mfrow=c(2,2))
plot(ll_glm2)
```

0.893514693170665



[8]:
```
#Cook's Distance

#no values exceed 1 -> thus, there are no influential points
plot(cooks.distance(ll_glm2), ylim=c(0,1), pch=19)
abline(h=1, lty=2, col=2)
```



b) Evaluating Effects **Research Question**: How does contact among customers interested in the same product affect the level of satisfaction?

GLM: Ordinal Regression (Proportional Odds Model)

As our goal is to evaluate the effects of an ordinal categorical variable on another, an ordinal regression model is best suited to answer the research question. Moreover, the results from the Brant test show that the proportional odds assumption holds, and thus, the proportional odds model is the preferred ordinal regression model to use on the data.

"Best" Model:

Based on the model selection procedure (backward elimination) and corresponding chi-square tests, the saturated proportional odds regression model, taking the following form, was determined to best fit the data.

For satisfaction level response groups "low" (1), "medium" (2), and "high" (3), let X_C be the contact random variable of two groups ("low" and "high") with "high" as the reference group; let X_B be the brand random variable of three groups ("1", "2", and "3") with "1" as the reference group; and let β_{0j} be the varying intercepts for each model $j = 1, 2$:

$$Y = logit(P(Y \leq j)) = log(\frac{P(Y \leq j)}{P(Y > j)}) = X\beta_j$$

$$= \beta_{0j} - 0.3193X_{C:\text{low}} + 0.6896X_{B:2} + 0.52318X_{B:3} + 0.3899X_{C:\text{low}}X_{B:2} + 0.0530X_{C:\text{low}}X_{B:3}$$

Model Fit:

To assess the overall fit of this model, Figures 1.1-1.3 below were created to illustrate how close the predicted or "fitted" values (in this case, proportions of being in any combination of covariate categories) came to the ones observed in the data. In Figures 1.1 and 1.2, we observe that the proportions fitted by the model are relatively close to those observed. Moreover, Figure 1.3 illustrates that there is high positive correlation between observed and predicted values, indicating that this model provides an adequate fit for the data.

Coefficient Interpretation:

`(Intercept)1: -1.13653`

`(Intercept)2: -0.02074`

Given that the proportional odds assumption holds, each constant slope coefficient in the model gives the change in log-odds of having a satisfaction level that is low or low/medium as opposed to high (that is, being at or below the cutoff Y_j) given that the corresponding predictor is true (has a value of 1). Moreover, since the predictor reference groups are "high" (for contact level) and "1" (for brand), the intercept coefficients in both models represent the log-odds of having a satisfaction level that is low (model 1) or low/medium (model 2) as opposed to high, when customer contact level is high and brand 1 is the preferred brand. Similarly, exponentiating them yields these odds rather than log-odds. For a specific example, given that there is low customer contact and brand 2 is the preferred brand, the odds of low satisfaction level as opposed to medium/high satisfaction level is `exp(- 1.13653 - 0.3193 + 0.6896 + 0.3899)` ≈ 0.6864. Specifically, the odds of having a low satisfaction level as opposed to medium/high satisfaction level when customer contact level is low and brand 2 is preferred is (`exp(- 0.3193 + 0.6896 + 0.3899)` ≈ 2.1388) times that for which customer contact level is high and brand 1 is preferred (`exp(- 1.13653)` ≈ 0.3209). Similar interpretations can be made for coefficients in the second model (which give the same odds, but for low/medium satisfaction levels as opposed to high).

[9]:
```
#Factorizing and Ordering Variables

#satisfaction
survey$satisfaction_ord <- factor(survey$satisfaction, levels=c("low",␣
 ↪"medium", "high"), ordered=TRUE) #ordinal
#levels(survey$satisfaction_ord)

#contact
```

```r
survey$contact_ord <- factor(survey$contact, levels=c("low", "high"),␣
↪ordered=TRUE) #ordial
#levels(survey$contact_ord)

#brand
survey$brand <- as.factor(survey$brand) #nominal

survey <- survey %>% arrange(brand, satisfaction)
```

[10]:
```r
exp(- 1.13653 - 0.31927 + 0.68962 + 0.38990)
exp(- 0.31927 + 0.68962 + 0.38990)
exp(- 1.13653)
exp(- 1.13653)*exp(- 0.31927 + 0.68962 + 0.38990)
```

0.686410111301349

2.13881085637864

0.320930721505479

0.686410111301349

[11]:
```r
#Checking Proportional Odds Assumption

po <- polr(satisfaction_ord ~ contact_ord*brand, data=survey, weights=frequency)
brant(po) #proportional odds assumption holds
```

```
-------------------------------------------------------
Test for                X2      df      probability
-------------------------------------------------------
Omnibus                 0       5       1
contact_ord.L           0       1       1
brand2                  0       1       1
brand3                  0       1       1
contact_ord.L:brand2    0       1       1
contact_ord.L:brand3    0       1       1
-------------------------------------------------------

H0: Parallel Regression Assumption holds
```

[12]:
```r
#PROPORTIONAL ODDS MODEL (for research question)

#saturated model <- best model
po_glm1 <- vglm(satisfaction_ord ~ contact_ord*brand,␣
↪family=cumulative(parallel=TRUE), data=survey, weights=frequency)
summary(po_glm1)
#additive model
po_glm2 <- vglm(satisfaction_ord ~ contact_ord+brand,␣
↪family=cumulative(parallel=TRUE), data=survey, weights=frequency)
```

```
#most significant main effect
po_glm3 <- vglm(satisfaction_ord ~ contact_ord,␣
 ↪family=cumulative(parallel=TRUE), data=survey, weights=frequency)
#null model
po_glm0 <- vglm(satisfaction_ord ~ 1, family=cumulative(parallel=TRUE),␣
 ↪data=survey, weights=frequency)
```

```
Call:
vglm(formula = satisfaction_ord ~ contact_ord * brand, family =␣
 ↪cumulative(parallel = TRUE),
    data = survey, weights = frequency)

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept):1        -1.13653    0.10070 -11.286  < 2e-16 ***
(Intercept):2        -0.02074    0.09675  -0.214   0.8303
contact_ord.L        -0.31927    0.13509  -2.363   0.0181 *
brand2                0.68962    0.12856   5.364 8.12e-08 ***
brand3                0.52318    0.11733   4.459 8.24e-06 ***
contact_ord.L:brand2  0.38990    0.18135   2.150   0.0316 *
contact_ord.L:brand3  0.05298    0.16561   0.320   0.7490
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Names of linear predictors: logitlink(P[Y<=1]), logitlink(P[Y<=2])

Residual deviance: 3604.091 on 29 degrees of freedom

Log-likelihood: -1802.045 on 29 degrees of freedom

Number of Fisher scoring iterations: 3

No Hauck-Donner effect found in any of the estimates


Exponentiated coefficients:
       contact_ord.L               brand2               brand3
           0.7266776            1.9929655            1.6873780
contact_ord.L:brand2 contact_ord.L:brand3
           1.4768365            1.0544103
```

[13]:
```
#LRT/Chi-Square Tests

#lrtest(po_glm0, po_glm3) #main effect model equal to null model
#lrtest(po_glm3, po_glm2) #additive model better than main effect model
lrtest(po_glm2, po_glm1) #saturated model better than additive model
```

```
#thus, interaction term is needed and `po_glm1` is the "best" model
```

```
Likelihood ratio test

Model 1: satisfaction_ord ~ contact_ord + brand
Model 2: satisfaction_ord ~ contact_ord * brand
  #Df  LogLik Df  Chisq Pr(>Chisq)
1  31 -1805.1
2  29 -1802.0 -2 6.1955    0.04515 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

[14]:
```
#Model Fit

#Observed Proportions
observed <- survey %>%
             group_by(brand, contact_ord) %>%
             summarize(brand=brand,
                       satisfaction_ord=satisfaction_ord,
                       frequency=frequency,
                       total=sum(frequency)) %>%
             ungroup() %>%
             mutate(observed=frequency/total) %>%
             arrange(brand, satisfaction_ord)

#Fitted Proportions
fitted=fitted(po_glm1)

#Observed & Fitted Proportions
po_props <- cbind(observed, fitted) %>%
            mutate(fitted=case_when(satisfaction_ord == "low" ~ low,
                                    satisfaction_ord == "medium" ~ medium,
                                    satisfaction_ord == "high" ~ high))

#Observed & Fitted Proportions for Low Contact Level (Fig. 1.1)
low <- po_props %>% filter(contact_ord=="low")
ggplot(low, aes(x=brand, y=observed)) +
       geom_line(aes(group=satisfaction_ord, color=satisfaction_ord)) +
       geom_point(aes(color=satisfaction_ord, shape="observed"), size=4) +
       geom_point(aes(y=fitted, color=satisfaction_ord, shape="fitted"),␣
 ↪size=4) +
       scale_color_manual(name="Satisfaction Level", values=c("green3",␣
 ↪"purple", "orange")) +
       scale_shape_manual(name="Shape", values=c(18,20)) +
       labs(x="Brand", y="Proportion", title="Figure 1.1: Observed & Fitted␣
 ↪Proportions (Contact Level: Low)")
```

```r
#Observed & Fitted Proportions for High Contact Level (Fig. 1.2)
high <- po_props %>% filter(contact_ord=="high")
ggplot(high, aes(x=brand, y=observed)) +
        geom_line(aes(group=satisfaction_ord, color=satisfaction_ord)) +
        geom_point(aes(color=satisfaction_ord, shape="observed"), size=4) +
        geom_point(aes(y=fitted, color=satisfaction_ord, shape="fitted"),␣
 ↪size=4) +
        scale_color_manual(name="Satisfaction Level", values=c("green3",␣
 ↪"purple", "orange")) +
        scale_shape_manual(name="Shape", values=c(18,20)) +
        labs(x="Brand", y="Proportion", title="Figure 1.2: Observed & Fitted␣
 ↪Proportions (Contact Level: High)")

#Observed vs. Fitted Proportions (Fig. 1.6)
plot(po_props$observed, po_props$fitted,
     xlim=c(0.1, 0.5),
     ylim=c(0.1, 0.5),
     xlab="Observed",
     ylab="Fitted",
     main="Figure 1.3: Observed vs. Fitted Proportions") +
abline(coef=c(0,1), lty=2, col=2)
```

```
`summarise()` has grouped output by 'brand', 'contact_ord'. You can override
using the `.groups` argument.
```

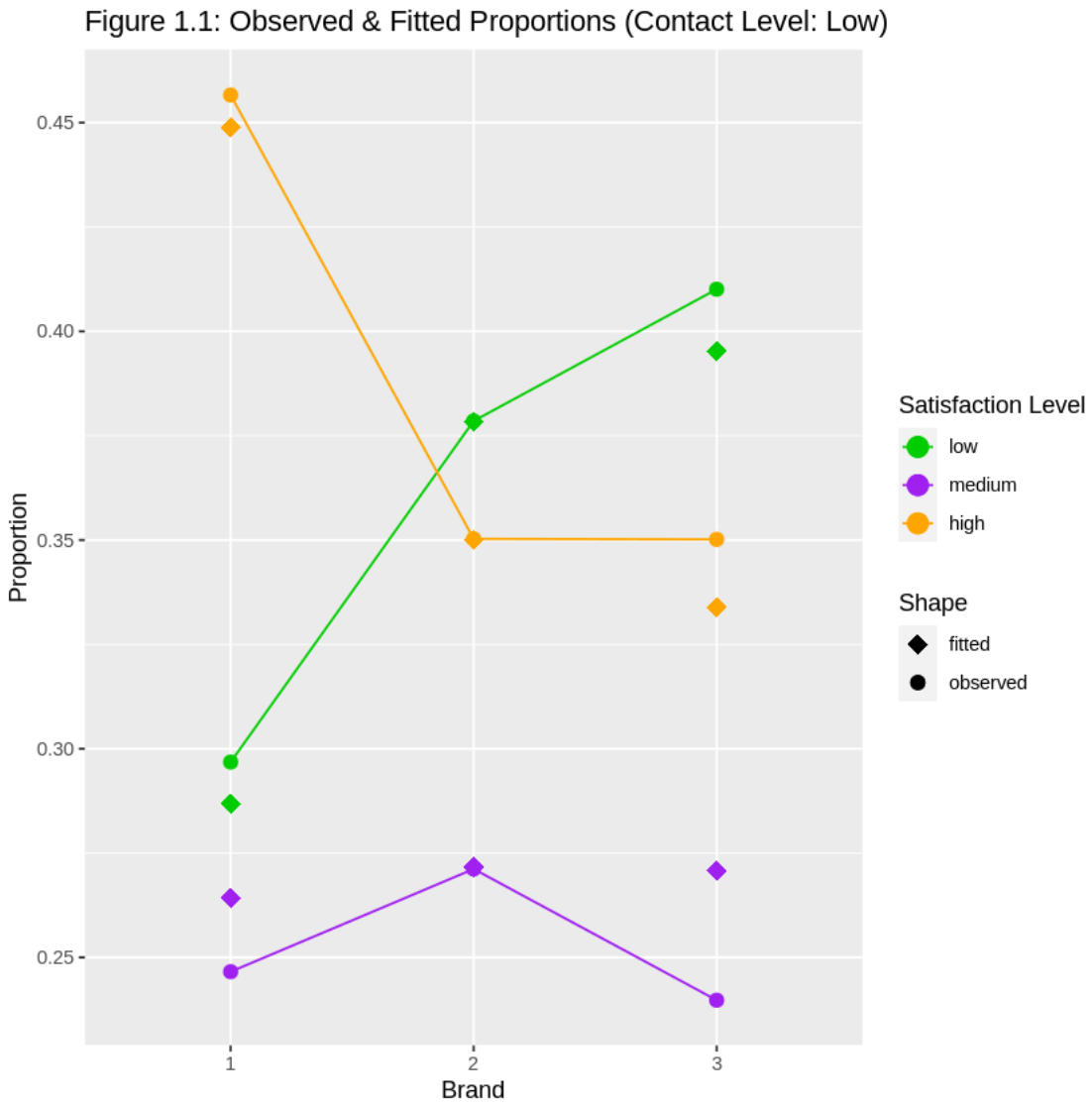


Figure 1.1: Observed & Fitted Proportions (Contact Level: Low)
Proportion
0.45
0.40
0.35
0.30
0.25
1
2
3
Brand
Satisfaction Level
low
medium
high
Shape
fitted
observed

Figure 1.2: Observed & Fitted Proportions (Contact Level: High)

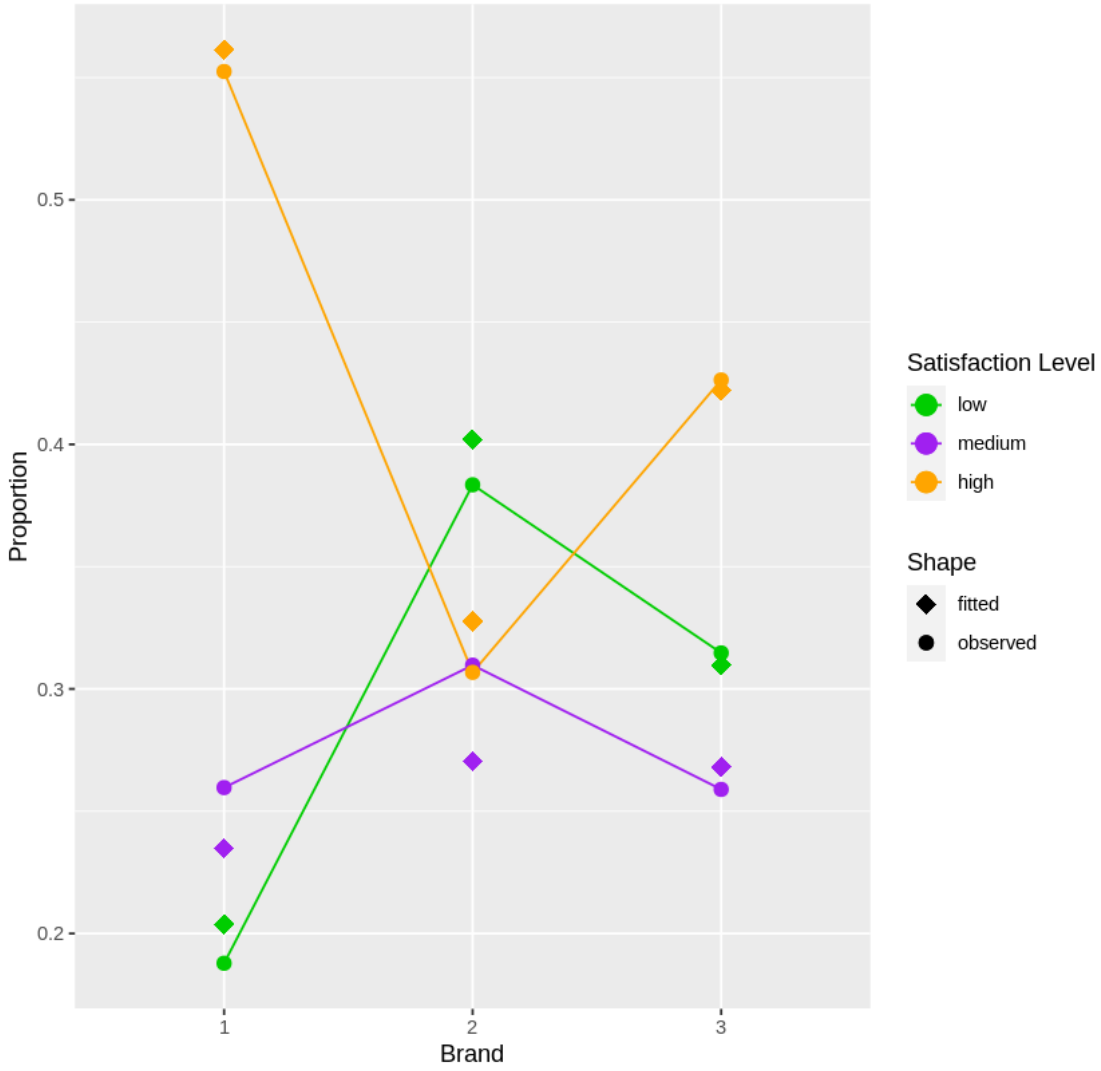
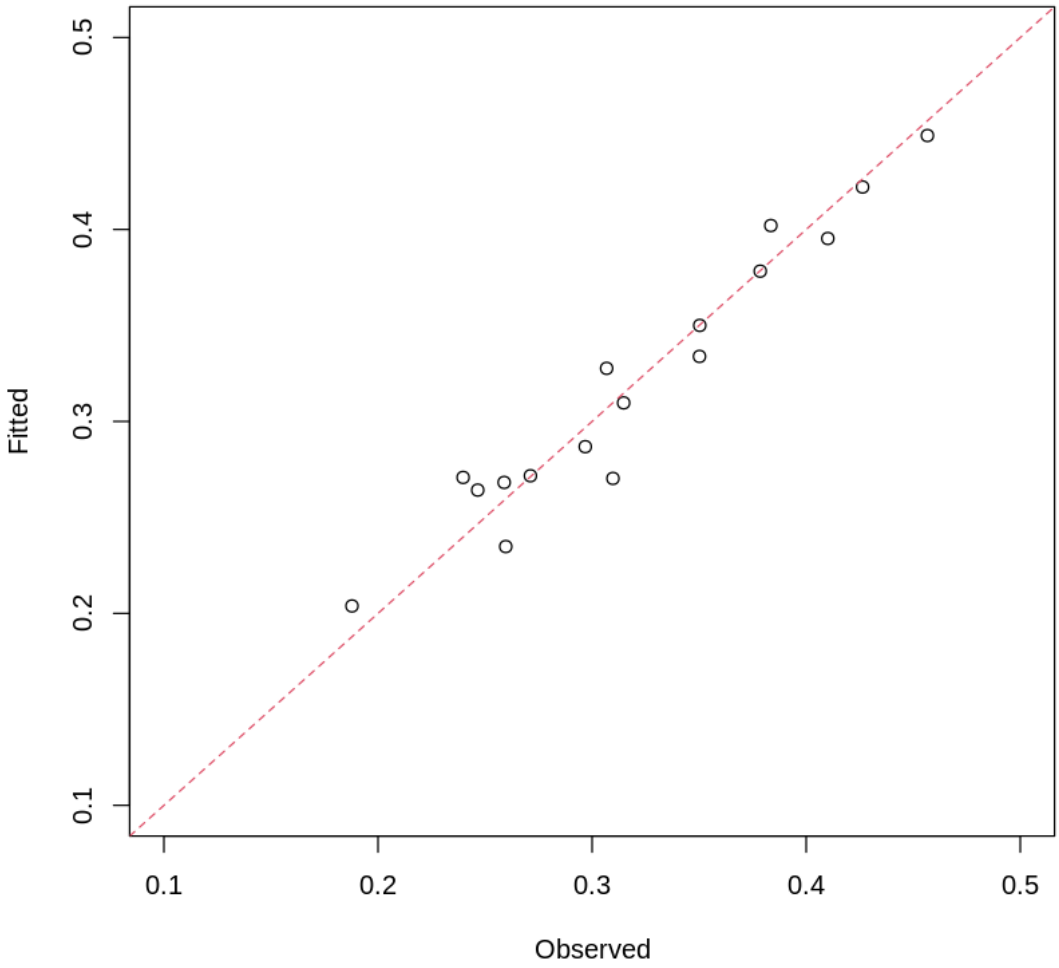


Figure 1.3: Observed vs. Fitted Proportions



c) Evaluating Effects Research Question:

How does contact among customers interested in the same product affect brand preference?

GLM: Multinomial Regression

As our goal is to evaluate the effects of an ordinal variable on a nominal variable, a multinomial regression model is best suited to answer the research question.

"Best" Model:

Based on the model selection procedure (backward elimination) and corresponding chi-square tests, the multinomial regression model that best fits the data has the following additive form:

$$Y = log(\frac{P(Y=j)}{P(Y=\text{brand 1})}) = X\beta_j$$

$$= \beta_{0j} + \beta_{1j}X_{C:\text{low}} + \beta_{2j}X_{S:\text{low}} + \beta_{3j}X_{S:\text{medium}}$$

for each of the brands $j \in \{2, 3\}$ (resulting in two models) not including the reference group, brand 1 (to which the others are being compared).

Model Fit:

To assess the overall fit of this model, Figures 1.4-1.6 below were created to illustrate how close the predicted or "fitted" values (in this case, proportions of being in any combination of covariate categories) came to the ones observed in the data. In Figures 1.4 and 1.5, we observe that the proportions fitted by the model are relatively close to those actually observed. Moreover, Figure 1.6 illustrates the strong linear relationship between observed and predicted values, indicating that the model provides a good fit for the data.

Coefficient Interpretation:

Model 1: `(Intercept)1: 0.2573 contact_ord.L: 0.6298 satisfaction_ord.L: -0.6687 satisfaction_ord.Q: -0.1108`

Model 2: `(Intercept)2: 0.6780 contact_ord.L: 0.4062 satisfaction_ord.L: -0.4537 satisfaction_ord.Q: -0.0702`

Given that the reference groups for both contact and satisfaction levels is "high", the intercept coefficients in both models represent the log-odds of preferring brands 2 and 3 over brand 1, respetively, when contact and satisfaction levels are "high". Exponentiating them, similarly, yields these odds rather than log-odds. The following coefficents, when exponentiated, give the marginal change in such odds for low contact, low satisfaction, and medium satisfaction levels, respectively. That is for example, `exp(contact_ord.L: 0.6298)` ≈ 1.88 in the first model above, tells us that the odds of preferring brand 2 over brand 1 is is multiplied by a factor of roughly 1.88 for those who have low (as opposed to high) contact level, assumming that their satisfaction level is high. Likewise, `exp(satisfaction_ord.L: -0.6687)` ≈ 0.512 tells us that the odds of preferring brand 2 over brand 1 is multiplied by a factor of about 0.512 for those who have low (as opposed to high or medium) satisfaction level, assumming that their contact level is high. A similar interpretation follows for a medium satisfcation level, `satisfaction_ord.Q: -0.1108`, in the first model, and for all such coefficients in the second model (which give the same odds, but for preference of brand 3 over brand 1 instead).

[15]:
```r
#MULTINOMIAL REGRESSION MODEL

#saturated model
mr1 <- multinom(brand ~ contact_ord*satisfaction_ord, data=survey,␣
 ↪weights=frequency) #saturated model
#summary(mr1)
mr2 <- multinom(brand ~ contact_ord + satisfaction_ord, data=survey,␣
 ↪weights=frequency) #additive model <- best model
summary(mr2)
```

```
mr3 <- multinom(brand ~ contact_ord, data=survey, weights=frequency) #most␣
 ↪significant main effect
#summary(mr3)
mr0 <- multinom(brand ~ 1, data=survey, weights=frequency) #null model
#summary(mr0)
```

```
# weights:  21 (12 variable)
initial  value 1846.767257
iter  10 value 1753.167036
final  value 1743.836217
converged
# weights:  15 (8 variable)
initial  value 1846.767257
iter  10 value 1750.076200
final  value 1747.282731
converged
```

```
Call:
multinom(formula = brand ~ contact_ord + satisfaction_ord, data = survey,
    weights = frequency)

Coefficients:
  (Intercept) contact_ord.L satisfaction_ord.L satisfaction_ord.Q
2   0.2572865     0.6297634         -0.6686554        -0.11082774
3   0.6779582     0.4061693         -0.4536753         0.07018268

Std. Errors:
  (Intercept) contact_ord.L satisfaction_ord.L satisfaction_ord.Q
2  0.07065536    0.09810450          0.1163085          0.1254377
3  0.06487477    0.08878074          0.1061206          0.1180804

Residual Deviance: 3494.565
AIC: 3510.565
```

```
# weights:  9 (4 variable)
initial  value 1846.767257
final  value 1766.418472
converged
# weights:  6 (2 variable)
initial  value 1846.767257
final  value 1785.947366
converged
```

[16]:
```
#Chi-Square Tests

#anova(mr0, mr3, test="Chisq") #main effect model better than null
#anova(mr3, mr2, test="Chisq") #additive model bettwer than main effect
anova(mr2, mr1, test="Chisq") #additive model equal to saturated model
```

```
#thus, additive model (mr2) is the "best" model
```

A Anova: 2 × 7

Model <chr>	Resid. df <dbl>	Resid. Dev <dbl>	Test <chr>	Df <dbl>	LR stat. <dbl>	Pr(Chi <dbl>
contact_ord + satisfaction_ord	28	3494.565		NA	NA	NA
contact_ord * satisfaction_ord	24	3487.672	1 vs 2	4	6.893028	0.14165

[17]:
```
#Model Fit

#Observed Proportions
observed2 <- survey %>%
             group_by(satisfaction_ord, contact_ord) %>%
             summarize(satisfaction_ord=satisfaction_ord,
                       brand=brand,
                       frequency=frequency,
                       total=sum(frequency)) %>%
             ungroup() %>%
             mutate(observed=frequency/total) %>%
             arrange(brand, satisfaction_ord)

#Fitted Proportions
fitted2=fitted(mr2)
colnames(fitted2)=c("one","two","three")

#Observed & Fitted Proportions
mn_props <- cbind(observed2, fitted2) %>%
            mutate(fitted=case_when(brand == "1" ~ one,
                                    brand == "2" ~ two,
                                    brand == "3" ~ three))

#Observed & Fitted Proportions for Low Contact Level (Fig. 1.4)
low2 <- mn_props %>% filter(contact_ord=="low")
ggplot(low2, aes(x=satisfaction_ord, y=observed)) +
        geom_line(aes(group=brand, color=brand)) +
        geom_point(aes(color=brand, shape="observed"), size=4) +
        geom_point(aes(y=fitted, color=brand, shape="fitted"), size=4) +
        scale_color_manual(name="Brand", values=c("green3", "purple",␣
 ↪"orange")) +
        scale_shape_manual(name="Shape", values=c(18,20)) +
        labs(x="Satisfaction Level", y="Proportion", title="Figure 1.4:␣
 ↪Observed & Fitted Proportions (Contact Level: Low)")

#Observed & Fitted Proportions for High Contact Level (Fig. 1.5)
high2 <- mn_props %>% filter(contact_ord=="high")
ggplot(high2, aes(x=satisfaction_ord, y=observed)) +
        geom_line(aes(group=brand, color=brand)) +
        geom_point(aes(color=brand, shape="observed"), size=4) +
```

```
       geom_point(aes(y=fitted, color=brand, shape="fitted"), size=4) +
       scale_color_manual(name="Brand", values=c("green3", "purple",␣
↪"orange")) +
       scale_shape_manual(name="Shape", values=c(18,20)) +
       labs(x="Satisfaction Level", y="Proportion", title="Figure 1.5:␣
↪Observed & Fitted Proportions (Contact Level: High)")

#Observed vs. Fitted Proportions (Fig. 1.6)
plot(mn_props$observed, mn_props$fitted,
     xlim=c(0.1, 0.5),
     ylim=c(0.1, 0.5),
     xlab="Observed",
     ylab="Fitted",
     main="Figure 1.6: Observed vs. Fitted Proportions") +
abline(coef=c(0,1), lty=2, col=2)
```

```
`summarise()` has grouped output by 'satisfaction_ord', 'contact_ord'. You can
override using the `.groups` argument.
```

Figure 1.4: Observed & Fitted Proportions (Contact Level: Low)

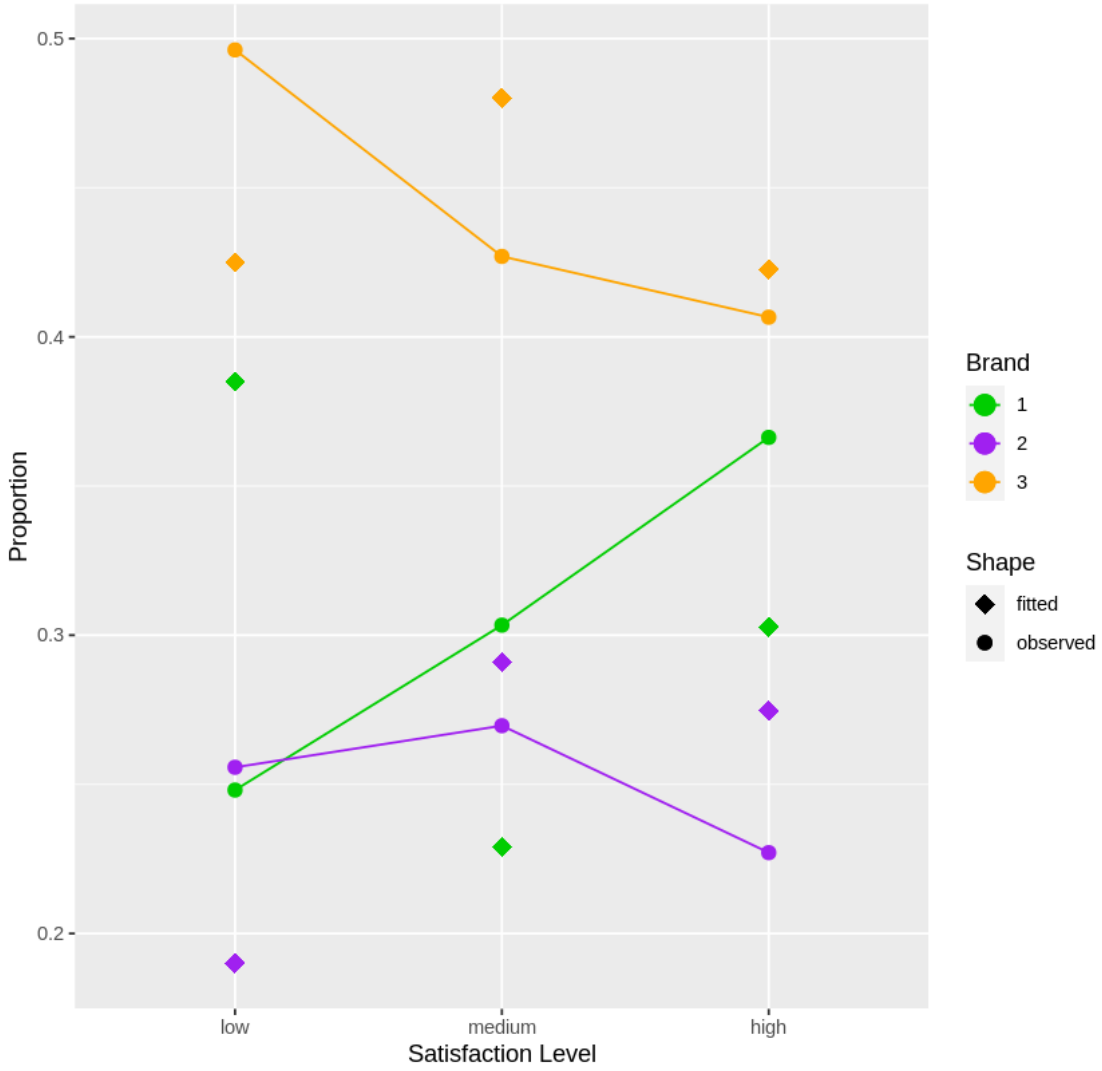


Figure 1.5: Observed & Fitted Proportions (Contact Level: High)

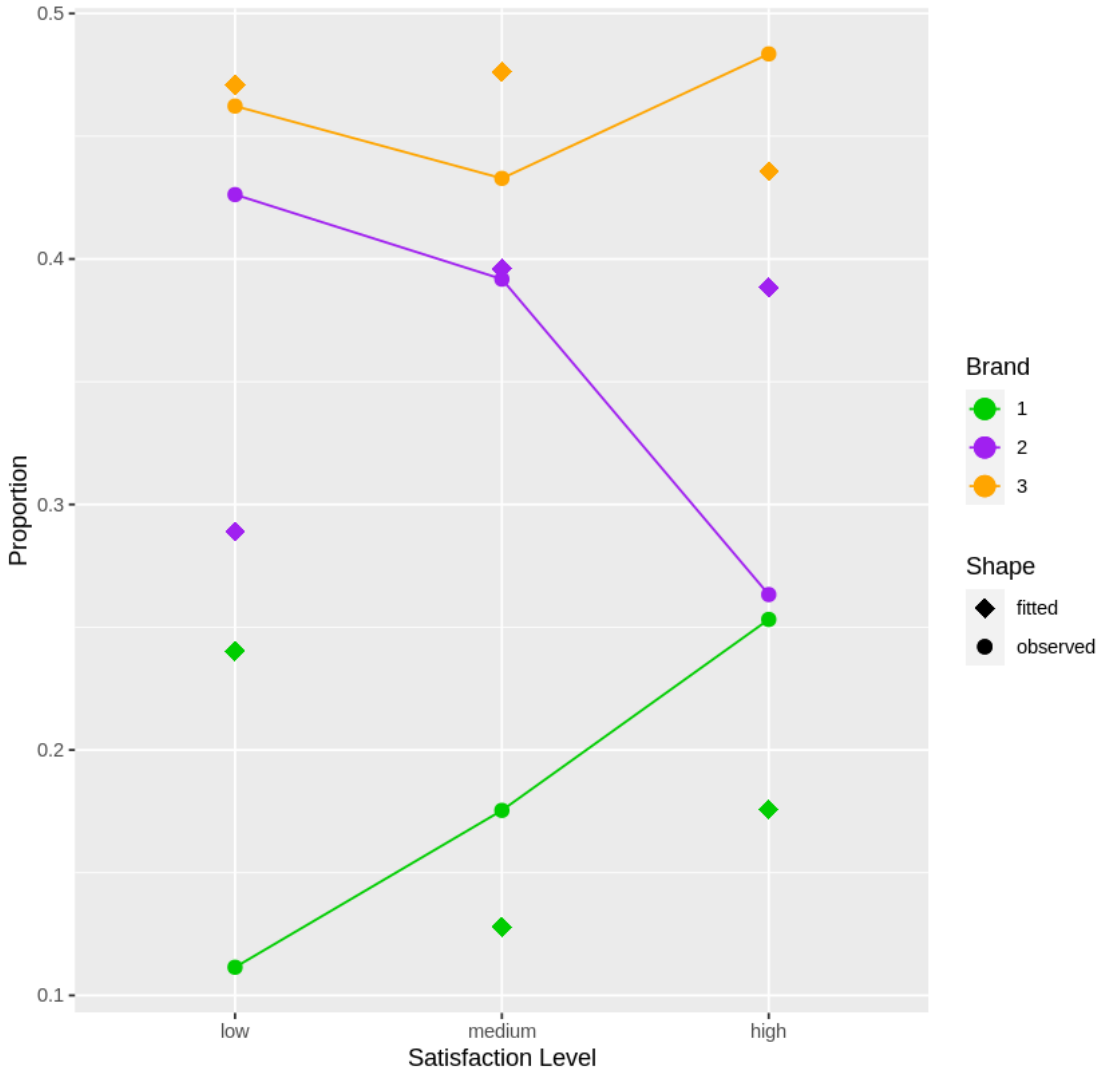
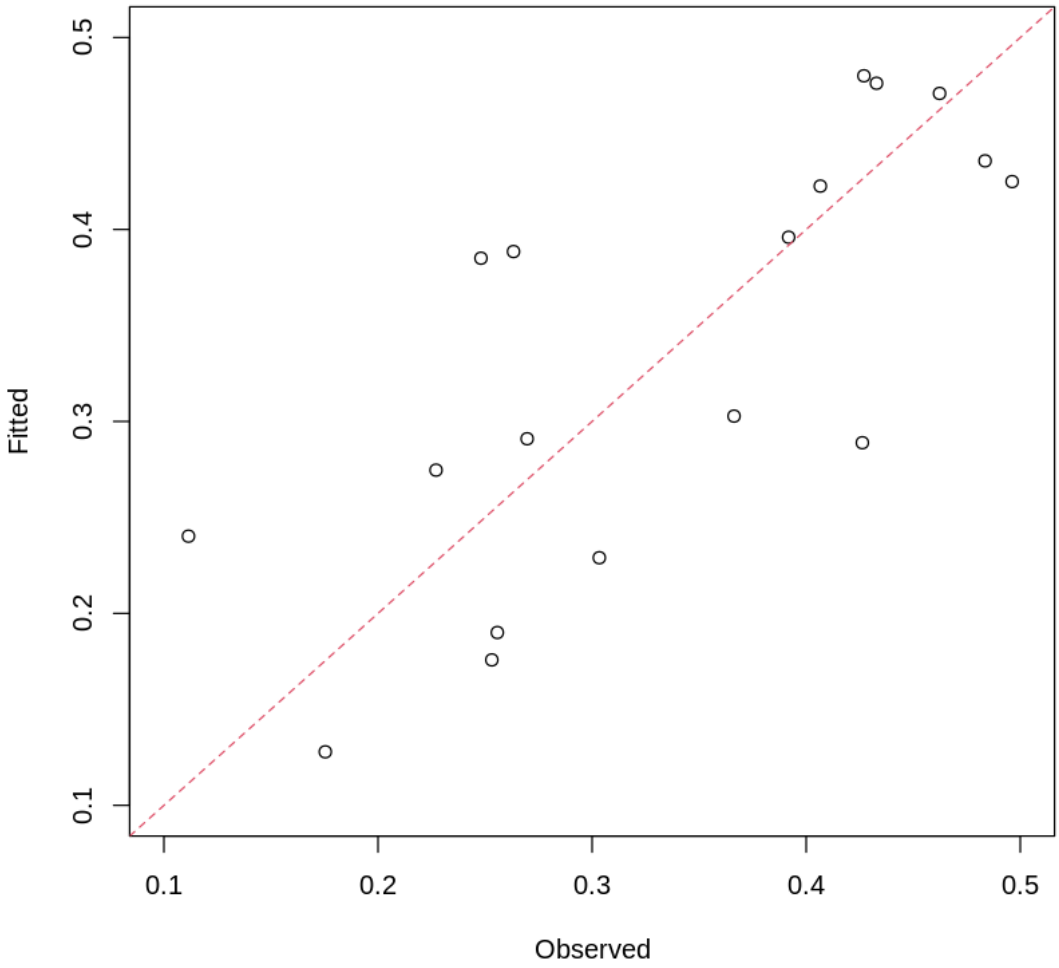


Figure 1.6: Observed vs. Fitted Proportions



1.1.2 Question 2:

The dataset "hepatitis.csv" contains data collected from a randomized control trial in which n patients with chronic active hepatitis were randomized to receive prednisolone or placebo. Data include the total observation time (in months), the treatment group, and an assessment of the status at the end of the follow-up period, for each patient in the study.

a) Conduct a comprehensive Exploratory Data Analysis (EDA) to inspect, understand and describe the information collected in this dataset. Use appropriate summary statistics and plots to present your results from the EDA.

b) Suppose that you want to evaluate the effectiveness of the treatment.

- i. Suggest an appropriate regression model that you could use for this purpose.

- ii. State the form and implement the model to fit the available data.
- iii. Test the model's assumptions and comment on the overall fit to the data.
- iv. Interpret the regression coefficients of this model.

[18]:
```
#importing "hepatitis" data
#adding status column for event of interest
hepatitis <- read.csv("/home/jovyan/AGLM/HW4/hepatitis.csv") %>%
             dplyr::select(time, censor, group) %>%
             mutate(status = case_when(censor == "censored" | censor == "loss␣
 ↪to follow-up" ~ 0, censor == "died" ~ 1))
#hepatitis
```

a) Exploratory Data Analysis (EDA) The variables in this dataset are as follows: - Outcome: $Y = [T, C]$ - time until event of interest (death) occurs - Covariate: X_1 - group (prednisolone, no treatment) - Categorical and binary with nominal scale

This EDA consists of: - Descriptive Statistics - Plots - Kaplan Meier Curves, Survival Descriptions, & Survival Comparison

Descriptive Statistics:

- Summary statistics for survival time ignoring and accounting for censoring (minimum value, 1st quartile, median, mean, 3rd quartile, maximum value)
- Summary statistics for survival time accounting for censoring for each treatment group (minimum value, 1st quartile, median, mean, 3rd quartile, maximum value)
- Average hazard rates ignoring and accounting for censoring

[19]:
```
#Descriptive Statistics

#summary statistics for survival time (ignoring censoring)
summary(hepatitis$time)
#summary statistics for survival time (accounting for censoring)
summary(hepatitis[which(hepatitis[,4]==1),1])
#summary statistics for survival time for those that died (with treatment)
summary(hepatitis[which(hepatitis[,4]==1 & hepatitis[,3]=="prednisolone"),1])
#summary statistics for survival time for those that died (without treatment)
summary(hepatitis[which(hepatitis[,4]==1 & hepatitis[,3]=="no treatment"),1])
#average hazard rate (ignoring censoring)
sum(hepatitis$status)/sum(hepatitis$time)
#average hazard rate (accounting for censoring)
sum(hepatitis$status)/sum(hepatitis[which(hepatitis[,4]==1),1])
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.00   31.25   80.00   87.14  143.50  182.00
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.00   11.00   40.00   51.26   69.50  168.00
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    2.0    33.0    89.0    80.0   119.5   168.0

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.00    9.25   30.50   31.50   44.25   71.00
```

0.00704225352112676

0.0195086705202312

Plots:

- Box Plot (Figure 2.1): Survival Time for Censored (0) and Uncensored (1) Observations
- Box Plot (Figure 2.2): Survival Time for Treatment and Nontreatment Groups
- Scatter Plot (Figure 2.3): Survival Time for Treatment and Nontreatment Groups by Status
- Bar Graph (Figure 2.4): Number of Deaths by Treatment Group

[20]:
```r
#Plots

#Box Plot (Fig. 2.1)
ggplot(hepatitis, aes(x=time, y=as.factor(status)), color=as.factor(status)) +
    geom_boxplot(color=c("darkgreen", "skyblue")) +
    labs(x="Survival Time", y="Status", title="Figure 2.1: Survival Time for␣
 ↪Censored (0) and Uncensored (1) Observations")

#Box Plot (Fig. 2.2)
ggplot(hepatitis, aes(x=time, y=group), color=group) +
    geom_boxplot(color=c("red", "orange")) +
    labs(x="Survival Time", y="Status", title="Figure 2.2: Survival Time for␣
 ↪Treatment and Nontreatment Groups")

#Scatter Plot (Fig. 2.3)
ggplot(hepatitis, aes(x=time, y=as.factor(status), color=group)) +
    geom_point(size=2) +
    scale_color_manual(values=c("red", "orange")) +
    labs(x="Survival Time", y="Status", title="Figure 2.3: Survival Time for␣
 ↪Treatment and Nontreatment Groups by Status")

#Bar Graph (Fig. 2.4)
ggplot(hepatitis[which(hepatitis$status==1), ], aes(x=group, fill=group)) +
    geom_bar(fill=c("red", "orange")) +
    labs(x="Treatment", y="Deaths", title="Figure 2.4: Number of Deaths by␣
 ↪Treatment Group")
```

Figure 2.1: Survival Time for Censored (0) and Uncensored (1) Observations

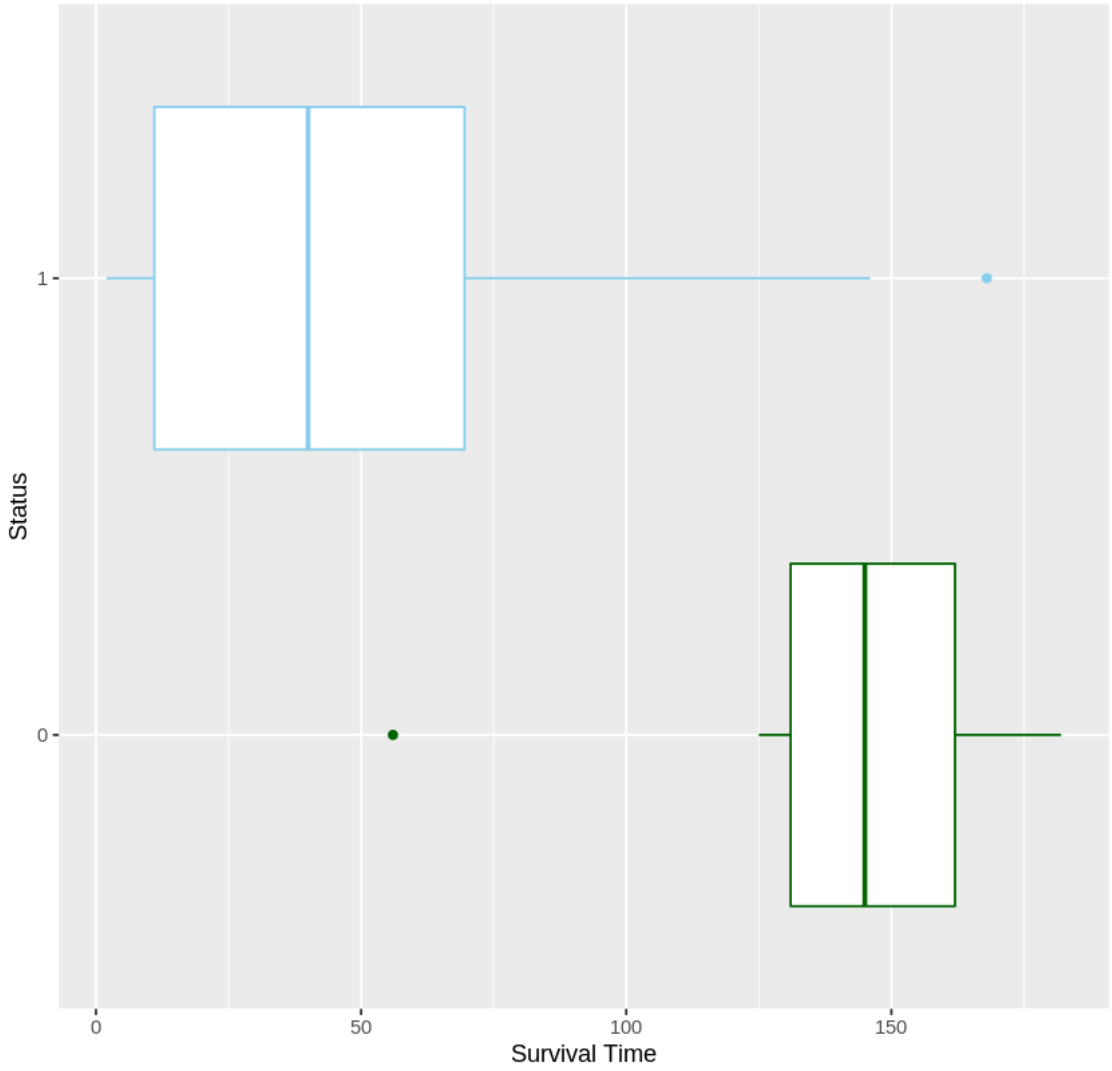


Figure 2.2: Survival Time for Treatment and Nontreatment Groups

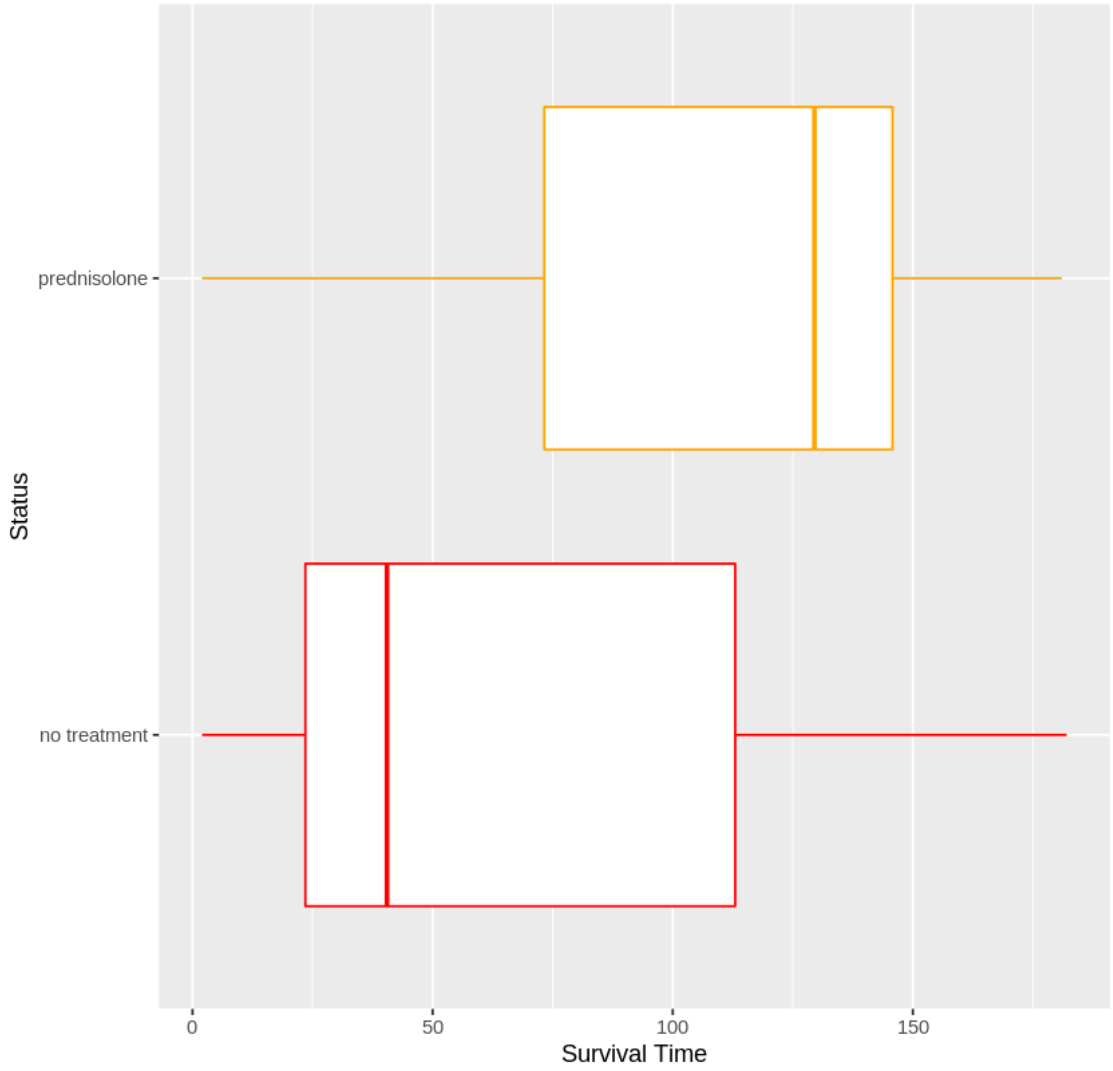


Figure 2.3: Survival Time for Treatment and Nontreatment Groups by Status

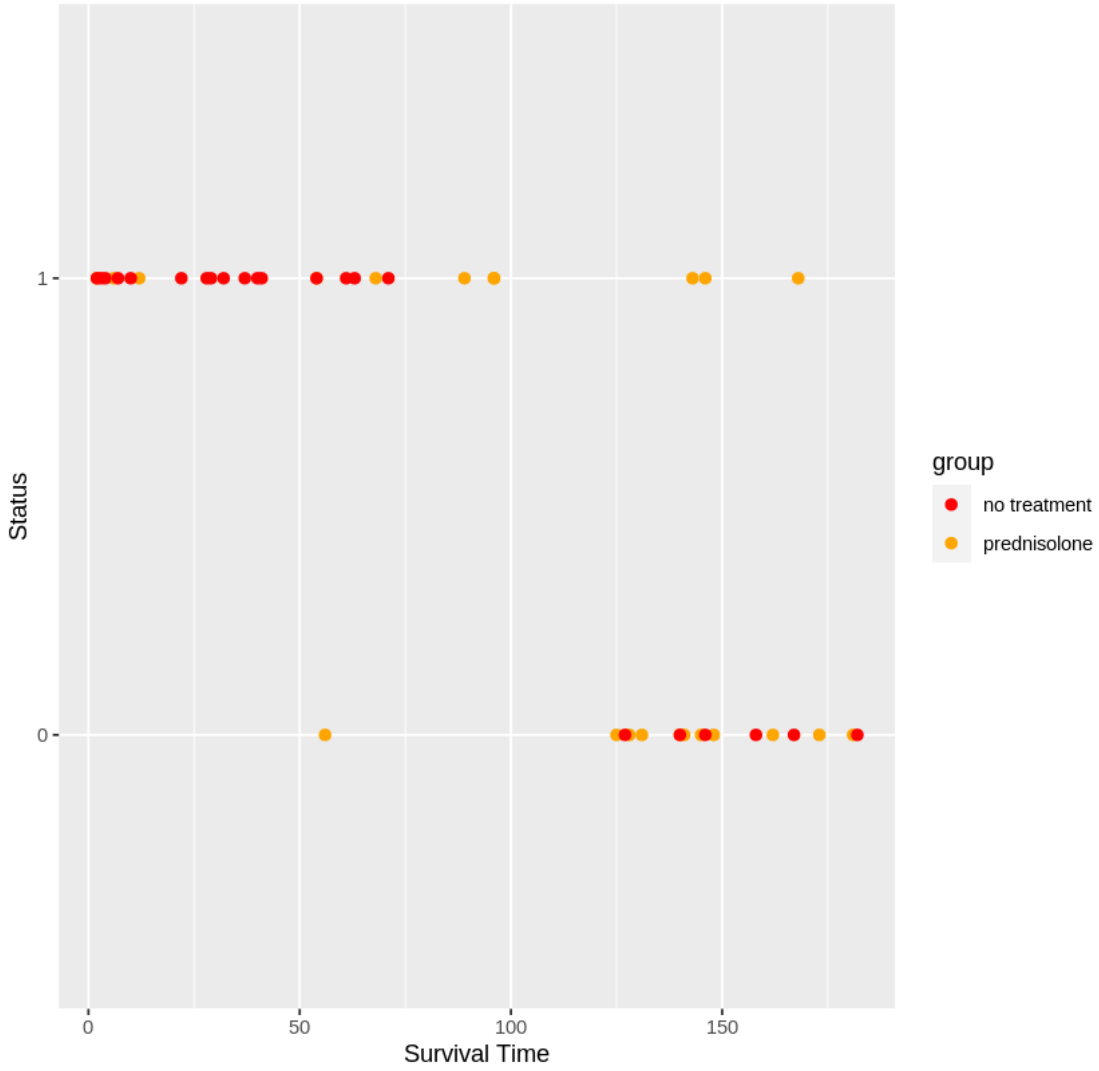
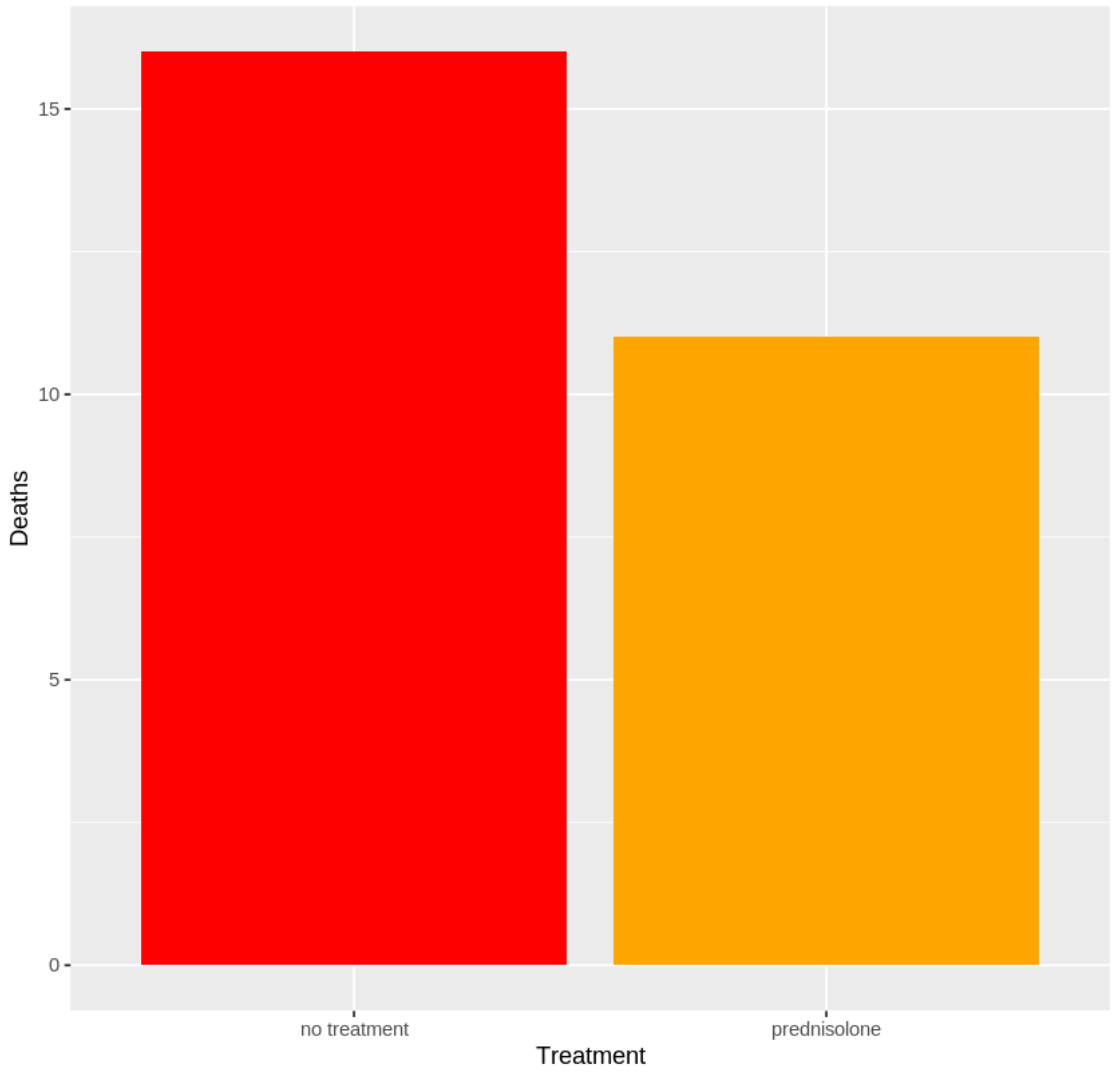


Figure 2.4: Number of Deaths by Treatment Group



Kaplan Meier Curves:

- Figure 2.5: Kaplan Meier Curve (Ignoring Censoring)
- Figure 2.6: Kaplan Meier Curves

Survival Descriptions:

- Number of observations, number of events (deaths), and median survival probability (in months) for the whole data (ignoring censoring)
- Number of observations, number of events (deaths), and median survival probability (in months) for each treatment group

Survival Comparison:

- **Log-Rank Test**: Testing the null hypothesis that there is no difference between survival curves for each treatment group. With a p-value of $0.03 < \alpha = 0.05$, we reject the null in favor of the alternative hypothesis that the treatment group survival curves are not statistically equivalent. Thus, we have statistically significant reason to believe that the survival probability of patients differs with treatment.

[21]:
```
#Kaplan Meier Curves

#Figure 2.5
ggsurvplot(survfit(Surv(time, status) ~ 1, data=hepatitis),
           surv.median.line = "hv",
           xlab="Months",
           ylab="Survival Probability",
           title="Figure 2.5: Kaplan Meier Curve (Ignoring Censoring)")

#Figure 2.6
ggsurvplot(survfit(Surv(time, status) ~ group, data=hepatitis),
           surv.median.line = "hv",
           xlab="Months",
           ylab="Survival Probability",
           title="Figure 2.6: Kaplan Meier Curves")
```

Figure 2.5: Kaplan Meier Curve (Ignoring Censoring)

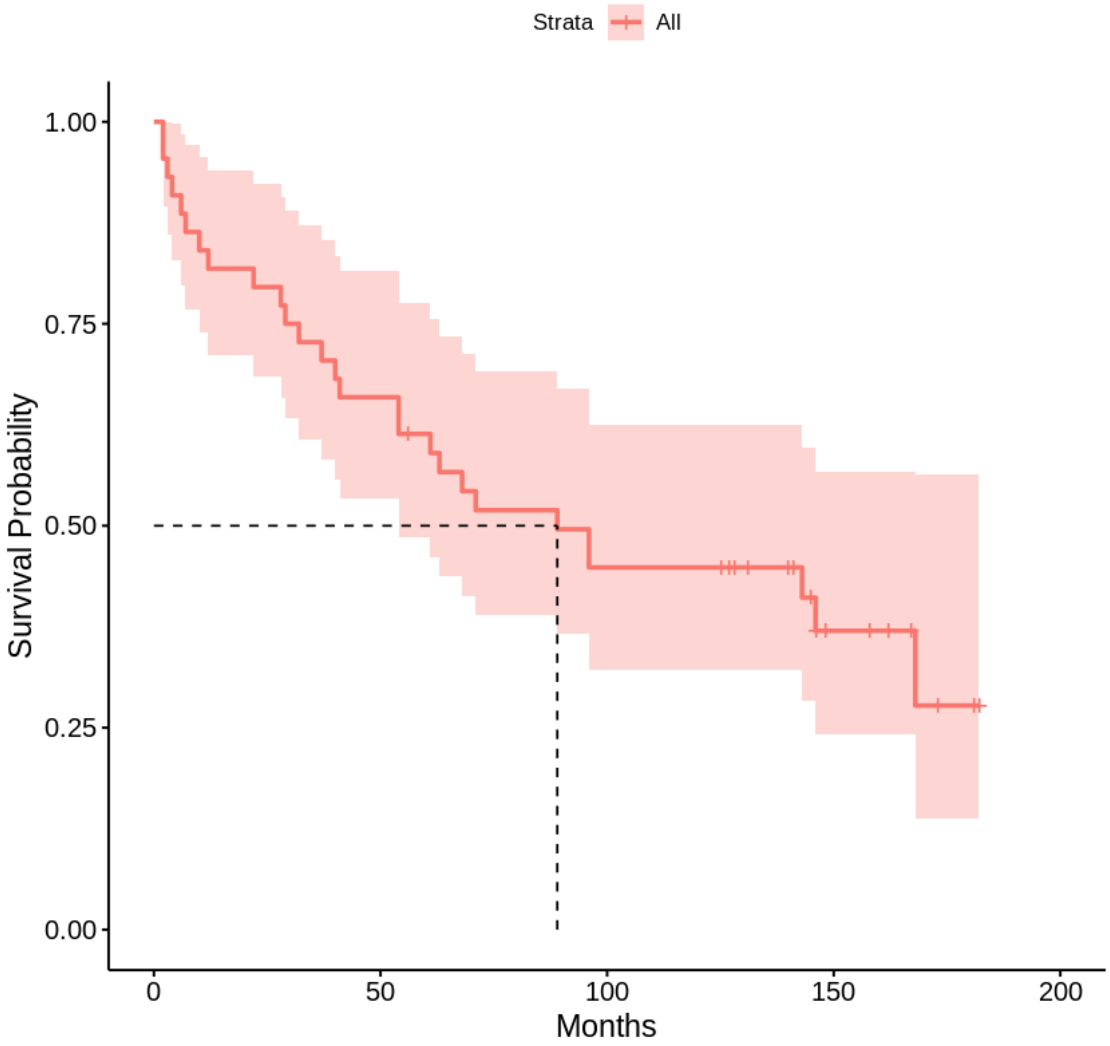
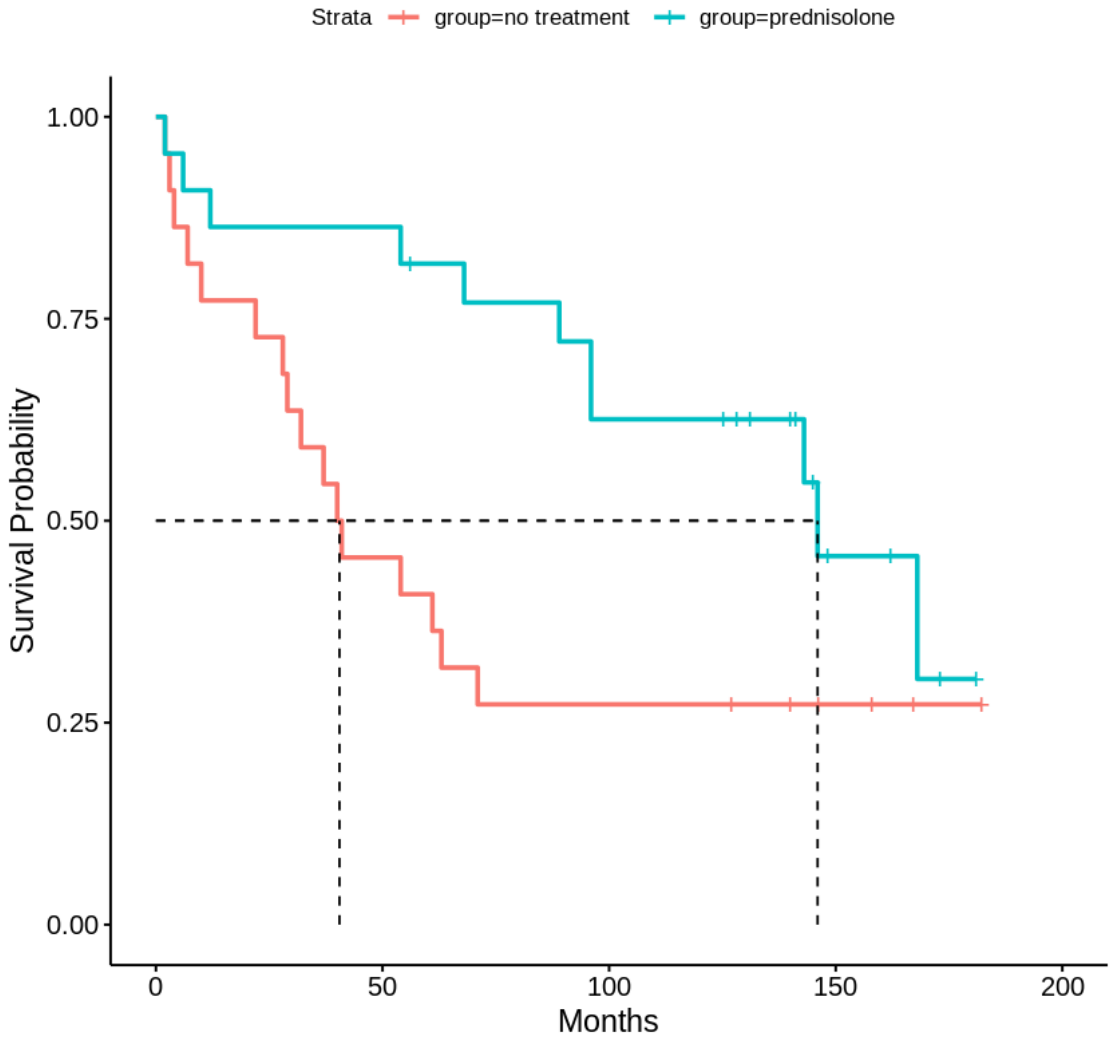


Figure 2.6: Kaplan Meier Curves



[22]:
```
#Survival Descriptions

survfit(Surv(time, status) ~ 1, data=hepatitis) #for Figure 2.5
survfit(Surv(time, status) ~ group, data=hepatitis) #for Firgure 2.6
```

```
Call: survfit(formula = Surv(time, status) ~ 1, data = hepatitis)

      n events median 0.95LCL 0.95UCL
[1,] 44     27     89      54      NA
```

```
Call: survfit(formula = Surv(time, status) ~ group, data = hepatitis)

                    n events median 0.95LCL 0.95UCL
group=no treatment 22     16   40.5      29      NA
```

```
group=prednisolone 22     11  146.0      96      NA
```

[23]:
```r
#Survival Comparison: Log-Rank Test

#Null Hypothesis: no difference between curves
survdiff(Surv(time, status) ~ group, data=hepatitis) #p=0.03<0.05 -> thus, we␣
↪reject the null
```

```
Call:
survdiff(formula = Surv(time, status) ~ group, data = hepatitis)

                     N Observed Expected (O-E)^2/E (O-E)^2/V
group=no treatment  22       16     10.6      2.73      4.66
group=prednisolone 22       11     16.4      1.77      4.66

 Chisq= 4.7  on 1 degrees of freedom, p= 0.03
```

b) Evaluating Effectiveness of Treatment Regression Model: Cox Proportional Hazard (PH) - Given that we are interested solely in evaluating the effect of treatment on survival, it is best to employ a Cox PH model to the data, as it is used to compare relative hazards (hazard ratios) between groups without making any assumptions about the baseline hazard $h_0(t)$.

Model Form:

$$h(t) = h_0(t) \cdot e^{\beta_1 X_{\text{prednisolone}}}$$

PH Assumption: - Graphically: Log-log Survival Curves - Plotting the Kaplan Meier survival estimates against time (Figure 2.7), we see that the curves are reasonably parallel. Therefore, the PH assumption holds and we assume that the Cox PH model provides an adequate fit to the data. - Hypothesis Test: GOF Test - Performing a goodness of fit test based on the Schoenfeld residuals, we obtain a p-value of $0.15 > \alpha = 0.05$. Thus, we fail to reject the null hypothesis that the PH assumption holds and assume that the Cox PH model is a good fit for the data.

Coefficient Interpretation:

coef	exp(coef)
-0.8324	0.435

Given that the reference group is the "no treatment" group, the exponentiated beta coefficient for treatment represents the hazard ratio (HR) of treatment (*prednisolone*) to no treatment:

$$\hat{HR} = \frac{\hat{h}_{\text{prednisolone}}(t)}{\hat{h}_{\text{no treatment}}(t)} = e^{\hat{\beta}_1} = e^{-0.8324} = 0.435$$

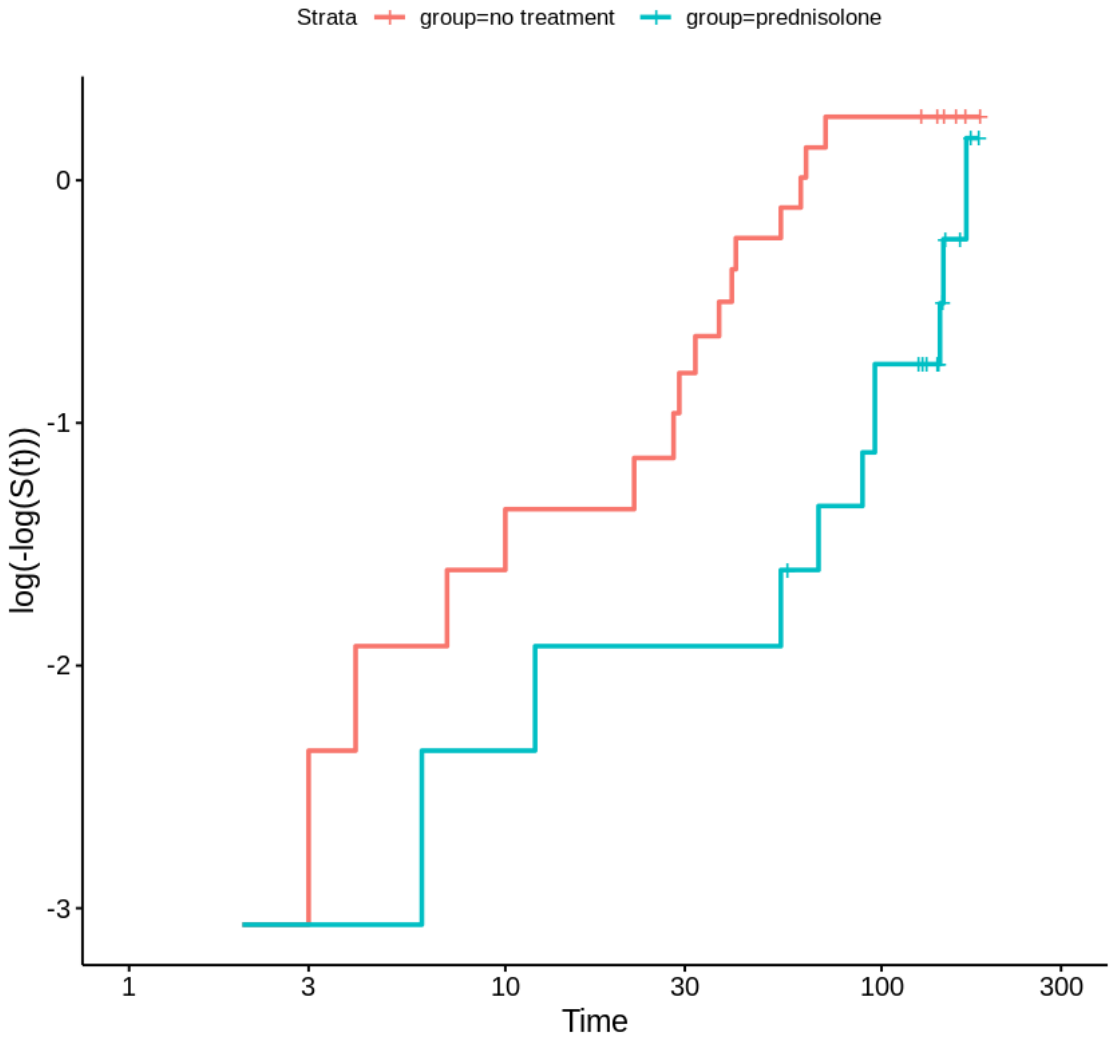
Obtaining a HR of 0.435, thus indicates that the hazard rate for treatment is about 0.435 times the hazard rate for no treatment. That is, based on the data, the hazard rate for patients who recieve prednisolone is about half of that for those who do not.

[24]:
```
#Cox PH Model (to test effectiveness of treatment)

m1 <- coxph(Surv(time, status) ~ group, data=hepatitis, ties="breslow")
summary(m1)
```

```
Call:
coxph(formula = Surv(time, status) ~ group, data = hepatitis,
    ties = "breslow")

  n= 44, number of events= 27

                     coef exp(coef) se(coef)      z Pr(>|z|)
groupprednisolone -0.8324    0.4350   0.3974 -2.095   0.0362 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

                  exp(coef) exp(-coef) lower .95 upper .95
groupprednisolone     0.435      2.299    0.1996    0.9479

Concordance= 0.633  (se = 0.048 )
Likelihood ratio test= 4.49  on 1 df,   p=0.03
Wald test            = 4.39  on 1 df,   p=0.04
Score (logrank) test = 4.62  on 1 df,   p=0.03
```

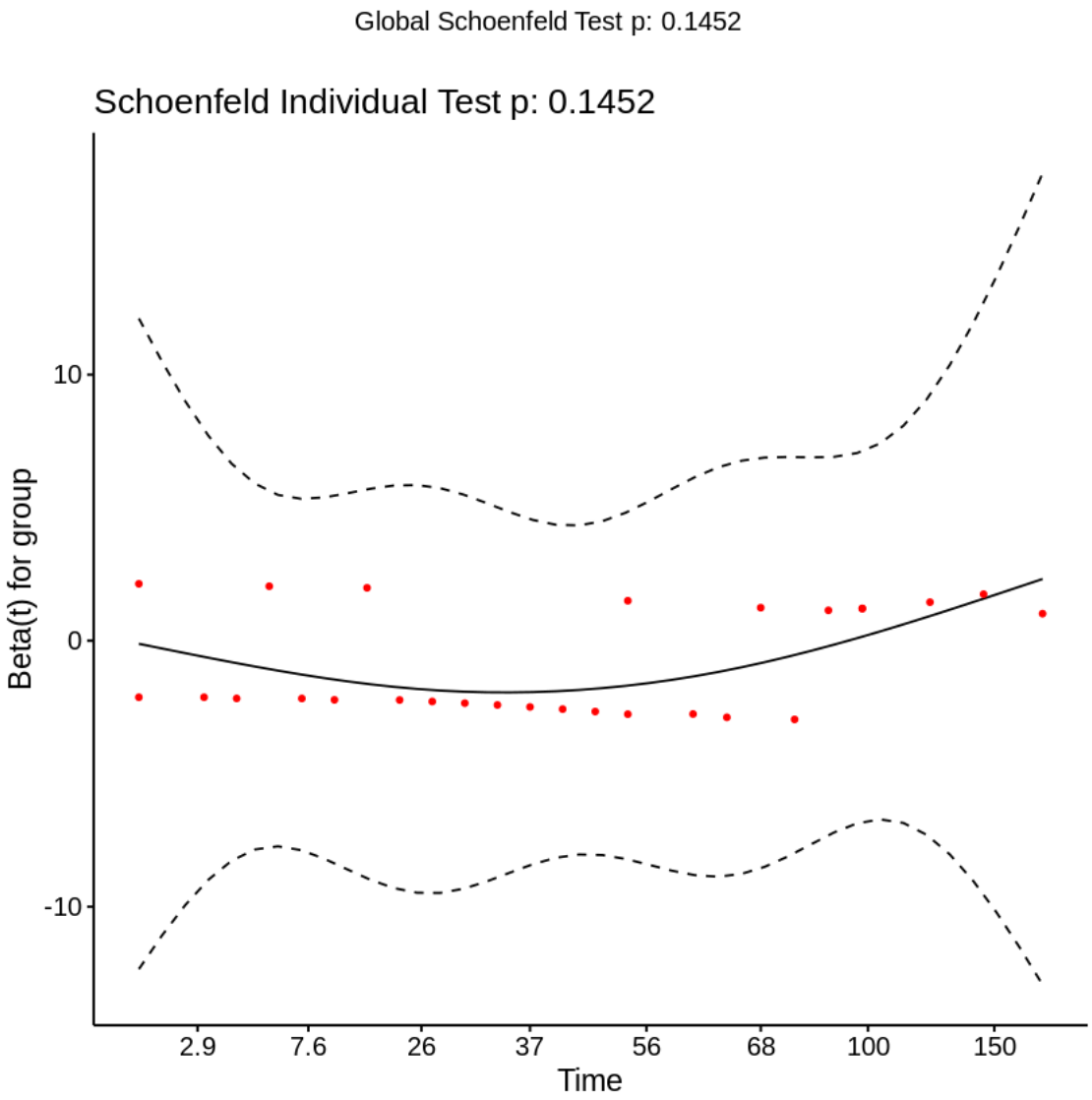
[25]:
```
#Log-log Survival Curves (Fig. 2.7)

ggsurvplot(survfit(Surv(time, status) ~ group, data=hepatitis),
           fun="cloglog",
           title="Figure 2.7: Log-log Kaplan Meier Survival Curves")
```

Figure 2.7: Log-log Kaplan Meier Survival Curves



[26]:
```
#GOF Test - based on Schoenfeld Residuals

#Null Hypothesis: PH assumption holds
cox.zph(m1) #p=0.15>0.05 -> thus, we do not reject the null

#Graph for GOF Test - Schoenfeld Residuals
ggcoxzph(cox.zph(m1))
```

```
       chisq df    p
group   2.12  1 0.15
GLOBAL  2.12  1 0.15
```



1.1.3 Question 3:

The "leukemia.csv" file contains data collected from leukemia patients including total time to death (in weeks) from diagnosis, white blood cell count (WBC), and the result from an AG test (positive or negative) for each patient in the study. The AG test is related to the type of leukemic cells found in the bone marrow at diagnosis.

a) Conduct a comprehensive Exploratory Data Analysis (EDA) to inspect, understand and describe the information collected in this dataset. Use appropriate summary statistics and plots to present your results from the EDA.

b) Is a positive AG test associated with better or worse prognosis? Explain.

c) Suppose that you want to assess the effect of both WBC and AG test result on the survival.

- i. Suggest 2 appropriate regression models that you could use for this purpose.
- ii. Perform a model selection procedure to find the model that best fits the data with each of the 2 approaches you suggested.
- iii. State the form of the 2 "best" models.
- iv. Test the models' assumptions and compare their fit to the data.
- v. Which of the 2 models best fits the data?
- vi. Interpret the regression coefficients of the "best" model.

[27]:
```
#importing "leukemia" data
#adding a column for categorical WBC
leukemia <- read.csv("/home/jovyan/AGLM/HW4/leukemia.csv") %>%
             mutate(WBC_group = case_when(WBC < 25 ~ 0,
                                          WBC >= 25 & WBC < 50 ~ 1,
                                          WBC >= 50 & WBC < 75 ~ 2,
                                          WBC >= 75 ~ 3))
#leukemia
```

a) Exploratory Data Analysis (EDA) The variables in this dataset are as follows: - Outcome: $Y = [T, C]$ - time until event of interest (death) occurs - Covariates: X_1 - white blood cells (count), X_2 - result from an AG test (positive or negative), X_3 - white blood cells (group 0-3) - X_1 is continuous with ratio scale - X_2 is categorical and binary with nominal scale - X_3 is categorical with four groups and ordinal scale

This EDA consists of: - Descriptive Statistics - Plots - Kaplan Meier Curves, Survival Descriptions, & Survival Comparison

Descriptive Statistics:

- Summary statistics for survival time (minimum value, 1st quartile, median, mean, 3rd quartile, maximum value)
- Summary statistics for survival time for each test group (minimum value, 1st quartile, median, mean, 3rd quartile, maximum value)
- Summary statistics for WBC (minimum value, 1st quartile, median, mean, 3rd quartile, maximum value)
- Summary statistics for WBC for each test group (minimum value, 1st quartile, median, mean, 3rd quartile, maximum value)
- Average hazard rates for the whole data and for each test group

[28]:
```
#Descriptive Statistics

#summary statistics for survival time
summary(leukemia$time)
#summary statistics for survival time (for + AG test)
summary(leukemia[which(leukemia[,3]=="+"),1])
#summary statistics for survival time (for - AG test)
```

```
summary(leukemia[which(leukemia[,3]=="-"),1])
#summary statistics for WBC
summary(leukemia$WBC)
#summary statistics for WBC (for + AG test)
summary(leukemia[which(leukemia[,3]=="+"),2])
#summary statistics for WBC (for - AG test)
summary(leukemia[which(leukemia[,3]=="-"),2])
#average hazard rate
length(leukemia$time)/sum(leukemia$time)
#average hazard rate for + AG test
length(leukemia[which(leukemia[,3]=="+"),1])/
 ↪sum(leukemia[which(leukemia[,3]=="+"),1])
#average hazard rate for - AG test
length(leukemia[which(leukemia[,3]=="-"),1])/
 ↪sum(leukemia[which(leukemia[,3]=="-"),1])
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    4.00   22.00   40.88   65.00  156.00
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00   16.00   56.00   62.47  108.00  156.00
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2.00    3.75    7.50   17.94   24.00   65.00
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.75    5.30   10.50   29.17   32.00  100.00
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.75    5.40   10.00   29.07   35.00  100.00
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.500   5.075  20.000  29.262  28.750 100.000
```

0.0244625648628614

0.0160075329566855

0.0557491289198606

Plots:

- Scatter Plot (Figure 3.1): Survival Time Given WBC & AG Test Result
- Box Plot (Figure 3.2): Survival Time Given WBC Group & AG Test Result
- Box Plot (Figure 3.3): WBC by AG Test Result
- Violin Plot (Figure 3.4): Survival Time Given AG Test Result

[29]:
```
#Plots

#Scatter Plot (Fig. 3.1)
ggplot(leukemia) +
    geom_point(aes(x=WBC, y=time, color=AG)) +
```

```
    labs(x="White Blood Cell Count (WBC)", y="Survival Time", title="Figure 3.1:
 ↪ Survival Time Given WBC & AG Test Result")

#Box Plot (Fig. 3.2)
ggplot(leukemia, aes(x=as.factor(WBC_group), y=time, color=AG)) +
    geom_boxplot(position=position_dodge(1)) +
    labs(x="White Blood Cell Group", y="Survival Time", title="Figure 3.2:␣
 ↪Survival Time Given WBC Group & AG Test Result")

#Box Plot (Fig. 3.3)
ggplot(leukemia, aes(x=AG, y=WBC, color=AG)) +
    geom_boxplot() +
    labs(x="AG Test Result", y="White Blood Cell Count (WBC)", title="Figure 3.
 ↪3: WBC by AG Test Result")

#Violin Plot (Fig. 3.4)
ggplot(leukemia, aes(x=AG, y=time, fill=AG)) +
    geom_violin() +
    labs(x="AG Test Result", y="Survival Time", title="Figure 3.4: Survival␣
 ↪Time Given AG Test Result")
```

Figure 3.1: Survival Time Given WBC & AG Test Result

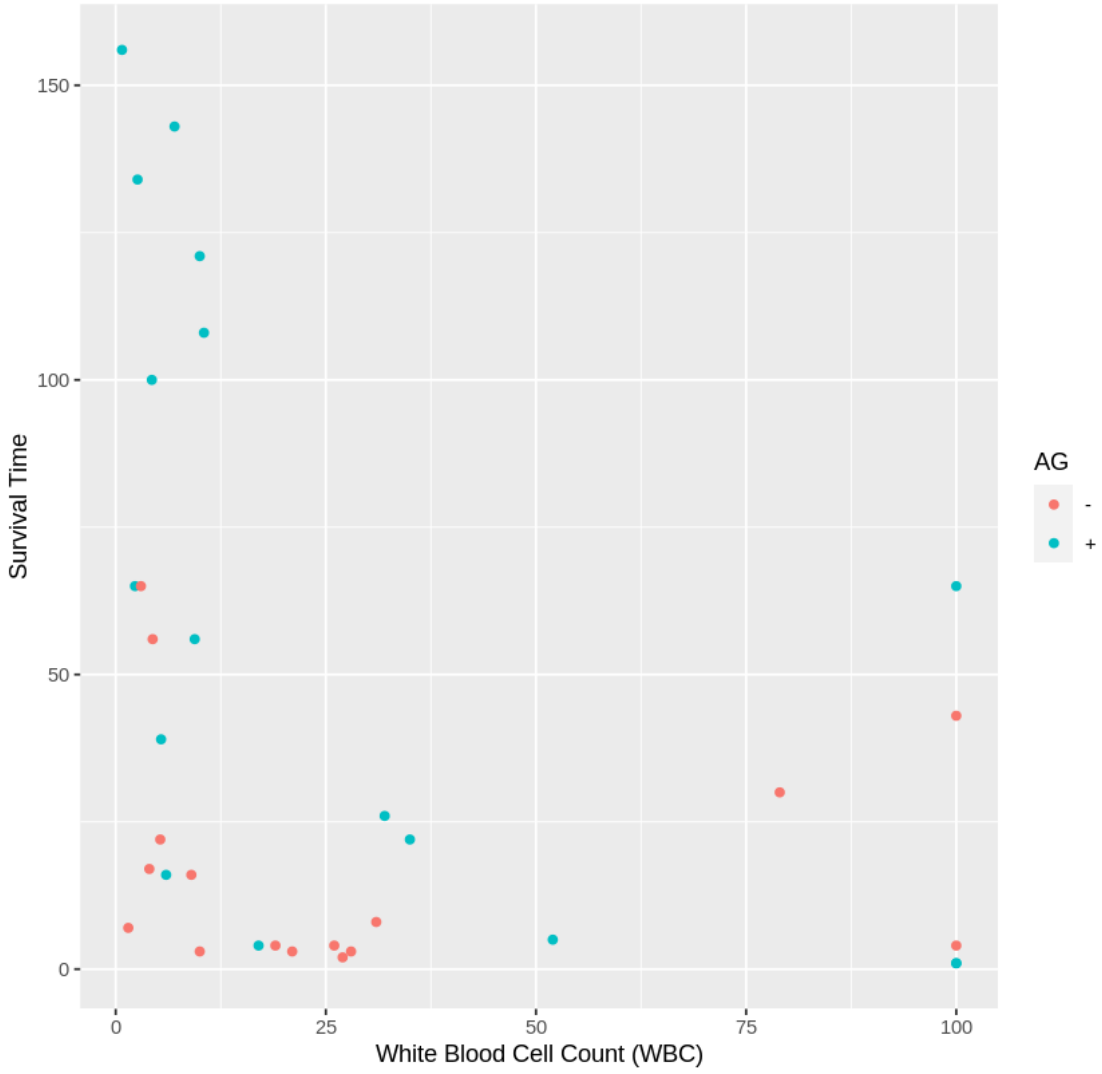


Figure 3.2: Survival Time Given WBC Group & AG Test Result

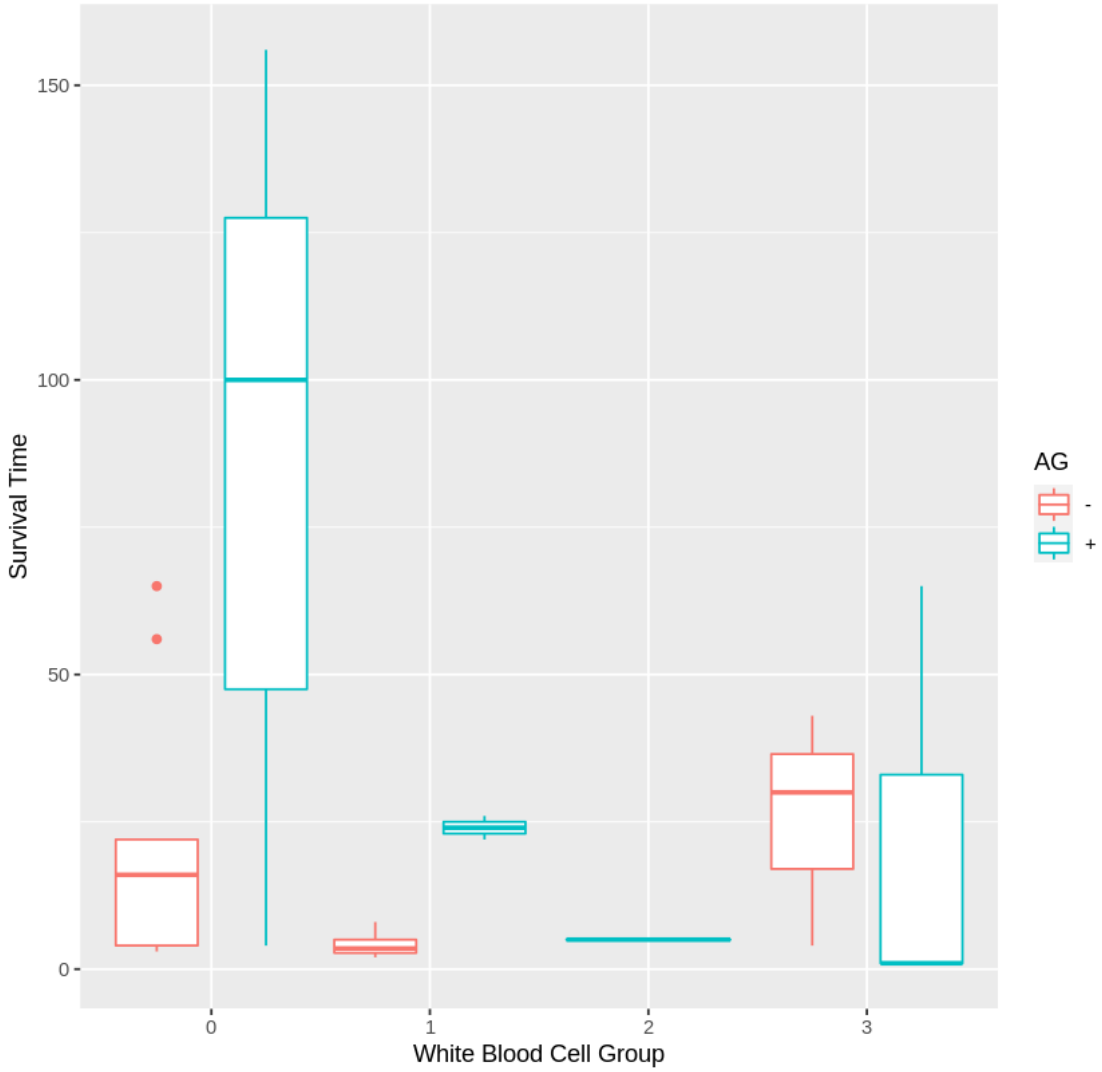
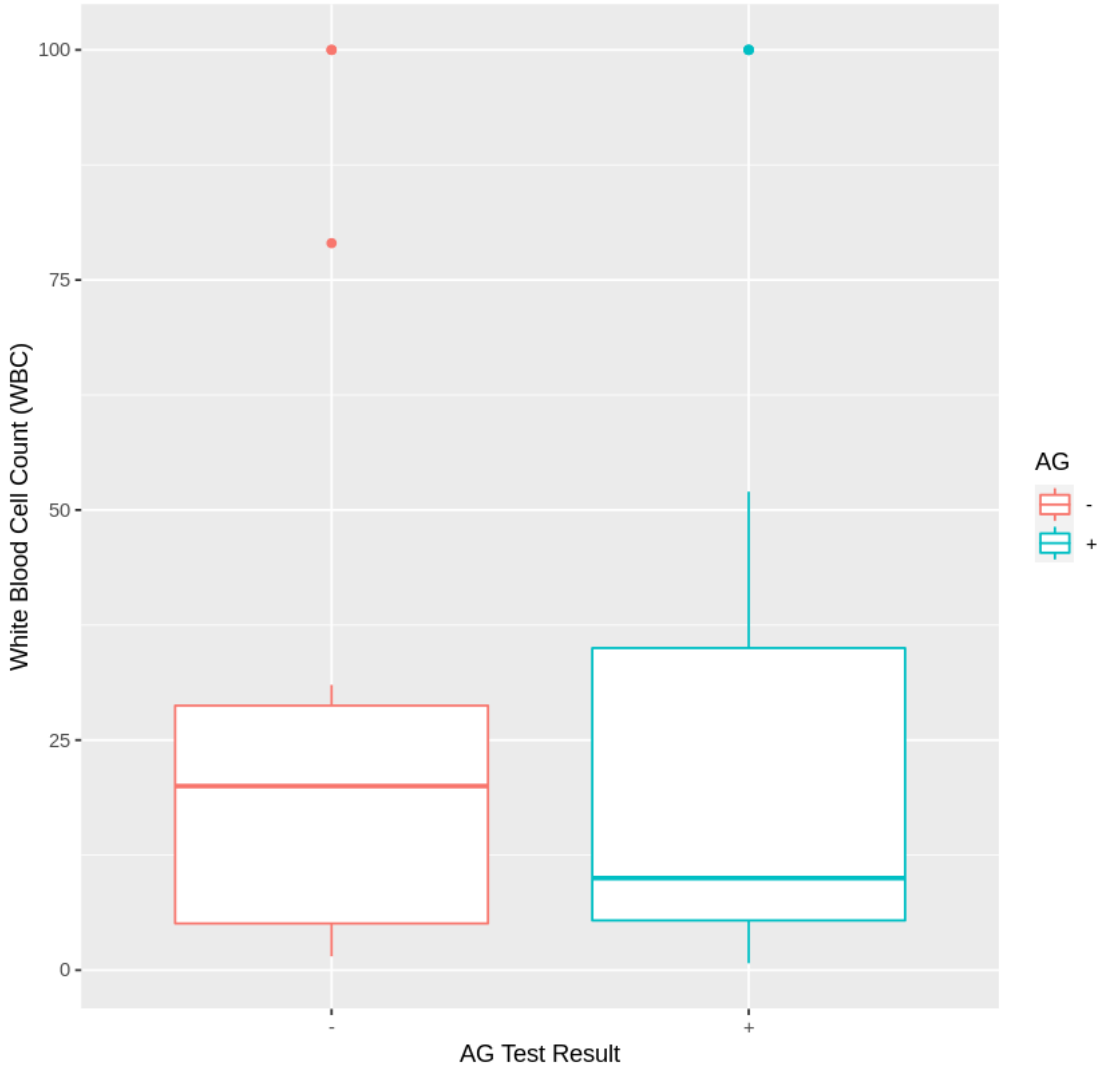
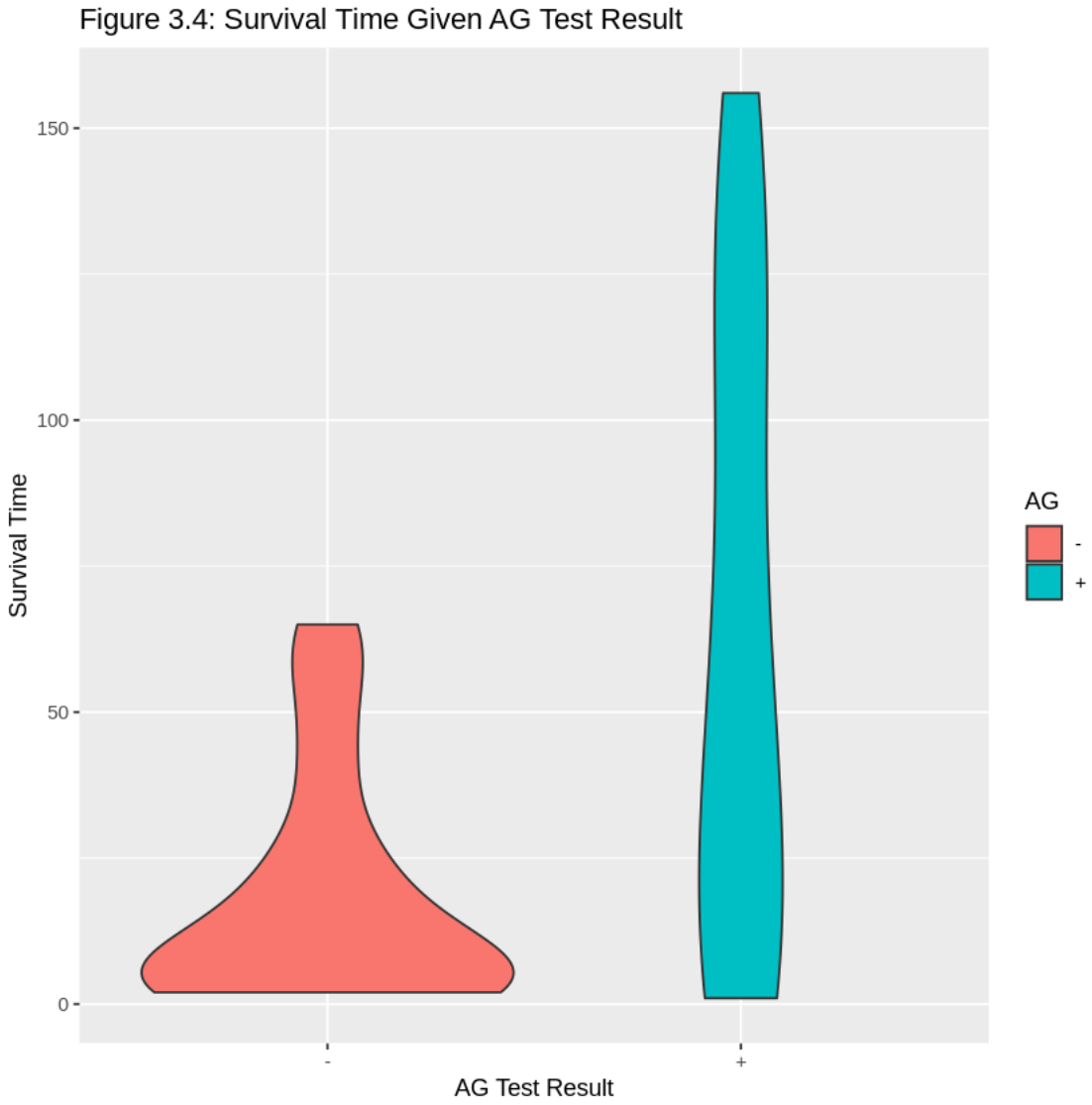


Figure 3.3: WBC by AG Test Result





Kaplan Meier Curves:

- Figure 3.5: Kaplan Meier Curve
- Figure 3.6: Kaplan Meier Curves (given test group)

Survival Descriptions:

- Number of observations, number of events (deaths), and median survival probability (in weeks) for the whole data
- Number of observations, number of events (deaths), and median survival probability (in weeks) for each AG test group

Survival Comparison:

- **Log-Rank Test**: Testing the null hypothesis that there is no difference between survival curves for each AG test group. With a p-value of $0.004 < \alpha = 0.05$, we reject the null in favor of the alternative hypothesis that the test group survival curves are not statistically equivalent. Thus, we have statistically significant reason to believe that the survival probability of patients differs with test result.

[30]:
```
#Kaplan Meier Curves

#Figure 3.5
km2_1 <- ggsurvplot(survfit(Surv(time) ~ 1, data=leukemia),
                    surv.median.line = "hv",
                    xlab="Weeks",
                    ylab="Overall Survival Probability",
                    title="Figure 3.5: Kaplan Meier Curve")

#Figure 3.6
km2_2 <- ggsurvplot(survfit(Surv(time) ~ AG, data=leukemia),
                    surv.median.line = "hv",
                    xlab="Weeks",
                    ylab="Overall Survival Probability",
                    title="Figure 3.6: Kaplan Meier Curves (given test group)")

km2_1
km2_2
```

Figure 3.5: Kaplan Meier Curve

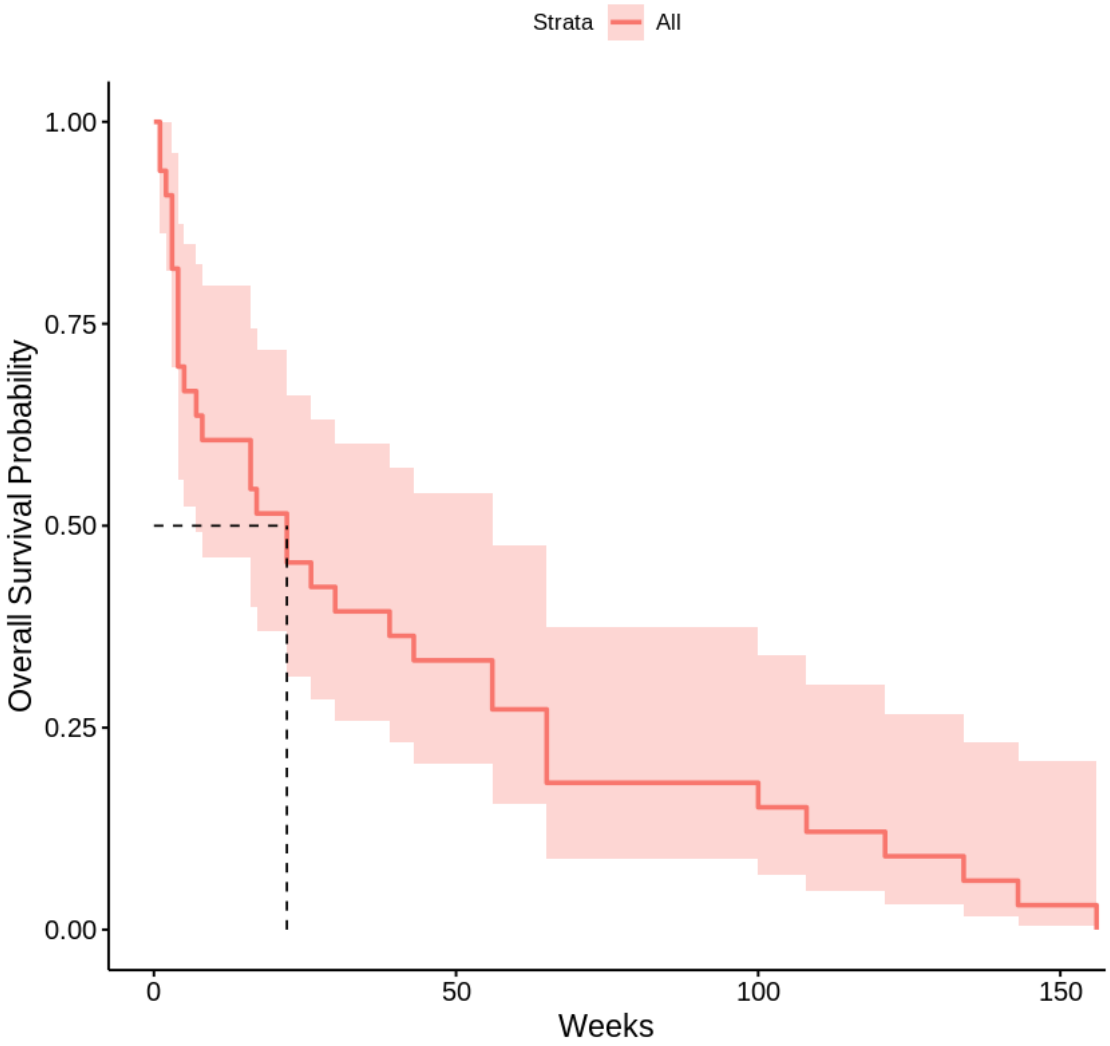
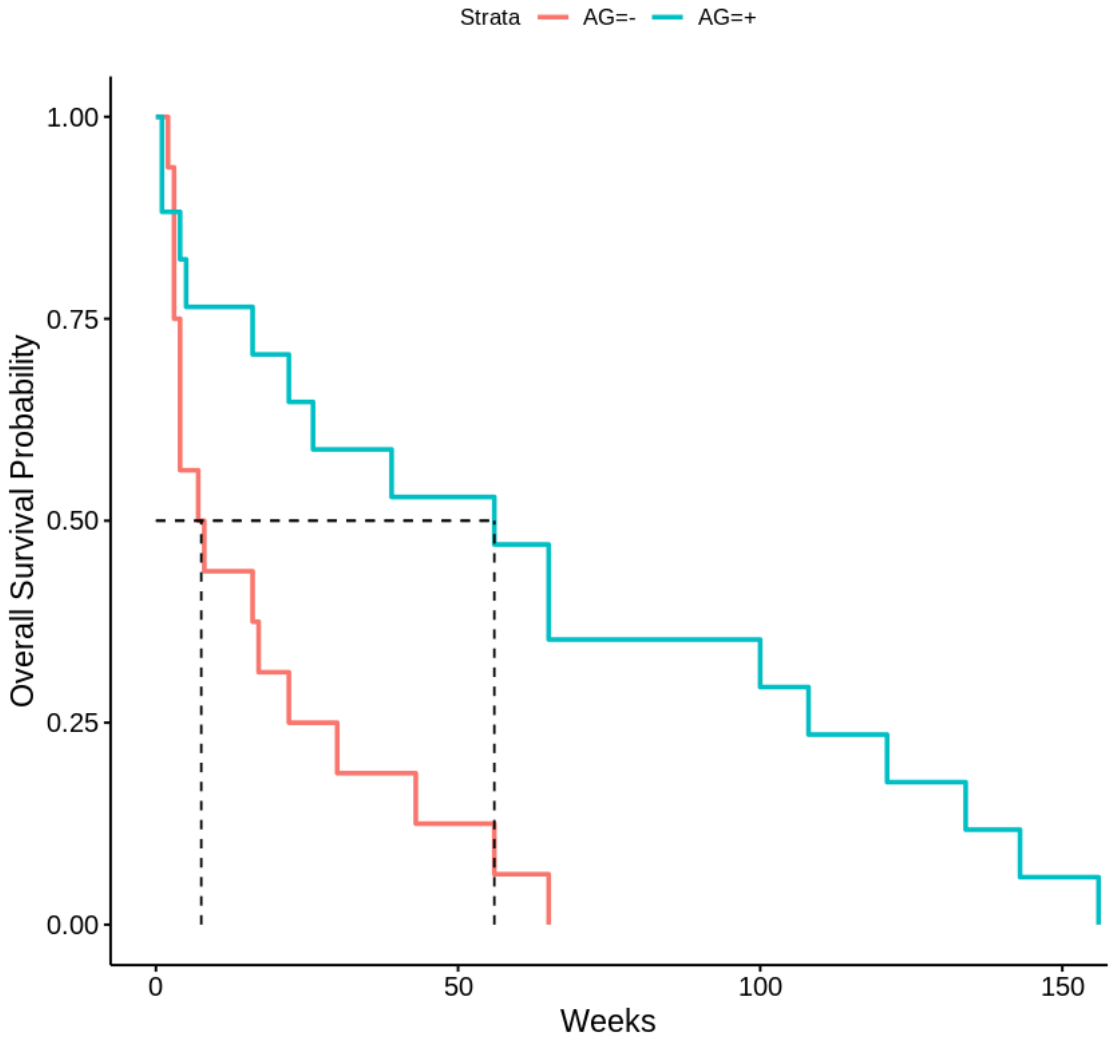


Figure 3.6: Kaplan Meier Curves (given test group)



[31]:
```
#Survival Descriptions

survfit(Surv(time) ~ 1, data=leukemia) #for Figure 3.5
survfit(Surv(time) ~ AG, data=leukemia) #for Firgure 3.6
```

```
Call: survfit(formula = Surv(time) ~ 1, data = leukemia)

      n events median 0.95LCL 0.95UCL
[1,] 33     33     22       7      56
```

```
Call: survfit(formula = Surv(time) ~ AG, data = leukemia)

      n events median 0.95LCL 0.95UCL
AG=- 16     16    7.5       4      43
```

```
AG=+ 17     17    56.0      22     121
```

[32]:
```
#Survival Comparison: Log-Rank Test

#Null Hypothesis: no difference between curves
survdiff(Surv(time) ~ AG, data=leukemia) #p=0.004<0.05 -> thus, we reject the␣
↪null
```

```
Call:
survdiff(formula = Surv(time) ~ AG, data = leukemia)

      N Observed Expected (O-E)^2/E (O-E)^2/V
AG=- 16       16      9.3      4.83      8.45
AG=+ 17       17     23.7      1.90      8.45

 Chisq= 8.4  on 1 degrees of freedom, p= 0.004
```

b) Association Between AG Test Result and Prognosis The violin plot above (Figure 3.4) illustrates the difference in the spread of suvival times (in weeks) for each AG test result group. Noticably, those who've recieved a positive AG test result generally displayed greater survival times, while those who recieved a negative AG test result displayed survival times that were at most half of the largest survival times in the former group. Although this is not indicative of a strong relationship between AG test result and survival time, it does imply that a postive AG test may result in a better prognosis. Moreover, the Kaplan Meier curves in Figure 3.6 and corresponding Log-Rank test, show us that there is a statistically significant difference between survival probabilities among AG test groups. Specifically, given that survival probabilities are significantly larger for those who tested positive, it is safe to conclude that a positive AG test is associated with an overall better prognosis.

c) Assessing the Effect of WBC and AG Test Result on Survival Regression Models: 1. Cox Proportional Hazard (PH) - Provides a comparison of hazards between groups within each predictor. 2. Exponential - Allows us to predict survival and assess the corresponding impacts of WBC and AG test result.

Model Selection: Bakcward Elimination

Based on the model selection procedure and corresponding hypothesis tests and AIC values, the two models that best fit the data have the following additive forms:

1. Cox PH:
$$h(t) = h_0(t) \cdot e^{\beta_1 X_{\text{AG}+} + \beta_2 log(\text{WBC})}$$

2. Exponential (AFT model):
$$t = e^{\alpha_0} \cdot e^{\alpha_1 X_{\text{AG}+} + \alpha_2 log(\text{WBC})}$$

where exponentiated α denotes a survival time ratio (TR), and exponentiated $-\alpha = \beta$ denotes a hazard ratio (HR).

Model Assumptions: - PH Assumption: - Both models above assume the PH assumption (as both either exclude or maintain a constant baseline hazard), which we observe holds for the data

in performing a goodness of fit test based on the Schoenfeld residuals. A corresponding p-value of $0.62 > \alpha = 0.05$ indicates that we fail to reject the null hypothesis that the PH assumption holds, and verifies this assumption. - AFT Assumption: - Given the distribution of deviance residuals (Figure 3.10), the fit of the AFT parametric model, and the fact that both models yield the same HR's (when exponentiating $-\alpha$), we may assume that the AFT assumption holds.

"Best" Model: Cox PH Additive Model (`m2_2`)

Looking at the predicted vs. observed (KM) survival probability plots (Figure 3.7 & 3.8) for both models, as well as their deviance residuals, we see that the Cox PH model, provides a better fit to the data. This is further validated by their total sums of squared deviance residuals (≈ 29 for the Cox PH model and ≈ 40 for the exponential model), as well as the AIC values (≈ 161 for the Cox PH model and ≈ 299 for the exponential model).

Coefficient Interpretation:

predictor	coef	exp(coef)
AG+	-1.0176	0.3614
log(WBC)	0.3603	1.4337

Since the reference group for the AG test result (categorical predictor) is the positive ("+") group, its exponentiated beta coefficient represents the hazard ratio (HR) of a positive AG test result to a negative AG test result. Specifically, obtaining a HR of 0.3614, indicates that the hazard rate for a positive AG test is about 0.3614 times the hazard rate for a negative AG test. That is, based on the data, the hazard rate for patients who recieve a positive AG test is about a third of that for those who do not, irrespective of white blood cell count (WBC). Moreover, a HR of 1.4337 for our numerical predictor, indicates that the data roughly showed a 1.4337 unit increase in the expected relative hazard for each one unit increase in the log of WBC. Thus, for every one unit increase in WBC, we can expect to see a $e^{1.4337} \approx 4$ unit increase in hazard rate, holding AG test result constant.

Sources: - https://www.ripublication.com/ijss17/ijssv12n2_15.pdf

[33]:
```
#COX PH - Model Selection (Backward Elimination)

m2_1 <- coxph(Surv(time) ~ AG*log(WBC), data=leukemia, ties="breslow")
#summary(m2_1)
m2_2 <- coxph(Surv(time) ~ AG+log(WBC), data=leukemia, ties="breslow") #best␣
 ↪model
summary(m2_2)
m2_3 <- coxph(Surv(time) ~ AG, data=leukemia, ties="breslow")
#summary(m2_3)
m2_4 <- coxph(Surv(time) ~ 1, data=leukemia, ties="breslow")
#summary(m2_4)
```

```
Call:
coxph(formula = Surv(time) ~ AG + log(WBC), data = leukemia,
    ties = "breslow")
```

```
  n= 33, number of events= 33

            coef exp(coef) se(coef)      z Pr(>|z|)
AG+      -1.0176    0.3614   0.4235 -2.403  0.01626 *
log(WBC)  0.3603    1.4337   0.1355  2.659  0.00785 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

         exp(coef) exp(-coef) lower .95 upper .95
AG+         0.3614     2.7666    0.1576    0.8289
log(WBC)    1.4337     0.6975    1.0993    1.8699

Concordance= 0.728  (se = 0.047 )
Likelihood ratio test= 14.63  on 2 df,   p=7e-04
Wald test            = 14.12  on 2 df,   p=9e-04
Score (logrank) test = 15.32  on 2 df,   p=5e-04
```

[34]:
```
#LRT
anova(m2_4, m2_3, test="LRT")
anova(m2_3, m2_2, test="LRT")
anova(m2_2, m2_1, test="LRT") #both saturated and additive models fit the data␣
 ↪equally well

#AIC
AIC(m2_4, m2_3, m2_2, m2_1) #saturated model AIC ~ additive model AIC
```

A anova: 2 × 4

	loglik <dbl>	Chisq <dbl>	Df <dbl>	P(>|Chi|) <dbl>
1	-85.99694	NA	NA	NA
2	-82.23295	7.527989	1	0.006074769

A anova: 2 × 4

	loglik <dbl>	Chisq <dbl>	Df <int>	P(>|Chi|) <dbl>
1	-82.23295	NA	NA	NA
2	-78.68167	7.102561	1	0.007697387

A anova: 2 × 4

	loglik <dbl>	Chisq <dbl>	Df <int>	P(>|Chi|) <dbl>
1	-78.68167	NA	NA	NA
2	-77.03063	3.302072	1	0.06919254

A data.frame: 4 × 2

	df <dbl>	AIC <dbl>
m2_4	0	171.9939
m2_3	1	166.4659
m2_2	2	161.3633
m2_1	3	160.0613

[35]:
```
#EXPONENTIAL - Model Selection (Backward Elimination)

m3_1 <- survreg(Surv(time) ~ AG*log(WBC), data=leukemia, dist="exponential")
#summary(m3_1)
m3_2 <- survreg(Surv(time) ~ AG+log(WBC), data=leukemia, dist="exponential")␣
 ↪#best model
summary(m3_2)
m3_3 <- survreg(Surv(time) ~ AG, data=leukemia, dist="exponential")
#summary(m3_3)
m3_4 <- survreg(Surv(time) ~ 1, data=leukemia, dist="exponential")
#summary(m3_4)
```

```
Call:
survreg(formula = Surv(time) ~ AG + log(WBC), data = leukemia,
    dist = "exponential")
             Value Std. Error     z      p
(Intercept)  3.713      0.454  8.17  3e-16
AG+          1.018      0.364  2.80 0.0051
log(WBC)    -0.304      0.124 -2.45 0.0144

Scale fixed at 1

Exponential distribution
Loglik(model)= -146.5   Loglik(intercept only)= -155.5
        Chisq= 17.82 on 2 degrees of freedom, p= 0.00014
Number of Newton-Raphson Iterations: 5
n= 33
```

[36]:
```
#AIC

AIC(m3_4, m3_3, m3_2, m3_1) #saturated model AIC ~ additive model AIC
```

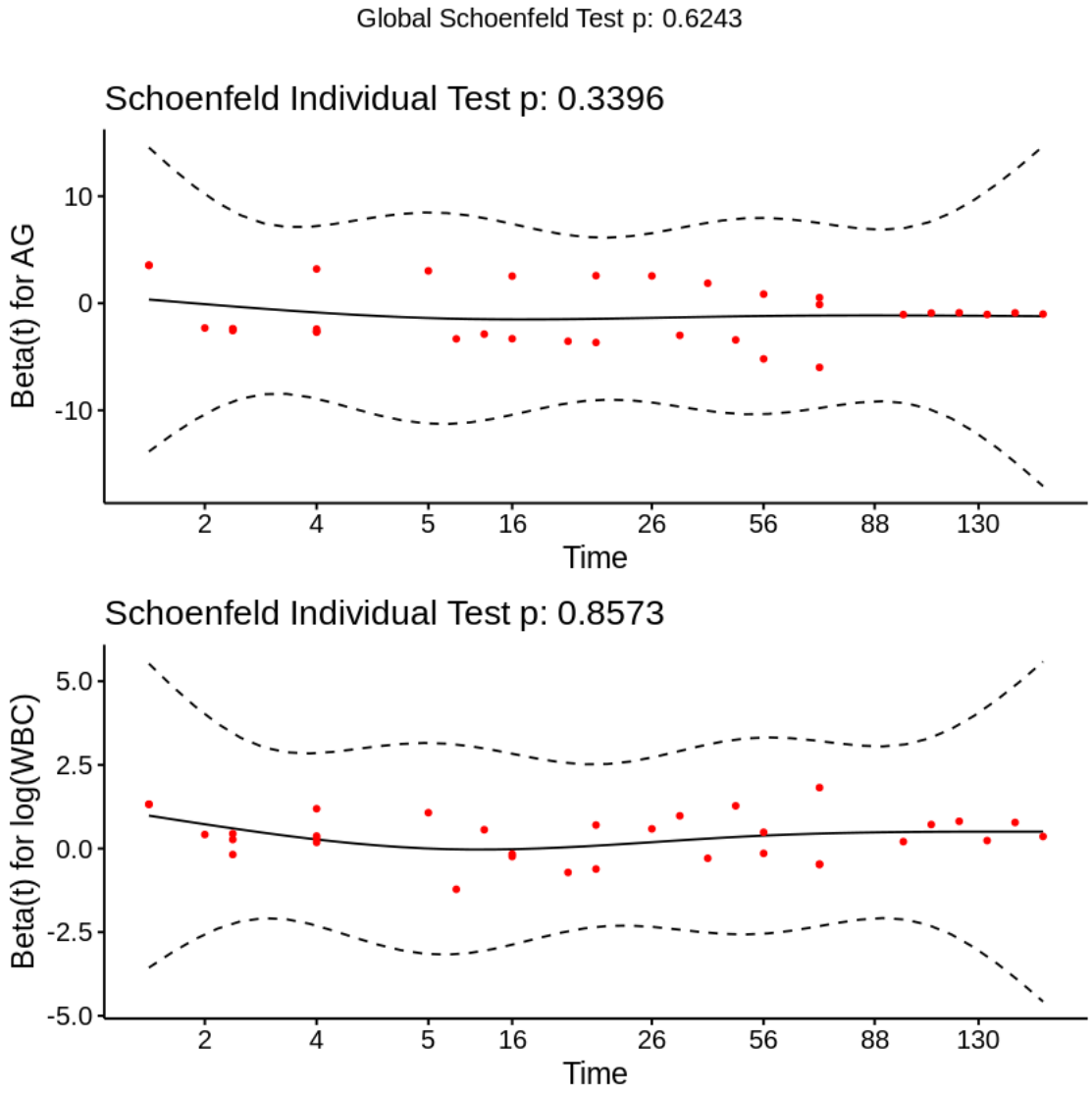
A data.frame: 4 × 2

	df <dbl>	AIC <dbl>
m3_4	1	312.9003
m3_3	2	302.9603
m3_2	3	299.0810
m3_1	4	299.3166

[37]:
```
#PH Assumption
#GOF Test - based on Schoenfeld Residuals

#Null Hypothesis: PH assumption holds
cox.zph(m2_2) #p=0.62>0.05 -> thus, we do not reject the null
```

```
#Graph for GOF Test - Schoenfeld Residuals
ggcoxzph(cox.zph(m2_2))
```

```
          chisq df    p
AG       0.9118  1 0.34
log(WBC) 0.0323  1 0.86
GLOBAL   0.9422  2 0.62
```



[38]:
```
#AFT Assumption

#AG test result coefficients
#Exponential -> e^alpha = 2.7677 TR
exp(1.018)
```

```
#Exponential -> e^-alpha = 0.3613 HR
exp(-1.018)
#Cox PH -> e^beta = 0.3615 HR
exp(-1.0176)
```

2.76765391713015

0.361316851724339

0.361461407374231

[39]:
```
#COMPARING COX PH AND EXP MODELS

#Predicted vs. Observed Survival Probabilities

#additive models
exp <- flexsurvreg(Surv(time)~as.factor(AG)+log(WBC), data=leukemia,␣
 ↪dist="exponential") #m3_2
plot(exp,
     col="red",
     xlab = "Time",
     ylab = "Survival Probability",
     main = "Figure 3.7: Cox PH & Exponentional Model Predictions")
points(leukemia$time, predict(m2_2, type="survival"), col="blue") #m2_2

#models with only AG as a covariate
exp2 <- flexsurvreg(Surv(time)~as.factor(AG), data=leukemia,␣
 ↪dist="exponential") #m3_3
plot(exp2,
     col=c("red", "orange"),
     xlab = "Time",
     ylab = "Survival Probability",
     main = "Figure 3.8: Cox PH & Exponentional Model Predictions")
points(leukemia$time, predict(m2_3, type="survival"), col="blue")
legend("topright", c("AG +", "AG -"), col = c("red", "orange"), lty = 1) #m2_3
```

Figure 3.7: Cox PH & Exponentional Model Predictions

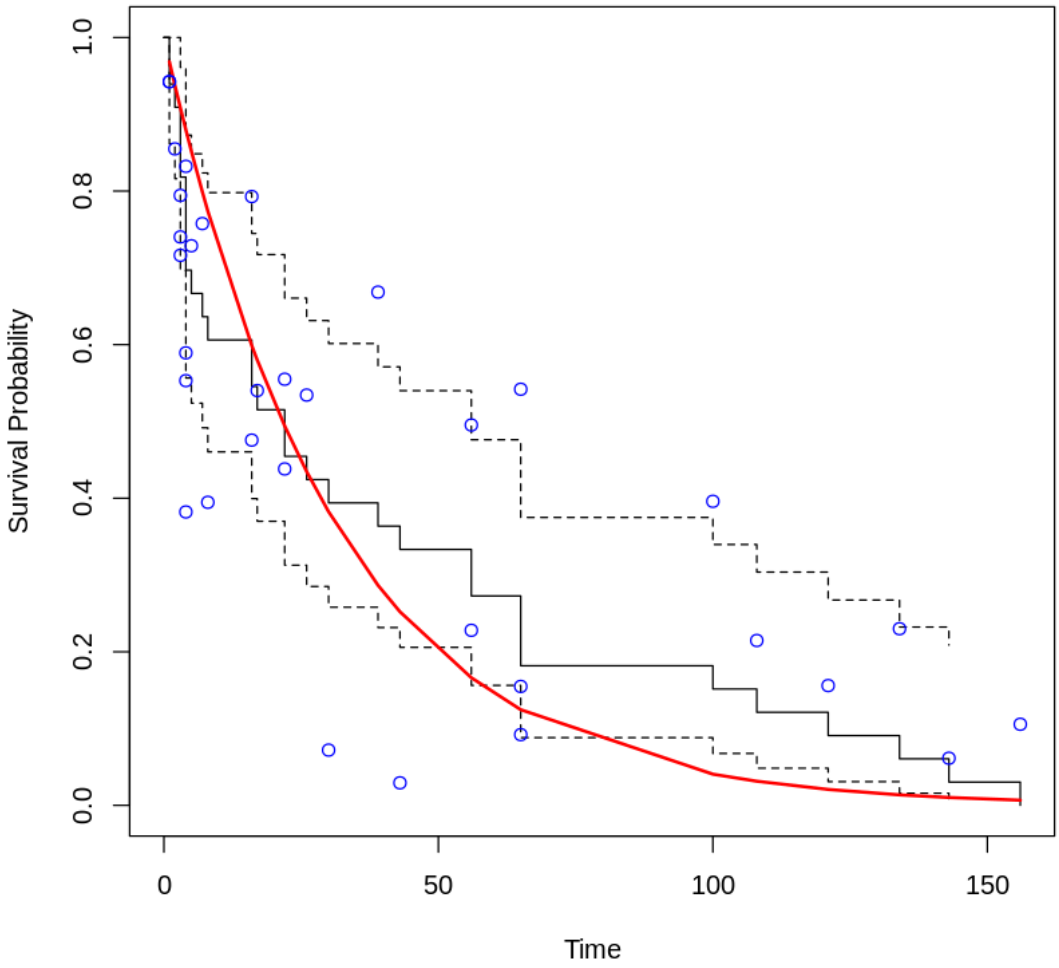
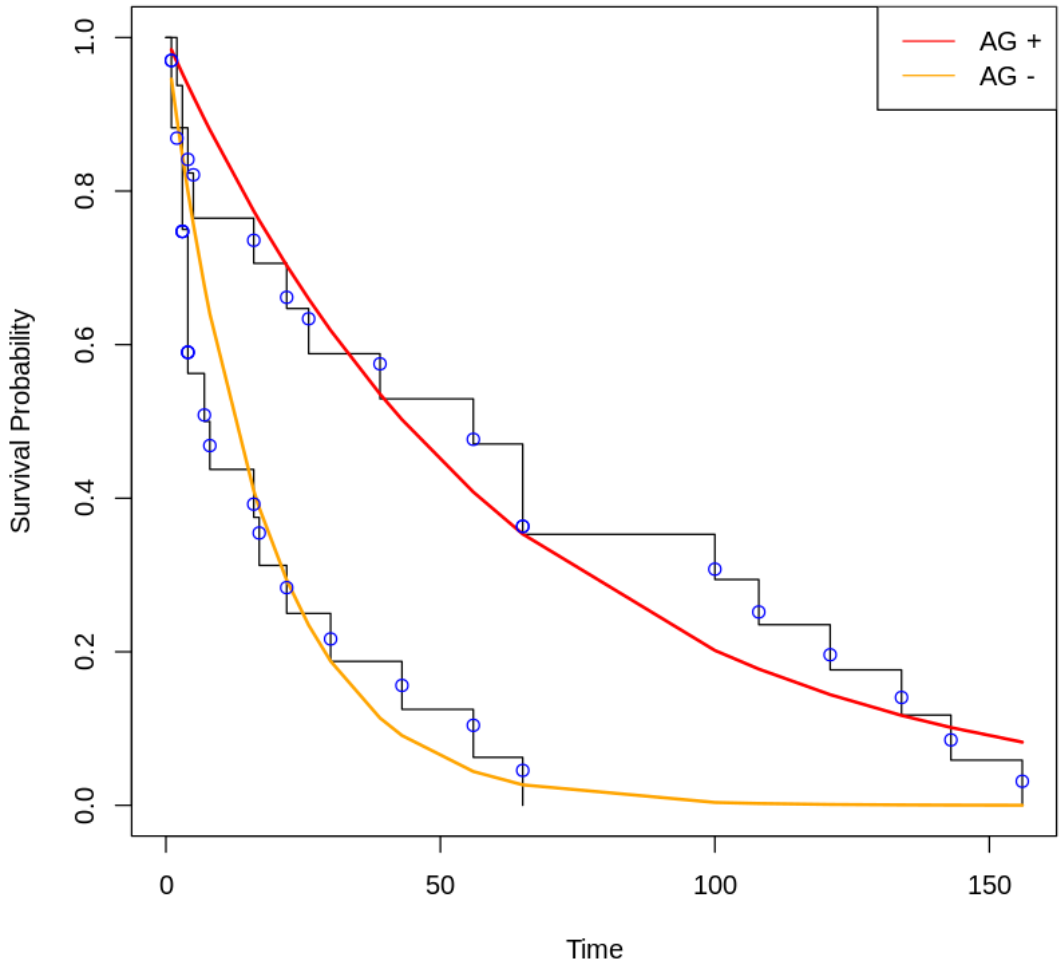


Figure 3.8: Cox PH & Exponentional Model Predictions



[40]:
```
#Deviance Residuals

#cox
ggcoxdiagnostics(m2_2,
                 type="deviance",
                 linear.predictions=FALSE,
                 title="Figure 3.9: Cox PH Model Deviance Residuals")

#exp
plot(residuals(m3_2, type="deviance"),
     main="Figure 3.10: Exponentional Model Deviance Residuals")
abline(h=0, lty=2, lwd=3, col="red")
```

`geom_smooth()` using formula 'y ~ x'

Figure 3.9: Cox PH Model Deviance Residuals

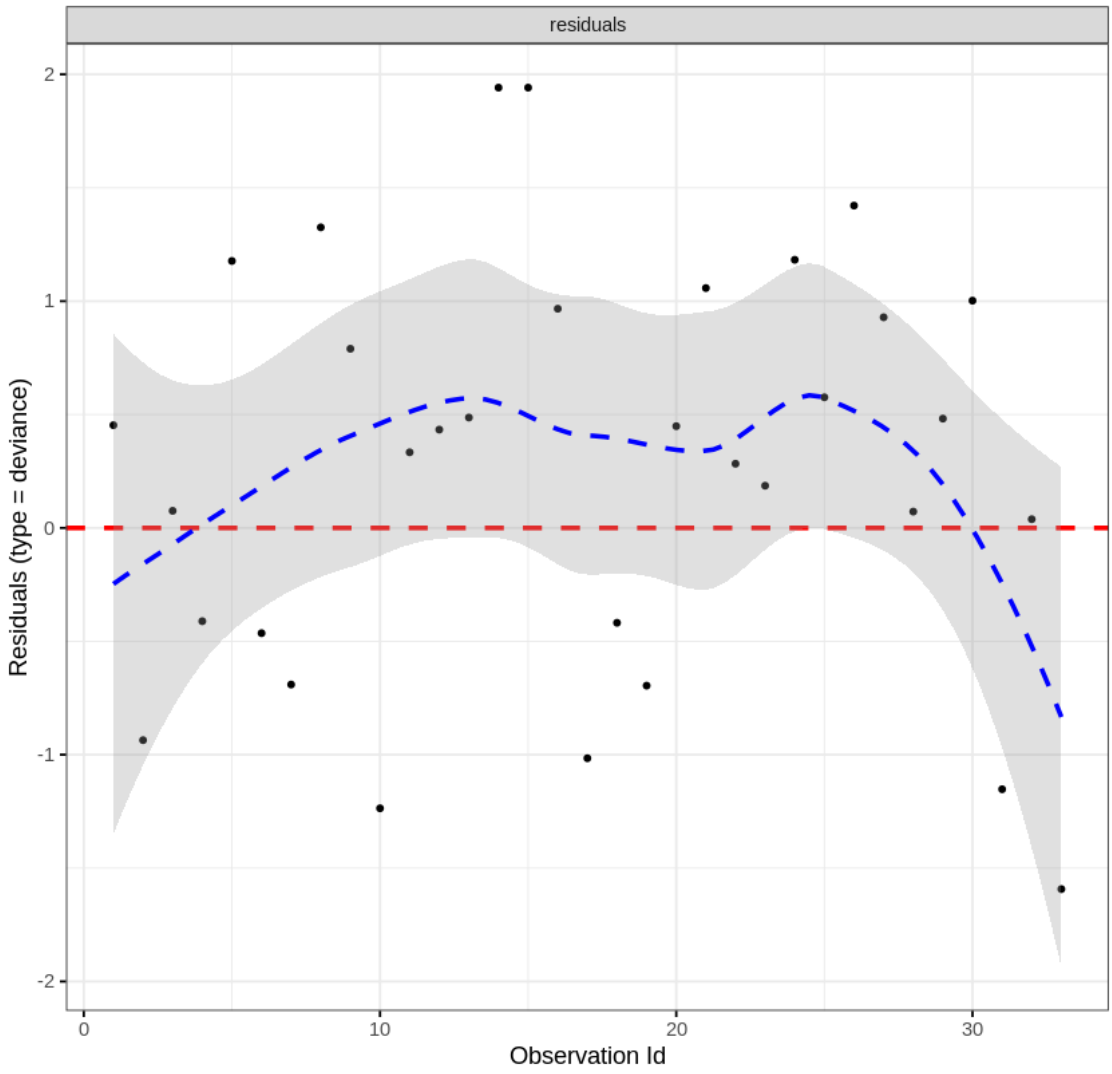
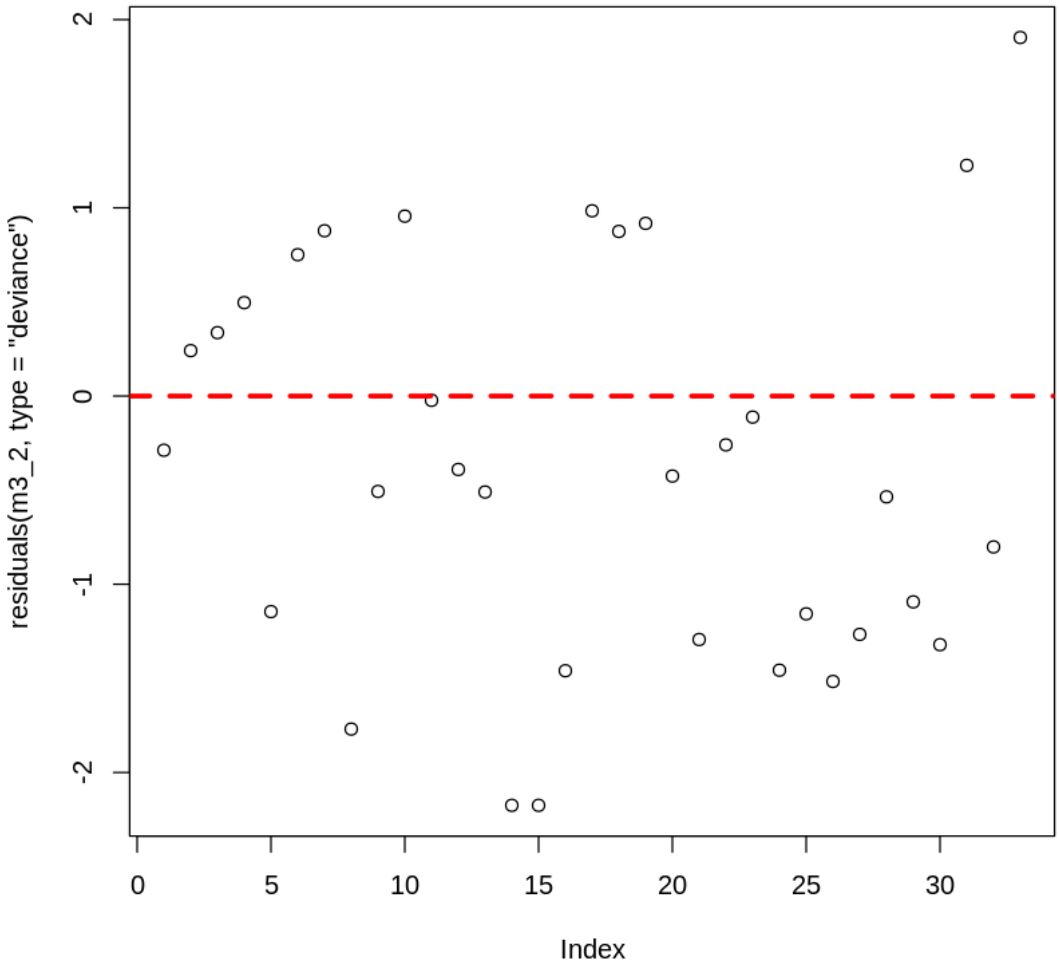


Figure 3.10: Exponentional Model Deviance Residuals



[41]:
```
#Sum of Squared Residuals

sum(residuals(m2_2, type="deviance")^2) #cox ph
sum(residuals(m3_2, type="deviance")^2) #exp
```

29.0977121289924

40.3190891122841

[42]:
```
#AIC

AIC(m2_2, m3_2)
```

A data.frame: 2 × 2

	df <dbl>	AIC <dbl>
m2_2	2	161.3633
m3_2	3	299.0810