Acosta, Raphael Zeth Acosta
Advincula, Audrey Johannah
Bautista, Anedy Johane
Guanlao, Cianne Paulette

1. Partition the data into 80:20 Train-Test

Unset

```
# Library ========
library(car)
library(caret
library(measures)
library(DescTools)

# Data Partitioning ========
indexsetframe = sample(2,nrow(SENIC),replace = T, prob = c(0.80,0.20))
trainsenic  = SENIC[indexsetframe==1,]
testsenic = SENIC[indexsetframe==2,]
```

2. Using an appropriate GLM, suggest at least two possible models. For each model, follow these steps.
    a. Choose and explain the use of the distribution and link for the GLM, perform variable selection. Justify the removal or inclusion of the variables. It is highly suggested to check the correlation matrix of the variables to check whether some of the predictors will be highly correlated. You can also use LRT and Wald's tests to check which variables are significant.

Unset

```
#### CORRELATIONAL MATRIX ####

corr_matrix <- cor(SENIC,method="spearman")

# HIGH CORRELATION:
# bed - census
# bed - nurses
# bed - facilities
# census - nurses
# census - facilities
# facilities - nurses

trainsenic$region = as.factor(trainsenic$region)
testsenic$region = as.factor(testsenic$region)
```

```
#============================ LOG ============================#

#### GLM SATURATED with many variables (LOG)
log_glmsat1a = glm(stay ~ age+inf+region+census, family = gaussian(link =
"log"), data = train_data)
summary(log_glmsat1a)

log_glmsat1b = glm(stay ~ age+inf+region+nurses, family = gaussian(link =
"log"), data = train_data)
summary(log_glmsat1b)

log_glmsat2a = glm(stay ~ age+region+census, family = gaussian(link = "log"),
data = train_data)
summary(log_glmsat2a)

log_glmsat2b = glm(stay ~ inf+region+census, family = gaussian(link = "log"),
data = train_data)
summary(log_glmsat2b)

# PROPOSED GLM with 2 variables
log_glmprop = glm(stay ~ inf+nurses, family = gaussian(link = "log"),data =
train_data)
summary(log_glmprop)

anova(log_glmprop,log_glmsat1a,test = "Chisq") #LOWEST - 6.152e-13
anova(log_glmprop,log_glmsat1b,test = "Chisq") #-5.9983
anova(log_glmprop,log_glmsat2a,test = "Chisq") #1.172e-07
anova(log_glmprop,log_glmsat2b,test = "Chisq") #0.004667

# DEVIANCE TEST
summary(log_glmsat1a)
summary(log_glmprop)

model_log = log_glmsat1a
prop_log = log_glmprop

deviance(model_log) #174.1627
deviance(prop_log) #132.0482

#AIC
AIC(model_log) #332.2699
AIC(prop_log) #308.5254
```

```
### LRT TEST INTERPRETATION (LOG) ###

a = 0.05
p-value of log_glmsat1a (LOWEST)  = 6.152e-13
Therefore, we REJECT the null hypothesis. There is sufficient evidence to state
that the saturated model(log_glmsat1a) with variables (age, inf, region,
census) contributes well to model fitness or the prediction of depedent
variable (stay)


#============================ INVERSE ============================#

inverse_glmsat1a = glm(stay ~ age+inf+region+census, family = gaussian(link =
"inverse"), data = train_data)
summary(inverse_glmsat1a)

inverse_glmsat1b = glm(stay ~ age+inf+region+nurses, family = gaussian(link =
"inverse"), data = train_data)
summary(inverse_glmsat1b)

inverse_glmsat2a = glm(stay ~ age+region+census, family = gaussian(link =
"inverse"), data = train_data)
summary(inverse_glmsat2a)

inverse_glmsat2b = glm(stay ~ inf+region+census, family = gaussian(link =
"inverse"), data = train_data)
summary(inverse_glmsat2b)

# INVERSE PROPOSED GLM with 2 variables
inverse_glmprop = glm(stay ~ age+inf+region+census, family = gaussian(link =
"inverse"), data = train_data)
summary(inverse_glmprop)

#Intercept Model
inverse_glmnull = glm(stay~1,family = gaussian(link = "inverse"),data =
train_data)
summary(inverse_glmnull)

#Full model
inverse_glmfull = glm(stay~.,family = gaussian(link = "inverse"),data =
train_data)
summary(inverse_glmfull)
```

```
#LRT TEST
anova(inverse_glmprop,inverse_glmsat1a,test = "Chisq") #LOWEST = -3.35e-12
anova(inverse_glmprop,inverse_glmsat1b,test = "Chisq") #-5.703
anova(inverse_glmprop,inverse_glmsat2a,test = "Chisq") #8.219e-09
anova(inverse_glmprop,inverse_glmsat2b,test = "Chisq") #0.005551

# DEVIANCE TEST
summary(inverse_glmsat1a)
summary(inverse_glmprop)

model_inverse = inverse_glmsat1a
prop_inverse = inverse_glmprop

deviance(model_inverse) #170.9868
deviance(prop_inverse) #124.1256

#AIC
AIC(model_inverse) #330.5584
AIC(prop_inverse) #302.7711

### LRT TEST INTERPRETATION (INVERSE) ###

a = 0.05
p-value = 3.35e-12

Therefore, we REJECT the null hypothesis. There is sufficient evidence to state
that the saturated model(inverse_glmsat1a) with variables (age, inf, region,
census) contributes well to model fitness or the prediction of depedent
variable (stay)



#============================ IDENTITY ============================#

identity_glmsat1a = glm(stay ~ age+inf+region+census, family = gaussian(link =
"identity"), data = train_data)
summary(identity_glmsat1a)

identity_glmsat1b = glm(stay ~ age+inf+region+nurses, family = gaussian(link =
"identity"), data = train_data)
summary(identity_glmsat1b)

identity_glmsat2a = glm(stay ~ age+region+census, family = gaussian(link =
"identity"), data = train_data)
```

```
summary(identity_glmsat2a)

identity_glmsat2b = glm(stay ~ inf+region+census, family = gaussian(link =
"identity"), data = train_data)
summary(identity_glmsat2b)

# identity PROPOSED GLM with 2 variables
identity_glmprop = glm(stay ~ age+inf+region+census, family = gaussian(link =
"identity"), data = train_data)
summary(identity_glmprop)

# PROPOSED GLM with 2 variables
identity_glmprop = glm(stay ~ age+inf+region+census, family = gaussian(link =
"identity"), data = train_data)
summary(identity_glmprop)

#Intercept Model
identity_glmnull = glm(stay~1,family = gaussian(link = "identity"),data =
train_data)
summary(identity_glmnull)

#Full model
glmfull = glm(stay~.,family = gaussian(link = "identity"),data = train_data)
summary(glmfull)

# LRT TEST
anova(identity_glmprop,identity_glmsat1a,test = "Chisq") #LOWEST - 2.615e-08
anova(identity_glmprop,identity_glmsat1b,test = "Chisq") #-6.2406
anova(identity_glmprop,identity_glmsat2a,test = "Chisq") #-36.941
anova(identity_glmprop,identity_glmsat2b,test = "Chisq") #-11.39

# DEVIANCE TEST
summary(identity_glmsat1a)
summary(identity_glmprop)

model_identity = identity_glmsat1a
prop_identity = identity_glmprop

deviance(model_identity) #177.6099
deviance(prop_identity) #140.6099

# AIC
AIC(model_identity) #334.0615
AIC(prop_identity) #314.3678
```

```
### LRT TEST INTERPRETATION (IDENTITY) ###

a = 0.05
p-value = 2.615e-08

Therefore, we REJECT the null hypothesis. There is sufficient evidence to state
that the saturated model(identity_glmsat1a) with variables (age, inf, region,
census) contributes well  to model fitness or the prediction of depedent
variable (stay)
```

b. Using the discussed evaluation measures for predictive ability, compare the performance of the two models. Based on this, choose your final model.

Unset

```
#### PREDICTION METRICS #####

glmlog = glm(stay ~ age+inf+region+census,
                   family = gaussian(link = "log"), data = trainsenic)

glmMLR = glm(stay ~ age+inf+region+census,
             family = gaussian(link = "identity"), data = trainsenic)

glminverse = glm(stay ~ age+inf+region+census,
                 family = gaussian(link = "inverse"), data = trainsenic)


AIC(glmlog,glmMLR,glminverse)

For the three models, glmlog, glmMLR and glminverse,
glminverse has a lower AIC compared to glmlog and glmMLR,
glminverse is considered the better model in terms of the trade-off
between goodness of fit and complexity.


#### Predictive Capability of LOG ####

predict_log = predict(glmlog, newdata = testsenic)

MAE_log = mean(abs(testsenic$stay-predict_log))

MSE_log = mean((testsenic$stay-predict_log)^2)
```

```
RMSE_log = sqrt(MSE_log)

MAPE_log = mean(abs(testsenic$stay-predict_log)/testsenic$stay)

log_list <-
list(c("MSE"=MSE_log,"RMSE"=RMSE_log,"MAE"=MAE_log,"MAPE"=MAPE_log))

#=====INTERPRETATION====#

The MAE value of 7.841 indicates that, on average, the absolute difference
between the predictions and actual values of the dependent variable ’stay’
is approximately 7.841 units.
The MSE value of 67.36 represents the average squared error between predicted
and actual values.
The RMSE value of 8.207, which is the square root of MSE, suggests that, on
average, the predictions are approximately 8.207 units close to the actual
values.
The MAPE of the GLM model is 76 percent which suggests that the model has a
substantial prediction
error, as it tends to deviate from the actual target values by 76% on average.

#### Predictive Capability of IDENTITY ####

predict_MLR = predict(glmMLR, newdata = testsenic)

MAE_MLR = mean(abs(testsenic$stay-predict_MLR))

MSE_MLR = mean((testsenic$stay-predict_MLR)^2)

RMSE_MLR = sqrt(MSE_MLR)

MAPE_MLR = mean(abs(testsenic$stay-predict_MLR)/testsenic$stay)

mlr_list <-
list(c("MSE"=MSE_MLR,"RMSE"=RMSE_MLR,"MAE"=MAE_MLR,"MAPE"=MAPE_MLR))

#=====INTERPRETATION====#

the MAE value of 1.133 indicates that, on average, the absolute difference
between the predictions and actual values of the dependent variable ’stay’
is approximately 1.133 units.
The MSE value of 3.541 represents the average squared error between predicted
and actual values.
```

```
The RMSE value of 1.88, which is the square root of MSE, suggests that, on
average, the
predictions are approximately 1.88 units close to the actual values.
The MAPE of the GLM model is 10 percent which suggests that the model has a
substantial prediction
error, as it tends to deviate from the actual target values by 10% on average.


#### Predictive Capability of INVERSE ####

predict_inverse = predict(glminverse, newdata = testsenic)

MAE_inverse = mean(abs(testsenic$stay-predict_inverse))

MSE_inverse = mean((testsenic$stay-predict_inverse)^2)

RMSE_inverse = sqrt(MSE_inverse)

MAPE_inverse = mean(abs(testsenic$stay-predict_inverse)/testsenic$stay)

inverse_list <-
list(c("MSE"=MSE_inverse,"RMSE"=RMSE_inverse,"MAE"=MAE_inverse,"MAPE"=MAPE_inve
rse))

#=====INTERPRETATION====#

The MAE value of 10.01 indicates that, on average, the absolute difference
between the predictions and actual values of the dependent variable ’stay’
is approximately 10.01 units.
The MSE value of 106.61 represents the average squared error between predicted
and actual values.
The RMSE value of 10.33, which is the square root of MSE, suggests that, on
average, the
predictions are approximately 10.33 units close to the actual values.
The MAPE of the GLM model is 99 percent which suggests that the model has a
substantial prediction
error, as it tends to deviate from the actual target values by 99% on average.
####### PREDICTIVE CAPABILITY COMPARISON #######

Based on the three predictive measures, it would seem that the MLR model would
be the better model
in terms of the predictive capability since it produced significantly lower
values.
```

```
#### PSEUDO R-SQUARED ####

PseudoR2(glmlog, which = c("McFadden","CoxSnell","Nagelkerke","Efron"))
PseudoR2(glmMLR, which = c("McFadden","CoxSnell","Nagelkerke","Efron"))
PseudoR2(glminverse, which = c("McFadden","CoxSnell","Nagelkerke","Efron"))

####### PPSEUDO R-SQUARED INTERPRETATION #######

Based on the Pseudo R-squared values produced by the different models, it
suggests that the
Inverse Model would be the better model in terms of the explainability of the
response variable "stay"


#=============================== OVERALL SUMMARY ===============================#

      The MLR model performs better based on the MSE, RMSE, and MAE,and MAPE
indicating it is more accurate in terms of prediction error.

      The inverse model performs better in terms of pseudo R² values,
suggesting it might explain more the variance in the data.
```

c. Provide an interpretation for each of the coefficient estimates. Show also your solution.

Unset

```
##### FINAL MODEL #####

final_glmMLR = glm(stay ~ age+inf+region+census,
             family = gaussian(link = "identity"), data = trainsenic)
```

Interpreting the coefficients of GLM linked with the IDENTITY function

Since the chosen model is a GLM with Identity Link, the equation for it is as follows:

- $\mu_0 = \beta_0 + \beta_1 x_0 + \beta_2 x_1 + \beta_3 x_2 + \beta_4 x_3 + \beta_5 x_4 + \beta_6 x_5$

Given the variables of the final model, let x_0 be the point *age*, x_1 be the point *inf*, x_2 be the point *region* when region = 2 (NC), x_3 be the point *region* when region = 3 (S), x_4 be the point *region* when region = 4 (W), and x_5 be the point *census.*

- $\mu_0 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_3(region2) + \beta_4(region3) + \beta_5(region4) + \beta_6(census)$

Interpretation for the *age* predictor:

- Increase the *age* variable by 1 unit; and let μ_1 be the new mean with *age+1,* while keeping the other variables fixed.
- $\mu_1 = \beta_0 + \beta_1(age + 1) + \beta_2(inf) + \beta_3(region2) + \beta_4(region3) + \beta_5(region4) + \beta_6(census)$
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_1 + \beta_2(inf) + \beta_3(region2) + \beta_4(region3) + \beta_5(region4) + \beta_6(census)$
- $\mu_1 = \mu_0 + \beta_1$
- Substitute the coefficient of ***age*** to β_1 ;
- $\mu_1 = \mu_0 + 0.0987453$
- **Conclusion:** For every unit increase in the **age predictor**, the mean length of ***stay*** of all patients in hospitals (in days) increases by ***0.0987453***

Interpretation for the *inf* predictor:

- Increase the *inf* variable by 1 unit; and let μ_1 be the new mean with *inf+1,* while keeping the other variables fixed.
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf + 1) + \beta_3(region2) + \beta_4(region3) + \beta_5(region4) + \beta_6(census)$
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_2 + \beta_3(region2) + \beta_4(region3) + \beta_5(region4) + \beta_6(census)$
- $\mu_1 = \mu_0 + \beta_2$
- Substitute the coefficient of ***inf*** to β_2 ;
- $\mu_1 = \mu_0 + 0.5635910$
- **Conclusion:** For every unit increase in the **inf predictor**, the mean length of ***stay*** of all patients in hospitals (in days) increases by ***0.5635910***

Interpretation for the *region* predictor (region2, region3, region4):

- Considering the region variable as a categorical attribute, we will set region1 (NE) as the dummy variable when all other dummy variables (region2, region3, region4) are equal to zero. Then, the equation will be simplified and then serve as the reference point:
  - $\mu_0 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_6(census)$

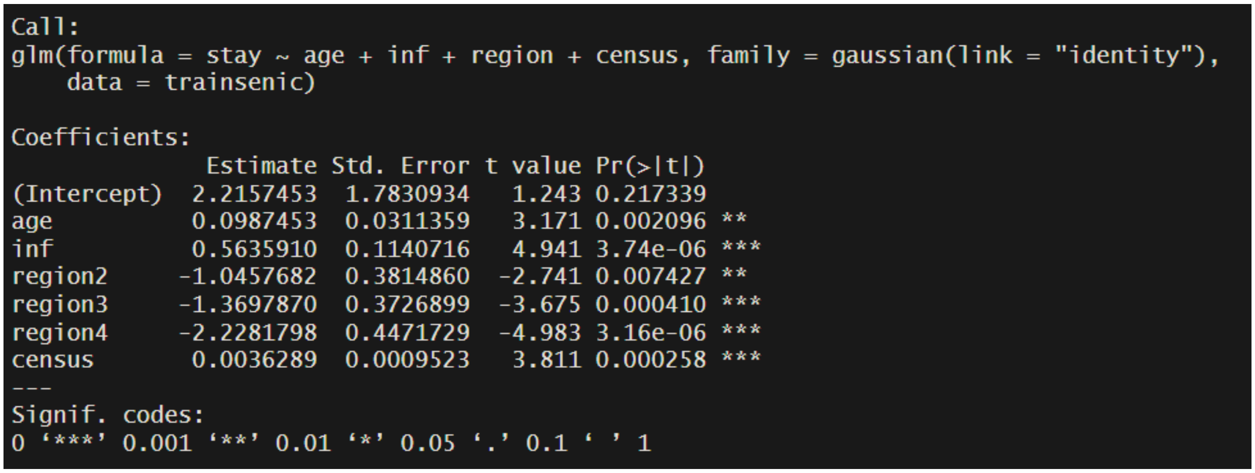
- **REGION2**
- The value of **region2 will be set to 1**, while region3 and region4 will be set to 0 to avoid collinearity problems.
- Let μ_1 be the new mean with *region2 = 1, region3 = 0, and region4 = 0,* while keeping the rest of the variables fixed.
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_3(1) + \beta_4(0) + \beta_5(0) + \beta_6(census)$
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_3 + \beta_6(census)$
- $\mu_1 = \mu_0 + \beta_3$
- Substitute the coefficient of ***region2*** to β_3 ;
- $\mu_1 = \mu_0 + (-1.0457682)$
- **Conclusion:** If a patient is located at geographic region (NC), the mean length of ***stay*** of all patients in hospitals (in days) is lower by ***1.0457682*** as compared to those located at geographic region (NE).

- **REGION3**
- The value of **region3 will be set to 1**, while region2 and region4 will be set to 0 to avoid collinearity problems.
- Let μ_1 be the new mean with *region3 = 1, region2 = 0, and region4 = 0,* while keeping the rest of the variables fixed.
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_3(0) + \beta_4(1) + \beta_5(0) + \beta_6(census)$
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_4 + \beta_6(census)$
- $\mu_1 = \mu_0 + \beta_4$
- Substitute the coefficient of ***region3*** to β_4;
- $\mu_1 = \mu_0 + (-\ 1.3697870)$
- **Conclusion:** If a patient is located at geographic region (S), the mean length of ***stay*** of all patients in hospitals (in days) is lower by ***1.3697870*** as compared to those located at geographic region (NE).

- **REGION4**
- The value of **region4 will be set to 1**, while region2 and region3 will be set to 0 to avoid collinearity problems.
- Let μ_1 be the new mean with *region4 = 1, region2 = 0, and region3 = 0,* while keeping the rest of the variables fixed.
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_3(0) + \beta_4(0) + \beta_5(1) + \beta_6(census)$
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_5 + \beta_6(census)$
- $\mu_1 = \mu_0 + \beta_5$
- Substitute the coefficient of ***region4*** to β_5;
- $\mu_1 = \mu_0 + (-\ 2.2281798)$
- **Conclusion:** If a patient is located at geographic region (W), the mean length of ***stay*** of all patients in hospitals (in days) is lower by ***2.2281798*** as compared to those located at geographic region (NE).

Interpretation for the *census* predictor:

- Increase the *census* variable by 1 unit; and let μ_1 be the new mean with *census+1,* while keeping the other variables fixed.
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_3(region2) + \beta_4(region3) + \beta_5(region4) + \beta_6(census + 1)$
- $\mu_1 = \beta_0 + \beta_1(age) + \beta_2(inf) + \beta_2 + \beta_3(region2) + \beta_4(region3) + \beta_5(region4) + \beta_6(census) + \beta_6$
- $\mu_1 = \mu_0 + \beta_6$
- Substitute the coefficient of ***census*** to β_6 ;
- $\mu_1 = \mu_0 + 0.0036289$
- **Conclusion:** For every unit increase in the **census predictor**, the mean length of ***stay*** of all patients in hospitals (in days) increases by ***0.0036289***



```
Call:
glm(formula = stay ~ age + inf + region + census, family = gaussian(link = "identity"),
    data = trainsenic)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.2157453  1.7830934   1.243 0.217339
age          0.0987453  0.0311359   3.171 0.002096 **
inf          0.5635910  0.1140716   4.941 3.74e-06 ***
region2     -1.0457682  0.3814860  -2.741 0.007427 **
region3     -1.3697870  0.3726899  -3.675 0.000410 ***
region4     -2.2281798  0.4471729  -4.983 3.16e-06 ***
census       0.0036289  0.0009523   3.811 0.000258 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```