logistic

STAT 6021

2024-08-07

Load In Data

```
sleep <- read.csv("Sleep_Efficiency.csv")

# Remove ID, bedtime, and wakeup time
sleep <- sleep[, -c(1, 4, 5)]
```

Cleaning

```
# Weighted sleep (Sleep.duration * Sleep.efficiency)
sleep <- sleep %>%
  mutate(weighted.sleep = Sleep.duration * Sleep.efficiency)

sleep <- na.omit(sleep[, c("Sleep.duration","Sleep.efficiency", "Age", "Caffeine.consumption", "Alcohol

sleep$Enough.Sleep <- ifelse(
  sleep$Sleep.duration >= 7.5,
  1,
  0)
```

Initial Logit with >= 7.5 Hours

```
logit <- glm(
  Enough.Sleep ~ Age + Caffeine.consumption +
    Alcohol.consumption + Smoking.status + Exercise.frequency,
  sleep,
  family="binomial"
)
summary(logit)
```

```
##
## Call:
## glm(formula = Enough.Sleep ~ Age + Caffeine.consumption + Alcohol.consumption +
##     Smoking.status + Exercise.frequency, family = "binomial",
##     data = sleep)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.484  -1.272   0.954   1.063   1.339
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)           0.6545409  0.3809445   1.718   0.0858 .
## Age                  -0.0086233  0.0076968  -1.120   0.2626
## Caffeine.consumption  0.0009779  0.0035708   0.274   0.7842
## Alcohol.consumption  -0.1097320  0.0629527  -1.743   0.0813 .
## Smoking.statusYes     0.2199600  0.2143809   1.026   0.3049
## Exercise.frequency   -0.0095662  0.0697793  -0.137   0.8910
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 557.81  on 406  degrees of freedom
## Residual deviance: 551.94  on 401  degrees of freedom
## AIC: 563.94
##
## Number of Fisher Scoring iterations: 4
```

It looks like we do not have many significant variables below the 0.1 threshold, only our intercept and alcohol consumption. What we can do is we can compare this with a full model forward selection and see which one this selects to select variables.

```r
# Null model (no predictors)
null_model <- glm(Enough.Sleep ~ 1, data = sleep, family = binomial)

# Full model (all predictors)
full_model <- glm(
  Enough.Sleep ~ Age + Caffeine.consumption +
    Alcohol.consumption + Smoking.status + Exercise.frequency,
  data = sleep,
  family = binomial
)

# Forward selection
forward_model <- step(null_model,
                      scope = list(lower = null_model, upper = full_model),
                      direction = "forward")
```

```
## Start:  AIC=559.81
## Enough.Sleep ~ 1
##
##                        Df Deviance    AIC
## + Alcohol.consumption   1   554.54 558.54
## <none>                      557.81 559.81
## + Age                   1   556.09 560.09
## + Smoking.status        1   556.97 560.97
## + Caffeine.consumption  1   557.33 561.33
## + Exercise.frequency    1   557.72 561.72
##
## Step:  AIC=558.54
## Enough.Sleep ~ Alcohol.consumption
##
##                        Df Deviance    AIC
## <none>                      554.54 558.54
## + Age                   1   553.14 559.14
## + Smoking.status        1   553.46 559.46
```

```
## + Caffeine.consumption  1   554.29 560.29
## + Exercise.frequency    1   554.46 560.46
```

```
# Summary of the selected model
summary(forward_model)
```

```
##
## Call:
## glm(formula = Enough.Sleep ~ Alcohol.consumption, family = binomial,
##     data = sleep)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.344  -1.295   1.019   1.019   1.254
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)           0.38391    0.12443   3.085  0.00203 **
## Alcohol.consumption -0.11229    0.06218  -1.806  0.07097 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 557.81  on 406  degrees of freedom
## Residual deviance: 554.54  on 405  degrees of freedom
## AIC: 558.54
##
## Number of Fisher Scoring iterations: 4
```

Now, we have a significant intercept, and the p value for alcohol consumption is lower. This does not give us many variables to work with, so it looks like we should go back and explore more, maybe with interaction effects. Exercise frequency continues to perform poorly, so let's remove it and rerun the logistic, but accounting for all interaction terms.

```
logit <- glm(
  Enough.Sleep ~ (Age + Caffeine.consumption + Alcohol.consumption + Smoking.status)^2,
  data = sleep,
  family = binomial
)

summary(logit)
```

```
##
## Call:
## glm(formula = Enough.Sleep ~ (Age + Caffeine.consumption + Alcohol.consumption +
##     Smoking.status)^2, family = binomial, data = sleep)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7960  -1.2341   0.8788   1.0817   1.5745
##
## Coefficients:
##                                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)                             8.883e-01  5.440e-01   1.633   0.1025
## Age                                    -1.597e-02  1.263e-02  -1.264   0.2062
```

```
## Caffeine.consumption                       1.560e-03  1.302e-02   0.120   0.9046
## Alcohol.consumption                       -4.294e-01  2.287e-01  -1.877   0.0605
## Smoking.statusYes                          9.285e-01  8.875e-01   1.046   0.2955
## Age:Caffeine.consumption                  -8.801e-05  3.138e-04  -0.280   0.7791
## Age:Alcohol.consumption                    8.946e-03  4.919e-03   1.819   0.0689
## Age:Smoking.statusYes                     -1.204e-02  1.777e-02  -0.678   0.4980
## Caffeine.consumption:Alcohol.consumption   1.502e-03  2.690e-03   0.558   0.5766
## Caffeine.consumption:Smoking.statusYes     5.217e-03  9.033e-03   0.578   0.5636
## Alcohol.consumption:Smoking.statusYes     -2.138e-01  1.386e-01  -1.542   0.1230
##
## (Intercept)
## Age
## Caffeine.consumption
## Alcohol.consumption                       .
## Smoking.statusYes
## Age:Caffeine.consumption
## Age:Alcohol.consumption                   .
## Age:Smoking.statusYes
## Caffeine.consumption:Alcohol.consumption
## Caffeine.consumption:Smoking.statusYes
## Alcohol.consumption:Smoking.statusYes
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 557.81  on 406  degrees of freedom
## Residual deviance: 543.48  on 396  degrees of freedom
## AIC: 565.48
##
## Number of Fisher Scoring iterations: 4
```

It looks like Age and Alcohol consumption have some interaction and Alcohol consumption and smoking too, though not at a 0.1 threshold. Let's include these variables in our model and see what happens.

```
logit <- glm(
  Enough.Sleep ~ Age*(Alcohol.consumption + Smoking.status),
  data = sleep,
  family = binomial
)

summary(logit)
```

```
##
## Call:
## glm(formula = Enough.Sleep ~ Age * (Alcohol.consumption + Smoking.status),
##     family = binomial, data = sleep)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6326  -1.2448   0.8897   1.0899   1.6031
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)              0.979366   0.447606   2.188  0.02867 *
```

```
## Age                      -0.017132   0.010750  -1.594  0.11100
## Alcohol.consumption      -0.549641   0.210003  -2.617  0.00886 **
## Smoking.statusYes         0.893349   0.741598   1.205  0.22835
## Age:Alcohol.consumption   0.010450   0.004812   2.172  0.02988 *
## Age:Smoking.statusYes    -0.014206   0.016876  -0.842  0.39990
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 557.81  on 406  degrees of freedom
## Residual deviance: 546.81  on 401  degrees of freedom
## AIC: 558.81
##
## Number of Fisher Scoring iterations: 4
```

Interesting. It also looks like alcohol on its own is very important on its own, given how low the p value is for the beta coefficient estimate. Let's run a forward selection with these variables to see what the best performing model is, since it looks like we have some more significant items.

```
null_model <- glm(Enough.Sleep ~ 1, data = sleep, family = binomial)

full_model <- glm(
  Enough.Sleep ~ Age*(Alcohol.consumption + Smoking.status),
  data = sleep,
  family = binomial
)

forward_model <- step(null_model,
                      scope = list(lower = null_model, upper = full_model),
                      direction = "forward")
```

```
## Start:  AIC=559.81
## Enough.Sleep ~ 1
##
##                        Df Deviance    AIC
## + Alcohol.consumption   1   554.54 558.54
## <none>                      557.81 559.81
## + Age                   1   556.09 560.09
## + Smoking.status        1   556.97 560.97
##
## Step:  AIC=558.54
## Enough.Sleep ~ Alcohol.consumption
##
##                   Df Deviance    AIC
## <none>                 554.54 558.54
## + Age              1   553.14 559.14
## + Smoking.status   1   553.46 559.46
```

```
# Summary of the selected model
summary(forward_model)
```

```
##
## Call:
## glm(formula = Enough.Sleep ~ Alcohol.consumption, family = binomial,
##     data = sleep)
```

```
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.344  -1.295   1.019   1.019   1.254
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)          0.38391    0.12443   3.085  0.00203 **
## Alcohol.consumption -0.11229    0.06218  -1.806  0.07097 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 557.81  on 406  degrees of freedom
## Residual deviance: 554.54  on 405  degrees of freedom
## AIC: 558.54
##
## Number of Fisher Scoring iterations: 4
```

Again, the most important variable is going to be using alcohol consumption, and it still falls below above the 0.05 p value, but the intercept and alcohol are both still significant at the 0.1 threshold, indicating that the model may be somewhat useful. This is the model with the lowest AIC, but let us compare this one and the one above (which had multiple beta coefficient estimates below $p = 0.05$ rather than one) for predictions.

Example Prediction

Instead of measuring by the AIC for the best model, let's compare the results from the residuals for our data.

```
new.data <- sleep %>%
  select(
    Age,
    Alcohol.consumption,
    Smoking.status
  )

new.data.forward <- data.frame(Alcohol.consumption=sleep$Alcohol.consumption)

pred.logit <- ifelse(
  predict(logit, newdata = new.data, type = "response") > 0.5,1,0)

pred.forward <- ifelse(
  predict(forward_model, newdata = new.data.forward, type = "response") > 0.5,1,0)
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```r
confusion_matrix_logit <- confusionMatrix(
  as.factor(pred.logit),
  as.factor(sleep$Enough.Sleep)
)

print(confusion_matrix_logit)
```

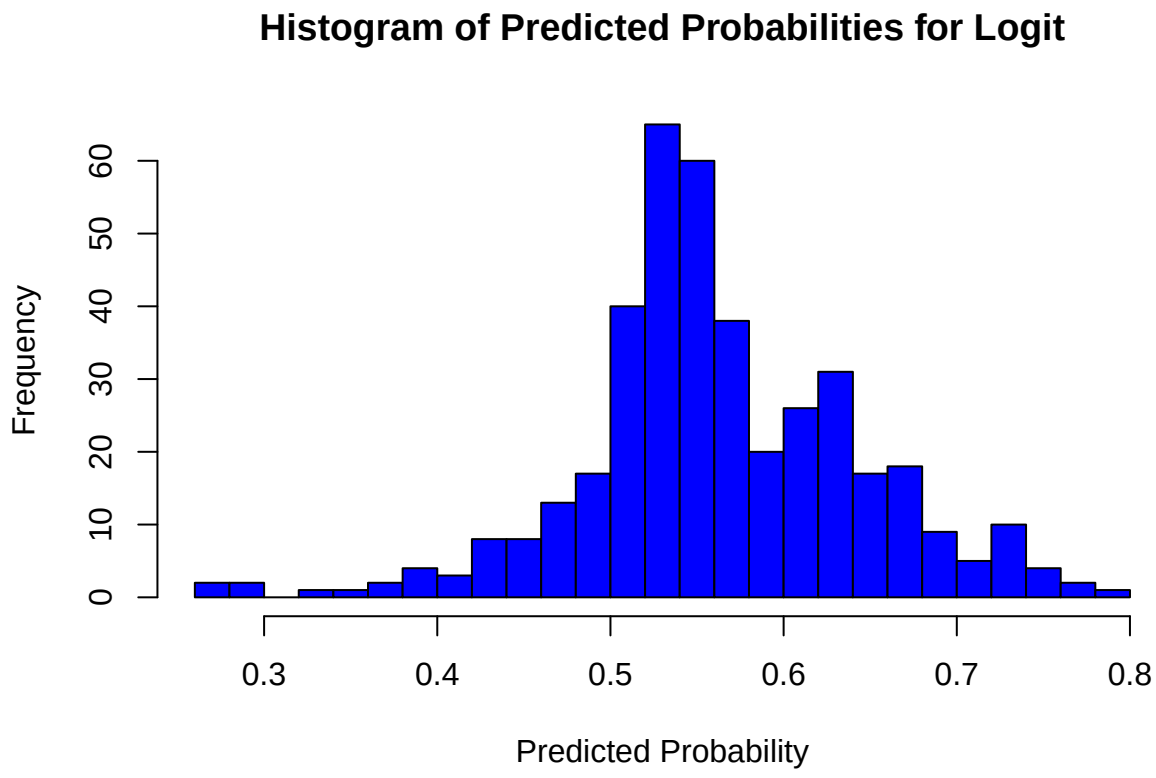
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0  30  31
##          1 148 198
##
##                Accuracy : 0.5602
##                  95% CI : (0.5105, 0.6091)
##     No Information Rate : 0.5627
##     P-Value [Acc > NIR] : 0.5604
##
##                   Kappa : 0.0358
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.16854
##             Specificity : 0.86463
##          Pos Pred Value : 0.49180
##          Neg Pred Value : 0.57225
##              Prevalence : 0.43735
##          Detection Rate : 0.07371
##    Detection Prevalence : 0.14988
##       Balanced Accuracy : 0.51658
##
##        'Positive' Class : 0
##
```

```r
confusion_matrix_forward <- confusionMatrix(
  as.factor(pred.forward),
  as.factor(sleep$Enough.Sleep)
)

print(confusion_matrix_forward)
```

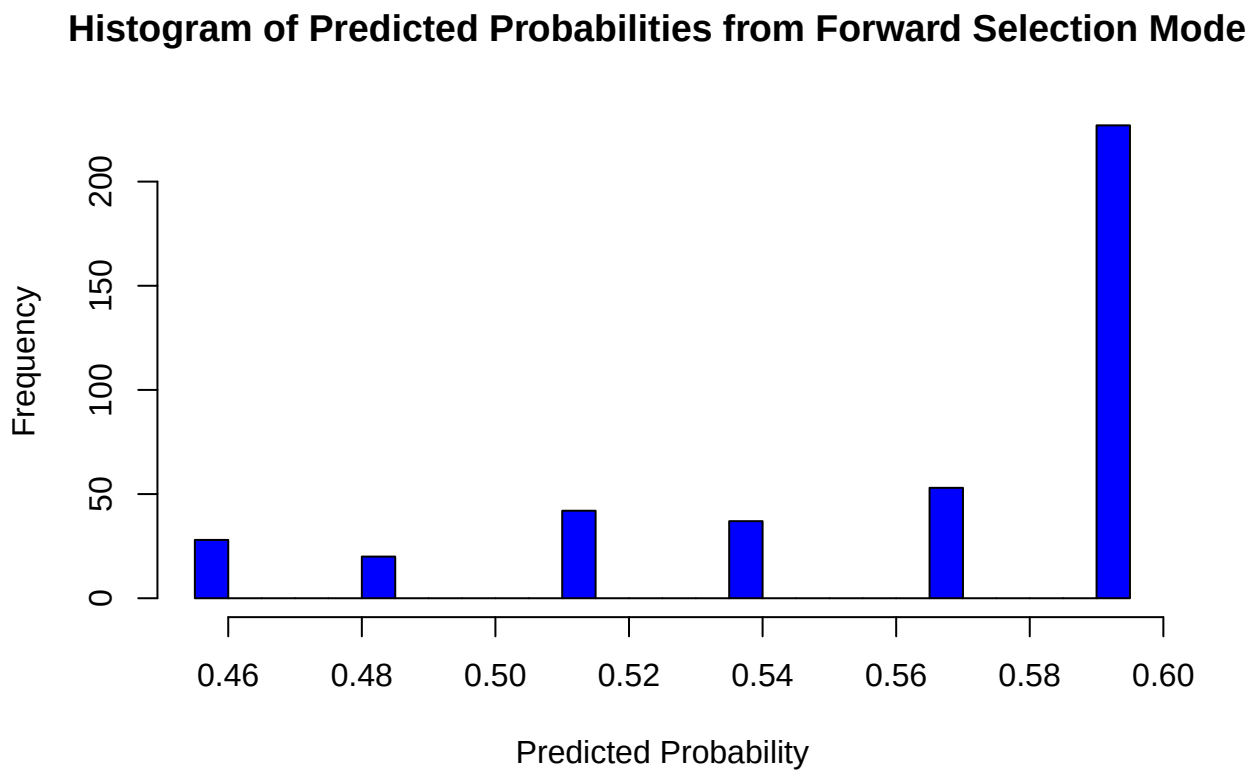
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0  30  18
##          1 148 211
##
##                Accuracy : 0.5921
##                  95% CI : (0.5426, 0.6403)
##     No Information Rate : 0.5627
##     P-Value [Acc > NIR] : 0.1251
##
```

```
##                  Kappa : 0.0979
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.16854
##             Specificity : 0.92140
##          Pos Pred Value : 0.62500
##          Neg Pred Value : 0.58774
##              Prevalence : 0.43735
##          Detection Rate : 0.07371
##    Detection Prevalence : 0.11794
##       Balanced Accuracy : 0.54497
##
##        'Positive' Class : 0
##
```

```r
library(ggplot2)

predicted_probabilities <- predict(logit, newdata = new.data, type = "response")

# Plot histogram
hist(predicted_probabilities, breaks = 20, col = "blue",
     main = "Histogram of Predicted Probabilities for Logit",
     xlab = "Predicted Probability", ylab = "Frequency",
     border = "black")
```



```r
predicted_probabilities <- predict(
  forward_model, newdata = new.data.forward, type = "response")

# Plot histogram
hist(predicted_probabilities, breaks = 20, col = "blue",
     main = "Histogram of Predicted Probabilities from Forward Selection Model",
     xlab = "Predicted Probability", ylab = "Frequency",
     border = "black")
```



Both models result in similarly low scores for accuracy, around 50%. Still, both models are above that mark, and the one that only uses alcohol has an accuracy around 59%, which is also higher than the model from logistic regression with multiple variables instead of only alcohol.

This above histogram plot starts on the right side for 0 drinks, and goes towards the left for each additional drink.

Example prediction

We can see the histogram from above to see what happens with the number of drinks and likeliness of not enough sleep. Our beta coefficient is -0.11702. That means for each additional drink, the odds of getting enough sleep decrease by 11.04%.

```r
(1 - exp(-0.11702) ) * 100
```

```
## [1] 11.04326
```

Suppose we are at a party and we are thinking how many drinks we can have in order to get enough sleep. Let's plot our model and see the number of drinks we can have to still be over 50% predicted probability.

```
sapply(seq(0, 5), function(x) {
  prob <- exp(0.39630 + -0.11702 * x) / (1 + exp(0.39630 + -0.11702 * x))
  cat(paste0("Drink ", x, ": ", round(prob,2), "\n"))
});
```

```
## Drink 0: 0.6
## Drink 1: 0.57
## Drink 2: 0.54
## Drink 3: 0.51
## Drink 4: 0.48
## Drink 5: 0.45
```

```
## [[1]]
## NULL
##
## [[2]]
## NULL
##
## [[3]]
## NULL
##
## [[4]]
## NULL
##
## [[5]]
## NULL
##
## [[6]]
## NULL
```

It looks like at drink 4, we are no longer predicting enough sleep.