

```
library(tidyverse)    # Data manipulation
```

```
## Warning: package 'tidyverse' was built under R version 4.4.3
```

```
## Warning: package 'readr' was built under R version 4.4.3
```

```
## Warning: package 'dplyr' was built under R version 4.4.3
```

```
## Warning: package 'forcats' was built under R version 4.4.3
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
```

```
## v dplyr      1.1.4      v readr      2.1.5
```

```
## v forcats    1.0.0      v stringr    1.5.1
```

```
## v ggplot2    3.5.1      v tibble     3.2.1
```

```
## v lubridate  1.9.4      v tidyr      1.3.1
```

```
## v purrr      1.0.4
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
```

```
## x dplyr::lag()     masks stats::lag()
```

```
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(brms)         # Bayesian regression modeling
```

```
## Warning: package 'brms' was built under R version 4.4.3
```

```
## Loading required package: Rcpp
```

```
## Loading 'brms' package (version 2.22.0). Useful instructions
```

```
## can be found by typing help('brms'). A more detailed introduction
```

```
## to the package is available through vignette('brms_overview').
```

```
##
```

```
## Attaching package: 'brms'
```

```
##
```

```
## The following object is masked from 'package:stats':
```

```
##
```

```
##      ar
```

```
library(bayesplot)    # MCMC diagnostics and posterior visualization
```

```
## Warning: package 'bayesplot' was built under R version 4.4.3
```

```
## This is bayesplot version 1.11.1
```

```
## - Online documentation and vignettes at mc-stan.org/bayesplot
```

```
## - bayesplot theme set to bayesplot::theme_default()
```

```
## * Does _not_ affect other ggplot2 plots
```

```
## * See ?bayesplot_theme_set for details on theme setting
```

```
##
```

```
## Attaching package: 'bayesplot'
```

```
##
```

```
## The following object is masked from 'package:brms':
```

```
##
```

```
##      rhat
```

```
library(ggplot2)
```

```
# Read the dataset from the CSV file
```

```
data <- read_csv("C:/Users/Sourav/Downloads/diabetes_data_upload.csv")
```

```
## Rows: 520 Columns: 17
```

```
## -- Column specification -----
```

```
## Delimiter: ","
```

```
## chr (16): Gender, Polyuria, Polydipsia, sudden weight loss, weakness, Polyph...
```

```
## dbl (1): Age
```

```
##
```

```
## i Use 'spec()' to retrieve the full column specification for this data.
```

```
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
# Inspect the data
```

```
print(head(data)) # View the first few rows
```

```
## # A tibble: 6 x 17
```

```
##   Age Gender Polyuria Polydipsia 'sudden weight loss' weakness Polyphagia
```

```
##   <dbl> <chr> <chr> <chr> <chr> <chr> <chr>
```

```
## 1 40 Male No Yes No Yes No
```

```
## 2 58 Male No No No Yes No
```

```
## 3 41 Male Yes No No Yes Yes
```

```
## 4 45 Male No No Yes Yes Yes
```

```
## 5 60 Male Yes Yes Yes Yes Yes
```

```
## 6 55 Male Yes Yes No Yes Yes
```

```
## # i 10 more variables: 'Genital thrush' <chr>, 'visual blurring' <chr>,
```

```
## # Itching <chr>, Irritability <chr>, 'delayed healing' <chr>,
```

```
## # 'partial paresis' <chr>, 'muscle stiffness' <chr>, Alopecia <chr>,
```

```
## # Obesity <chr>, class <chr>
```

```
str(data)
```

```
# Check the structure of the data
```

```
## spc_tbl_ [520 x 17] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
```

```
## $ Age : num [1:520] 40 58 41 45 60 55 57 66 67 70 ...
```

```
## $ Gender : chr [1:520] "Male" "Male" "Male" "Male" ...
```

```
## $ Polyuria : chr [1:520] "No" "No" "Yes" "No" ...
```

```
## $ Polydipsia : chr [1:520] "Yes" "No" "No" "No" ...
```

```
## $ sudden weight loss: chr [1:520] "No" "No" "No" "Yes" ...
```

```
## $ weakness : chr [1:520] "Yes" "Yes" "Yes" "Yes" ...
```

```
## $ Polyphagia : chr [1:520] "No" "No" "Yes" "Yes" ...
```

```
## $ Genital thrush : chr [1:520] "No" "No" "No" "Yes" ...
```

```
## $ visual blurring : chr [1:520] "No" "Yes" "No" "No" ...
```

```
## $ Itching : chr [1:520] "Yes" "No" "Yes" "Yes" ...
```

```
## $ Irritability : chr [1:520] "No" "No" "No" "No" ...
```

```
## $ delayed healing : chr [1:520] "Yes" "No" "Yes" "Yes" ...
```

```
## $ partial paresis : chr [1:520] "No" "Yes" "No" "No" ...
```

```
## $ muscle stiffness : chr [1:520] "Yes" "No" "Yes" "No" ...
```

```
## $ Alopecia : chr [1:520] "Yes" "Yes" "Yes" "No" ...
```

```
## $ Obesity : chr [1:520] "Yes" "No" "No" "No" ...
```

```
## $ class : chr [1:520] "Positive" "Positive" "Positive" "Positive" ...
```

```
## - attr(*, "spec")=
## .. cols(
## ..   Age = col_double(),
## ..   Gender = col_character(),
## ..   Polyuria = col_character(),
## ..   Polydipsia = col_character(),
## ..   'sudden weight loss' = col_character(),
## ..   weakness = col_character(),
## ..   Polyphagia = col_character(),
## ..   'Genital thrush' = col_character(),
## ..   'visual blurring' = col_character(),
## ..   Itching = col_character(),
## ..   Irritability = col_character(),
## ..   'delayed healing' = col_character(),
## ..   'partial paresis' = col_character(),
## ..   'muscle stiffness' = col_character(),
## ..   Alopecia = col_character(),
## ..   Obesity = col_character(),
## ..   class = col_character()
## .. )
## - attr(*, "problems")=<externalptr>
```

```
summary(data)           # Get summary statistics
```

```
##      Age      Gender      Polyuria      Polydipsia
## Min.   :16.00  Length:520  Length:520  Length:520
## 1st Qu.:39.00  Class :character  Class :character  Class :character
## Median :47.50  Mode  :character  Mode  :character  Mode  :character
## Mean   :48.03
## 3rd Qu.:57.00
## Max.   :90.00
## sudden weight loss  weakness      Polyphagia      Genital thrush
## Length:520          Length:520  Length:520      Length:520
## Class :character   Class :character  Class :character  Class :character
## Mode  :character   Mode  :character  Mode  :character  Mode  :character
##
##
## visual blurring      Itching      Irritability      delayed healing
## Length:520           Length:520  Length:520         Length:520
## Class :character     Class :character  Class :character   Class :character
## Mode  :character     Mode  :character  Mode  :character   Mode  :character
##
##
## partial paresis      muscle stiffness      Alopecia      Obesity
## Length:520           Length:520  Length:520      Length:520
## Class :character     Class :character  Class :character   Class :character
## Mode  :character     Mode  :character  Mode  :character   Mode  :character
##
##
##      class
## Length:520
```

```
## Class :character
## Mode :character
##
##
##
```

```
# Check for missing values
missing_vals <- sapply(data, function(x) sum(is.na(x)))
missing_vals
```

```
##           Age           Gender           Polyuria           Polydipsia
##           0             0             0             0
## sudden weight loss      weakness      Polyphagia      Genital thrush
##           0             0             0             0
## visual blurring        Itching      Irritability      delayed healing
##           0             0             0             0
## partial paresis      muscle stiffness      Alopecia      Obesity
##           0             0             0             0
##           class
##           0
```

```
# Remove duplicate rows
data <- distinct(data)

# Convert "Yes"/"No" columns to 1/0
# Identify the columns that contain Yes/No responses
yes_no_cols <- c(
  "Polyuria", "Polydipsia", "sudden weight loss", "weakness", "Polyphagia",
  "Genital thrush", "visual blurring", "Itching", "Irritability",
  "delayed healing", "partial paresis", "muscle stiffness",
  "Alopecia", "Obesity"
)

# Convert Yes -> 1, No -> 0
data[yes_no_cols] <- lapply(data[yes_no_cols], function(col) {
  ifelse(col == "Yes", 1, 0)
})

# Convert "Positive"/"Negative" in 'class' column to 1/0
data$class <- ifelse(data$class == "Positive", 1, 0)

# Inspect the transformed data
head(data)
```

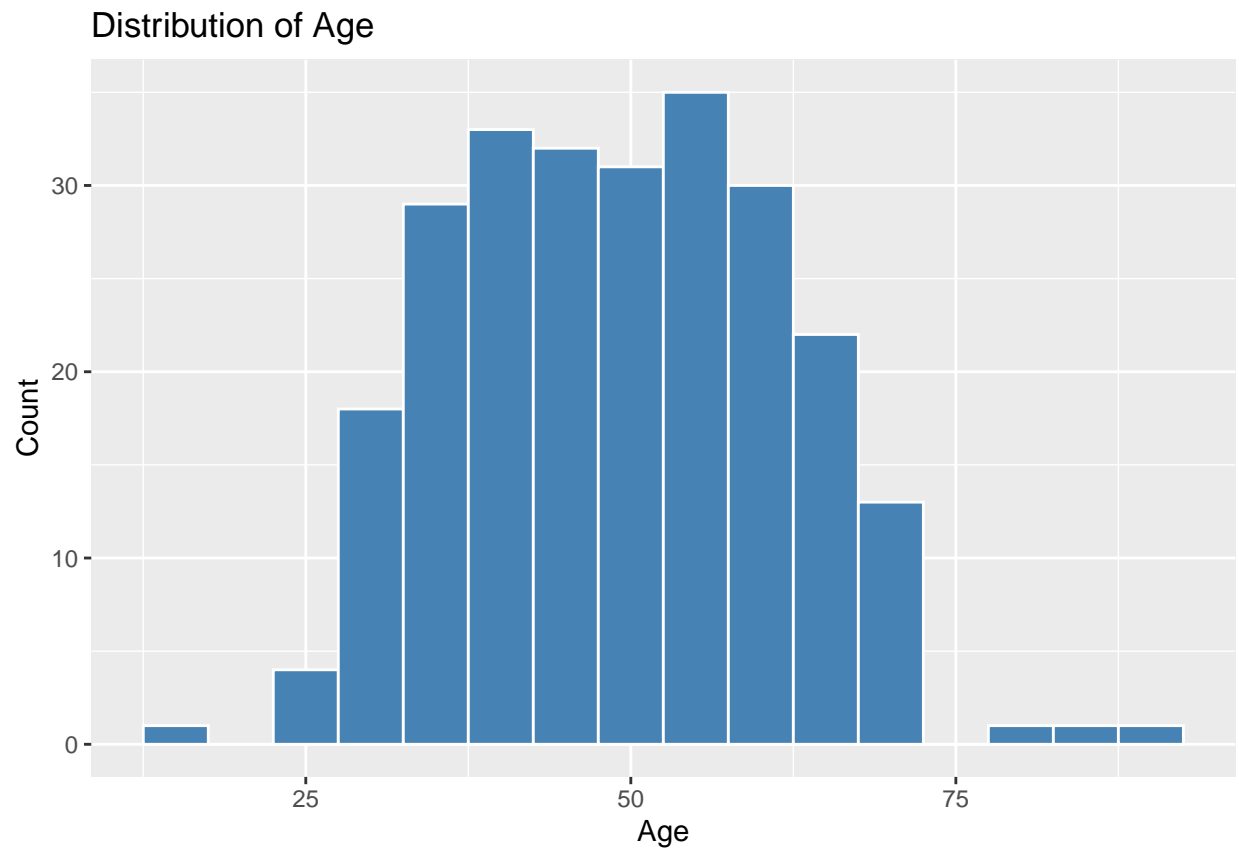
```
## # A tibble: 6 x 17
##   Age Gender Polyuria Polydipsia 'sudden weight loss' weakness Polyphagia
##   <dbl> <chr>   <dbl>      <dbl>          <dbl>      <dbl>      <dbl>
## 1   40 Male     0         1             0         1         0
## 2   58 Male     0         0             0         1         0
## 3   41 Male     1         0             0         1         1
## 4   45 Male     0         0             1         1         1
## 5   60 Male     1         1             1         1         1
## 6   55 Male     1         1             0         1         1
```

```
## # i 10 more variables: 'Genital thrush' <dbl>, 'visual blurring' <dbl>,
## #   Itching <dbl>, Irritability <dbl>, 'delayed healing' <dbl>,
## #   'partial paresis' <dbl>, 'muscle stiffness' <dbl>, Alopecia <dbl>,
## #   Obesity <dbl>, class <dbl>
```

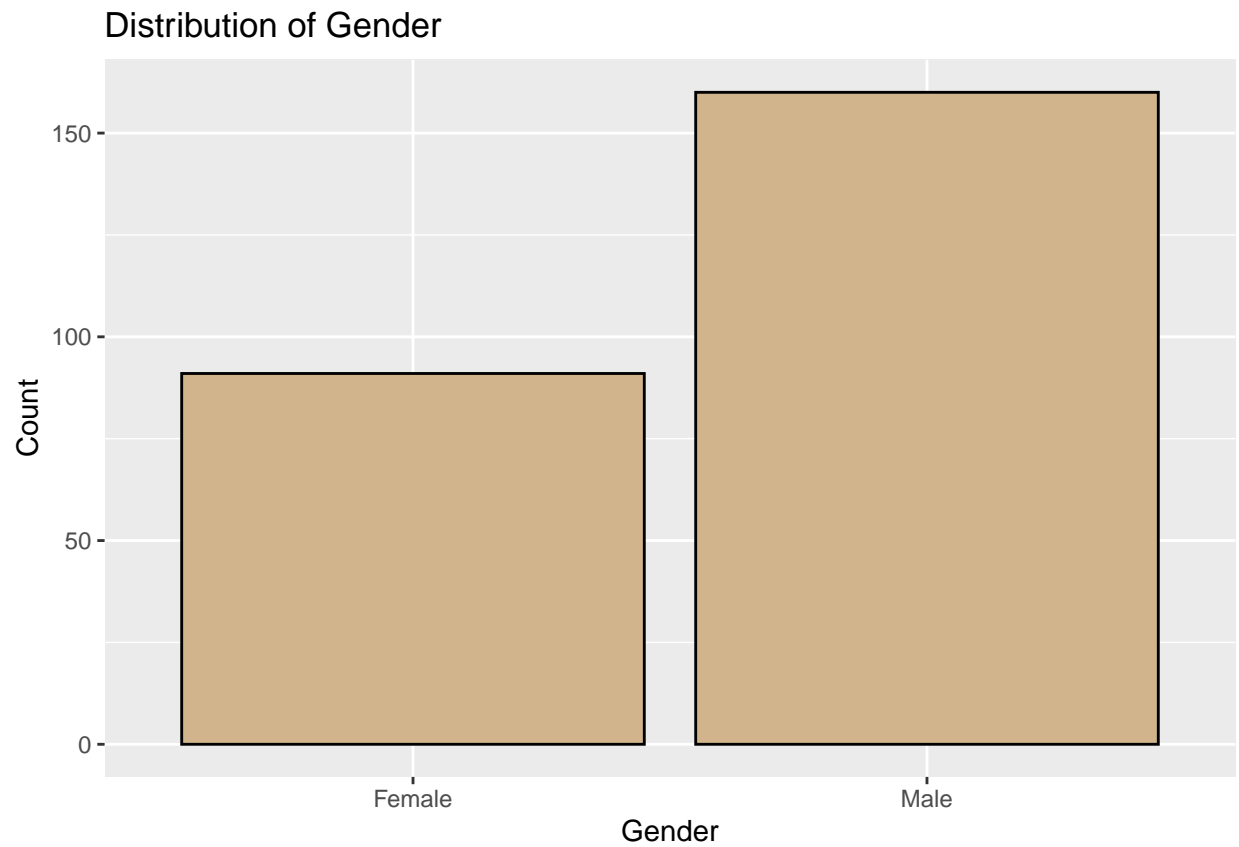
```
str(data)
```

```
## tibble [251 x 17] (S3: tbl_df/tbl/data.frame)
##  $ Age          : num [1:251] 40 58 41 45 60 55 57 66 67 70 ...
##  $ Gender       : chr [1:251] "Male" "Male" "Male" "Male" ...
##  $ Polyuria     : num [1:251] 0 0 1 0 1 1 1 1 1 0 ...
##  $ Polydipsia   : num [1:251] 1 0 0 0 1 1 1 1 1 1 ...
##  $ sudden weight loss: num [1:251] 0 0 0 1 1 0 0 1 0 1 ...
##  $ weakness     : num [1:251] 1 1 1 1 1 1 1 1 1 1 ...
##  $ Polyphagia   : num [1:251] 0 0 1 1 1 1 1 0 1 1 ...
##  $ Genital thrush : num [1:251] 0 0 0 1 0 0 1 0 1 0 ...
##  $ visual blurring : num [1:251] 0 1 0 0 1 1 0 1 0 1 ...
##  $ Itching      : num [1:251] 1 0 1 1 1 1 0 1 1 1 ...
##  $ Irritability  : num [1:251] 0 0 0 0 1 0 0 1 1 1 ...
##  $ delayed healing : num [1:251] 1 0 1 1 1 1 1 0 0 0 ...
##  $ partial paresis : num [1:251] 0 1 0 0 1 0 1 1 1 0 ...
##  $ muscle stiffness : num [1:251] 1 0 1 0 1 1 0 1 1 0 ...
##  $ Alopecia     : num [1:251] 1 1 1 0 1 1 0 0 0 1 ...
##  $ Obesity      : num [1:251] 1 0 0 0 1 1 0 0 1 0 ...
##  $ class        : num [1:251] 1 1 1 1 1 1 1 1 1 1 ...
```

```
# Histogram of Age
ggplot(data, aes(x = Age)) +
  geom_histogram(binwidth = 5, fill = "steelblue", color = "white") +
  labs(title = "Distribution of Age", x = "Age", y = "Count")
```

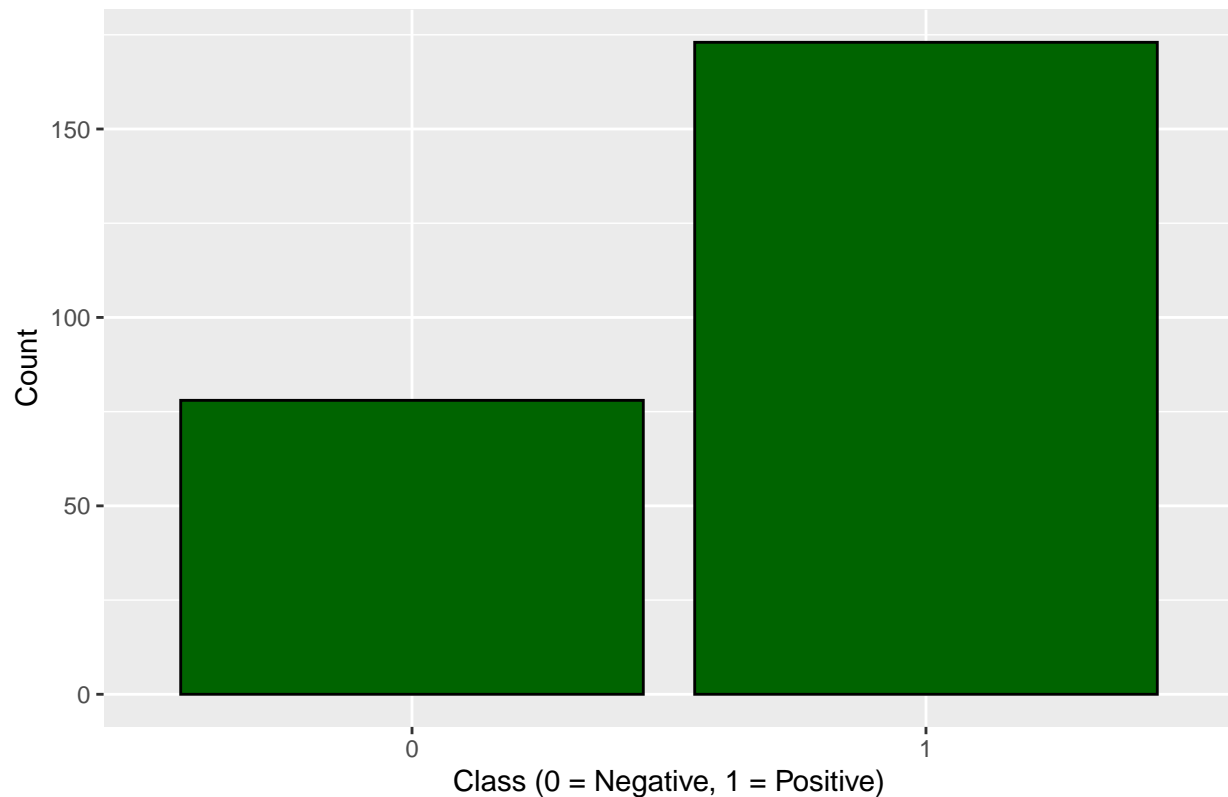


```
# Bar plot of Gender (if still categorical)  
# If you converted Gender to 1/0, you can factor it first or skip this step  
ggplot(data, aes(x = Gender)) +  
  geom_bar(fill = "tan", color = "black") +  
  labs(title = "Distribution of Gender", x = "Gender", y = "Count")
```



```
# Bar plot of 'class' (after conversion to 1/0)
ggplot(data, aes(x = factor(class))) +
  geom_bar(fill = "darkgreen", color = "black") +
  labs(title = "Distribution of Diabetes Class", x = "Class (0 = Negative, 1 = Positive)", y = "Count")
```

Distribution of Diabetes Class



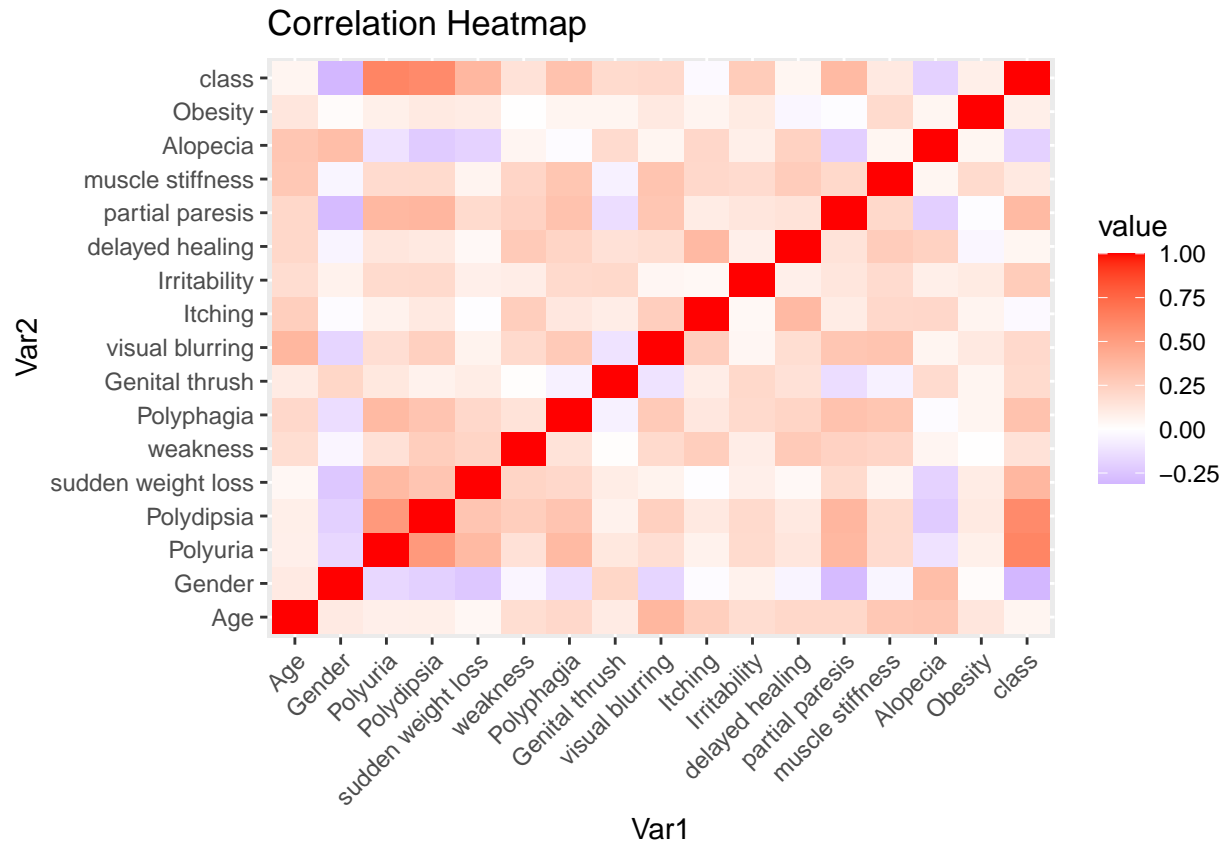
```
# Correlation Heatmap (Optional)
# For correlation, all columns must be numeric.
# If Gender is still categorical (M/F), convert it to 1/0 as shown above
numeric_data <- data %>%
  mutate(Gender = ifelse(Gender == "Male" | Gender == 1, 1, 0)) # ensure numeric
# Calculate correlation matrix
corr_mat <- cor(numeric_data)
# Melt the correlation matrix for ggplot
library(reshape2)
```

```
##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
## smiths
```

```
melted_corr <- melt(corr_mat)
# Plot heatmap
ggplot(melted_corr, aes(x = Var1, y = Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Correlation Heatmap")
```





```
data <- data %>%
  rename(
    sudden_weight_loss = `sudden weight loss`,
    Genital_thrush = `Genital thrush`,
    visual_blurring = `visual blurring`,
    delayed_healing = `delayed healing`,
    partial_paresis = `partial paresis`,
    muscle_stiffness = `muscle stiffness`
  )

# -----
# 1. Check Factor Levels
# -----
table(data$class)

##
##    0    1
## 78 173

# -----
# 2. Define Priors
# -----
my_priors <- c(
  set_prior("normal(0, 1)", class = "b"),      # Prior for coefficients
  set_prior("normal(0, 2.5)", class = "Intercept") # Prior for intercept
)
```

```

)

# -----
# 3. Fit the Bayesian Logit Regression Model
# -----
fit_logit <- brm(
  formula = class ~ Age + Polyuria + Polydipsia + sudden_weight_loss + weakness + Polyphagia +
    Genital_thrush + visual_blurring + Itching + Irritability +
    delayed_healing + partial_paresis + muscle_stiffness +
    Alopecia + Obesity,
  data = data,
  family = bernoulli(link = "logit"), # Logit regression for binary classification
  prior = my_priors,
  chains = 4, # Number of Markov Chains
  iter = 2000, # Total iterations per chain
  warmup = 1000, # Warm-up iterations (burn-in)
  cores = 4, # Parallel computation
  seed = 1234 # Ensure reproducibility
)

```

```
## Compiling Stan program...
```

```
## Start sampling
```

```

# -----
# 4. Model Summary and Posterior Estimates
# -----
summary(fit_logit) # Print model summary

```

```

## Family: bernoulli
## Links: mu = logit
## Formula: class ~ Age + Polyuria + Polydipsia + sudden_weight_loss + weakness + Polyphagia + Genital_
## Data: data (Number of observations: 251)
## Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
## total post-warmup draws = 4000
##
## Regression Coefficients:
##      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      -0.80      0.90   -2.56    0.96 1.00    4519    3230
## Age            -0.02      0.02   -0.06    0.02 1.00    4462    3211
## Polyuria        2.26      0.49    1.33    3.27 1.00    4302    2899
## Polydipsia      2.35      0.51    1.37    3.35 1.00    4271    3176
## sudden_weight_loss 0.92      0.45    0.06    1.78 1.00    4706    3347
## weakness        0.16      0.43   -0.69    1.00 1.00    4651    3021
## Polyphagia      0.69      0.45   -0.18    1.59 1.00    5292    3586
## Genital_thrush   1.19      0.50    0.22    2.18 1.00    4550    2884
## visual_blurring  0.83      0.50   -0.12    1.80 1.00    4265    2384
## Itching        -0.89      0.47   -1.83    0.04 1.00    4175    2882
## Irritability     0.96      0.48    0.00    1.89 1.00    4956    3280
## delayed_healing -0.34      0.48   -1.31    0.60 1.00    4897    3563
## partial_paresis  0.91      0.46    0.02    1.86 1.00    4927    3054
## muscle_stiffness -0.36      0.48   -1.29    0.58 1.00    4539    2988

```



```
## Polydipsia      2.35170517 0.50585099 1.369643000 3.35451904
## sudden_weight_loss 0.92497261 0.44542445 0.064998216 1.77874968
## weakness       0.16231782 0.42734848 -0.688729157 1.00453917
## Polyphagia     0.69318103 0.44988928 -0.179697091 1.59018486
## Genital_thrush 1.18862217 0.50333108 0.218080480 2.18152789
## visual_blurring 0.83076547 0.49701372 -0.123645622 1.79556494
## Itching        -0.88889485 0.47448139 -1.834356941 0.04340460
## Irritability   0.95679616 0.48175733 0.003176781 1.89471832
## delayed_healing -0.34497595 0.48211244 -1.314430574 0.59819604
## partial_paresis 0.91112851 0.45973601 0.024564559 1.85611671
## muscle_stiffness -0.35516524 0.48215407 -1.291760966 0.58412472
## Alopecia       -0.43707302 0.47684779 -1.401237888 0.49957054
## Obesity        0.15571816 0.51379889 -0.873471147 1.17664204
```

```
# Check the first few posterior samples
```

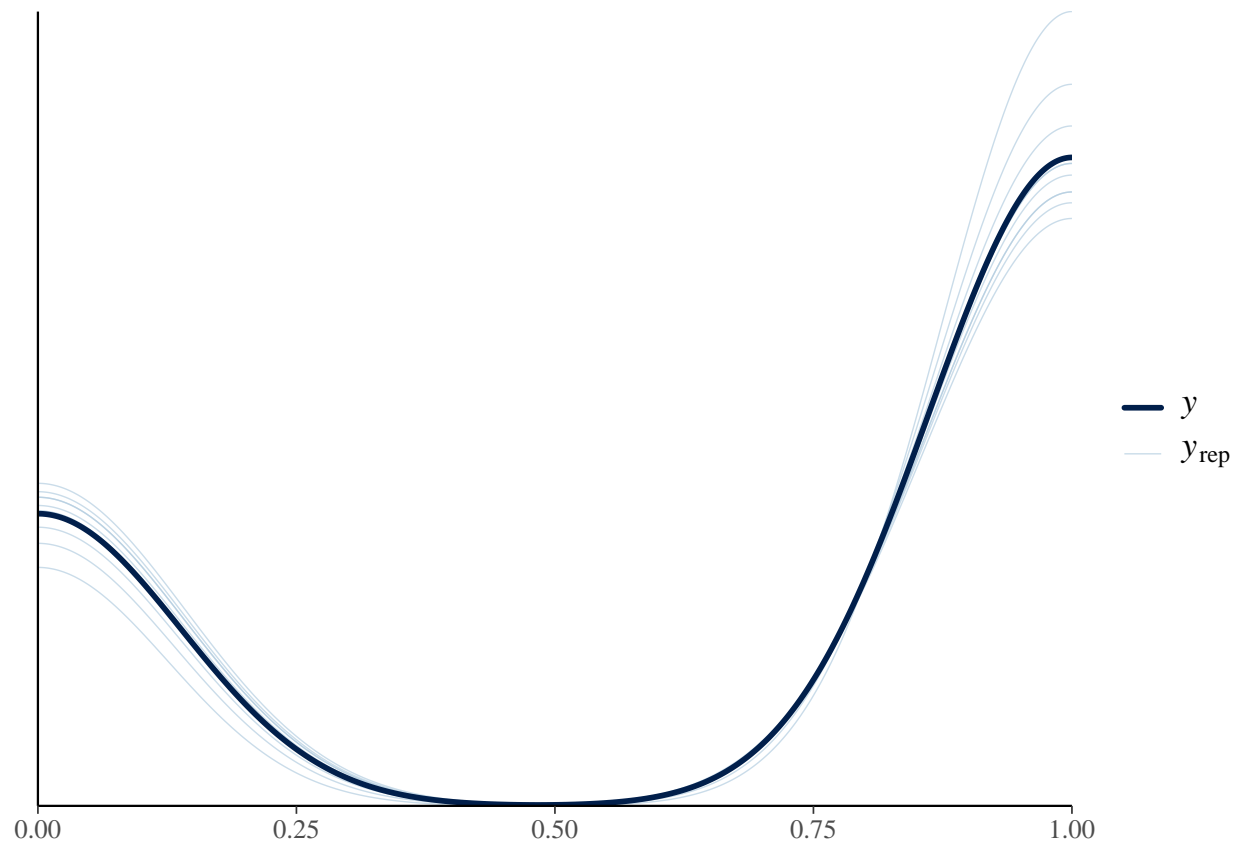
```
head(posterior_samples(fit_logit))
```

```
## Warning: Method 'posterior_samples' is deprecated. Please see ?as_draws for
## recommended alternatives.
```

```
##      b_Intercept      b_Age b_Polyuria b_Polydipsia b_sudden_weight_loss
## 1 -2.7746999 0.016796174 2.560970 2.750409 1.3507200
## 2 -2.4862688 0.013766010 1.787318 3.714879 0.5119741
## 3 0.1325893 -0.030498031 2.114115 2.484075 0.8992447
## 4 -1.9901908 -0.007830156 2.075145 2.446004 1.4587294
## 5 -0.5767134 -0.007190334 1.891896 3.050850 0.2233347
## 6 -0.4943746 -0.015838112 2.145437 1.496714 1.2706095
##      b_weakness b_Polyphagia b_Genital_thrush b_visual_blurring b_Itching
## 1 0.5686692 0.834398978 1.560081 1.1660117 -1.3920354
## 2 0.4056039 0.703410544 0.782191 0.1179227 -0.7898869
## 3 0.2360142 0.009231634 1.100071 0.3750265 -0.4932310
## 4 0.4631474 0.681767660 1.192488 0.8414611 -0.1256065
## 5 -0.4068958 0.627008340 1.219412 1.0310804 -1.6402234
## 6 0.4799626 0.611965059 1.281859 -0.1543572 -0.9739982
##      b_Irritability b_delayed_healing b_partial_paresis b_muscle_stiffness
## 1 0.4654567 -0.1134112 0.62587769 -0.5345490
## 2 1.1950757 -0.4905190 1.56807624 -0.8861569
## 3 1.2273998 -1.1390334 2.14974964 -0.5872135
## 4 0.4324161 -1.0169647 1.48733989 -0.4318473
## 5 1.4746561 0.2520516 0.04336771 -0.0857685
## 6 -0.2742894 -0.3088282 1.58623345 0.2286505
##      b_Alopecia b_Obesity Intercept lprior lp_
## 1 -1.32382152 1.2147389 2.172209 -29.41375 -95.86537
## 2 0.10179489 0.1188474 2.124626 -28.03220 -97.00803
## 3 -0.03457183 0.6057120 2.321360 -26.66930 -96.61435
## 4 -0.09597170 0.5750732 1.946688 -25.52352 -93.32933
## 5 -0.69165060 0.2145607 1.970815 -26.68394 -95.20039
## 6 -0.47091246 0.1062595 1.911711 -23.23686 -96.08259
```

```
# -----
# 5. Posterior Predictive Checks (Model Diagnostics)
# -----
# Density overlay of observed vs predicted values
pp_check(fit_logit, type = "dens_overlay")
```

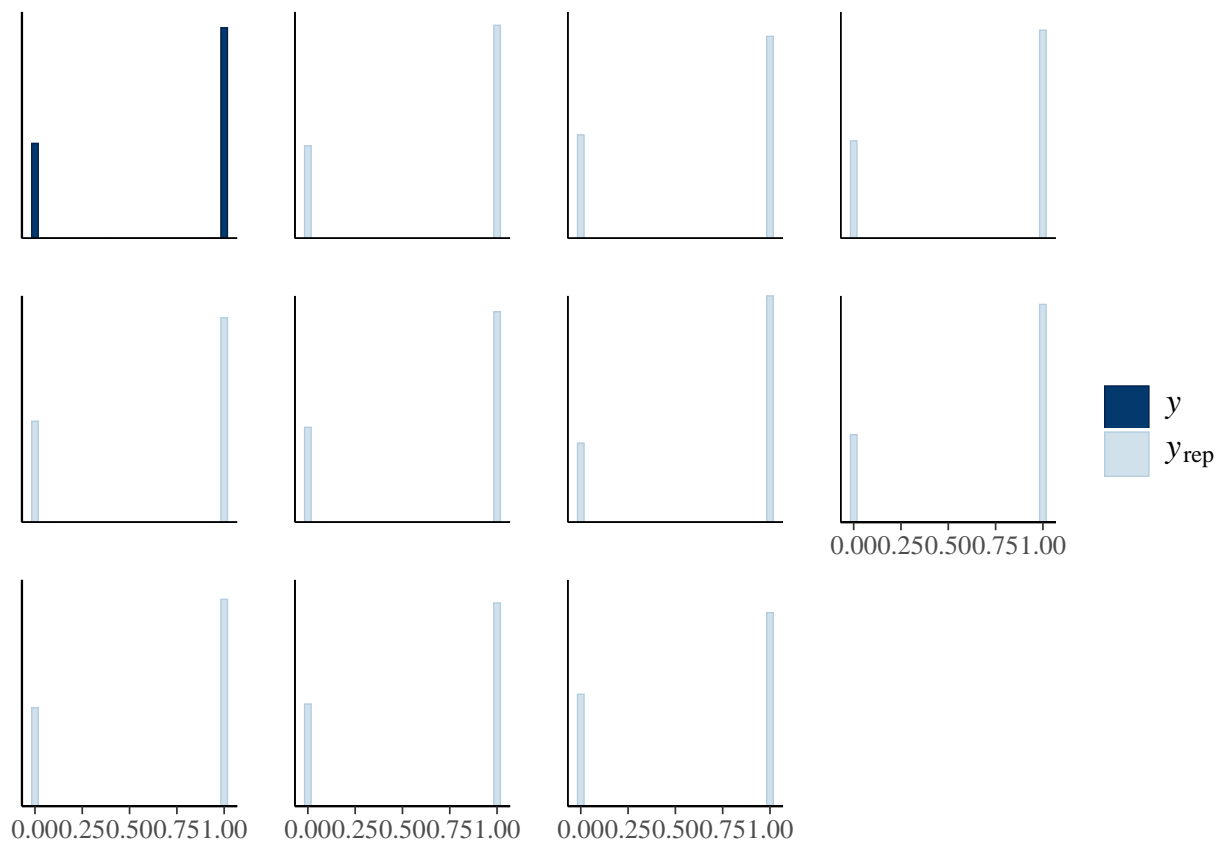
```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
# Histogram of simulated posterior predictions  
pp_check(fit_logit, type = "hist")
```

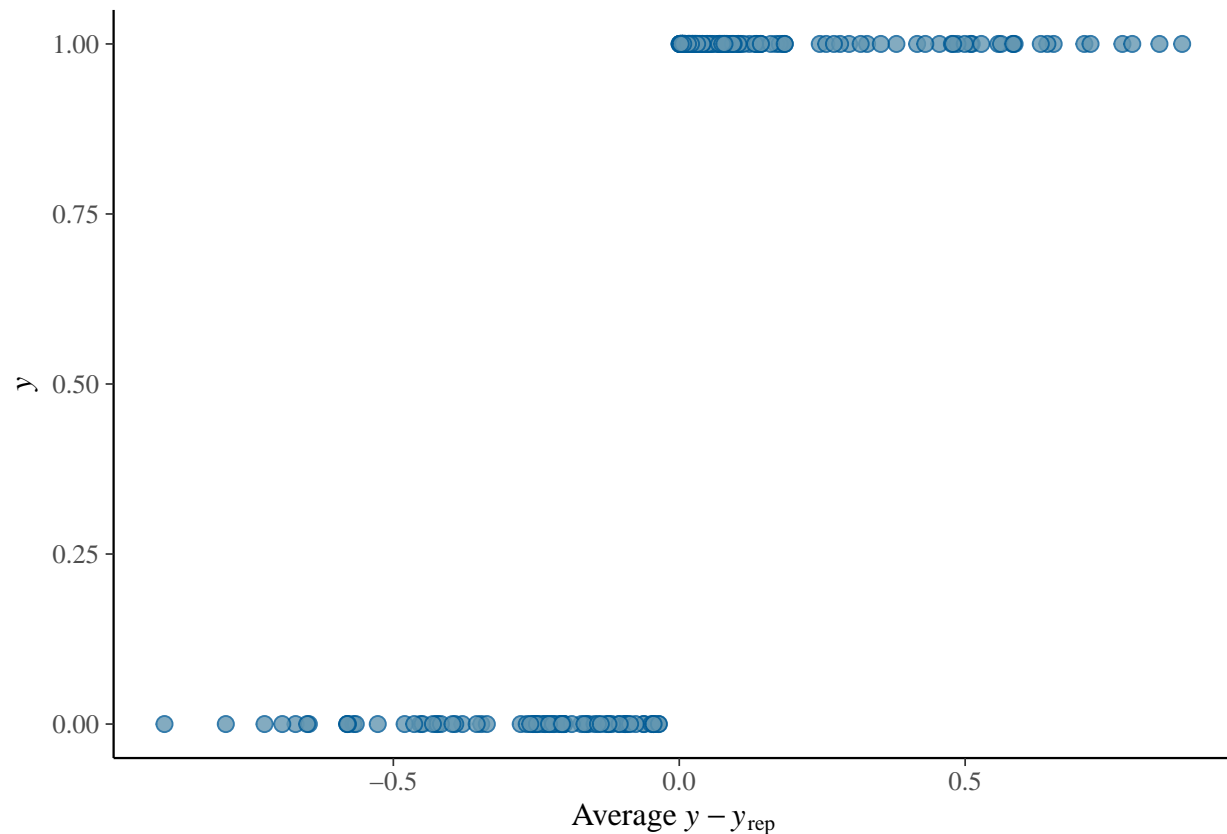
```
## Using 10 posterior draws for ppc type 'hist' by default.
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



```
pp_check(fit_logit, type = "error_scatter_avg")
```

```
## Using all posterior draws for ppc type 'error_scatter_avg' by default.
```



```
# Check if chains have converged (Trace Plots)
```

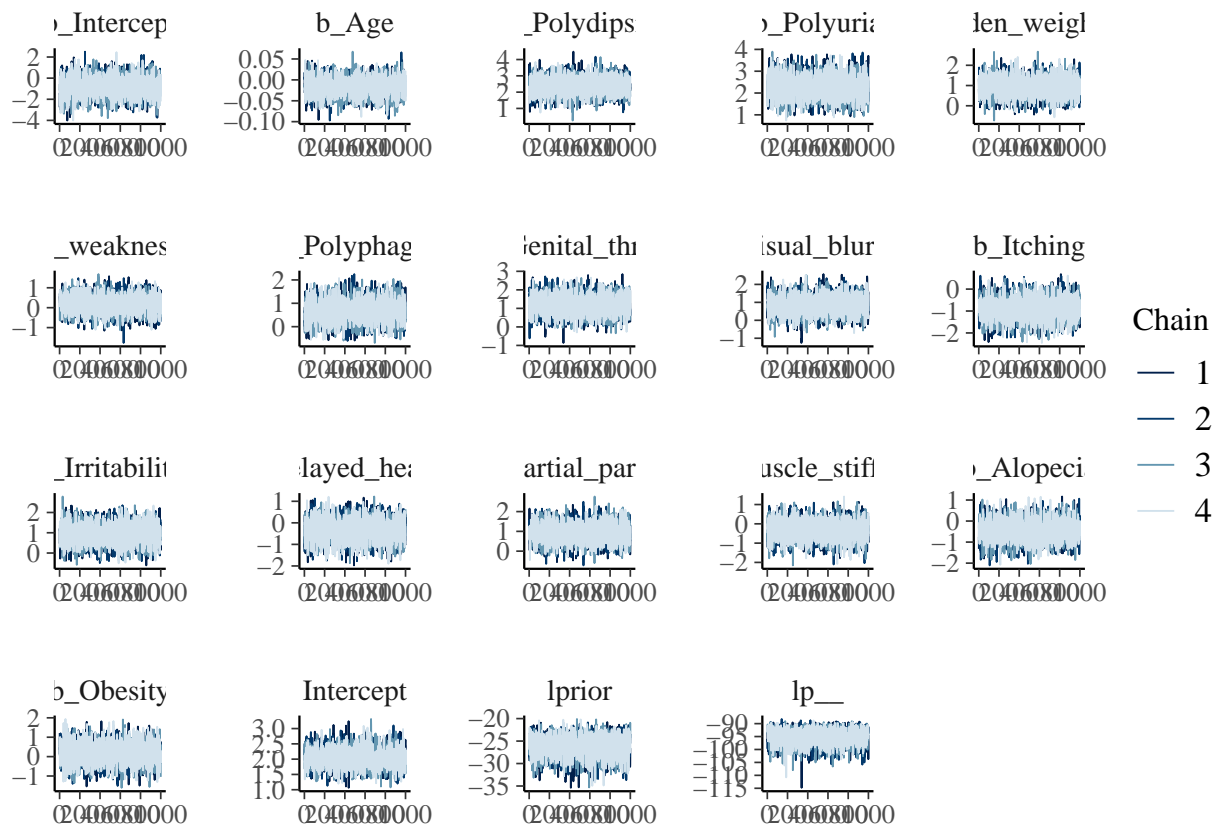
```
# Get all parameter names in the model
```

```
variables(as_draws(fit_logit))
```

```
## [1] "b_Intercept"      "b_Age"            "b_Polyuria"
## [4] "b_Polydipsia"     "b_sudden_weight_loss" "b_weakness"
## [7] "b_Polyphagia"     "b_Genital_thrush"  "b_visual_blurring"
## [10] "b_Itching"        "b_Irritability"    "b_delayed_healing"
## [13] "b_partial_paresis" "b_muscle_stiffness" "b_Alopecia"
## [16] "b_Obesity"        "Intercept"         "lprior"
## [19] "lp_"
```

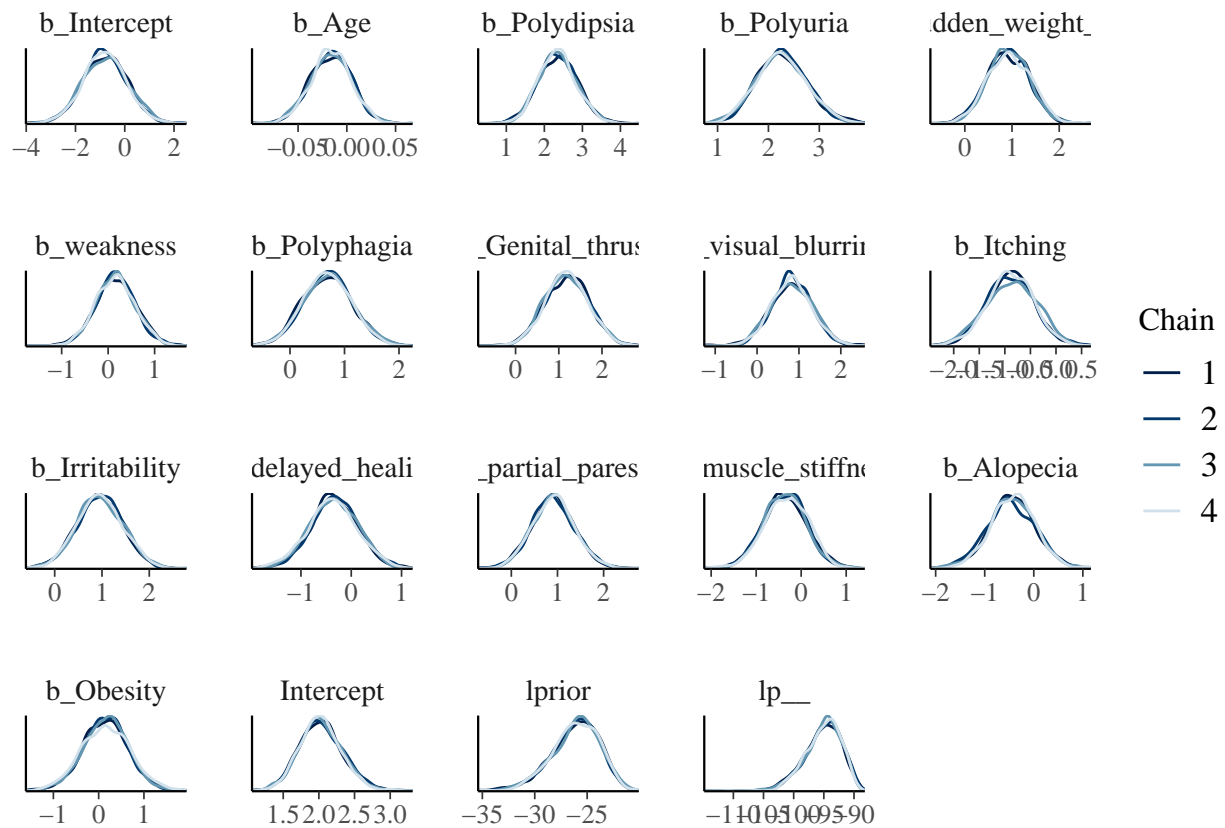
```
mcmc_trace(as_draws(fit_logit),
```

```
  pars = c("b_Intercept" , "b_Age", "b_Polydipsia", "b_Polyuria", "b_sudden_weight_loss", "b_
```

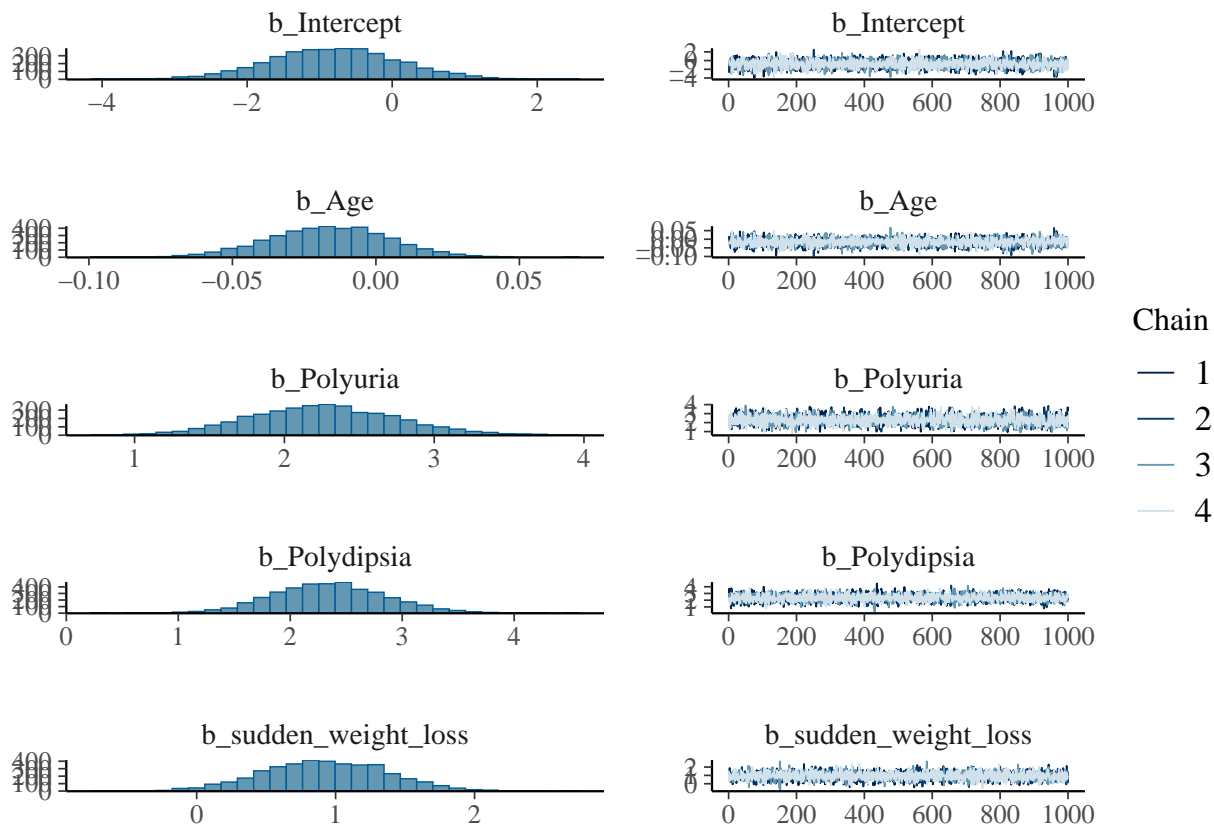


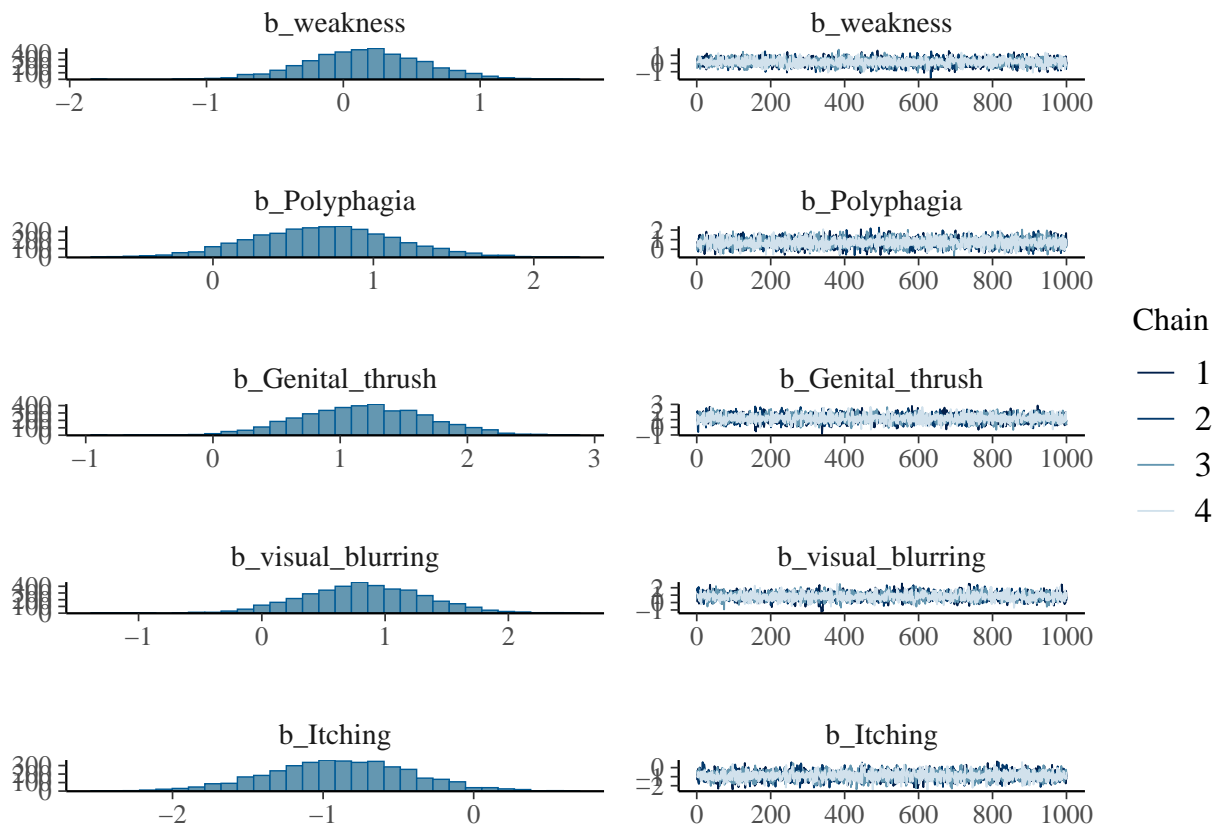
```
# Posterior Density Overlays
mcmc_dens_overlay(as.array(fit_logit),
                  pars = c("b_Intercept", "b_Age", "b_Polydipsia", "b_Polyuria", "b_sudden_weight_loss"))
```

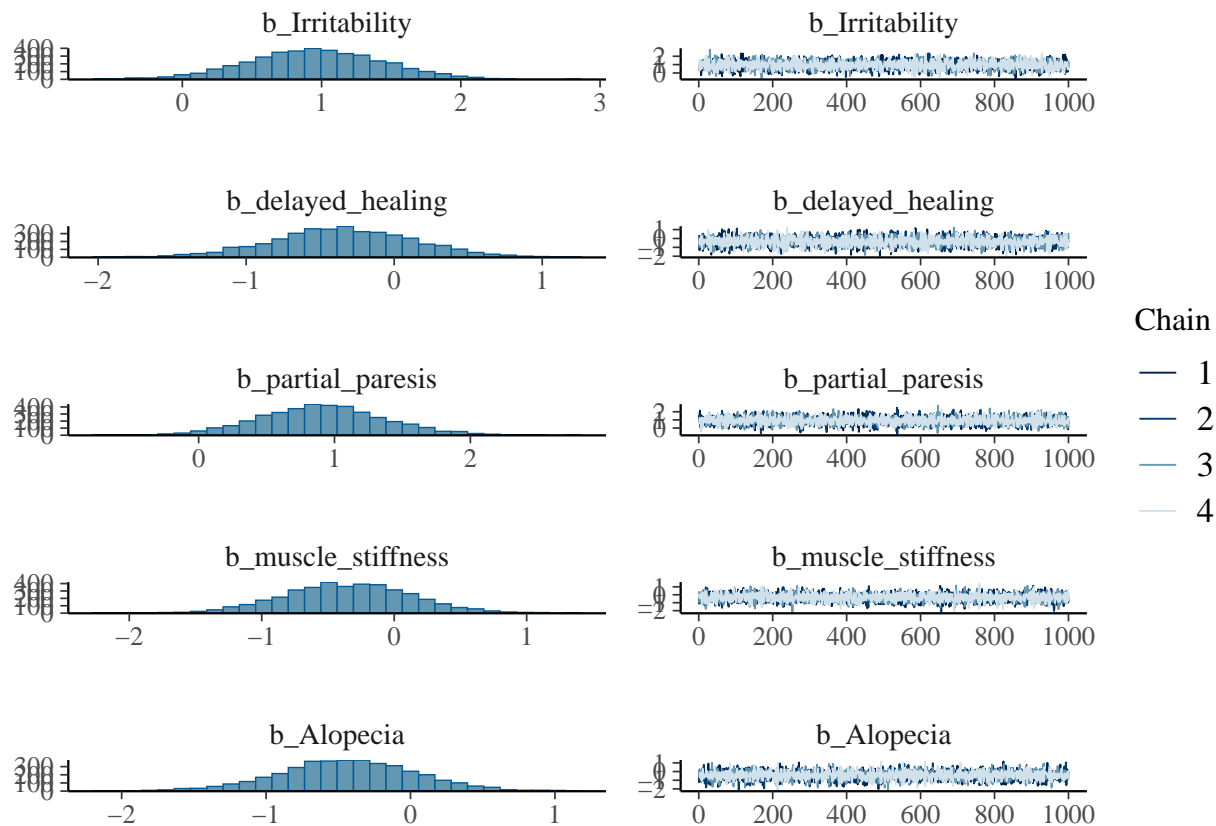


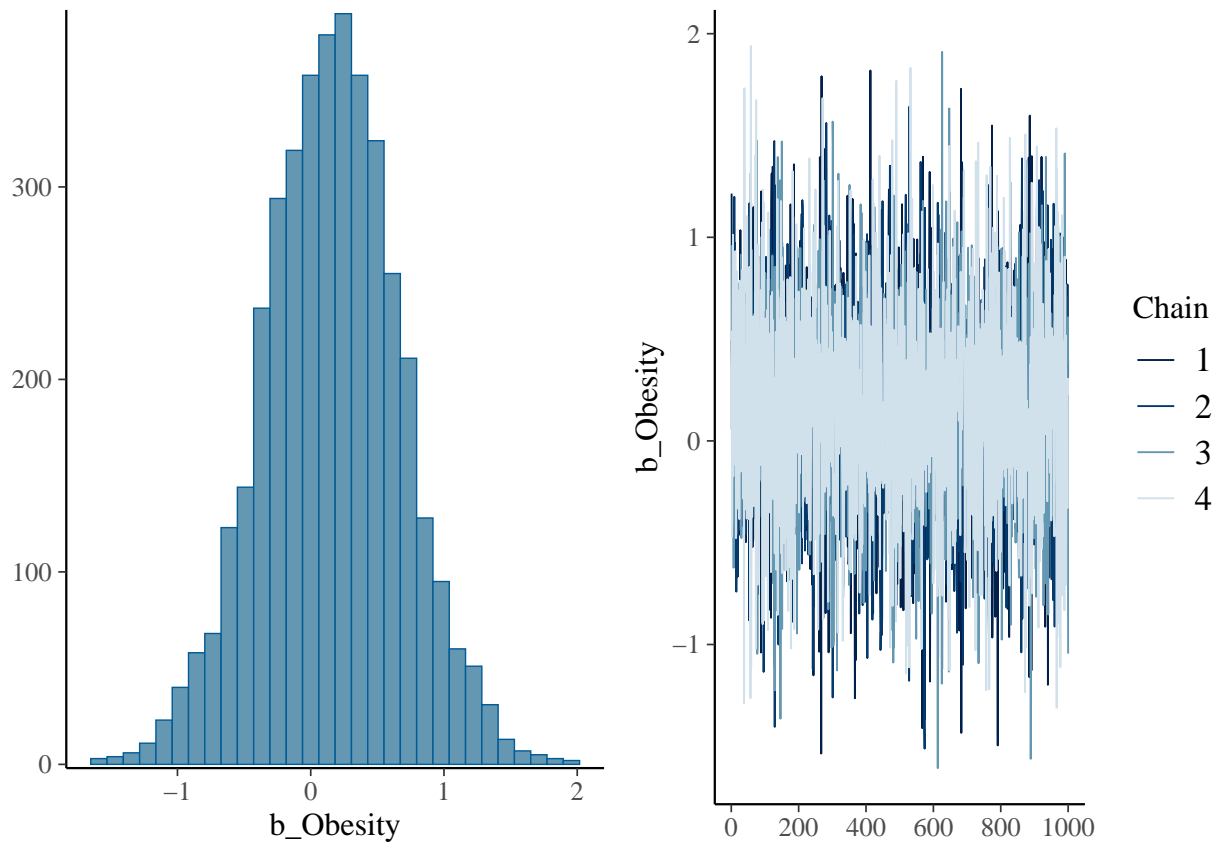


```
# Posterior Intervals
plot(fit_logit)
```

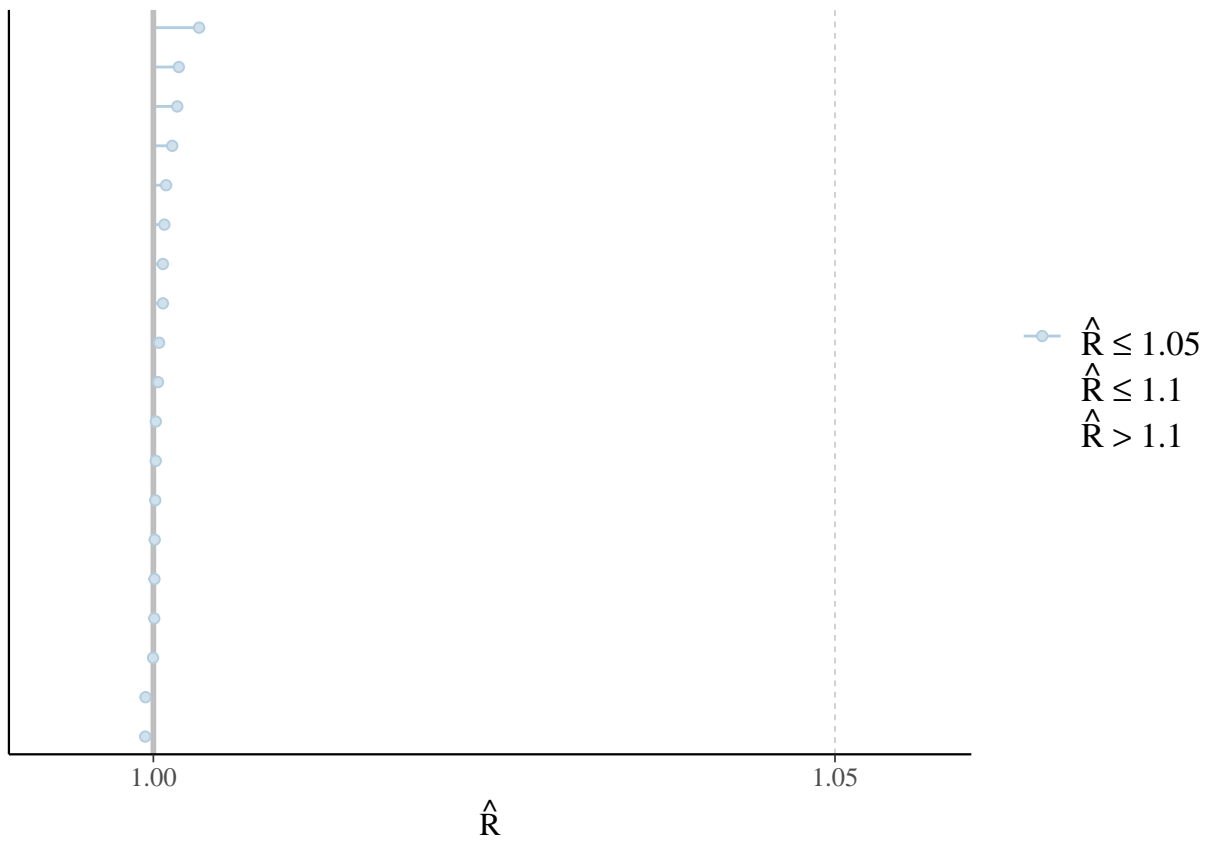




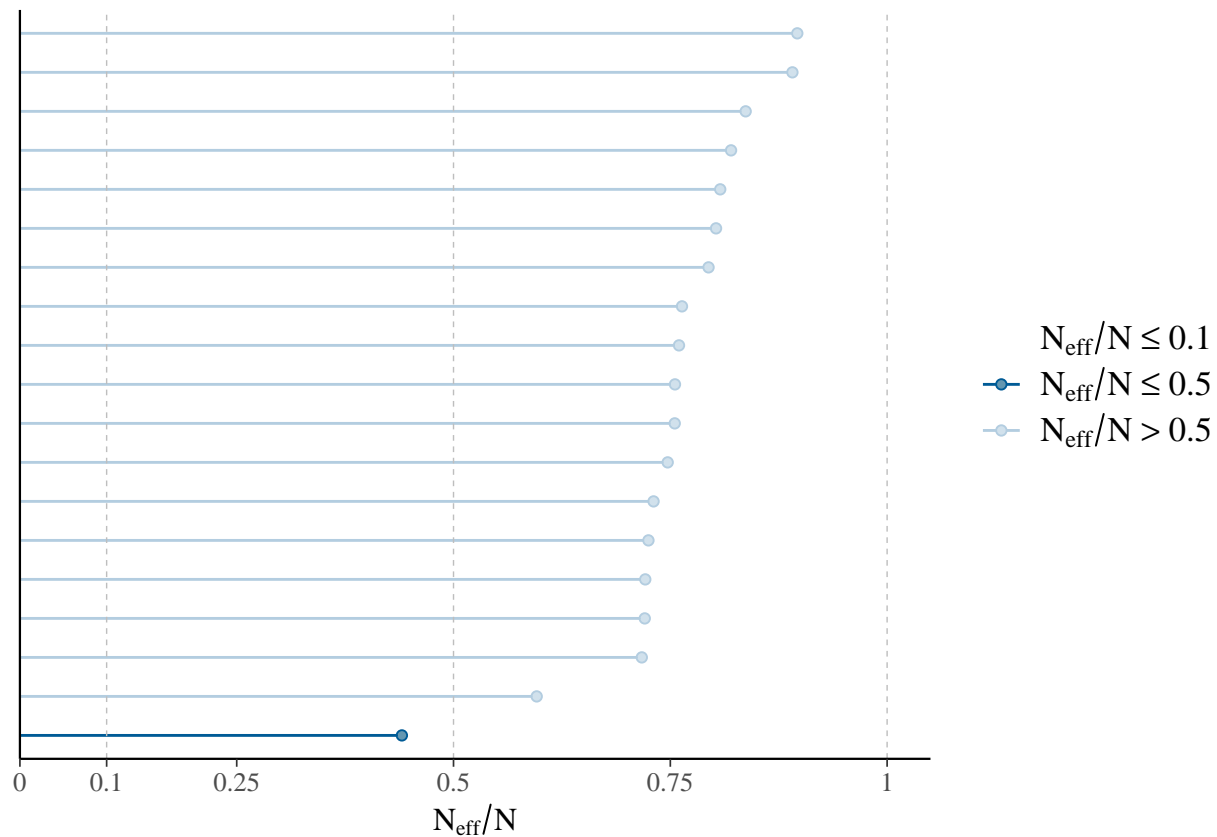




```
mcmc_plot(fit_logit, type = "rhat")
```



```
mcmc_plot(fit_logit, type = "neff")
```

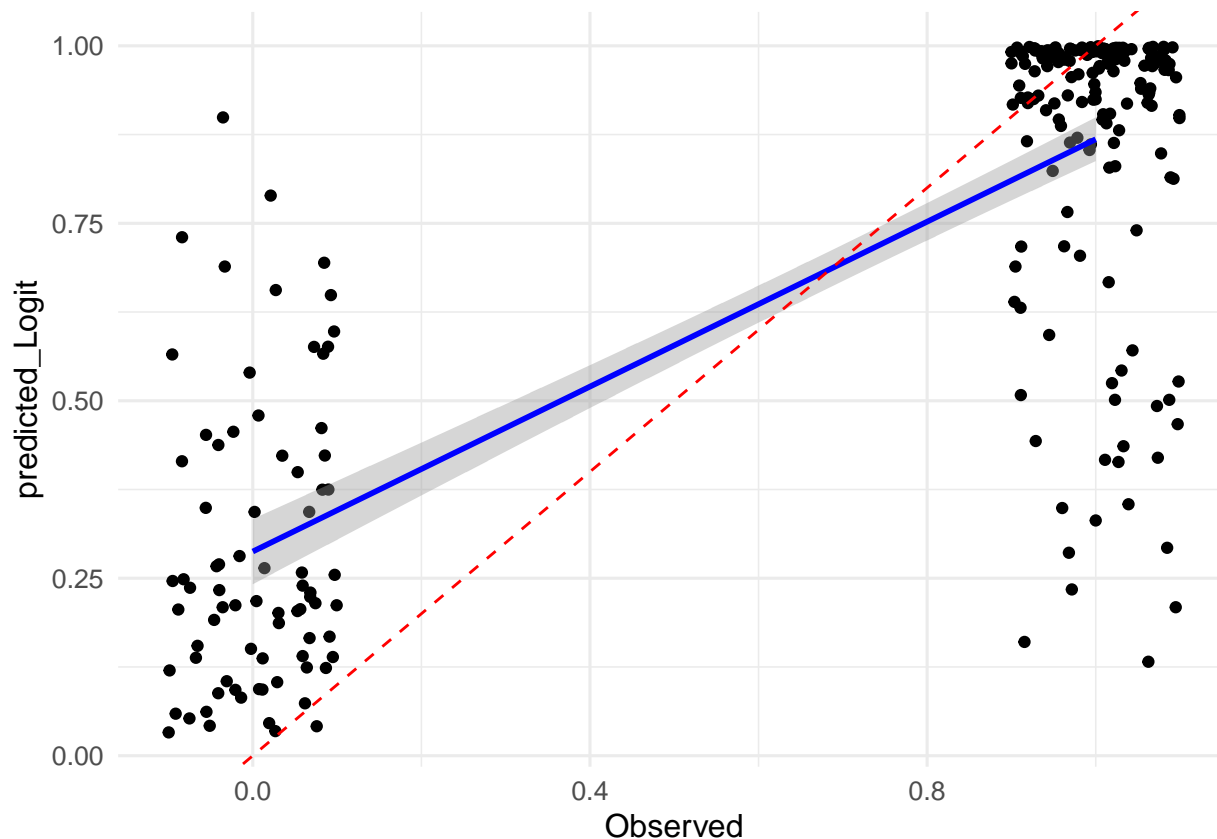


*#Plot the relationship between observed and predicted values*

```
data$predicted_Logit <- fitted(fit_logit)[, "Estimate"]
```

*# Logistic Regression*

```
ggplot(data, aes(x = class, y = predicted_Logit)) +
  geom_jitter(width = 0.1, height = 0) +
  geom_smooth(method = "lm", formula = y ~ x, color = "blue") +
  geom_abline(slope = 1, intercept = 0, color = "red", linetype = "dashed") +
  labs(x = "Observed", y = "predicted_Logit", color = "Observed Class") +
  theme_minimal(base_size = 12)
```



```
# -----
# 4. Fit the Bayesian Probit Regression Model
# -----
fit_probit <- brm(
  formula = class ~ Age + Polyuria + Polydipsia + sudden_weight_loss + weakness + Polyphagia +
    Genital_thrush + visual_blurring + Itching + Irritability +
    delayed_healing + partial_paresis + muscle_stiffness +
    Alopecia + Obesity,
  data = data,
  family = bernoulli(link = "probit"), # Probit regression for binary classification
  prior = my_priors,
  chains = 4, # Number of Markov Chains
  iter = 2000, # Total iterations per chain
  warmup = 1000, # Warm-up iterations (burn-in)
  cores = 4, # Parallel computation
  seed = 1234 # Ensure reproducibility
)
```

```
## Compiling Stan program...
```

```
## Start sampling
```

```
# -----
# 5. Model Summary and Posterior Estimates
# -----
summary(fit_probit) # Print model summary
```



```
## Family: bernoulli
## Links: mu = probit
## Formula: class ~ Age + Polyuria + Polydipsia + sudden_weight_loss + weakness + Polyphagia + Genital_
## Data: data (Number of observations: 251)
## Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
## total post-warmup draws = 4000
##
## Regression Coefficients:
##           Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept      -0.53      0.57   -1.64    0.58 1.00    5206    3442
## Age             -0.01      0.01   -0.04    0.01 1.00    5395    3139
## Polyuria        1.60      0.34    0.95    2.30 1.00    4550    3010
## Polydipsia      1.70      0.34    1.05    2.37 1.00    4499    3387
## sudden_weight_loss 0.56      0.31   -0.02    1.18 1.00    5822    3098
## weakness        0.13      0.29   -0.45    0.70 1.00    4932    3234
## Polyphagia      0.47      0.30   -0.13    1.08 1.00    6386    3272
## Genital_thrush  0.87      0.33    0.24    1.53 1.00    6027    3102
## visual_blurring  0.64      0.34   -0.04    1.32 1.00    5319    2967
## Itching        -0.78      0.33   -1.43   -0.14 1.00    4722    2975
## Irritability    0.67      0.33    0.03    1.34 1.00    5985    3448
## delayed_healing -0.24      0.32   -0.85    0.38 1.00    5498    2668
## partial_paresis  0.58      0.31   -0.03    1.17 1.00    5641    2919
## muscle_stiffness -0.33      0.33   -1.01    0.34 1.00    5894    2511
## Alopecia        -0.26      0.31   -0.89    0.35 1.00    5048    3124
## Obesity         0.12      0.33   -0.54    0.79 1.00    5500    2532
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

```
prior_summary(fit_probit) # Show priors used
```

```
##           prior      class      coef group resp dpar nlpar lb ub
## normal(0, 1)      b
## normal(0, 1)      b      Age
## normal(0, 1)      b      Alopecia
## normal(0, 1)      b delayed_healing
## normal(0, 1)      b Genital_thrush
## normal(0, 1)      b      Irritability
## normal(0, 1)      b      Itching
## normal(0, 1)      b muscle_stiffness
## normal(0, 1)      b      Obesity
## normal(0, 1)      b partial_paresis
## normal(0, 1)      b      Polydipsia
## normal(0, 1)      b      Polyphagia
## normal(0, 1)      b      Polyuria
## normal(0, 1)      b sudden_weight_loss
## normal(0, 1)      b visual_blurring
## normal(0, 1)      b      weakness
## normal(0, 2.5) Intercept
## source
## user
## (vectorized)
## (vectorized)
```

```
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## (vectorized)
## user
```

```
# Get fixed effects estimates (population-level effects)
fixef(fit_probit)
```

##	Estimate	Est.Error	Q2.5	Q97.5
## Intercept	-0.53452942	0.56731756	-1.64310124	0.57663078
## Age	-0.01070062	0.01316401	-0.03693990	0.01478971
## Polyuria	1.60471144	0.33625137	0.95207913	2.30091760
## Polydipsia	1.69641068	0.34004485	1.04901167	2.37469578
## sudden_weight_loss	0.56396894	0.30697888	-0.01932090	1.17742260
## weakness	0.12813227	0.29127280	-0.44743340	0.69962150
## Polyphagia	0.46546691	0.30301783	-0.12577730	1.08442908
## Genital_thrush	0.86608374	0.32893101	0.23601348	1.52851692
## visual_blurring	0.63886994	0.34243295	-0.03832870	1.32410097
## Itching	-0.78015321	0.32892290	-1.43414700	-0.13770585
## Irritability	0.67462636	0.33457287	0.03219570	1.33820059
## delayed_healing	-0.24237616	0.31502753	-0.84823252	0.38315609
## partial_paresis	0.58201967	0.30695768	-0.03336033	1.16795246
## muscle_stiffness	-0.33205944	0.33449676	-1.00858332	0.33844578
## Alopecia	-0.25808505	0.31493079	-0.89113241	0.34596849
## Obesity	0.11815689	0.33278913	-0.54409066	0.78871423

```
# Check the first few posterior samples
head(posterior_samples(fit_probit))
```

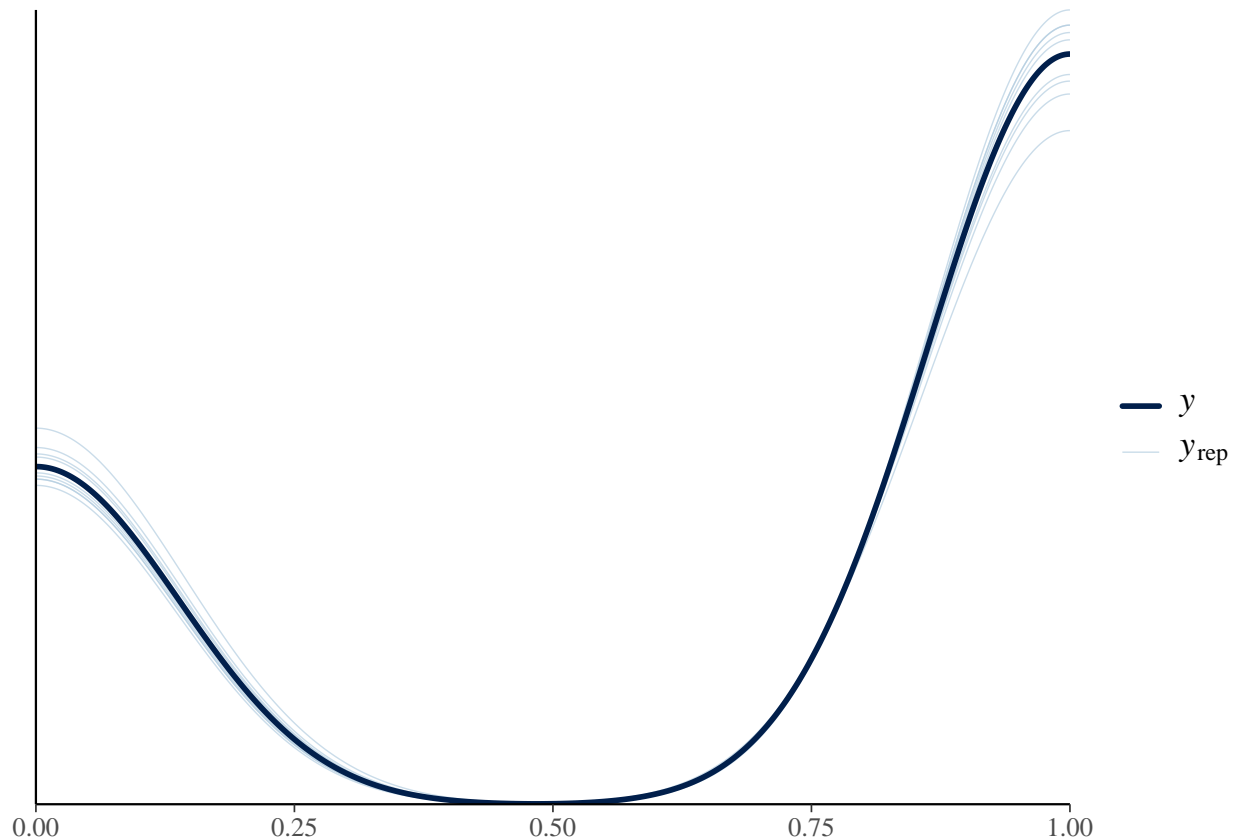
```
## Warning: Method 'posterior_samples' is deprecated. Please see ?as_draws for
## recommended alternatives.
```

##	b_Intercept	b_Age	b_Polyuria	b_Polydipsia	b_sudden_weight_loss
## 1	-1.10725382	0.001805965	1.7868771	0.8840681	0.2792649
## 2	-0.22130419	-0.018690806	1.4151176	2.7402095	1.1972684
## 3	-1.24341601	0.015692296	2.5262022	2.0125813	0.2559733
## 4	-0.15634801	-0.021614720	0.5235883	1.6842997	0.5523631
## 5	0.07799193	-0.024540500	0.7798000	2.3140430	0.1945644
## 6	-0.26745868	-0.021933839	0.9479691	2.0242256	0.3724910
##	b_weakness	b_Polyphagia	b_Genital_thrush	b_visual_blurring	b_Itching
## 1	0.33469874	0.4770060	0.9401970	0.5468016	-0.6446944
## 2	-0.06477899	0.5501483	0.6900196	0.6944521	-0.7558575

```
## 3  0.31764122    0.7019825    0.5515530    0.1034281 -1.1546823
## 4  0.13357991    0.5501434    1.2199354    1.3622858 -0.5841274
## 5 -0.25200574    0.6293881    1.8896803    0.8860027 -0.8895860
## 6 -0.22257127    0.9394300    1.5158065    0.8984474 -0.7533856
##   b_Irritability b_delayed_healing b_partial_paresis b_muscle_stiffness
## 1    0.1055772    -0.55724063    0.4726080    -0.09893532
## 2    1.5656869     0.17300588    0.6711435    -0.59254826
## 3    0.2637683    -0.56912282    1.0139957    -0.59143416
## 4    0.9863150    -0.04622703    0.3970872    -0.18992288
## 5    1.0047004    -0.20561502    0.7594751    -0.09084300
## 6    1.0766900    -0.28194448    0.5507098     0.06873542
##   b_Alopecia b_Obesity Intercept  lprior  lp__
## 1  0.04169338  0.1719188  1.043817 -18.99485 -87.52820
## 2 -0.15685114 -0.4075693  2.253126 -24.15306 -94.75451
## 3  0.08254778  0.2889975  2.182370 -23.30113 -96.24380
## 4 -0.46557936 -0.2505099  1.245508 -20.17870 -91.04152
## 5 -0.41522300 -0.4253321  1.354379 -22.56521 -93.52606
## 6 -0.17085121 -0.2094797  1.370037 -21.44933 -88.71173
```

```
# -----
# 6. Posterior Predictive Checks (Model Diagnostics)
# -----
# Density overlay of observed vs predicted values
pp_check(fit_probit, type = "dens_overlay")
```

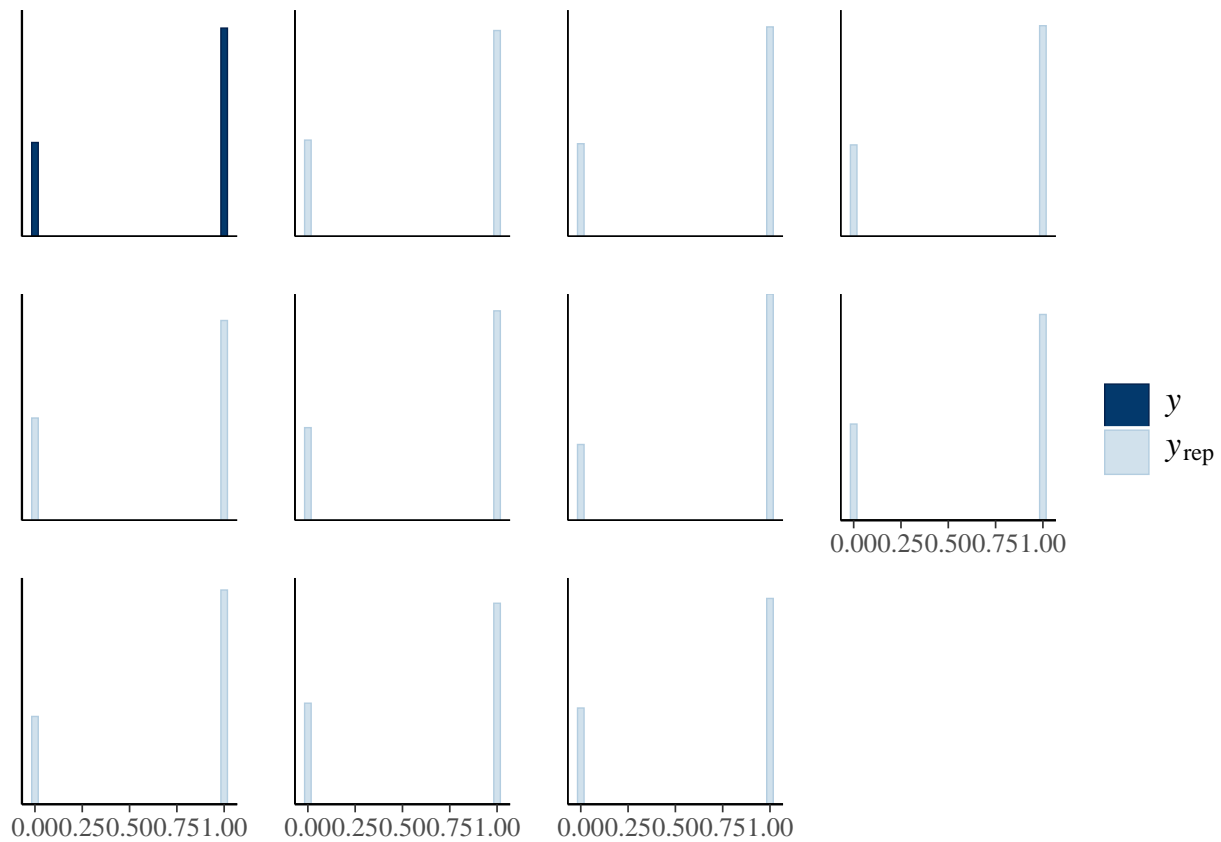
```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```



```
# Histogram of simulated posterior predictions
pp_check(fit_probit, type = "hist")
```

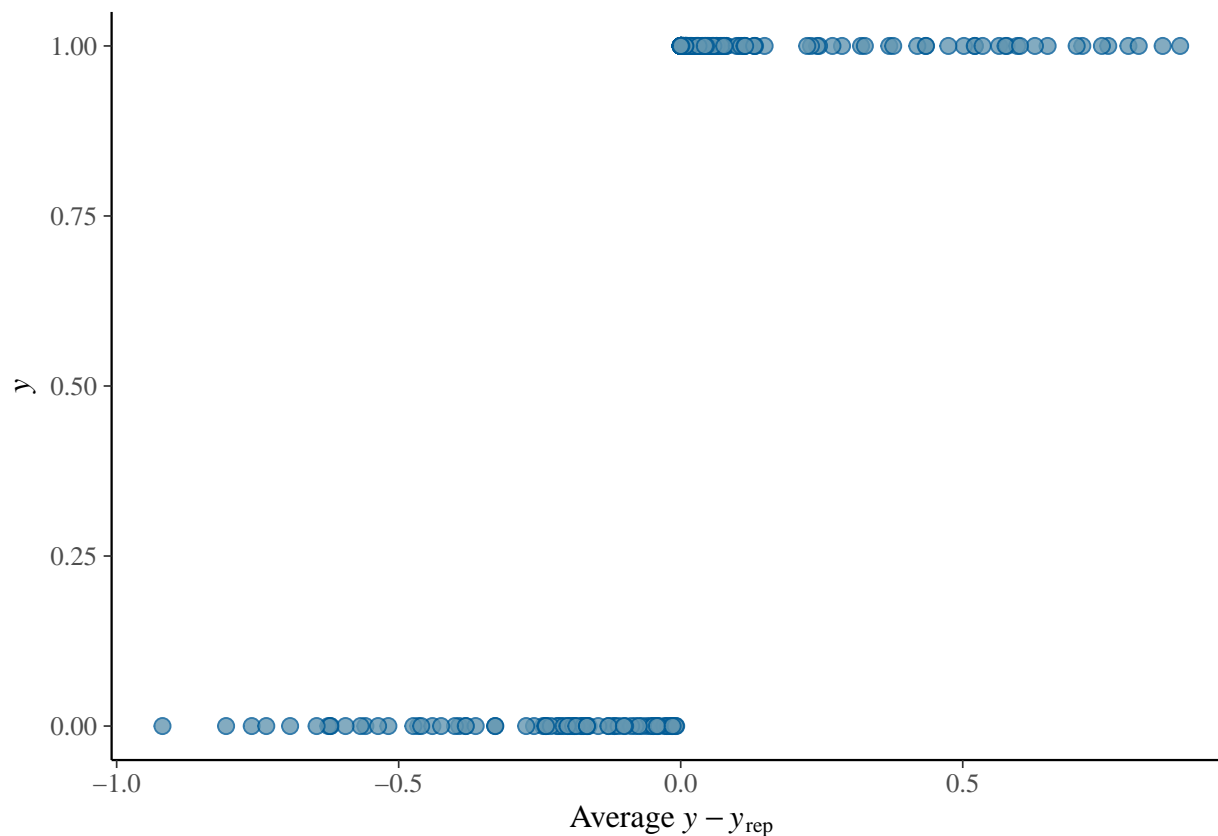
```
## Using 10 posterior draws for ppc type 'hist' by default.
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



```
pp_check(fit_probit, type = "error_scatter_avg")
```

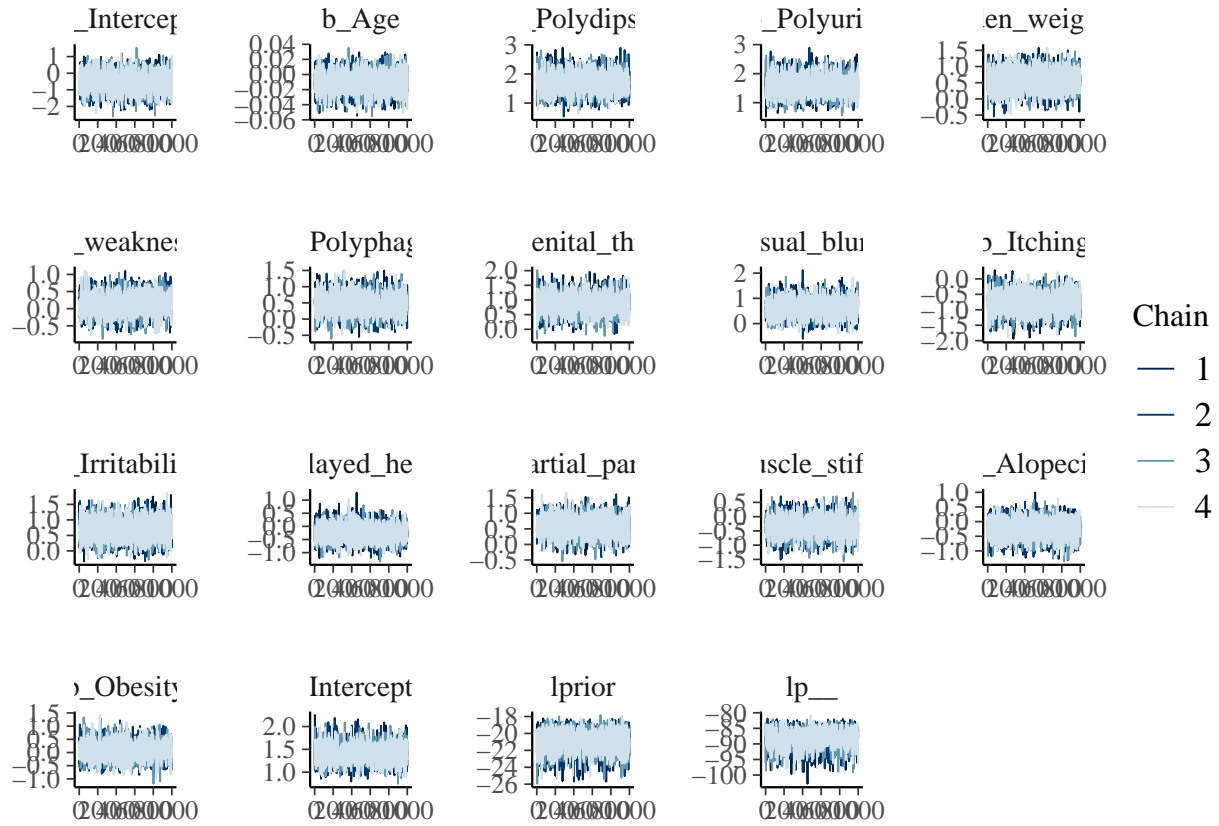
```
## Using all posterior draws for ppc type 'error_scatter_avg' by default.
```



```
# -----
# 7. Convergence Diagnostics and Visualization
# -----
# Check if chains have converged (Trace Plots)
# Get all parameter names in the model
variables(as_draws(fit_probit))
```

```
## [1] "b_Intercept"      "b_Age"            "b_Polyuria"
## [4] "b_Polydipsia"     "b_sudden_weight_loss" "b_weakness"
## [7] "b_Polyphagia"     "b_Genital_thrush"  "b_visual_blurring"
## [10] "b_Itching"        "b_Irritability"   "b_delayed_healing"
## [13] "b_partial_paresis" "b_muscle_stiffness" "b_Alopecia"
## [16] "b_Obesity"        "Intercept"        "lprior"
## [19] "lp_--"
```

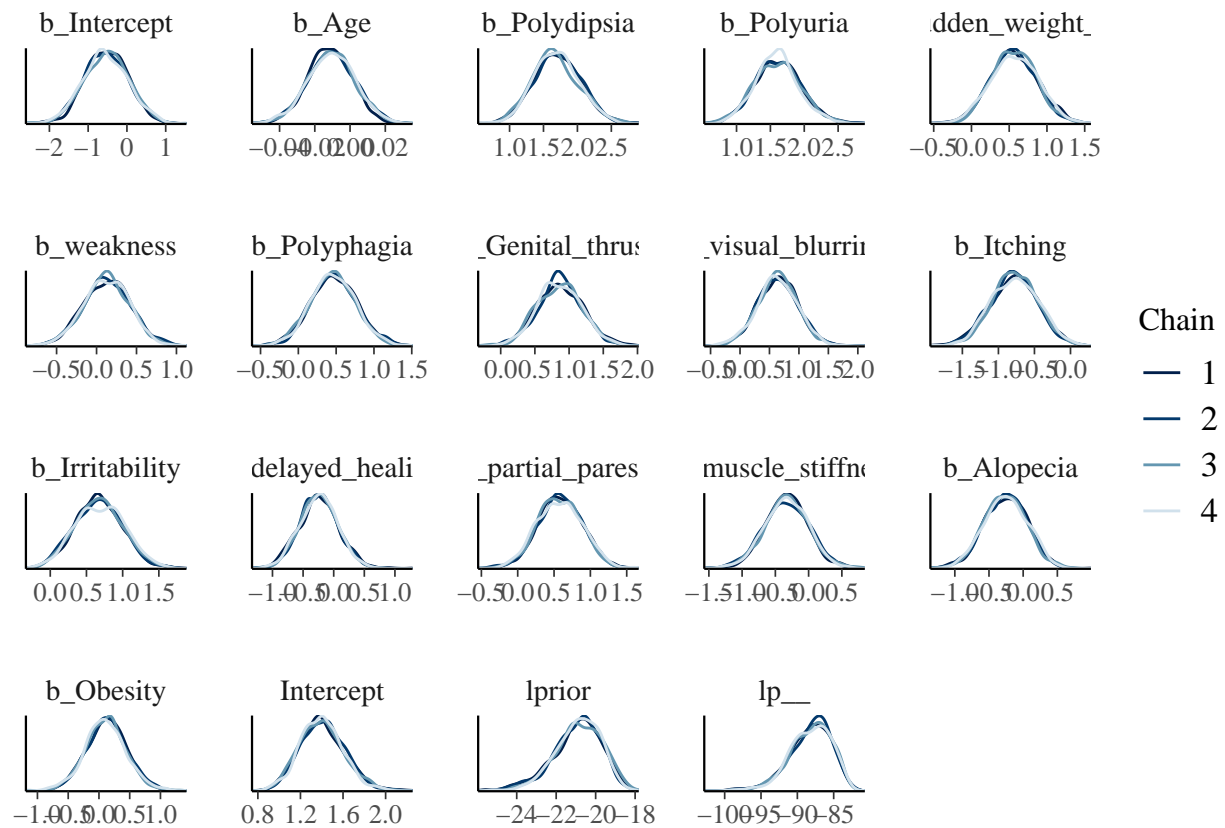
```
mcmc_trace(as_draws(fit_probit),
  pars = c("b_Intercept", "b_Age", "b_Polydipsia", "b_Polyuria", "b_sudden_weight_loss", "b_
```



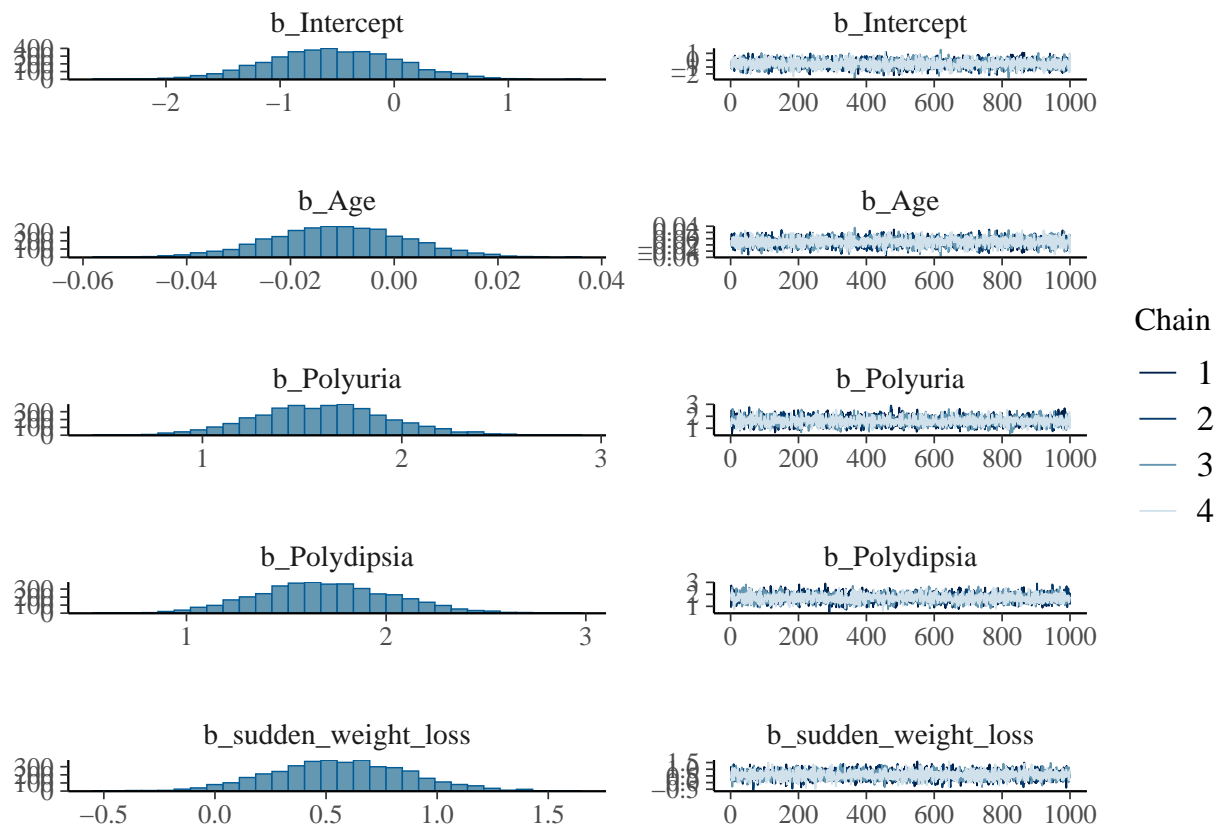
```
# Posterior Density Overlays
```

```
mcmc_dens_overlay(as.array(fit_probit),
```

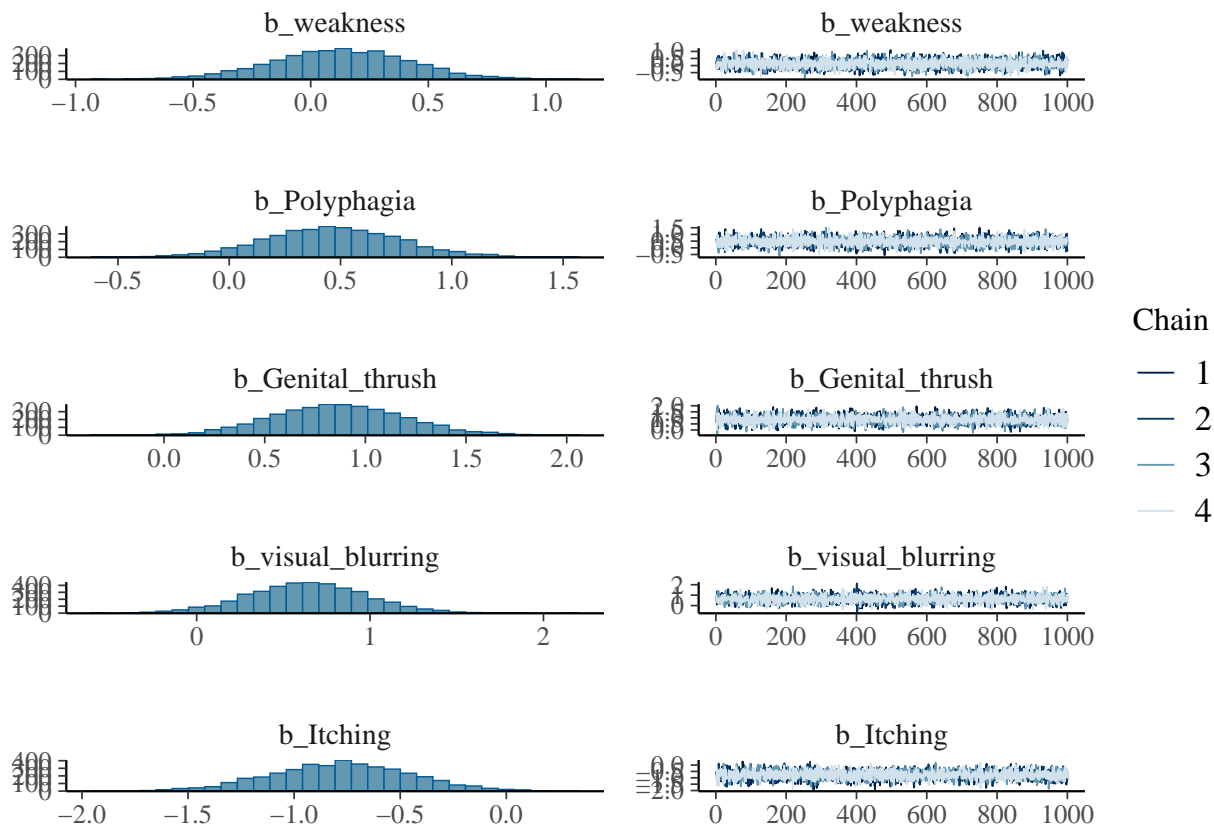
```
                  pars = c("b_Intercept", "b_Age", "b_Polydipsia", "b_Polyuria", "b_sudden_weight_loss",
```

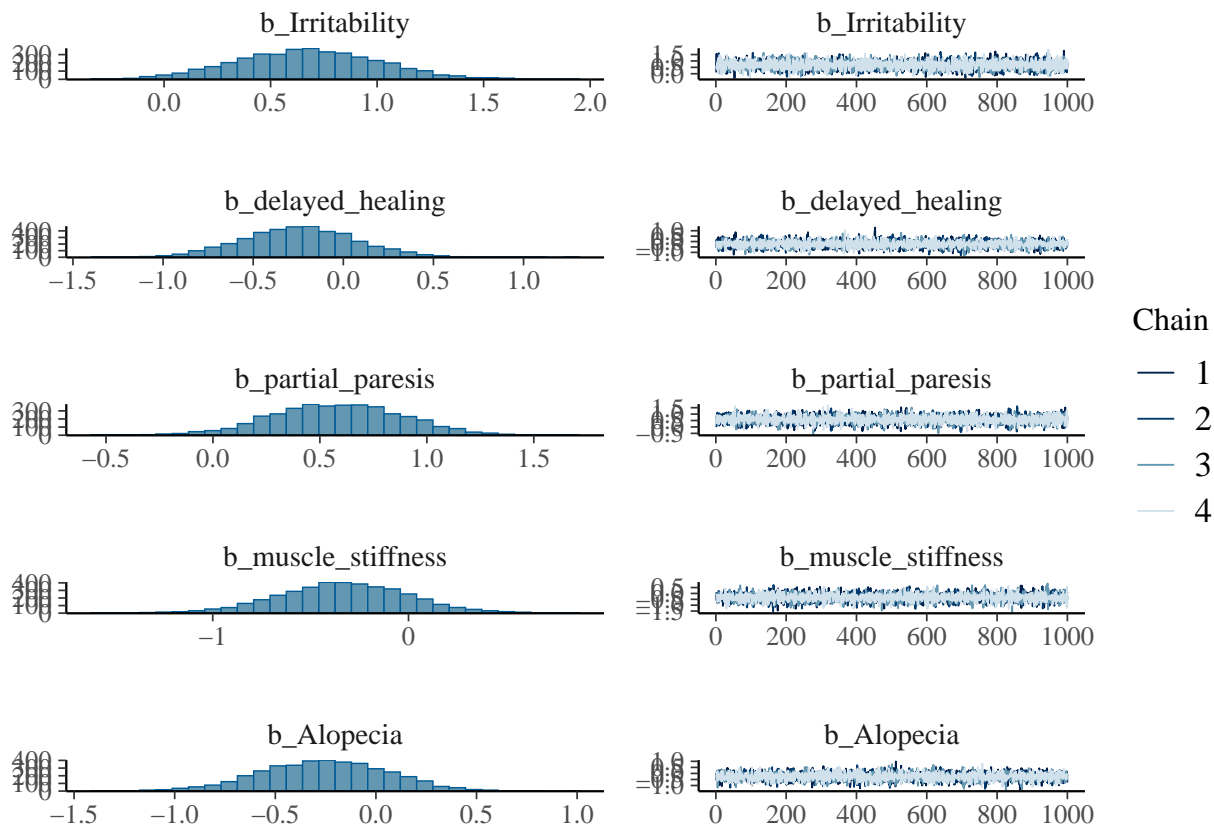


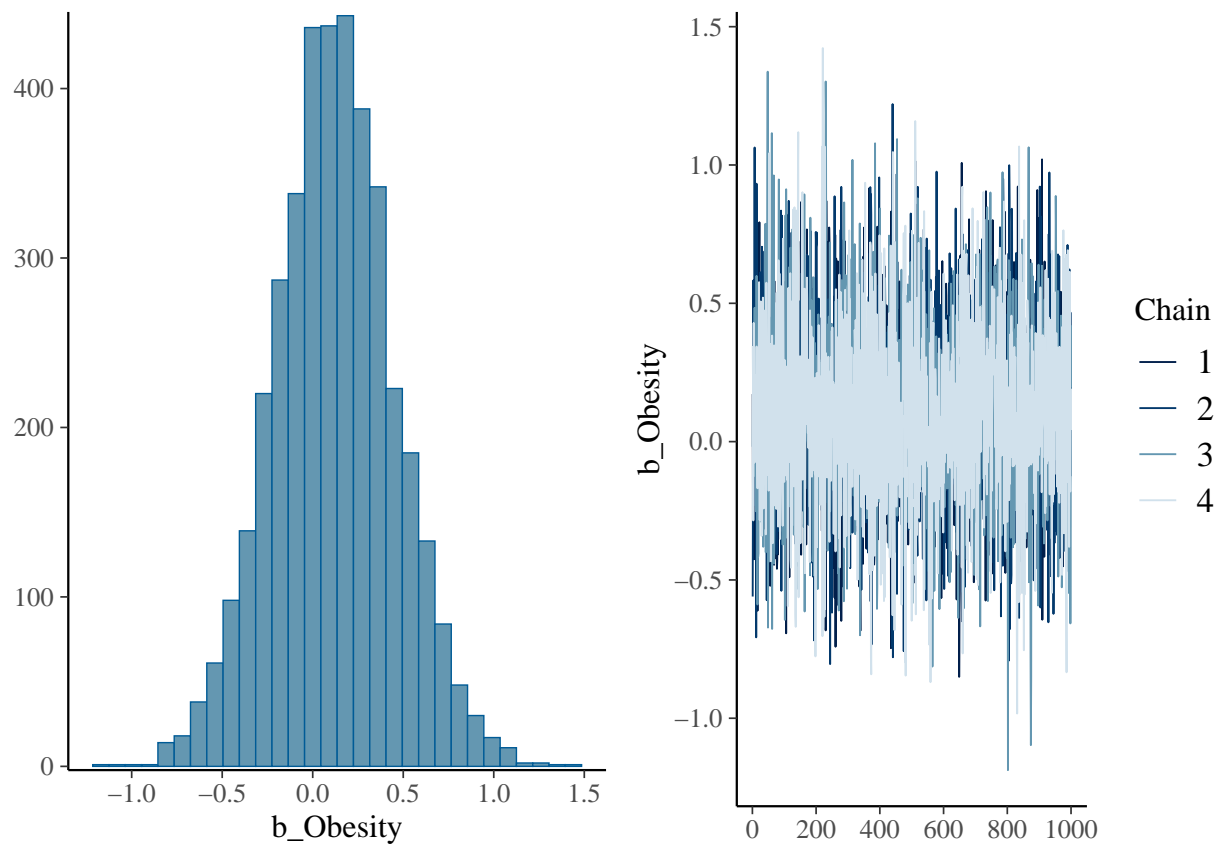
```
# Posterior Intervals
plot(fit_probit)
```



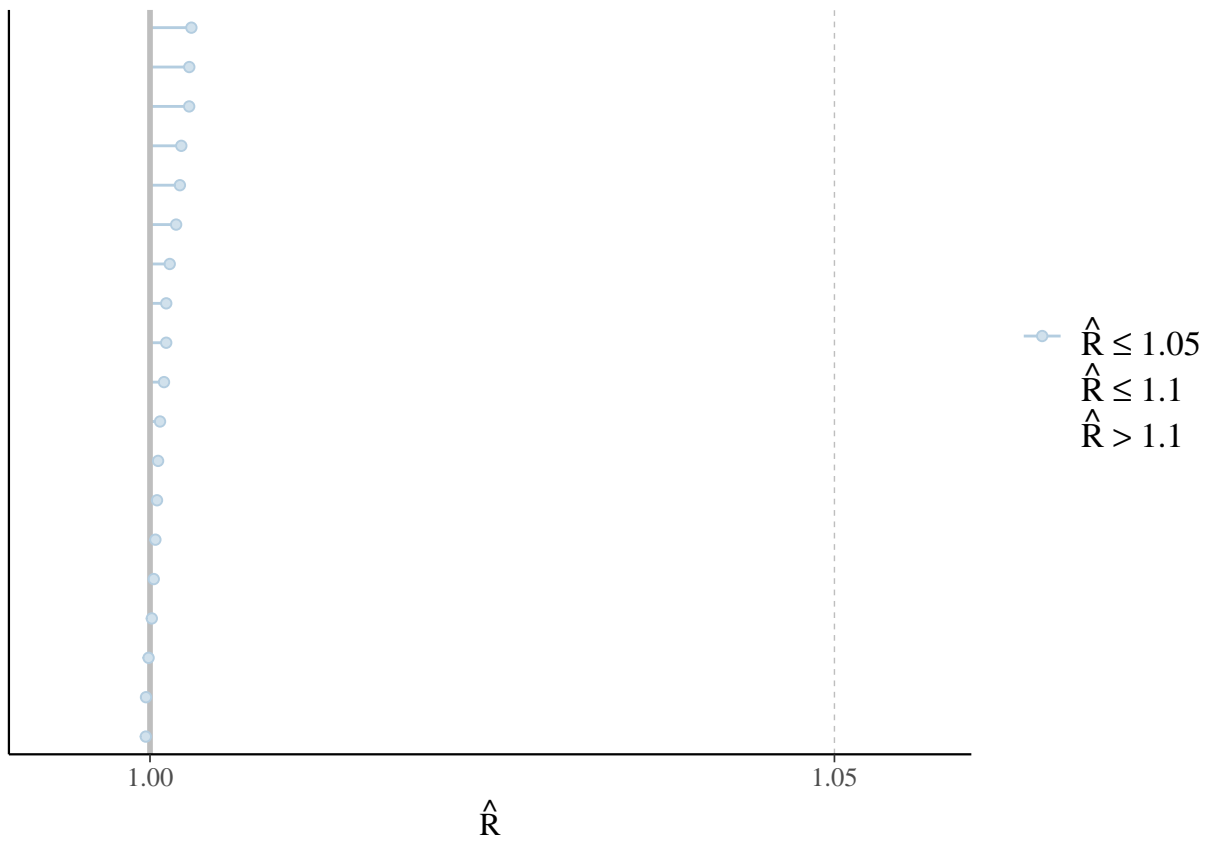




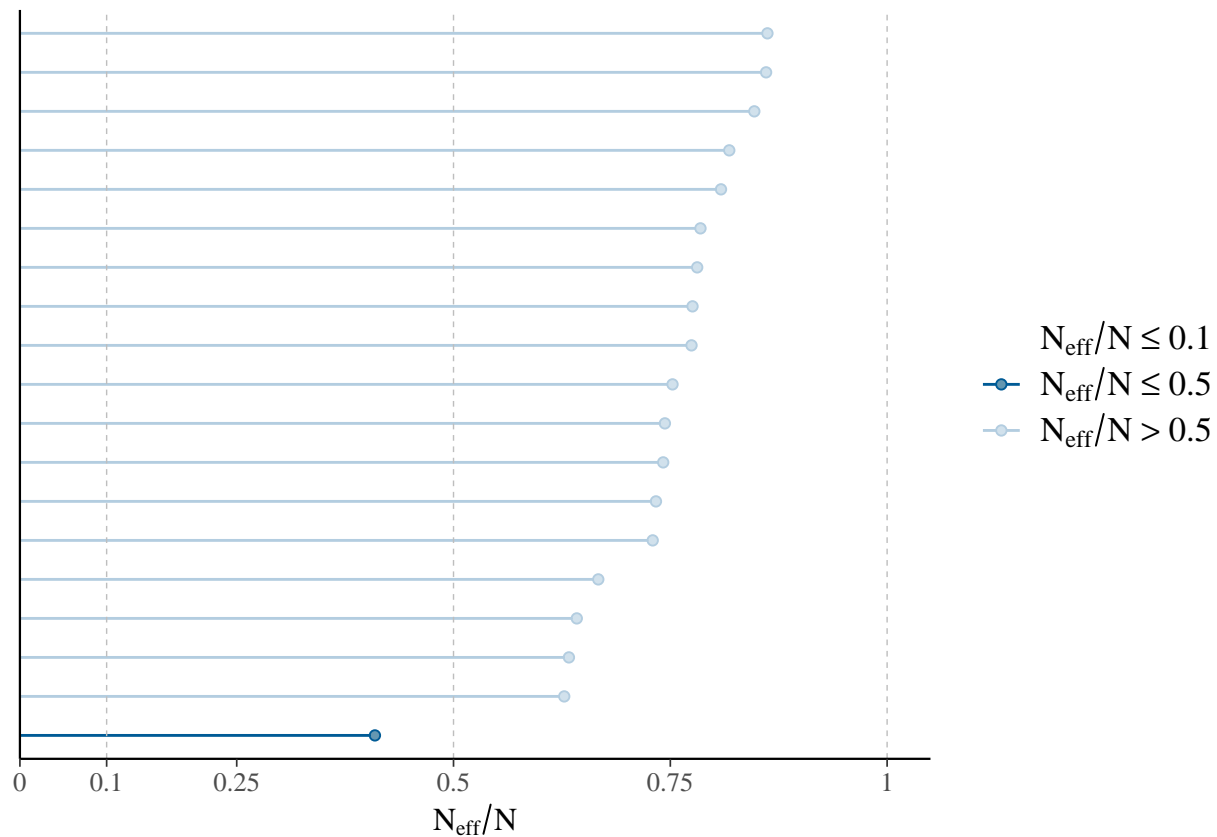




```
mcmc_plot(fit_probit, type = "rhat")
```



```
mcmc_plot(fit_probit, type = "neff")
```

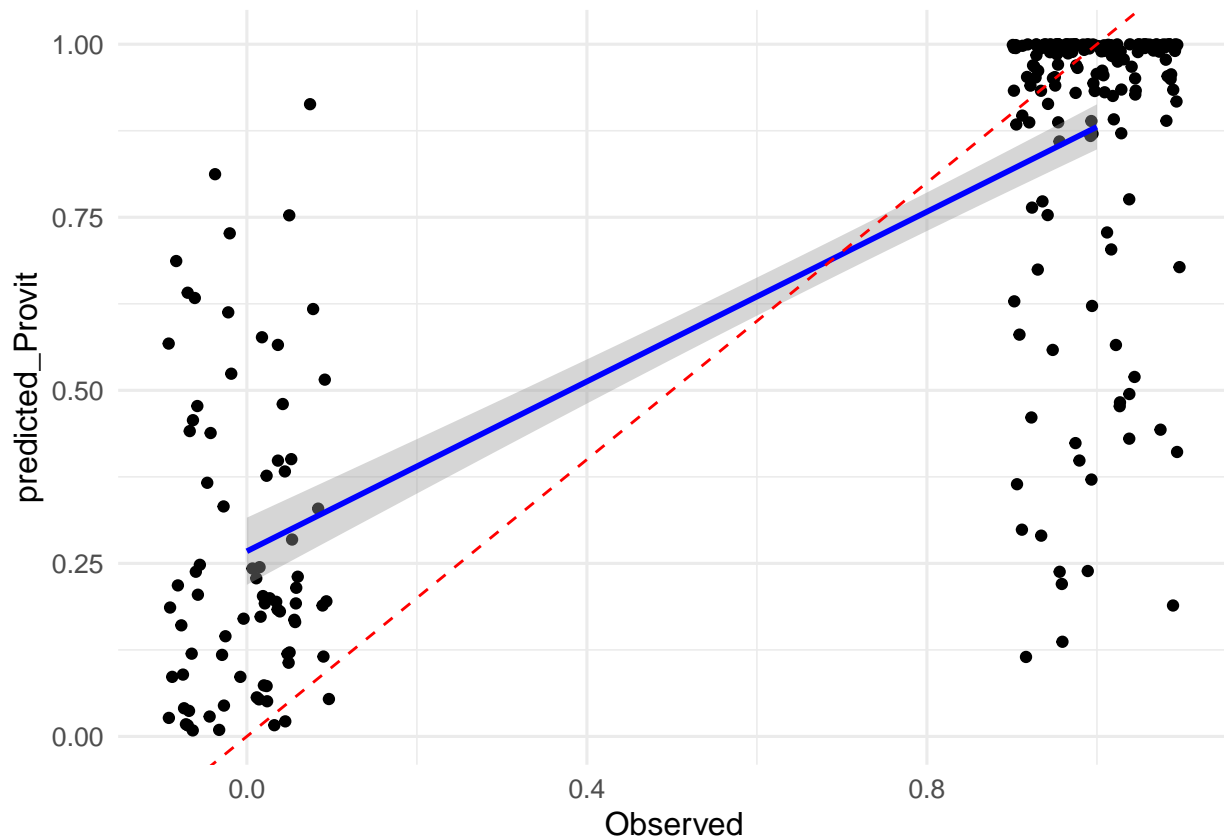


*#Plot the relationship between observed and predicted values*

```
data$predicted_Provit <- fitted(fit_probit)[, "Estimate"]
```

*# Logistic Regression*

```
ggplot(data, aes(x = class, y = predicted_Provit)) +  
  geom_jitter(width = 0.1, height = 0) +  
  geom_smooth(method = "lm", formula = y ~ x, color = "blue") +  
  geom_abline(slope = 1, intercept = 0, color = "red", linetype = "dashed") +  
  labs(x = "Observed", y = "predicted_Provit", color = "Observed Class") +  
  theme_minimal(base_size = 12)
```



```
# Compute Leave-One-Out Cross-Validation (LOO) for each model
loo_probit <- loo(fit_probit, moment_match = TRUE, reloo = TRUE)
```

```
## Warning: Some Pareto k diagnostic values are too high. See help('pareto-k-diagnostic') for details.
```

```
## 5 problematic observation(s) found.
```

```
## The model will be refit 5 times.
```

```
##
```

```
## Fitting model 1 out of 5 (leaving out observation 30)
```

```
## Start sampling
```

```
##
```

```
## Fitting model 2 out of 5 (leaving out observation 32)
```

```
## Start sampling
```

```
##
```

```
## Fitting model 3 out of 5 (leaving out observation 51)
```

```
## Start sampling
```

```
##
## Fitting model 4 out of 5 (leaving out observation 111)

## Start sampling

##
## Fitting model 5 out of 5 (leaving out observation 156)

## Start sampling

loo_logit <- loo(fit_logit, moment_match = TRUE, relloo = TRUE)

## No problematic observations found. Returning the original 'loo' object.
```

```
# Compare models
loo_compare(loo_probit, loo_logit)
```

```
##           elpd_diff se_diff
## fit_logit    0.0      0.0
## fit_probit -0.4      1.4
```

```
# Compare Model Coefficients (Fixed Effects Estimates)
# Extract fixed effects (posterior means & intervals)
logit_coef <- fixef(fit_logit)
probit_coef <- fixef(fit_probit)
```

```
# Print comparison of coefficients
print("Logit Model Coefficients:")
```

```
## [1] "Logit Model Coefficients:"
```

```
print(logit_coef)
```

```
##           Estimate Est.Error      Q2.5      Q97.5
## Intercept    -0.80120297 0.89545487 -2.564210370 0.95997235
## Age          -0.01642244 0.02109202 -0.058675101 0.02391545
## Polyuria      2.26495381 0.49294825  1.332048575 3.27174866
## Polydipsia    2.35170517 0.50585099  1.369643000 3.35451904
## sudden_weight_loss 0.92497261 0.44542445  0.064998216 1.77874968
## weakness      0.16231782 0.42734848 -0.688729157 1.00453917
## Polyphagia    0.69318103 0.44988928 -0.179697091 1.59018486
## Genital_thrush 1.18862217 0.50333108  0.218080480 2.18152789
## visual_blurring 0.83076547 0.49701372 -0.123645622 1.79556494
## Itching       -0.88889485 0.47448139 -1.834356941 0.04340460
## Irritability   0.95679616 0.48175733  0.003176781 1.89471832
## delayed_healing -0.34497595 0.48211244 -1.314430574 0.59819604
## partial_paresis 0.91112851 0.45973601  0.024564559 1.85611671
## muscle_stiffness -0.35516524 0.48215407 -1.291760966 0.58412472
## Alopecia      -0.43707302 0.47684779 -1.401237888 0.49957054
## Obesity       0.15571816 0.51379889 -0.873471147 1.17664204
```

```
print("Probit Model Coefficients:")
```

```
## [1] "Probit Model Coefficients:"
```

```
print(probit_coef)
```

##	Estimate	Est.Error	Q2.5	Q97.5
## Intercept	-0.53452942	0.56731756	-1.64310124	0.57663078
## Age	-0.01070062	0.01316401	-0.03693990	0.01478971
## Polyuria	1.60471144	0.33625137	0.95207913	2.30091760
## Polydipsia	1.69641068	0.34004485	1.04901167	2.37469578
## sudden_weight_loss	0.56396894	0.30697888	-0.01932090	1.17742260
## weakness	0.12813227	0.29127280	-0.44743340	0.69962150
## Polyphagia	0.46546691	0.30301783	-0.12577730	1.08442908
## Genital_thrush	0.86608374	0.32893101	0.23601348	1.52851692
## visual_blurring	0.63886994	0.34243295	-0.03832870	1.32410097
## Itching	-0.78015321	0.32892290	-1.43414700	-0.13770585
## Irritability	0.67462636	0.33457287	0.03219570	1.33820059
## delayed_healing	-0.24237616	0.31502753	-0.84823252	0.38315609
## partial_paresis	0.58201967	0.30695768	-0.03336033	1.16795246
## muscle_stiffness	-0.33205944	0.33449676	-1.00858332	0.33844578
## Alopecia	-0.25808505	0.31493079	-0.89113241	0.34596849
## Obesity	0.11815689	0.33278913	-0.54409066	0.78871423

```
#Compare Model Fit (LOO & WAIC)
```

```
# Compute model fit criteria
```

```
logit_fit <- loo(fit_logit)
```

```
probit_fit <- loo(fit_probit)
```

```
## Warning: Found 5 observations with a pareto_k > 0.7 in model 'fit_probit'. We  
## recommend to set 'moment_match = TRUE' in order to perform moment matching for  
## problematic observations.
```

```
waic_logit <- waic(fit_logit)
```

```
## Warning:  
## 3 (1.2%) p_waic estimates greater than 0.4. We recommend trying loo instead.
```

```
waic_probit <- waic(fit_probit)
```

```
## Warning:  
## 8 (3.2%) p_waic estimates greater than 0.4. We recommend trying loo instead.
```

```
# Compare LOO scores
```

```
print("Logit Model LOO Score:")
```

```
## [1] "Logit Model LOO Score:"
```



```
print(logit_fit)
```

```
##
## Computed from 4000 by 251 log-likelihood matrix.
##
##      Estimate   SE
## elpd_loo    -75.5  7.5
## p_loo       11.7  1.5
## looic       151.1 14.9
## -----
## MCSE of elpd_loo is 0.1.
## MCSE and ESS estimates assume MCMC draws (r_eff in [0.5, 1.6]).
##
## All Pareto k estimates are good (k < 0.7).
## See help('pareto-k-diagnostic') for details.
```

```
print("Probit Model L00 Score:")
```

```
## [1] "Probit Model L00 Score:"
```

```
print(probit_fit)
```

```
##
## Computed from 4000 by 251 log-likelihood matrix.
##
##      Estimate   SE
## elpd_loo    -76.0  8.6
## p_loo       14.9  2.2
## looic       151.9 17.2
## -----
## MCSE of elpd_loo is NA.
## MCSE and ESS estimates assume MCMC draws (r_eff in [0.6, 1.8]).
##
## Pareto k diagnostic values:
##              Count Pct.   Min. ESS
## (-Inf, 0.7] (good)   246  98.0%   335
##  (0.7, 1]   (bad)     5   2.0%   <NA>
##  (1, Inf)   (very bad) 0   0.0%   <NA>
## See help('pareto-k-diagnostic') for details.
```

```
# Compare WAIC scores
print("Logit Model WAIC Score:")
```

```
## [1] "Logit Model WAIC Score:"
```

```
print(waic_logit)
```

```
##
## Computed from 4000 by 251 log-likelihood matrix.
##
```

```
##           Estimate    SE
## elpd_waic    -75.4   7.4
## p_waic       11.5   1.5
## waic         150.8  14.9
##
## 3 (1.2%) p_waic estimates greater than 0.4. We recommend trying loo instead.
```

```
print("Probit Model WAIC Score:")
```

```
## [1] "Probit Model WAIC Score:"
```

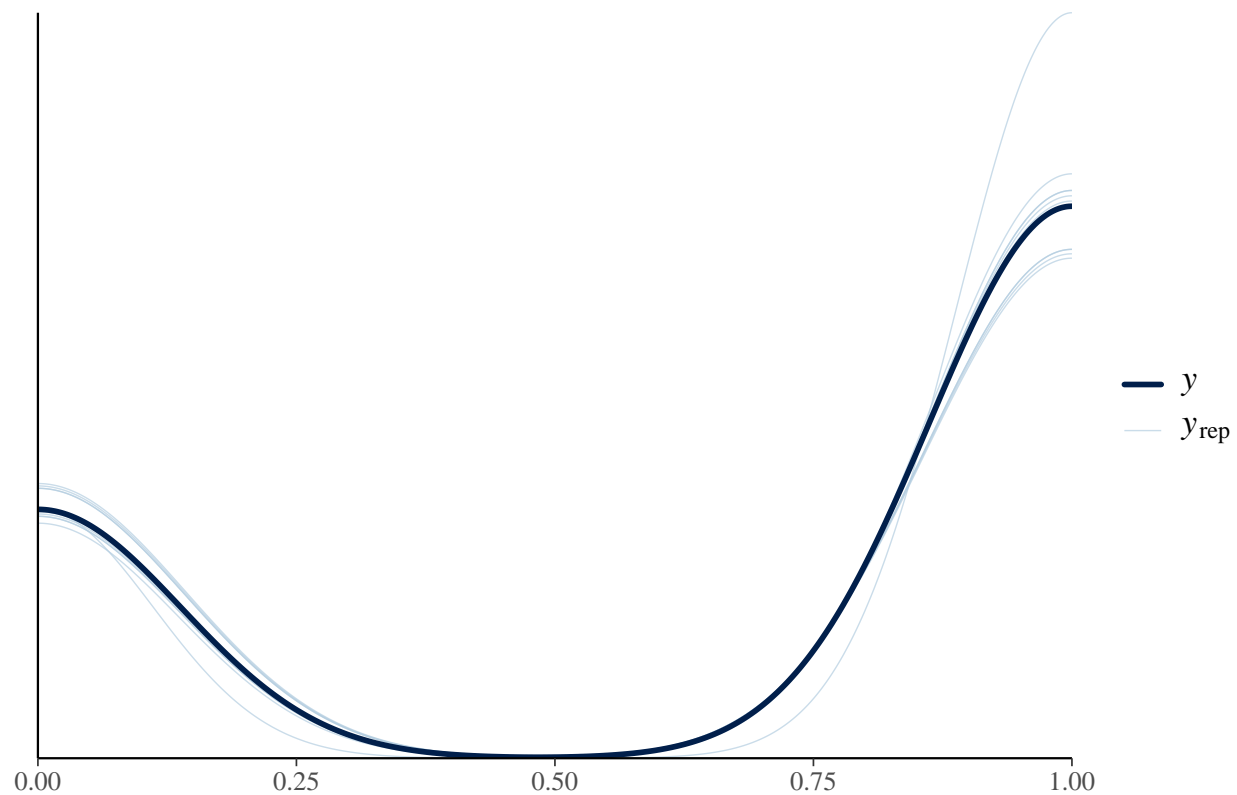
```
print(waic_probit)
```

```
##
## Computed from 4000 by 251 log-likelihood matrix.
##
##           Estimate    SE
## elpd_waic    -75.5   8.5
## p_waic       14.4   2.1
## waic         150.9  17.0
##
## 8 (3.2%) p_waic estimates greater than 0.4. We recommend trying loo instead.
```

```
#Compare Posterior Predictive Checks
# Posterior predictive checks for Logit Model
pp_check(fit_logit, type = "dens_overlay") +
  ggtitle("Posterior Predictive Check - Logit")
```

```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```

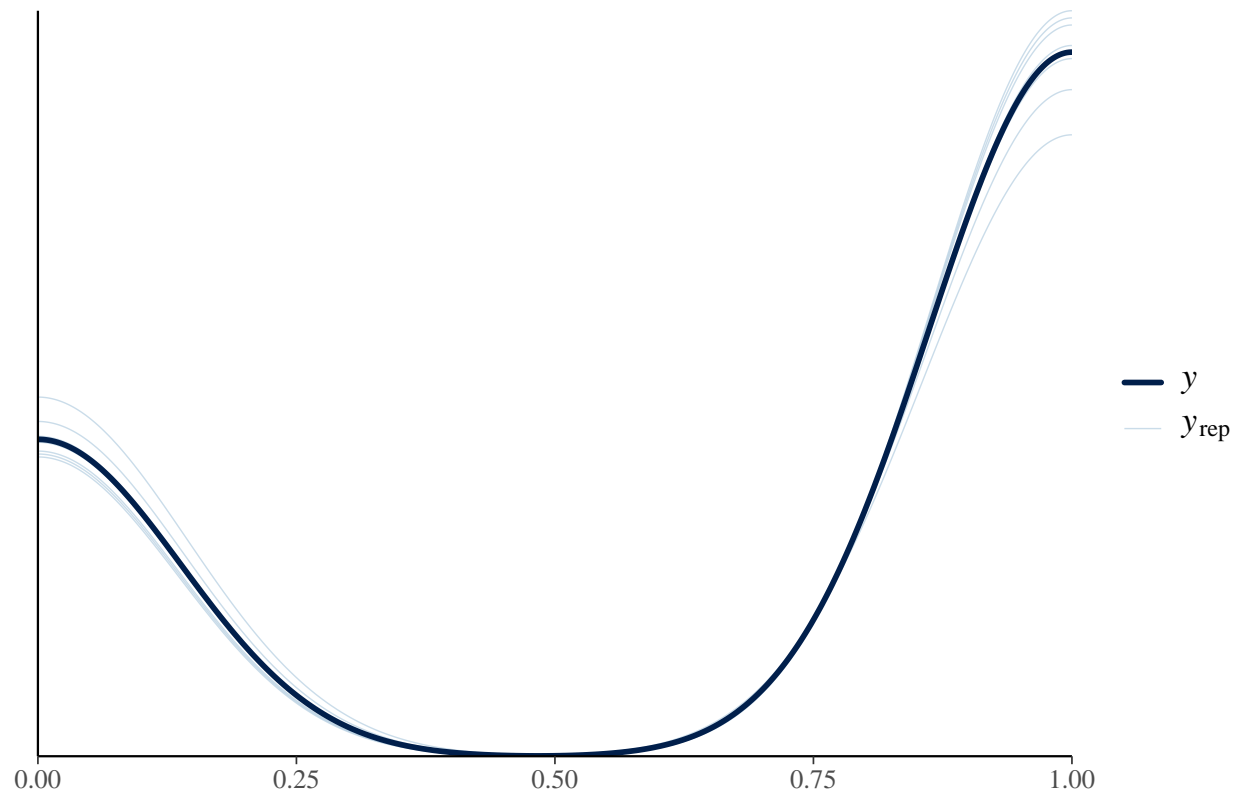
## Posterior Predictive Check – Logit



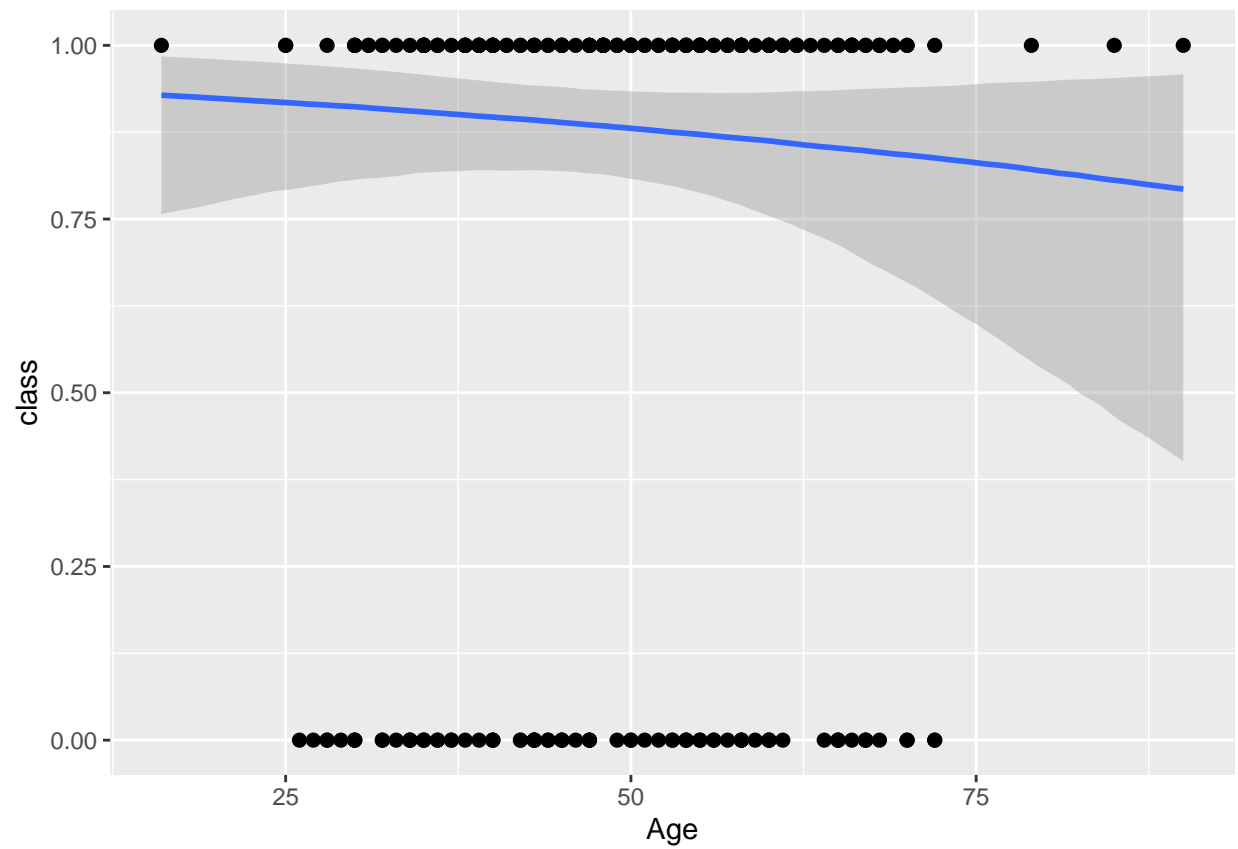
```
# Posterior predictive checks for Probit Model  
pp_check(fit_probit, type = "dens_overlay") +  
  ggtitle("Posterior Predictive Check - Probit")
```

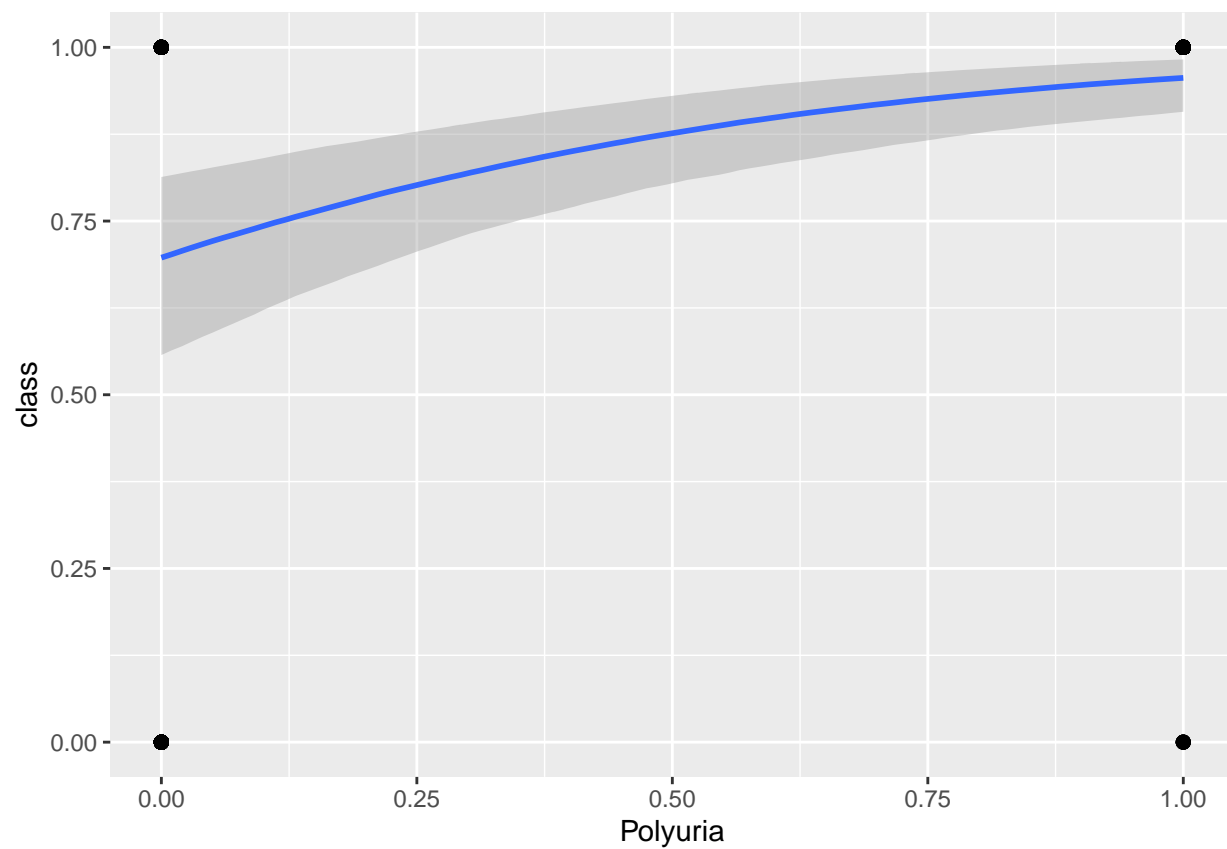
```
## Using 10 posterior draws for ppc type 'dens_overlay' by default.
```

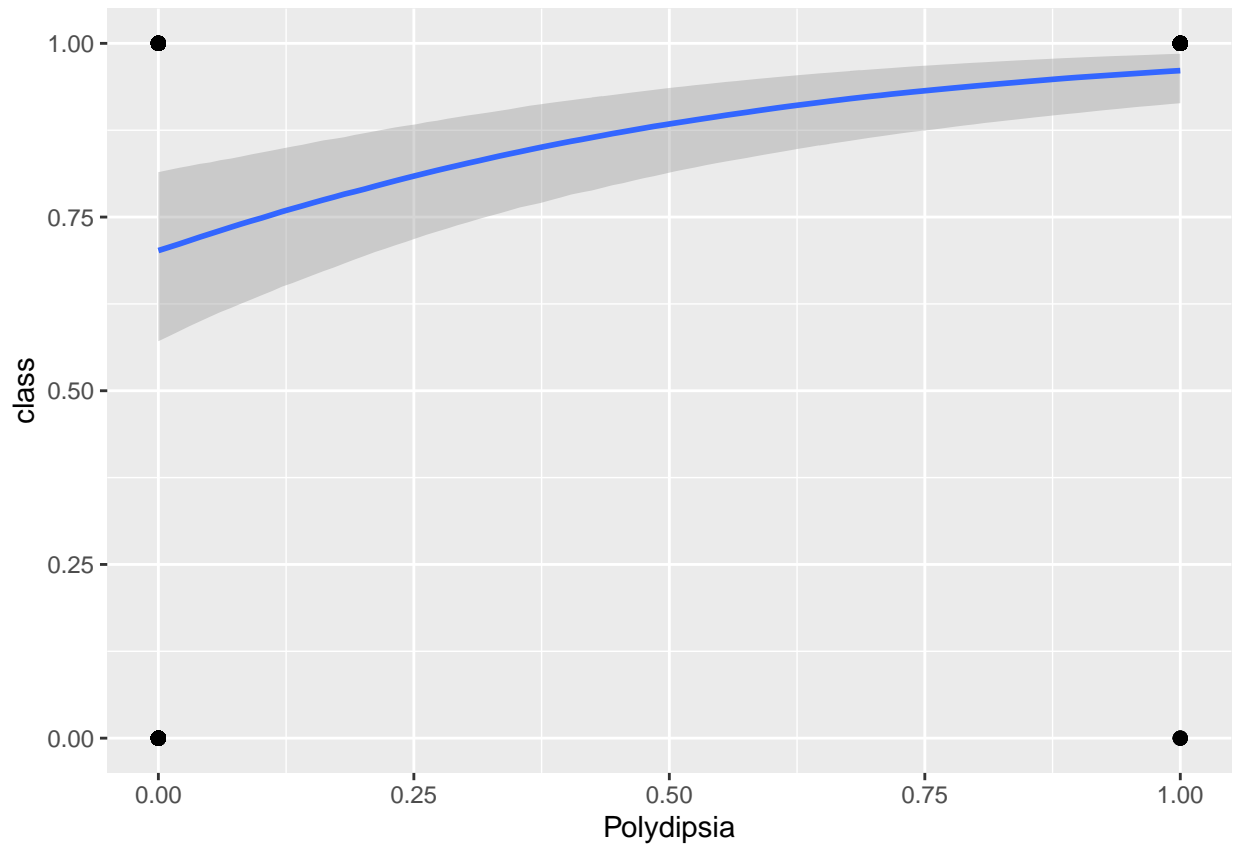
## Posterior Predictive Check – Probit

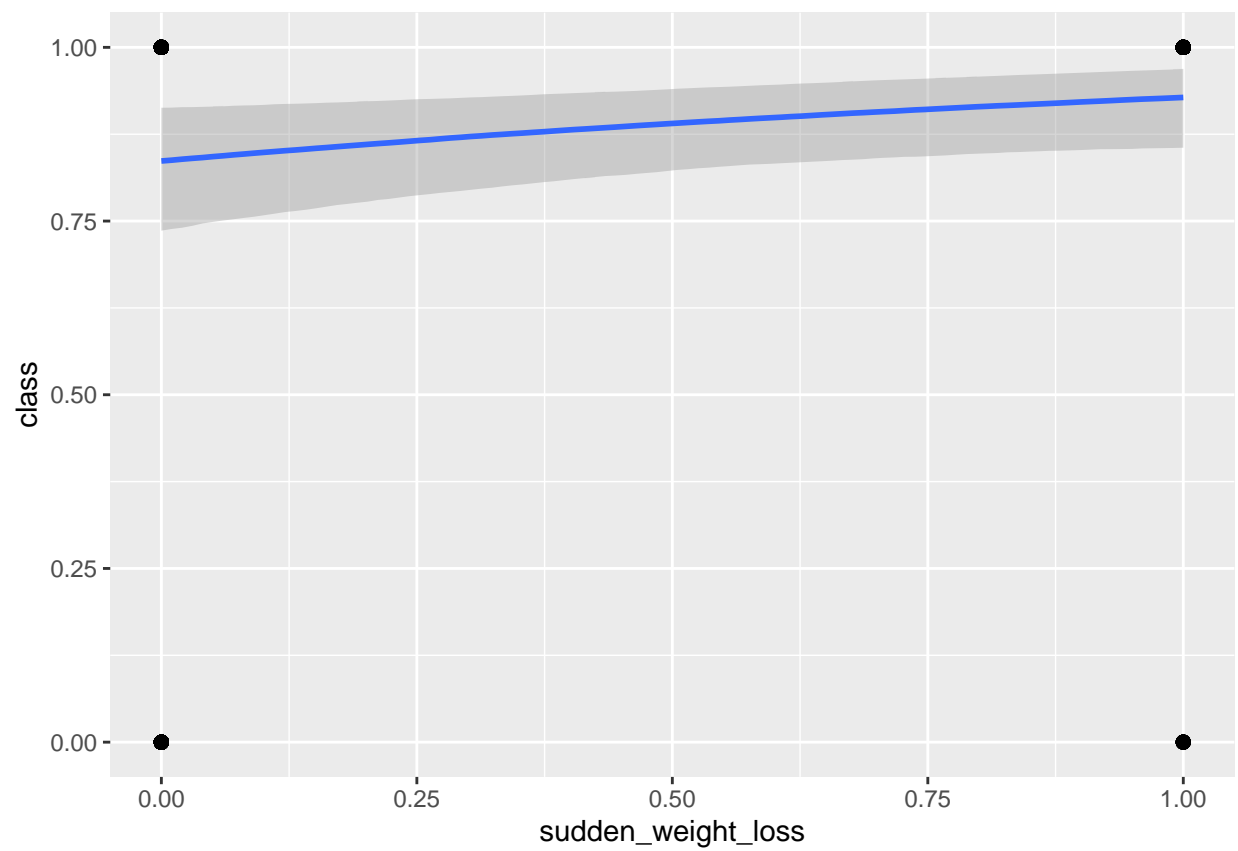


```
#Compare Marginal Effects  
# Marginal effects for Logit Model  
plot(conditional_effects(fit_logit), points = TRUE)
```

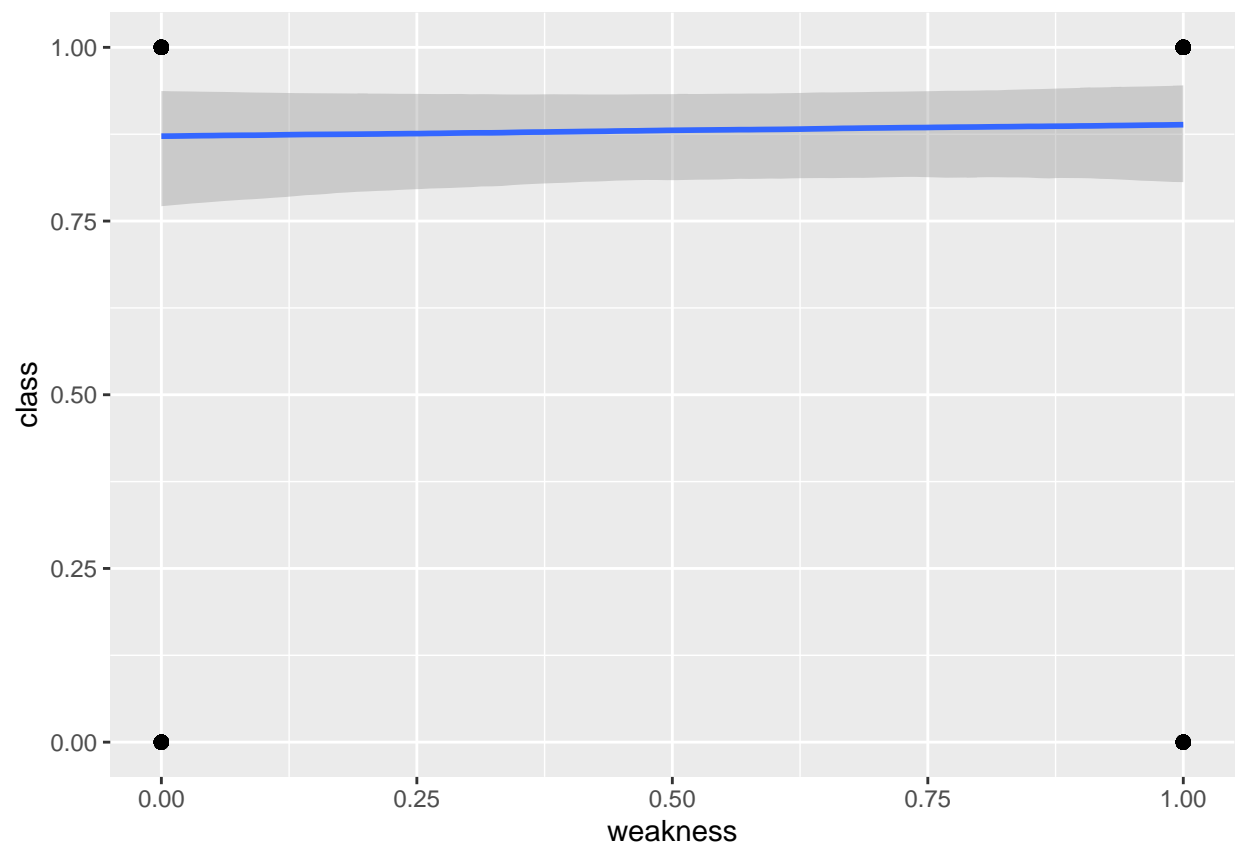


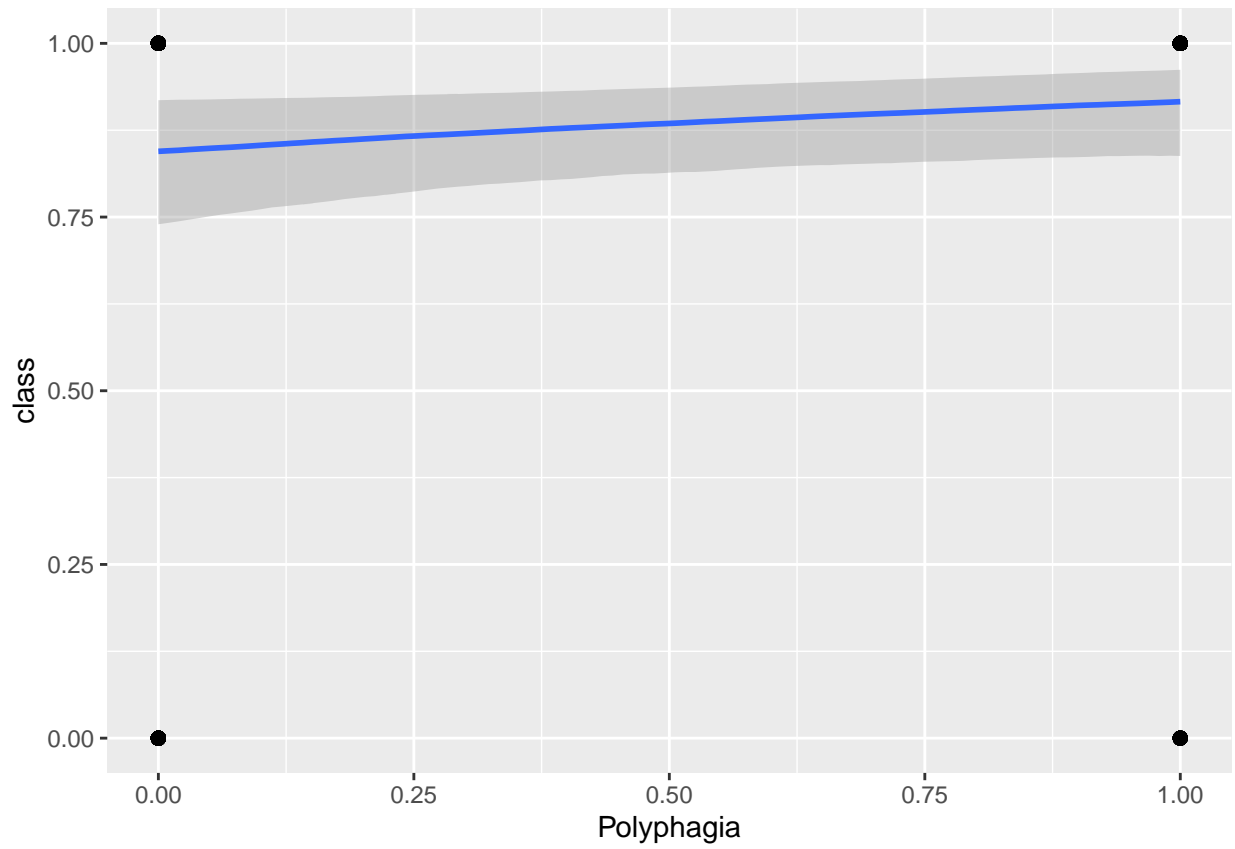


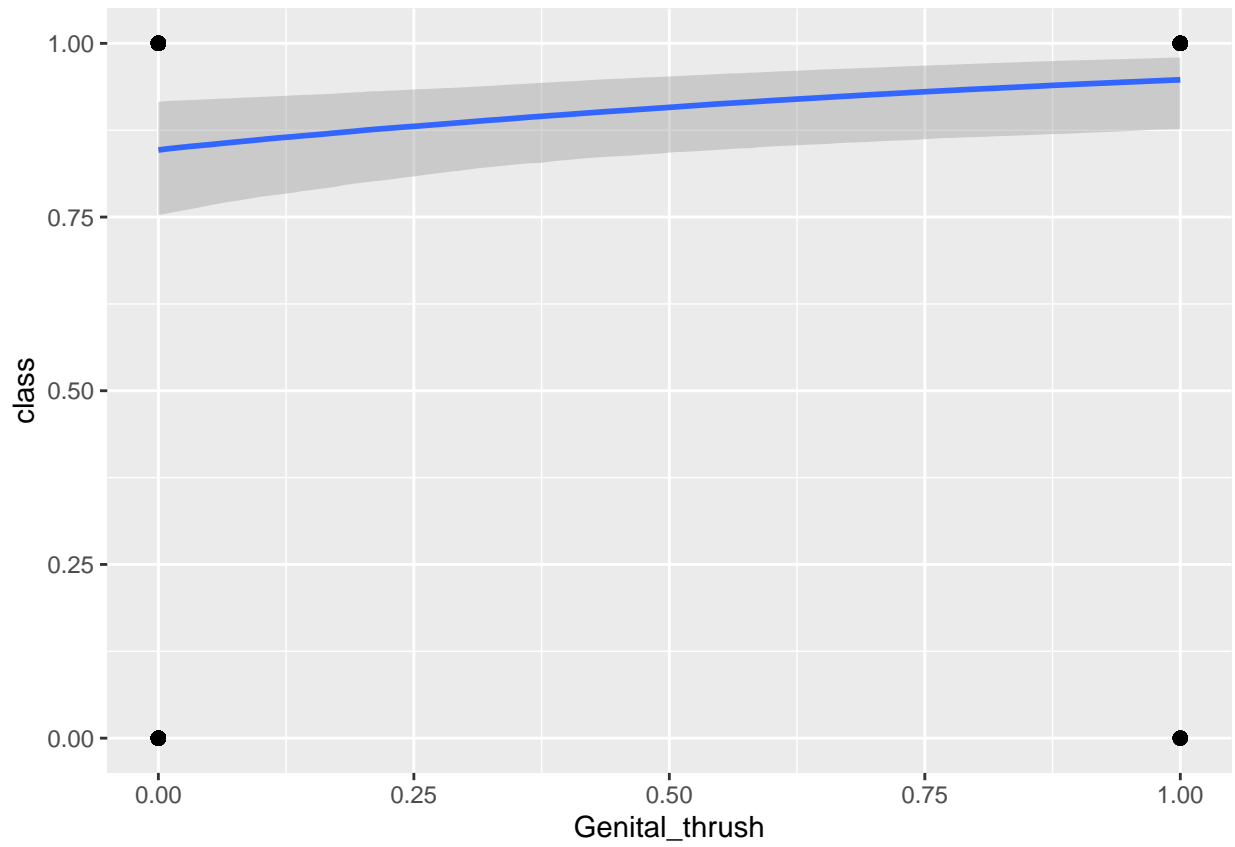


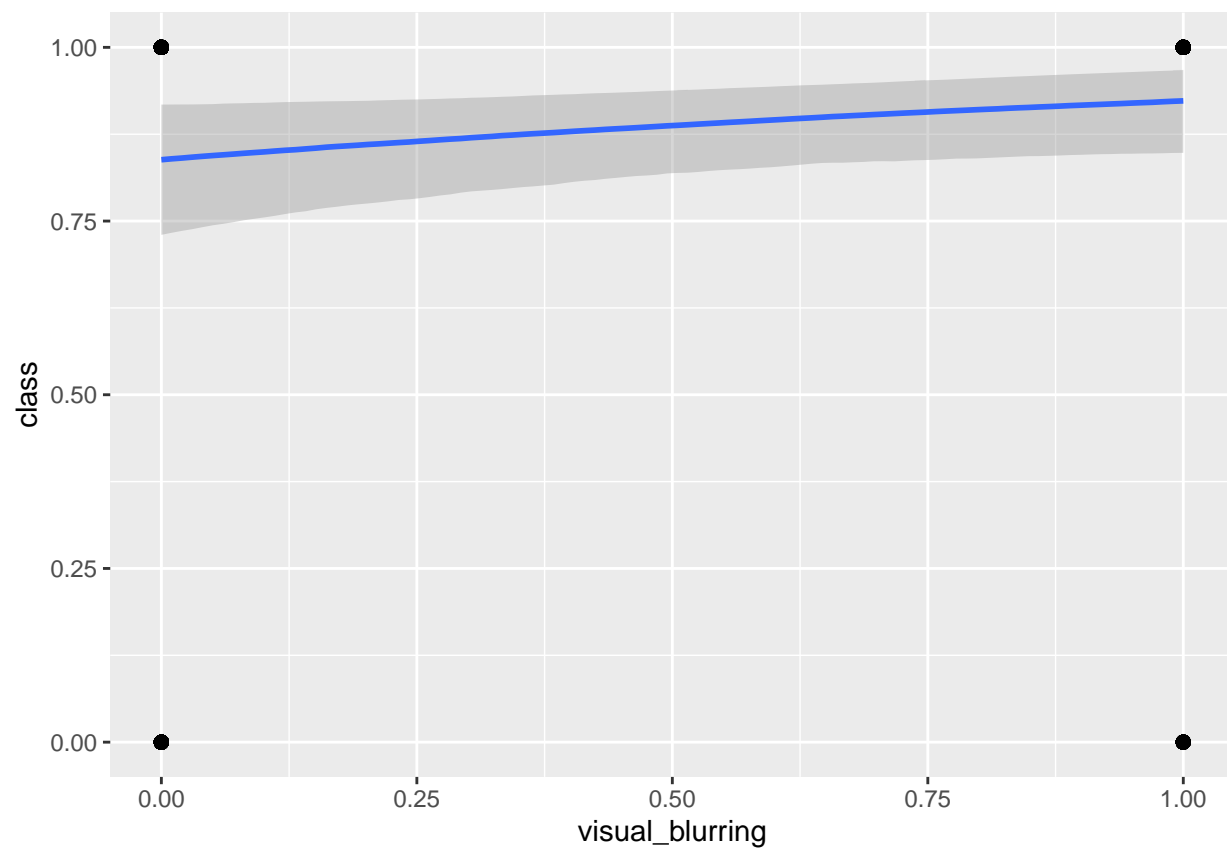


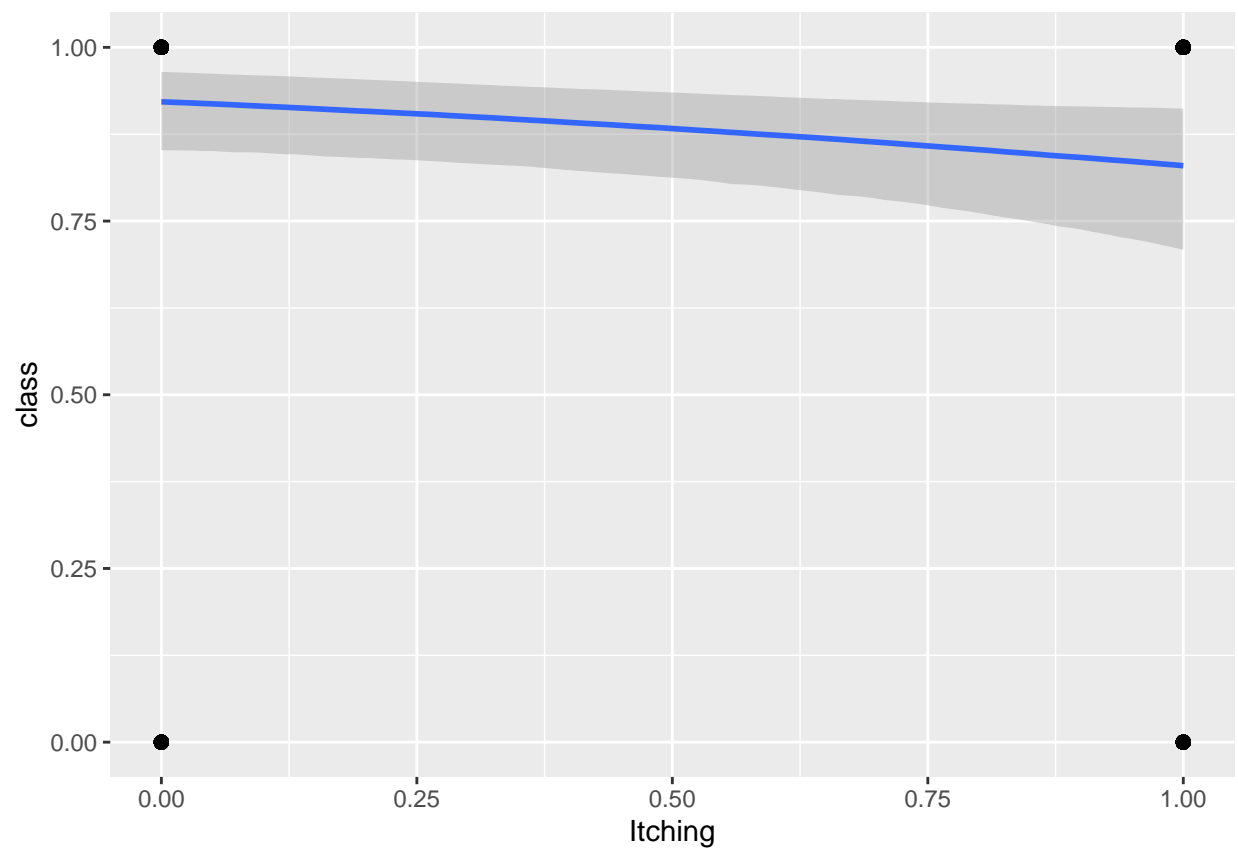


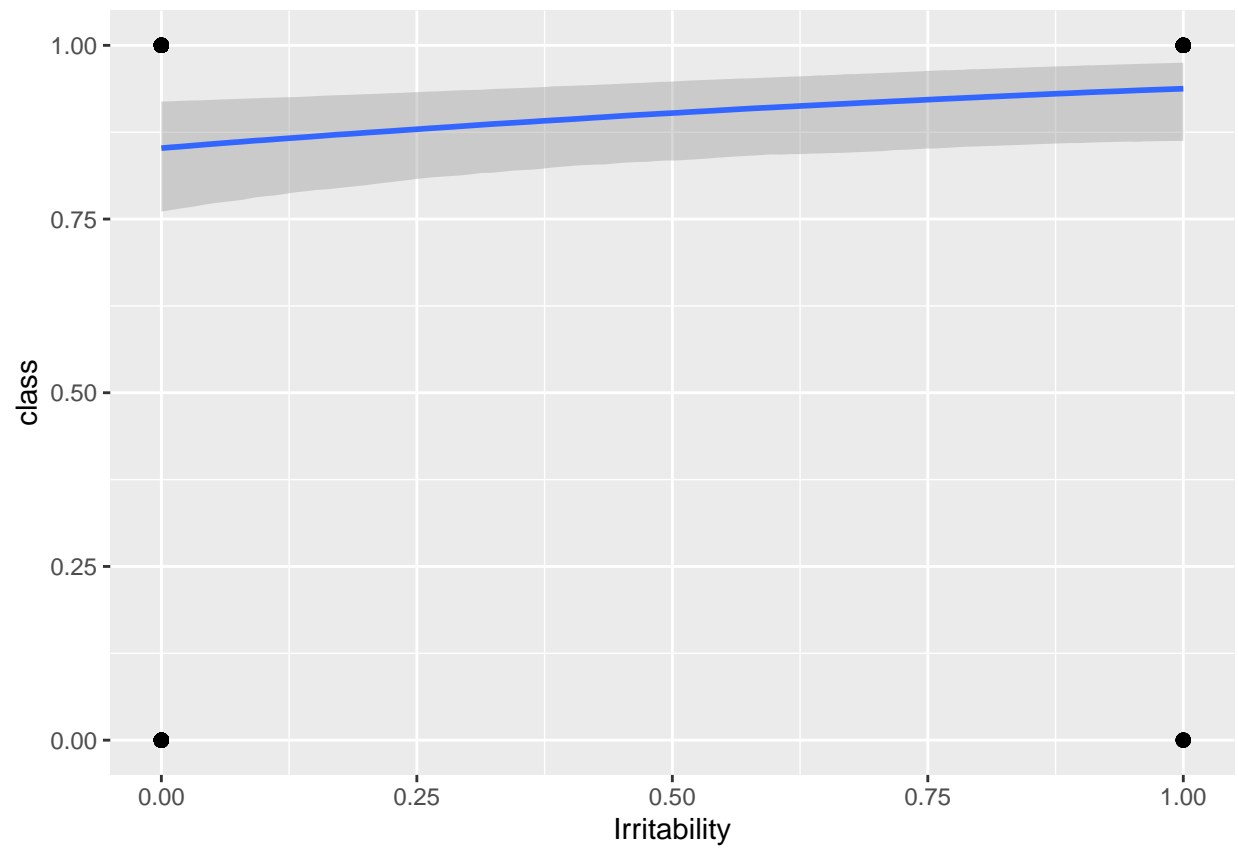


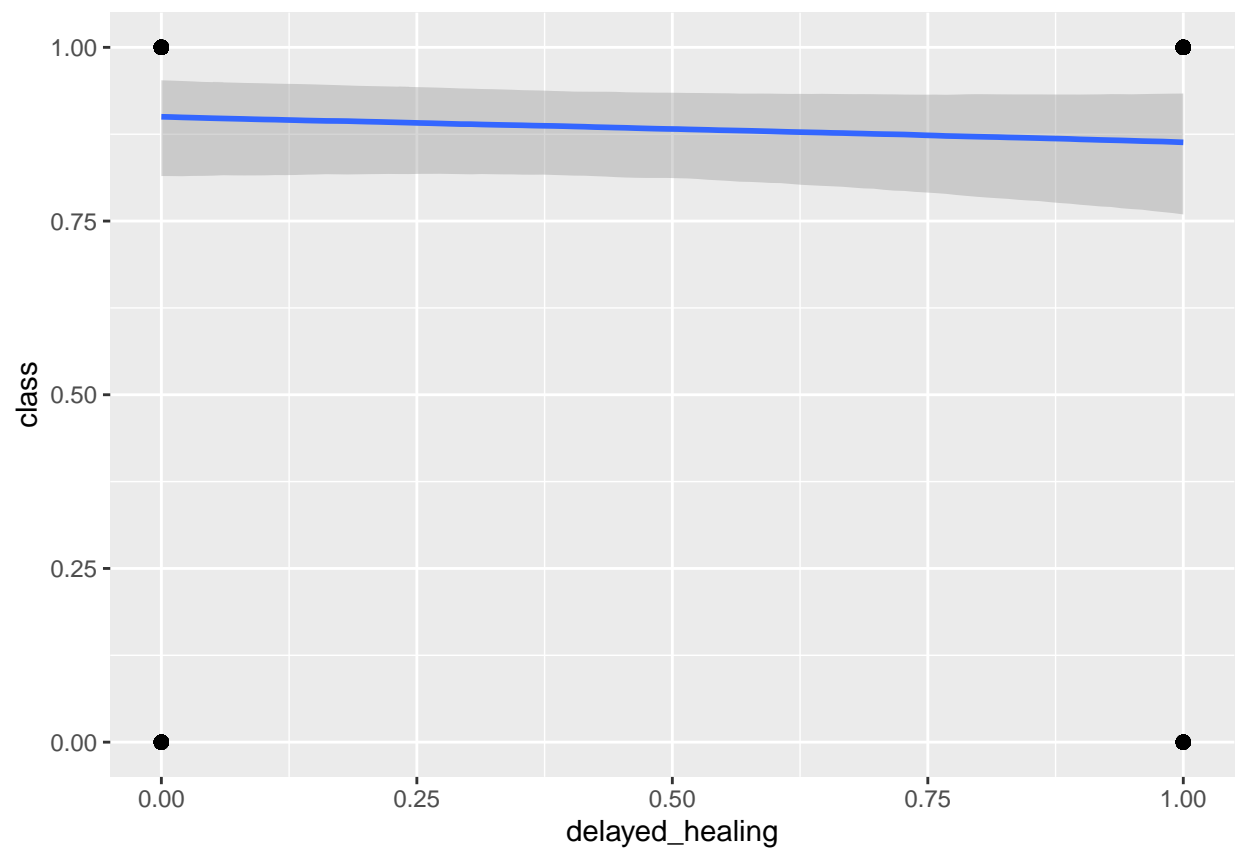


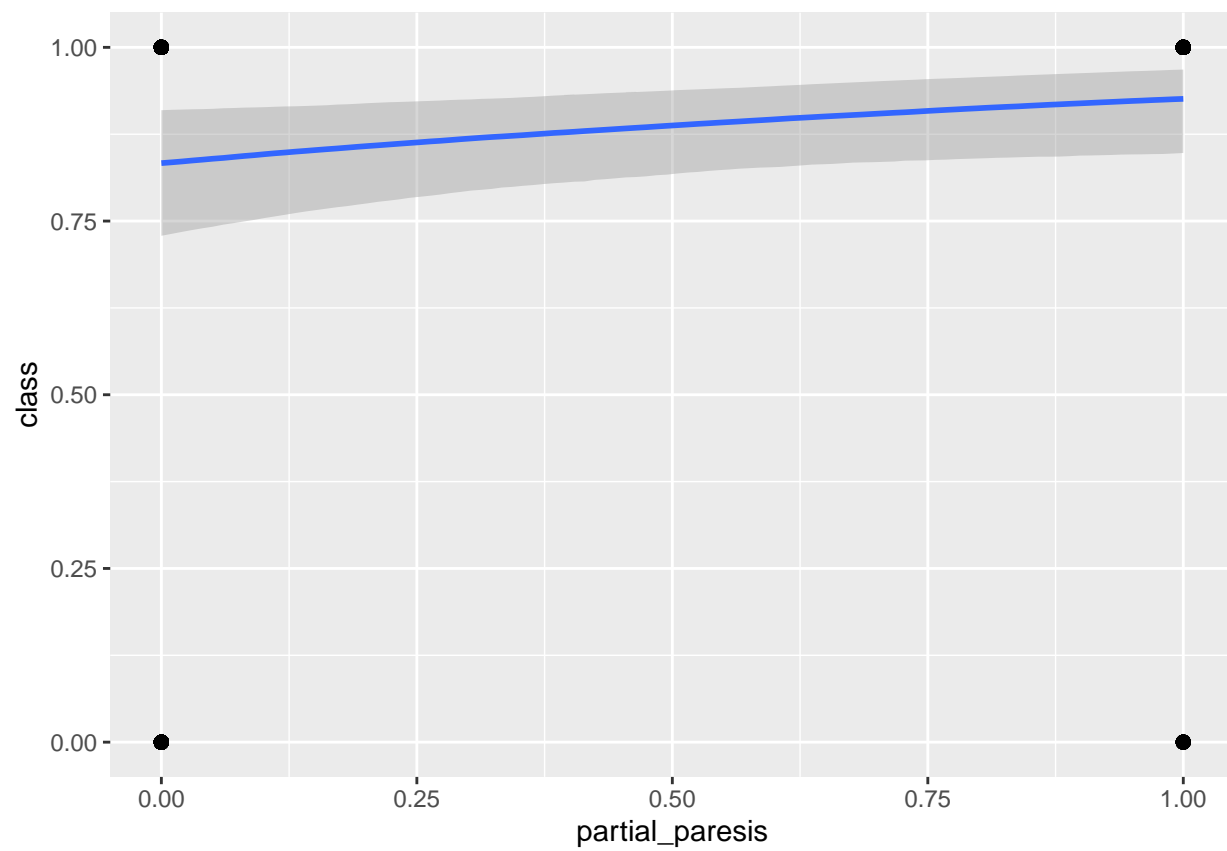




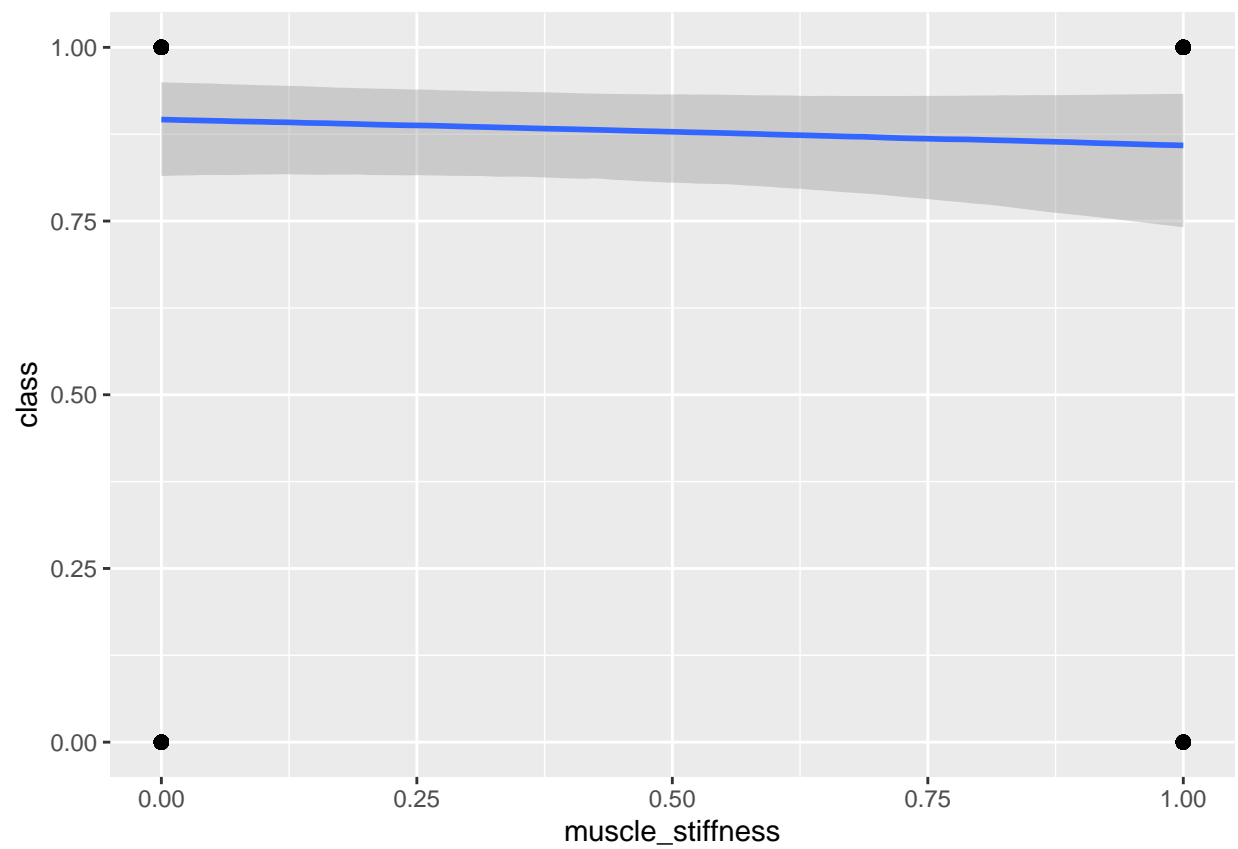


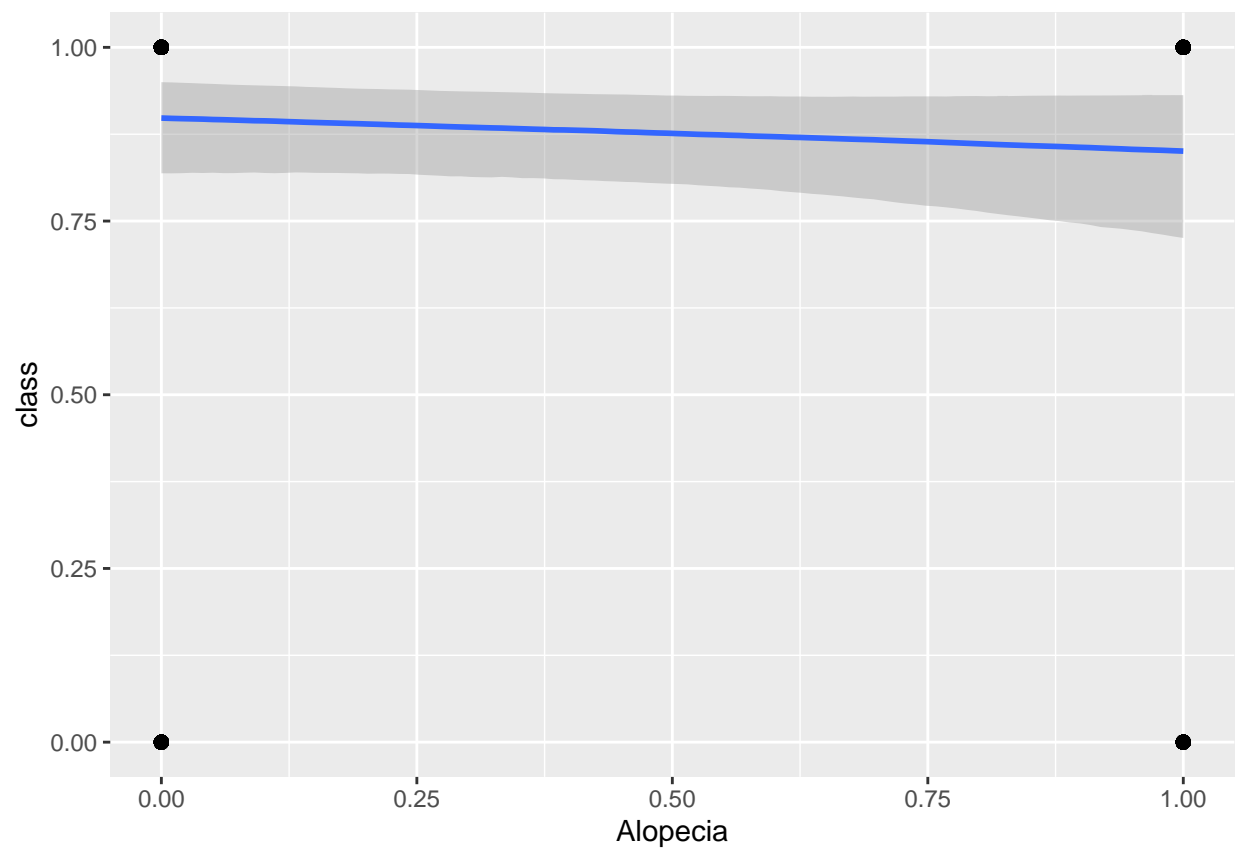


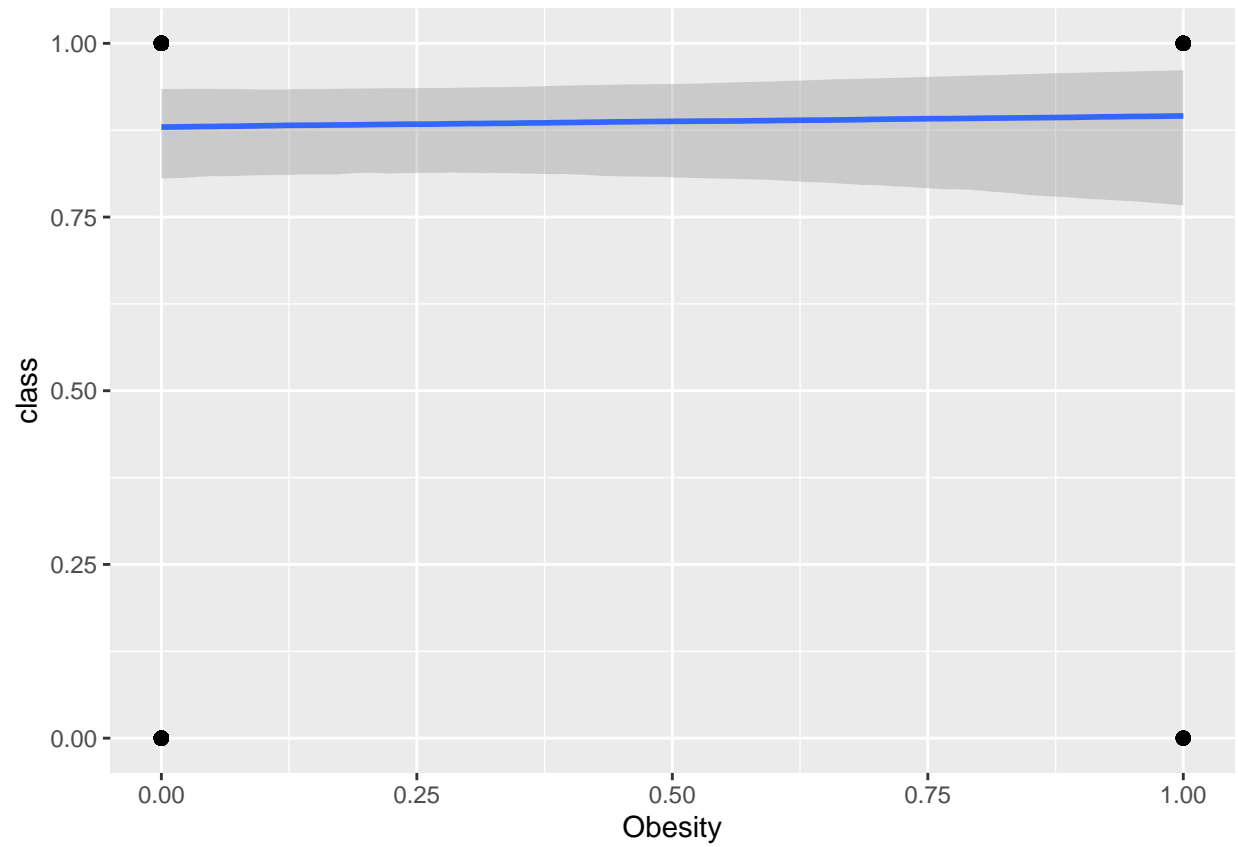




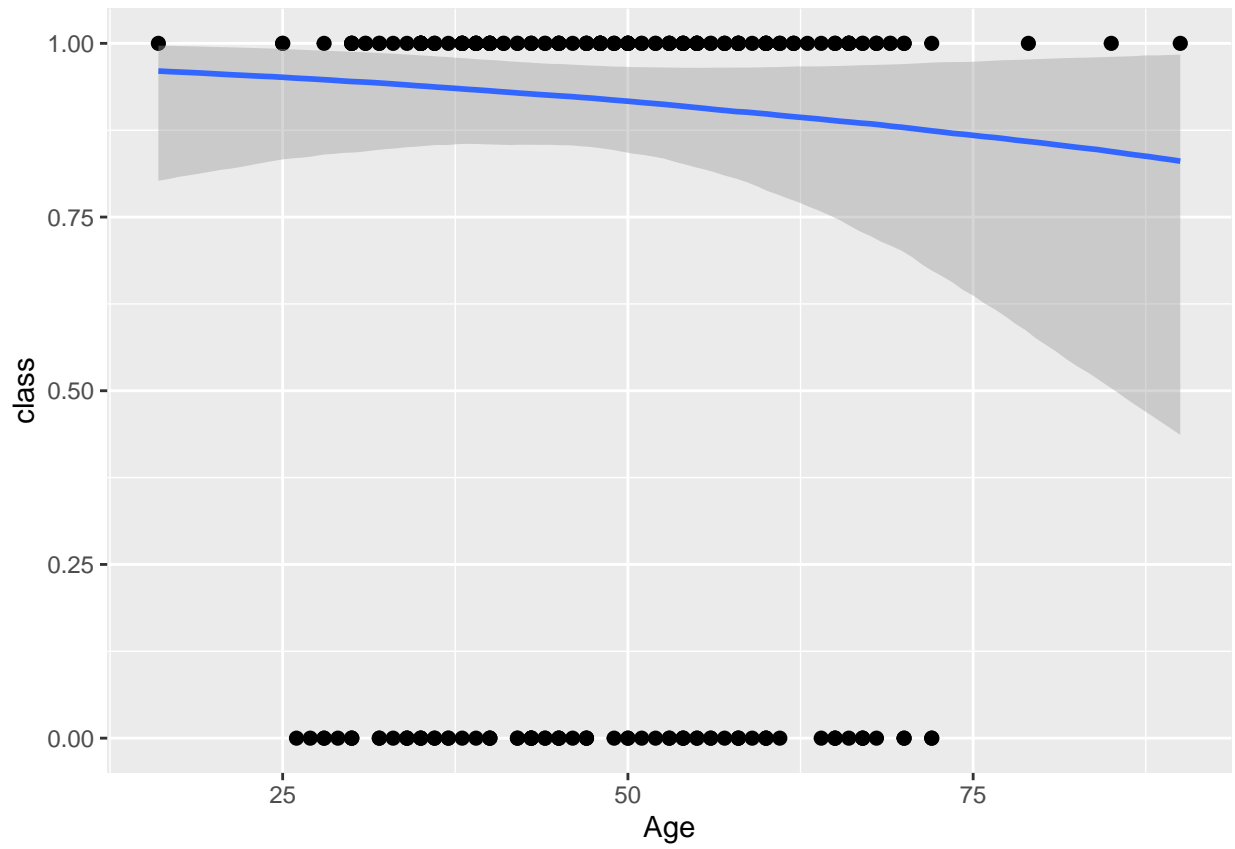


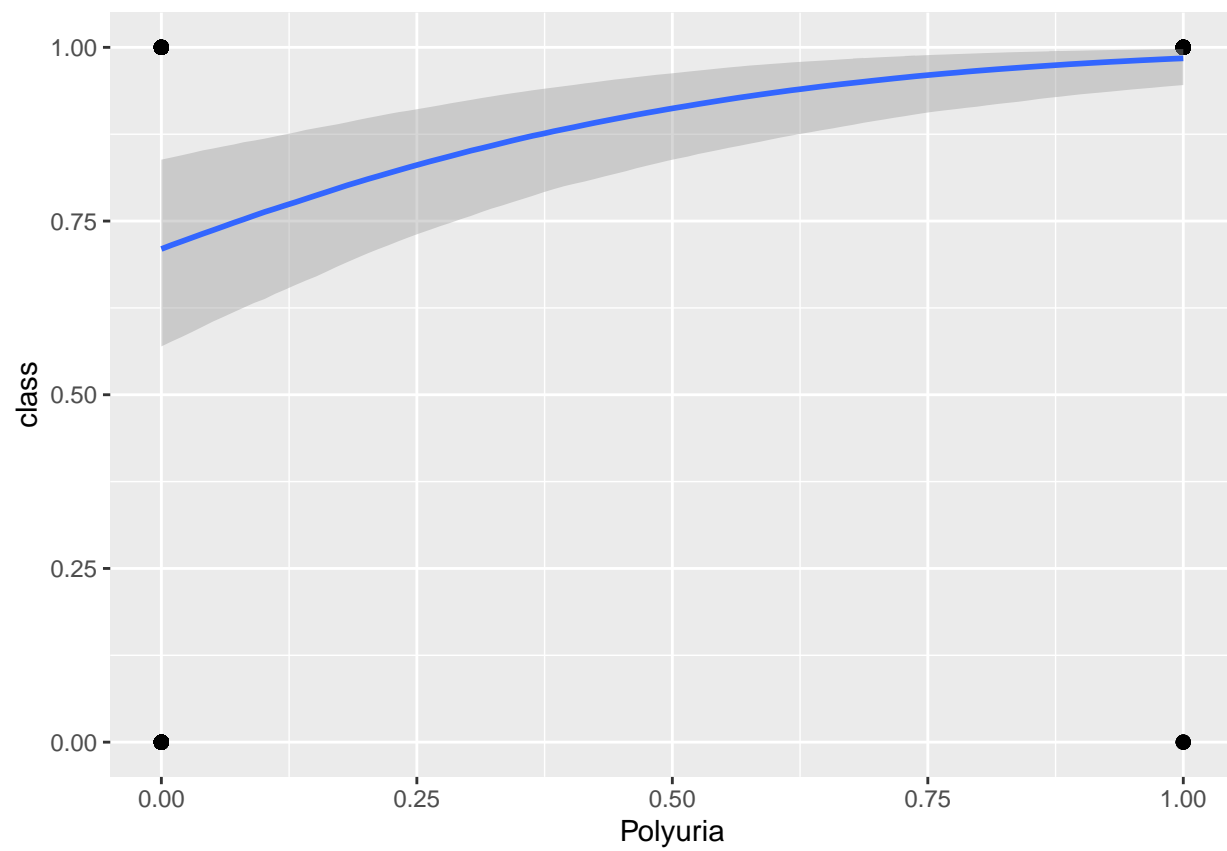


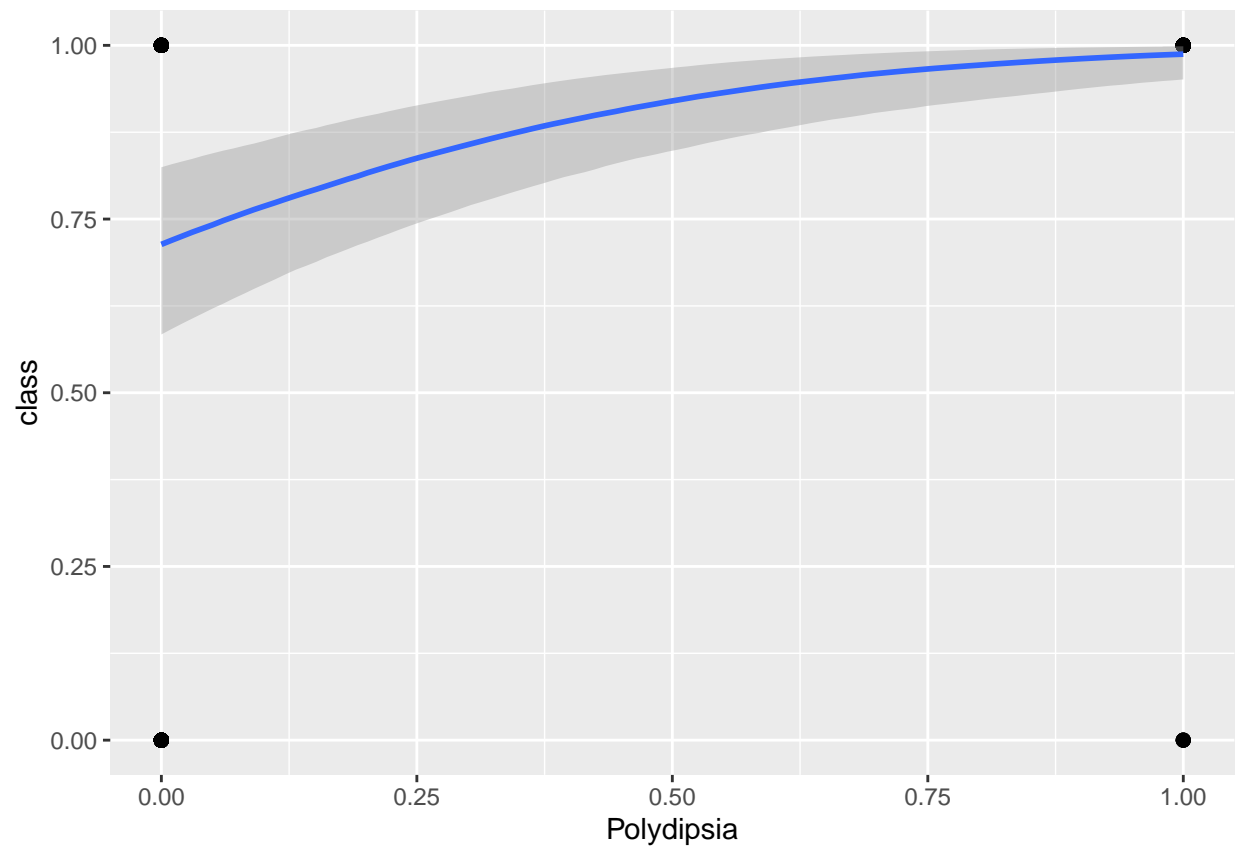


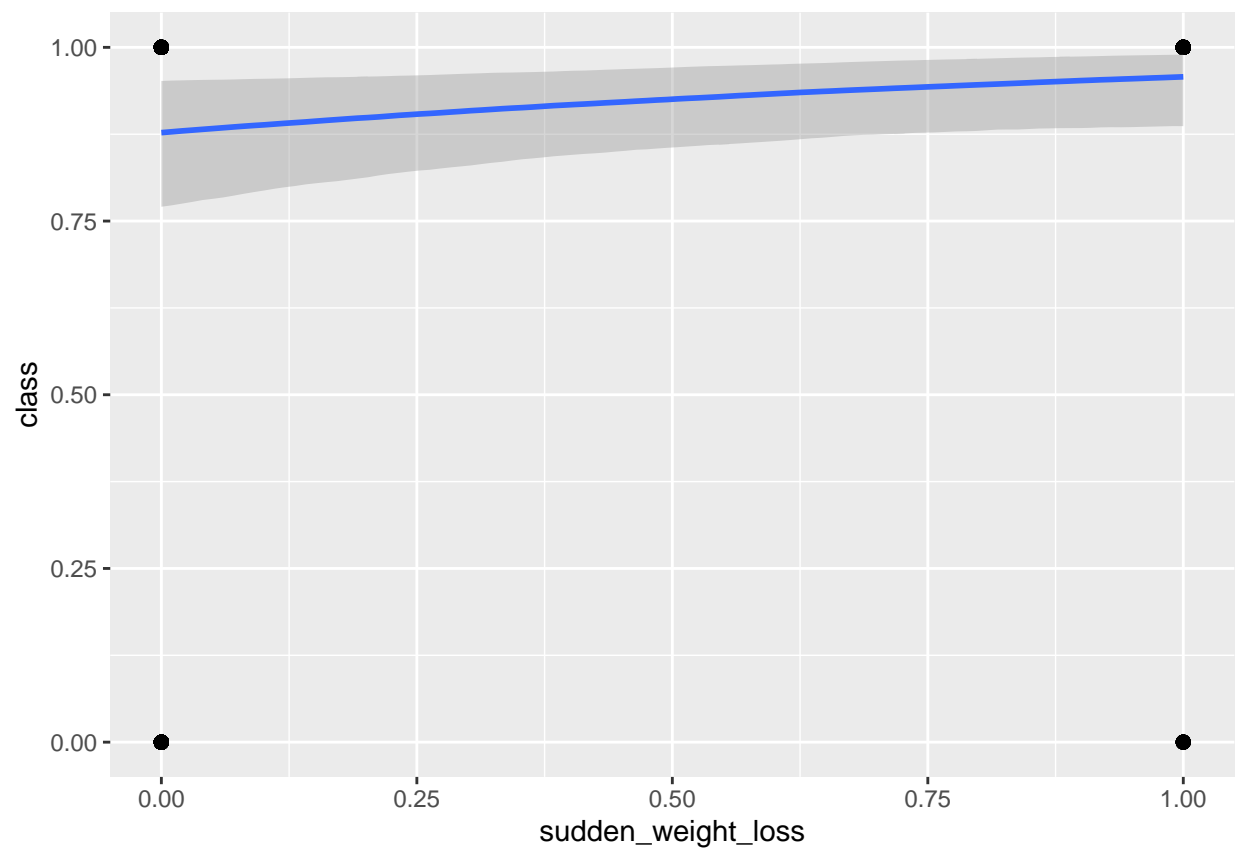


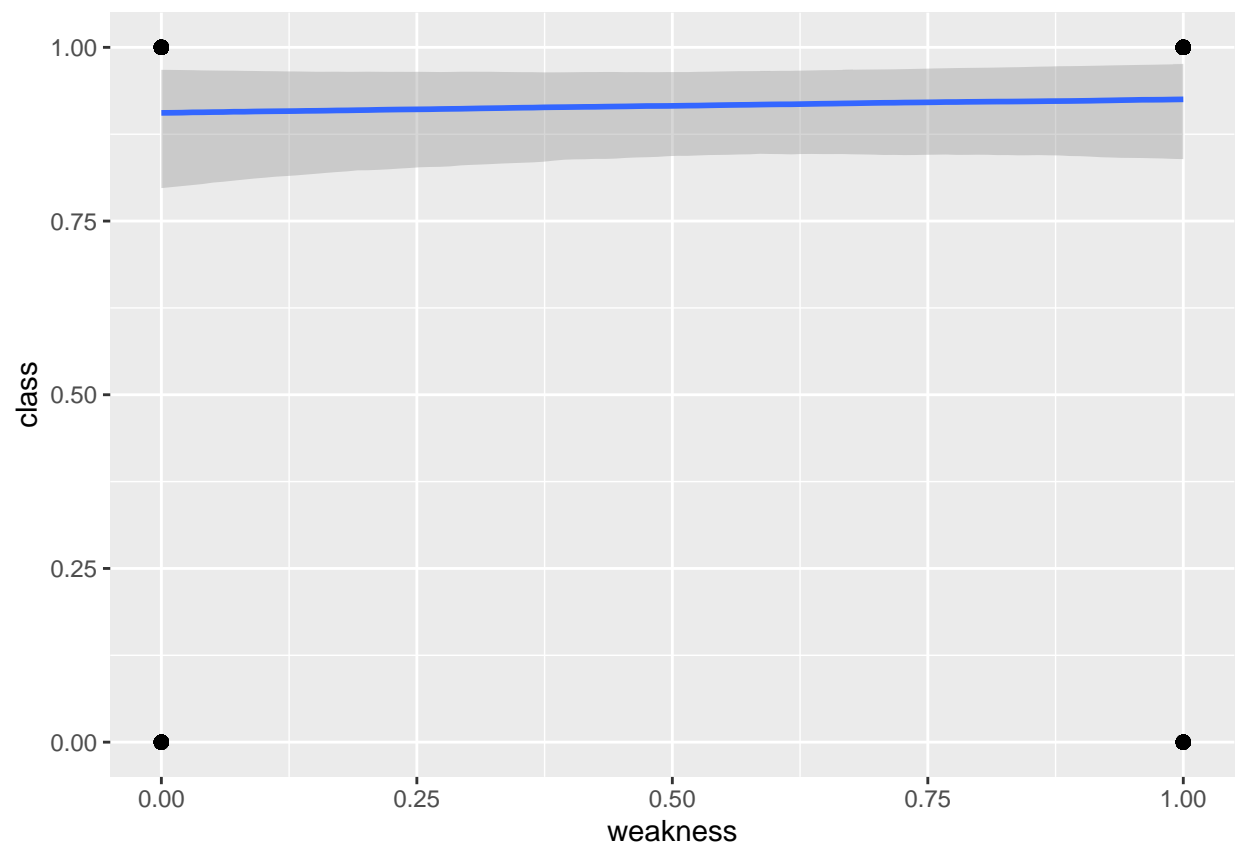
```
# Marginal effects for Probit Model  
plot(conditional_effects(fit_probit), points = TRUE)
```



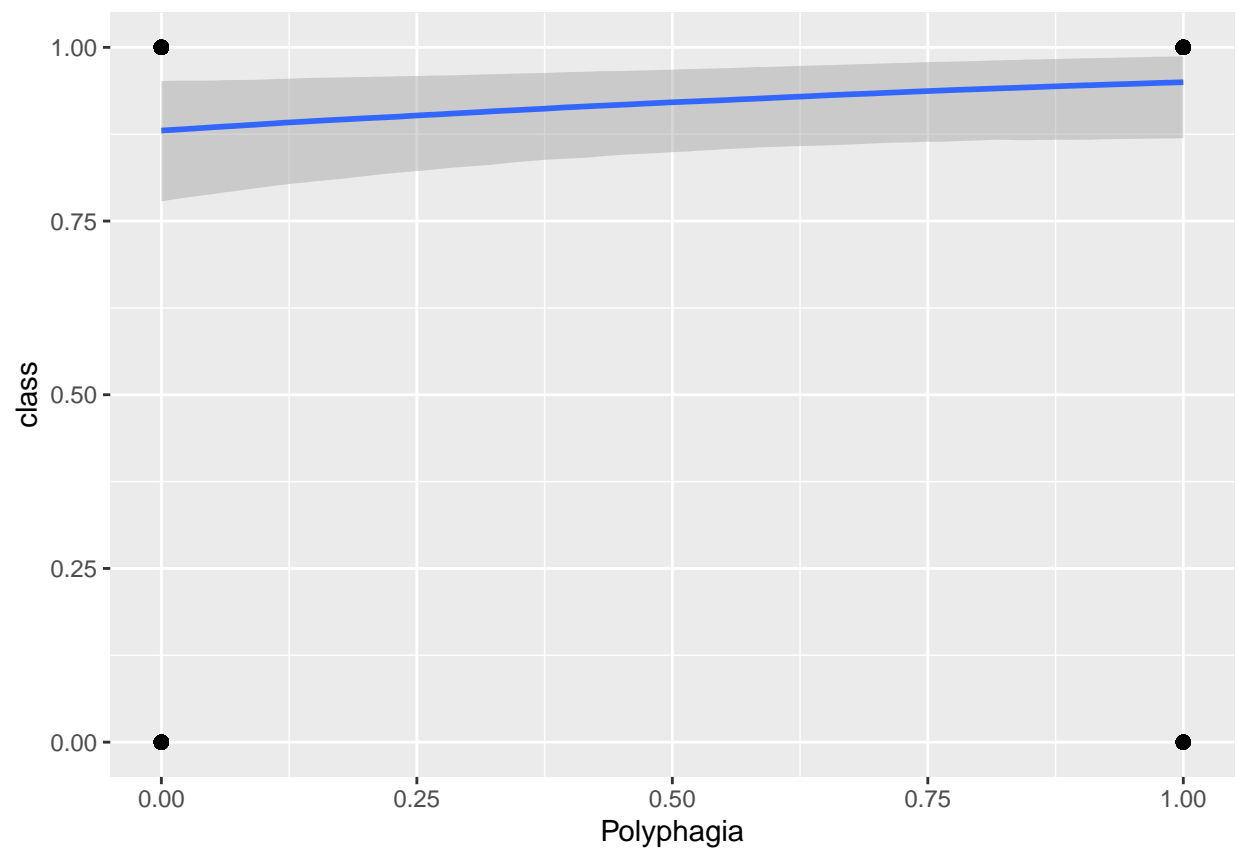


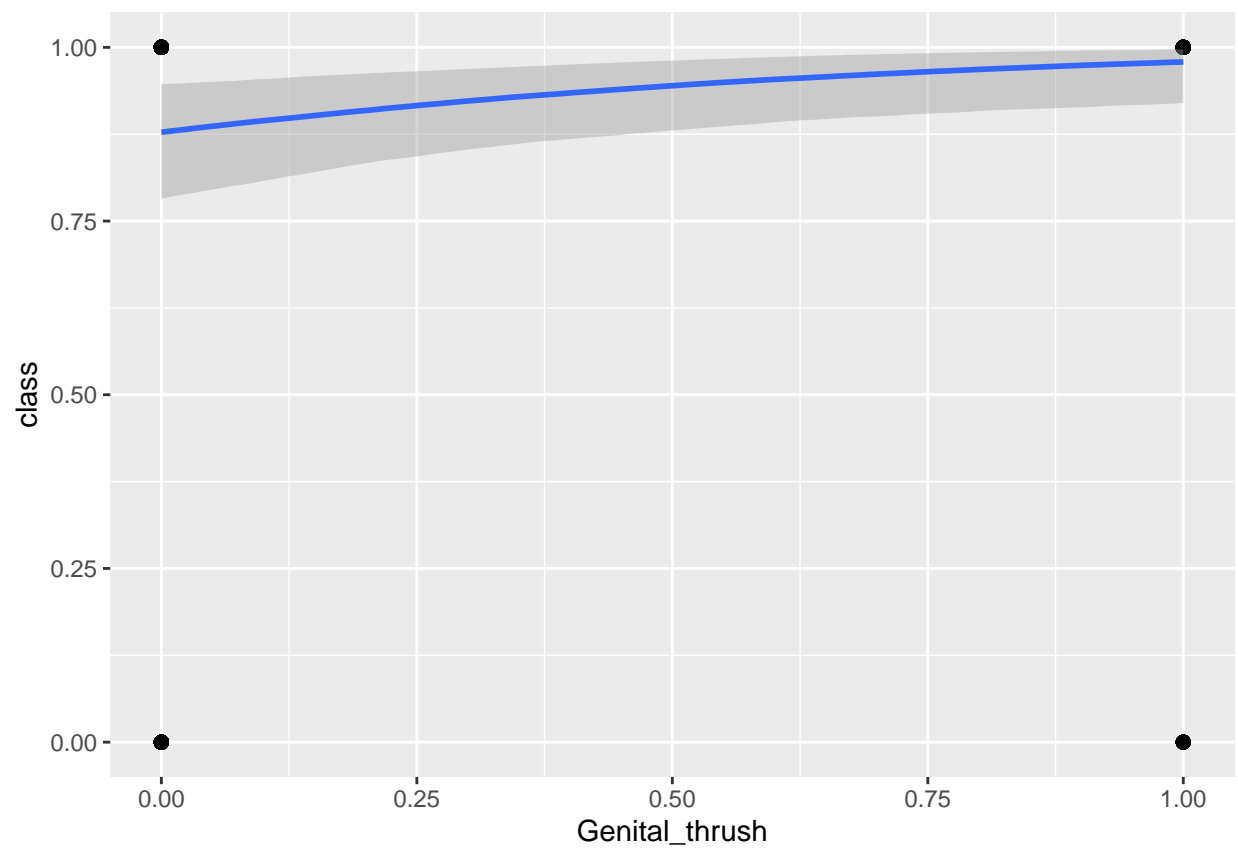


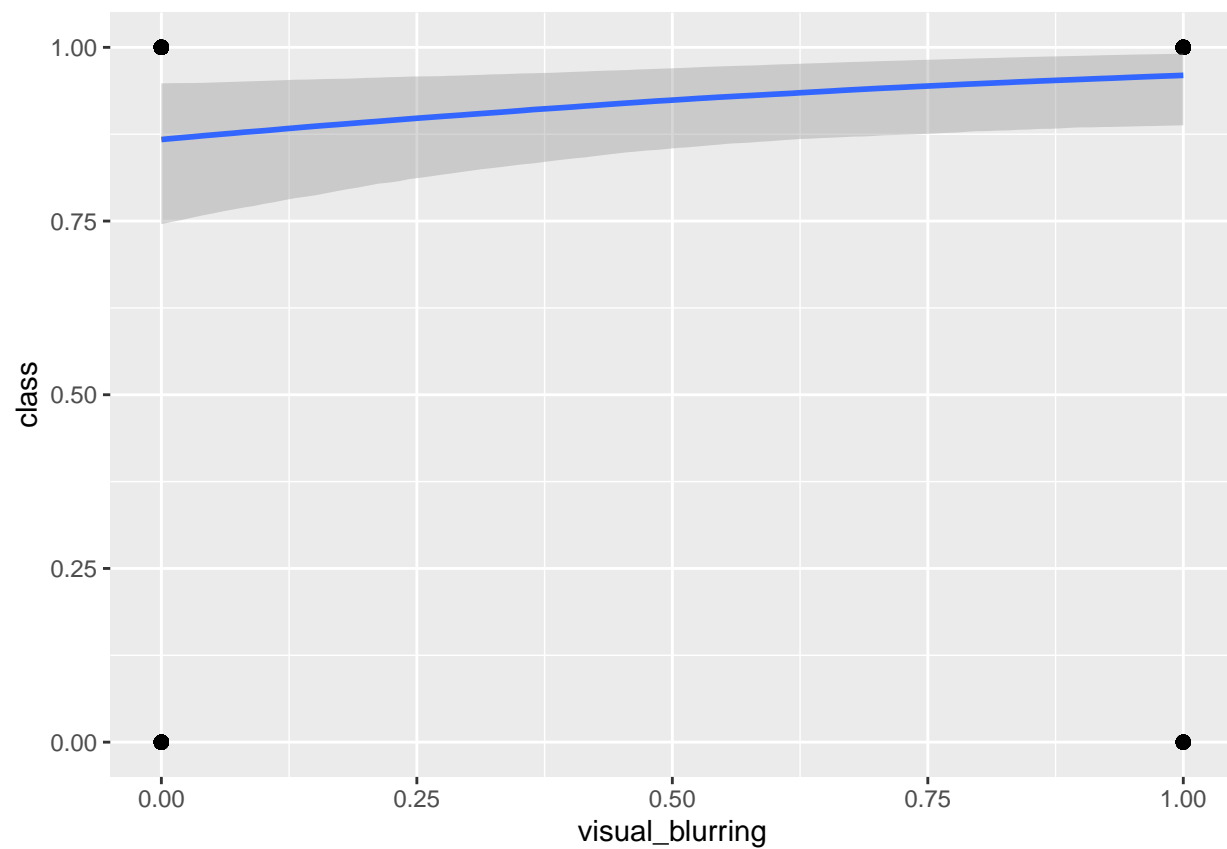


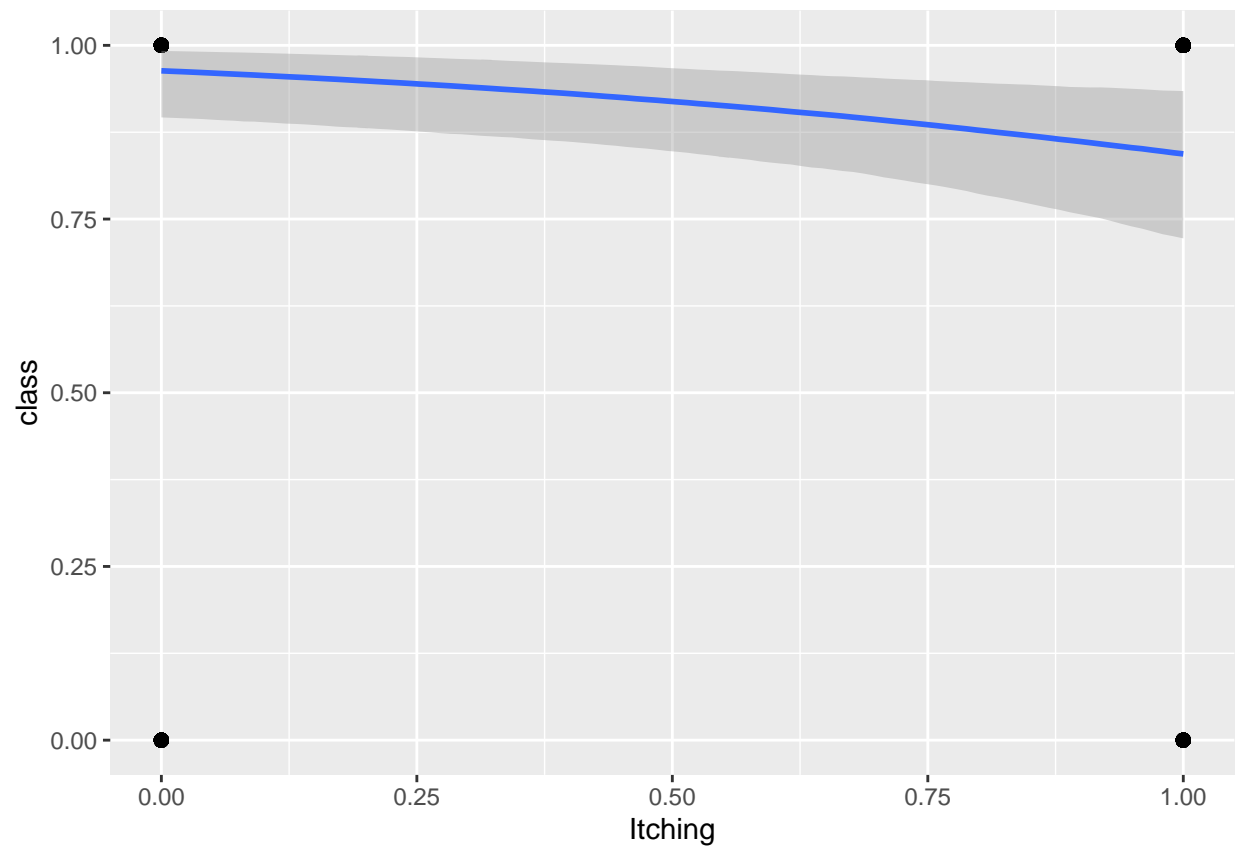


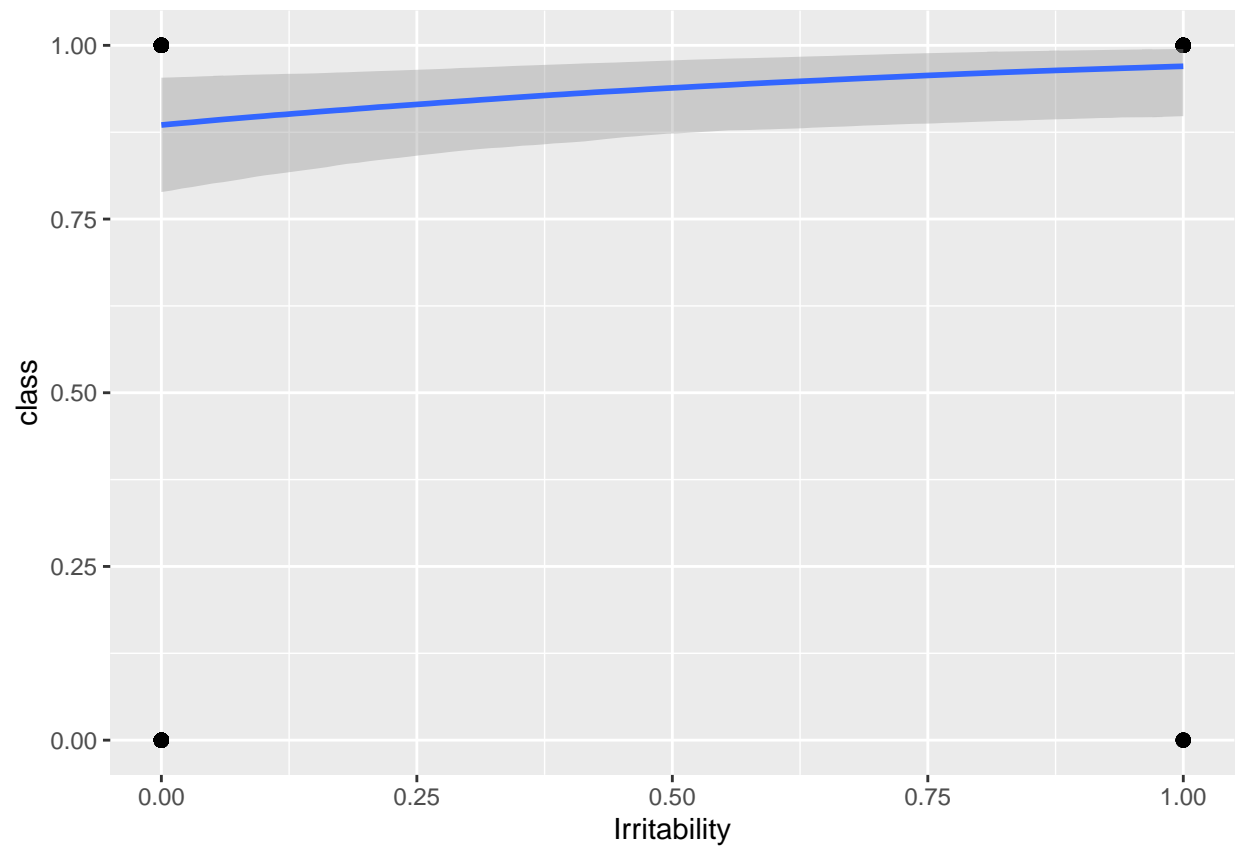


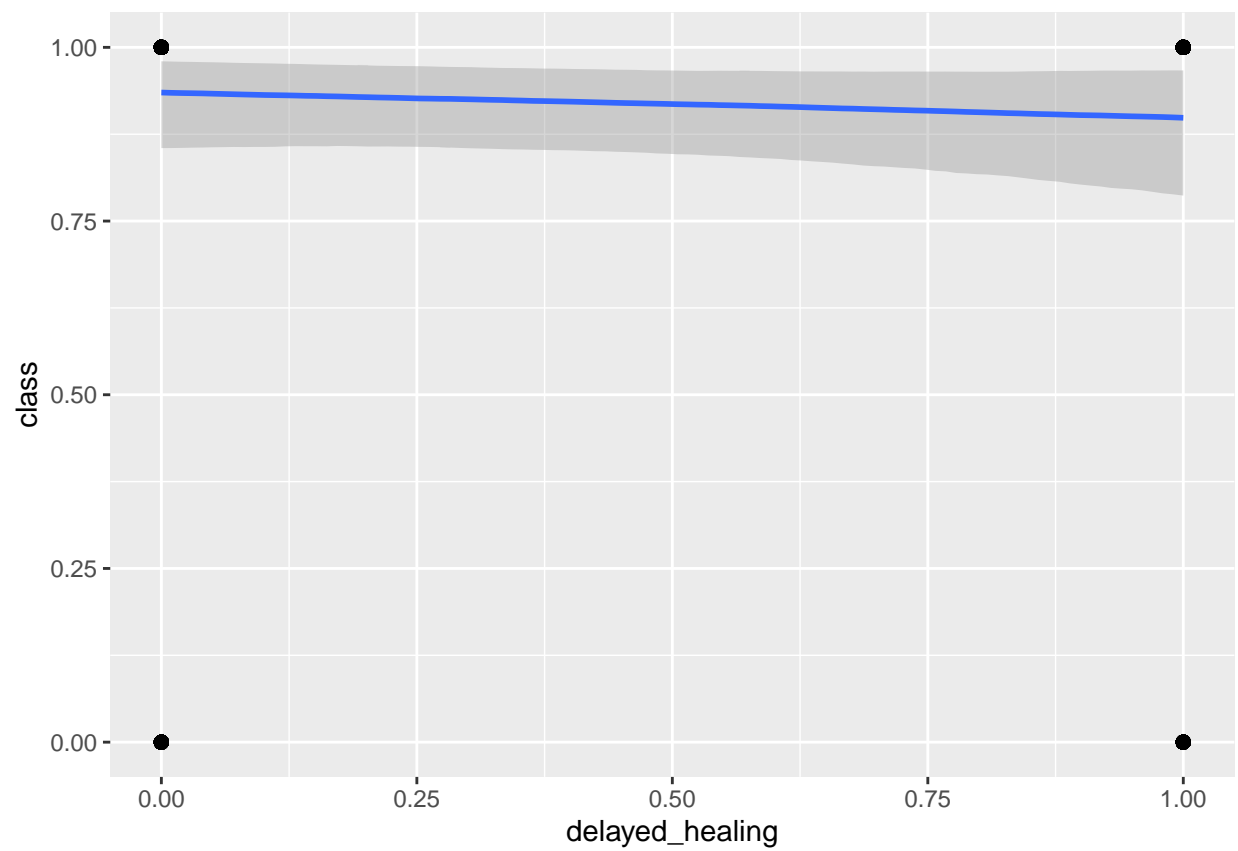


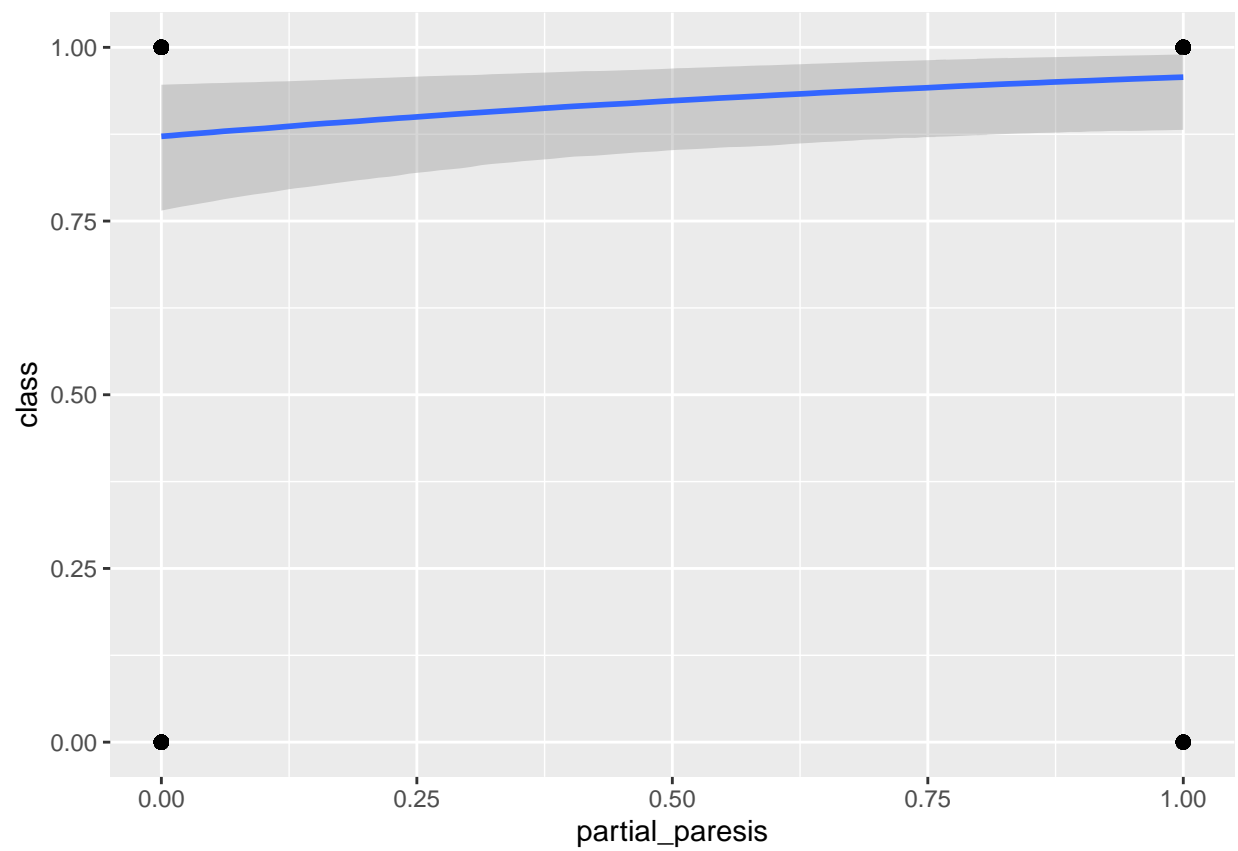


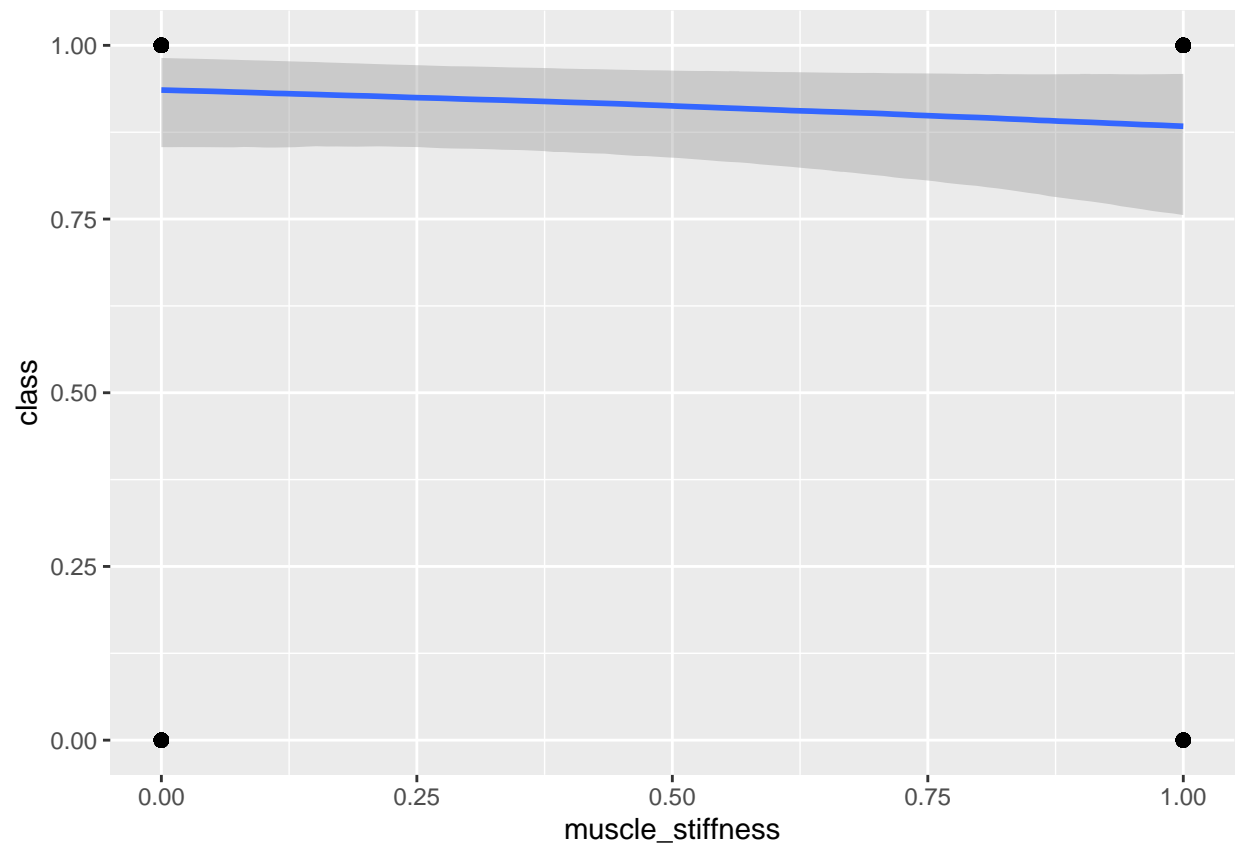




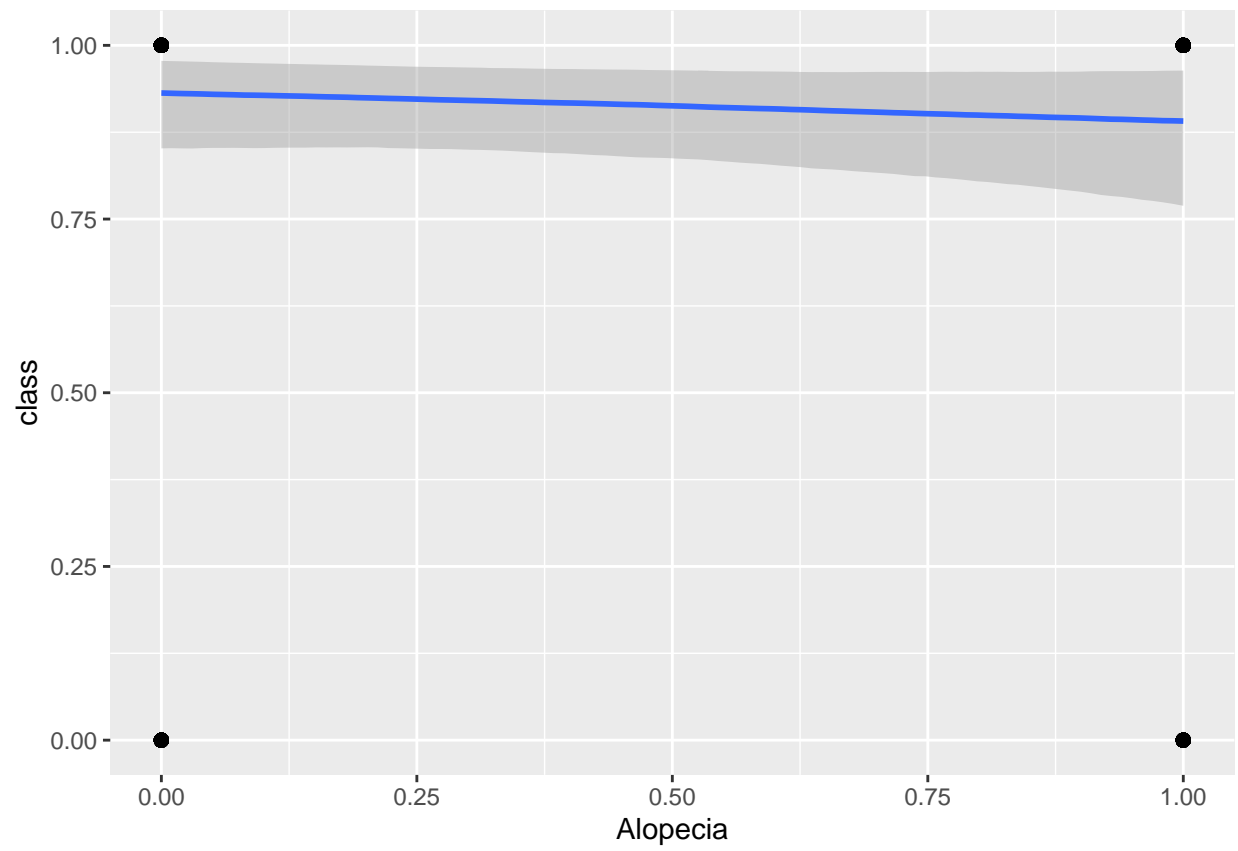


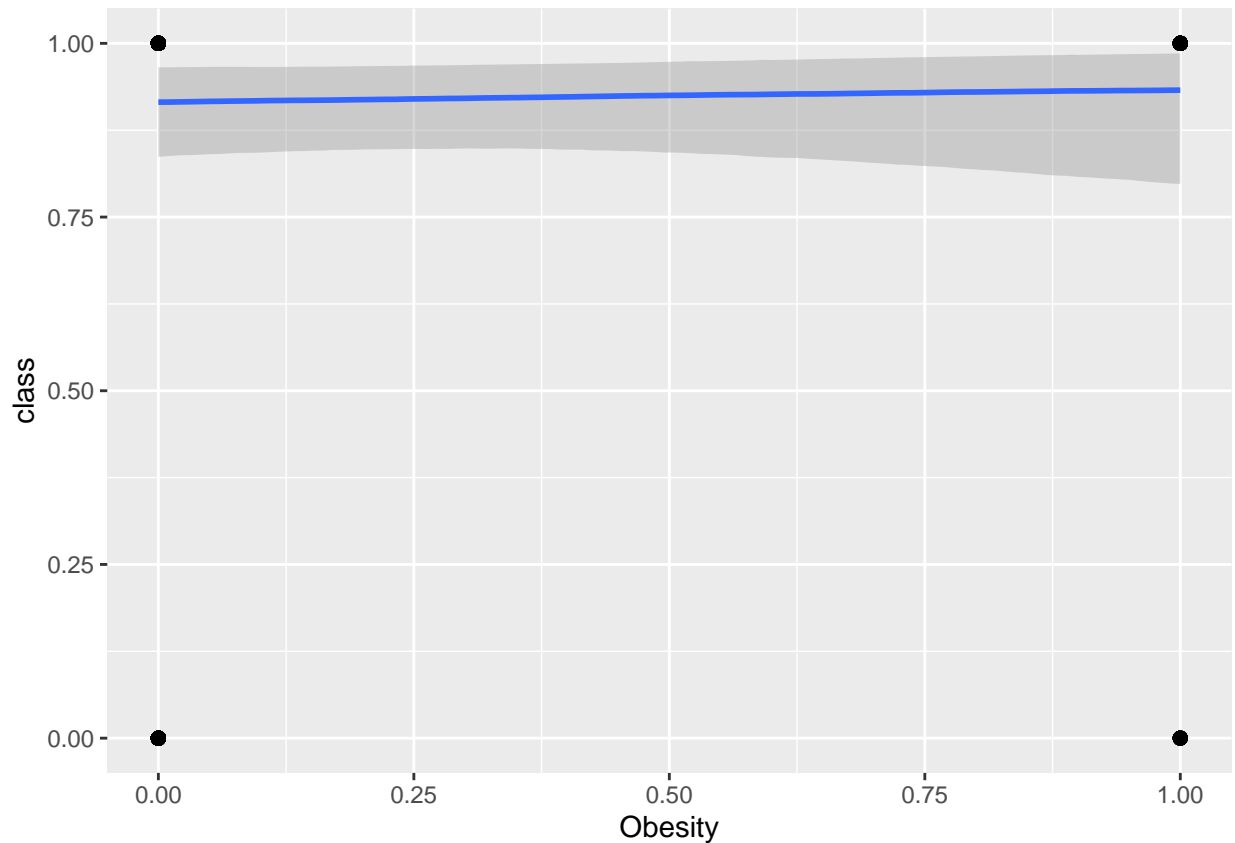










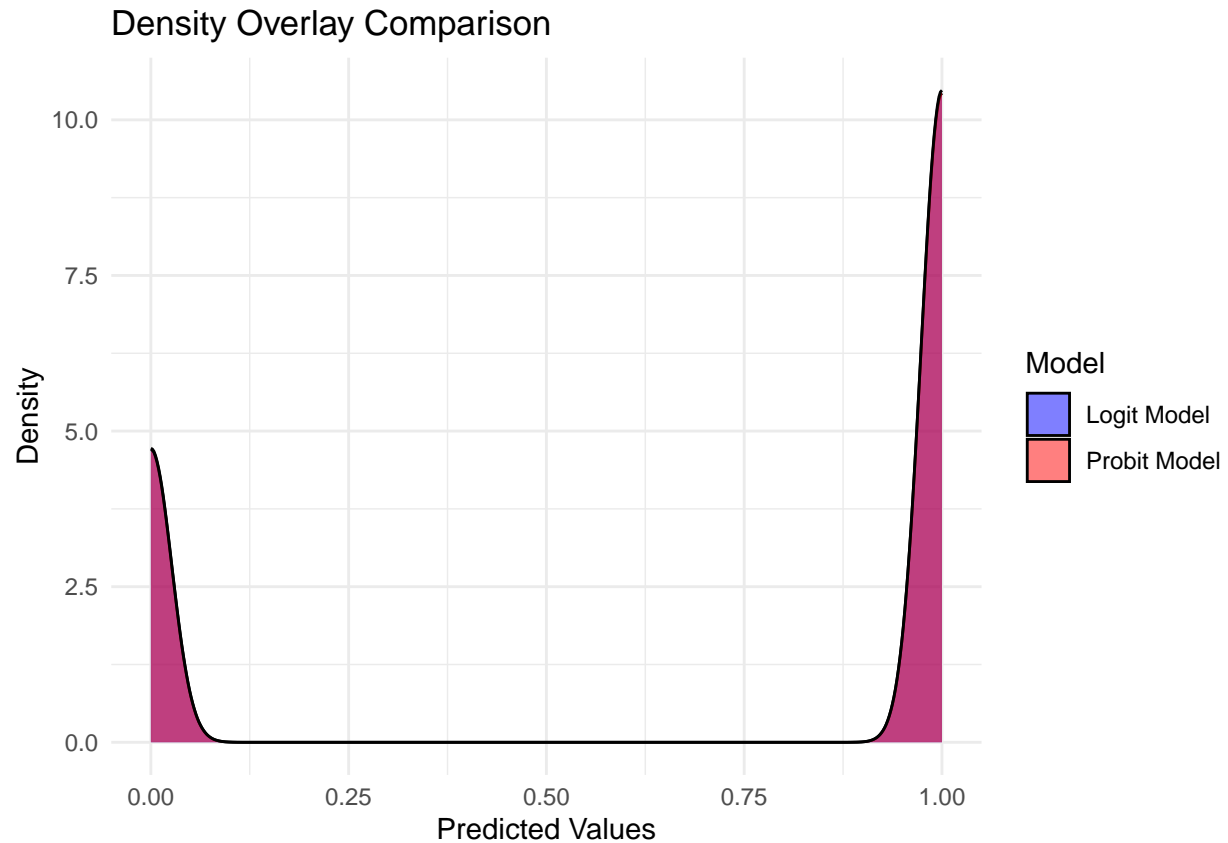


```
# -----
# 1. Generate Posterior Predictions for Both Models
# -----
logit_preds <- posterior_predict(fit_logit, ndraws = 4000) # Posterior samples for Logit
probit_preds <- posterior_predict(fit_probit, ndraws = 4000) # Posterior samples for Probit

# -----
# 2. Convert Predictions into Data Frames for ggplot
# -----
logit_df <- data.frame(Predicted = as.vector(logit_preds), Model = "Logit Model")
probit_df <- data.frame(Predicted = as.vector(probit_preds), Model = "Probit Model")

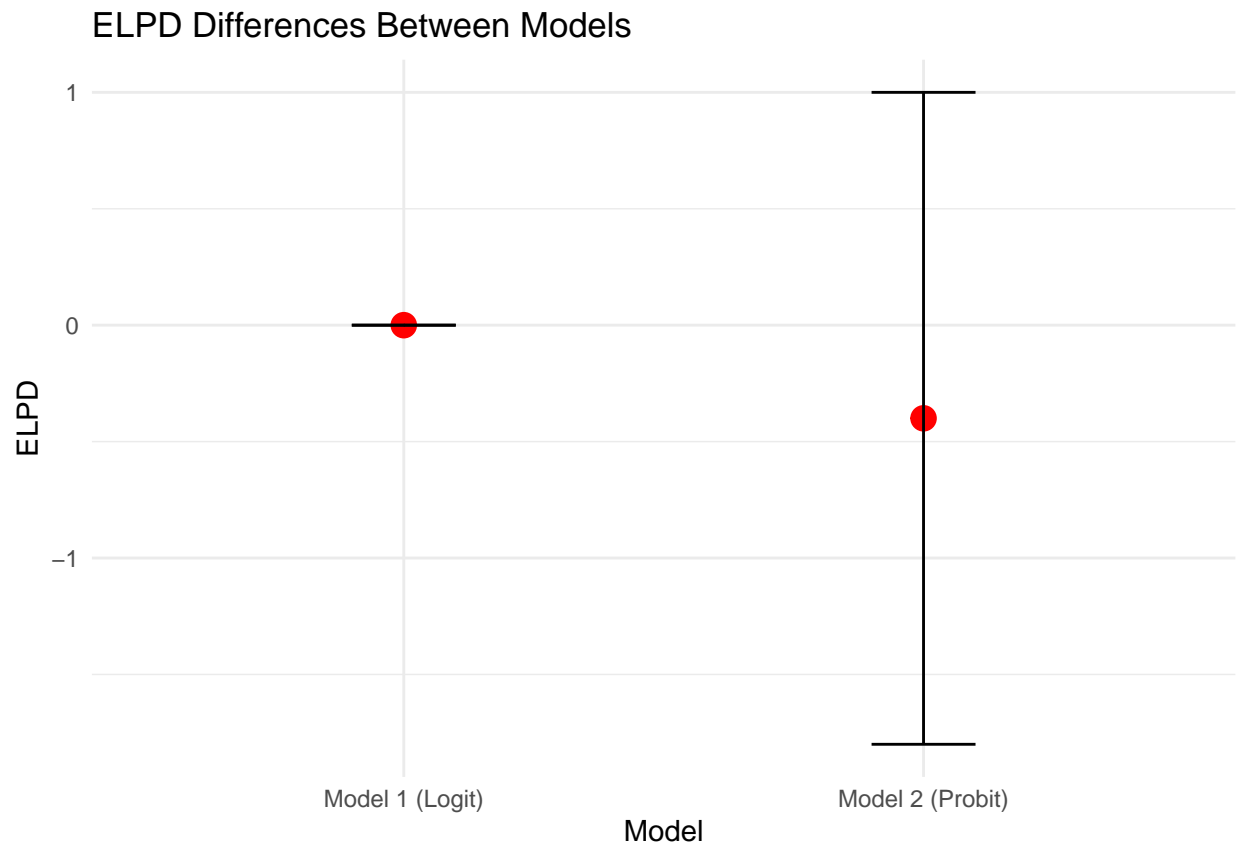
# Combine both predictions into one dataset
combined_preds <- bind_rows(logit_df, probit_df)

# -----
# 3. Plot Density Overlay Comparison
# -----
ggplot(combined_preds, aes(x = Predicted, fill = Model)) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Overlay Comparison", x = "Predicted Values", y = "Density") +
  scale_fill_manual(values = c("blue", "red")) +
  theme_minimal()
```



```
# -----
# 1. Create Data Frame for ELPD Visualization
# -----
comparison_data <- data.frame(
  Model = c("Model 1 (Logit)", "Model 2 (Probit)"), # Model names
  ELPD = c(0, -0.4), # elpd_diff values from your output
  SE = c(0.0, 1.4) # Standard error from your output
)

# -----
# 2. Plot ELPD Differences with Error Bars
# -----
ggplot(comparison_data, aes(x = Model, y = ELPD)) +
  geom_point(size = 4, color = "red") + # Blue points for ELPD
  geom_errorbar(aes(ymin = ELPD - SE, ymax = ELPD + SE), width = 0.2) + # Error bars
  ggtitle("ELPD Differences Between Models") +
  theme_minimal()
```



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.