PREDICTING THE SURVIVAL OF PASSENGERS IN THE TITANIC ACCIDENT

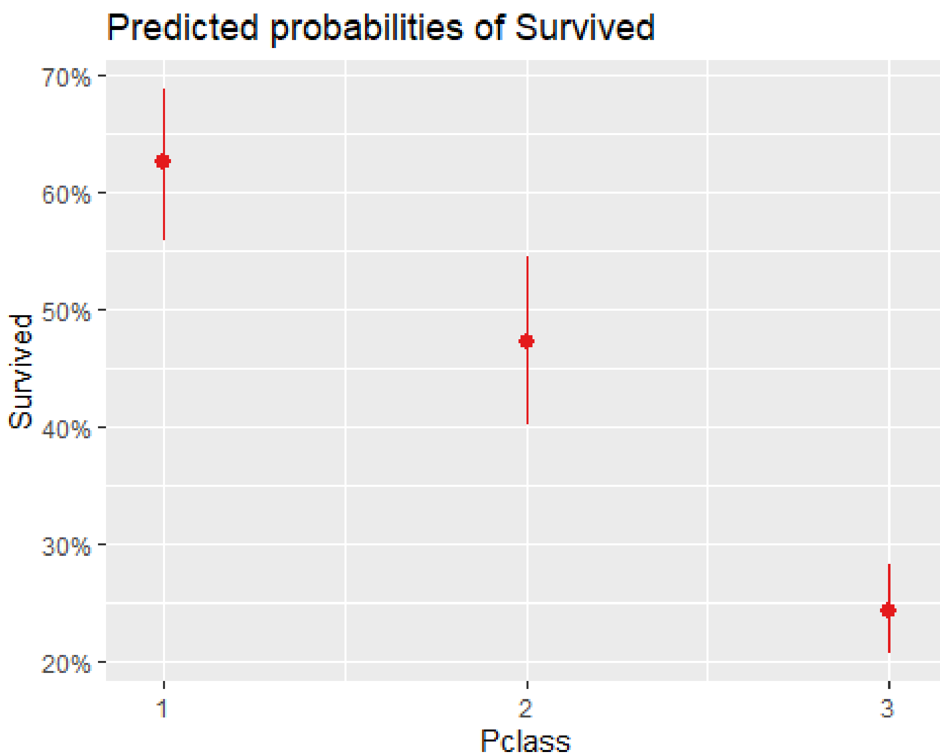
Enock Bereka

2024-11-24

```
Univariable logistic regression
Load the necessary library and the dataset
library(tidyverse)

titanic <- read_csv("C:/Users/PC/OneDrive/Desktop/Data Science/Datasets/titan
ic.csv")

titanic$Sex <- as.factor(titanic$Sex)
titanic$Sex <- as.numeric(titanic$Sex)
titanic$Pclass <- as.factor(titanic$Pclass)

Cross table for quick intuition
table(titanic$Survived, titanic$Pclass)


       1   2   3
   0  80  97 372
   1 134  87 119
```

- More people from the first class survived
- Many more people from the third class perished

```
Run logistic regression with categorical predictors
m1 <- glm(Survived~Pclass, titanic, family = binomial)

Plot predictions
library(sjPlot)

plot_model(m1, type = "eff", terms = c("Pclass"))
```



- The probability of survival in class one is more than 60%
- The probability of survival in the third class is less than 25%
- The probability of survival in class two is less than 45 %

Get probabilities and odds ratios

```
library(emmeans)

emmeans(m1, pairwise~Pclass, type = "response",
        infer = T)

 $emmeans
  Pclass  prob     SE  df asymp.LCL asymp.UCL null z.ratio p.value
  1      0.626 0.0331 Inf     0.559     0.688  0.5   3.651  0.0003
  2      0.473 0.0368 Inf     0.402     0.545  0.5  -0.737  0.4612
  3      0.242 0.0193 Inf     0.206     0.282  0.5 -10.822  <.0001

 Confidence level used: 0.95
 Intervals are back-transformed from the logit scale
 Tests are performed on the logit scale

 $contrasts
  contrast          odds.ratio    SE  df asymp.LCL asymp.UCL null z.ratio
  Pclass1 / Pclass2       1.87 0.382 Inf      1.16      3.01    1   3.056
  Pclass1 / Pclass3       5.24 0.923 Inf      3.46      7.91    1   9.395
  Pclass2 / Pclass3       2.80 0.509 Inf      1.83      4.29    1   5.684
```

```
 p.value
  0.0063
  <.0001
  <.0001

Confidence level used: 0.95
Conf-level adjustment: tukey method for comparing a family of 3 estimates
Intervals are back-transformed from the log odds ratio scale
P value adjustment: tukey method for comparing a family of 3 estimates
Tests are performed on the log odds ratio scale
```

- First class passengers are significantly more likely to survive than dying
- The probability of survival in third class is significantly lower than that of dying

Check

```
titanic2 <- titanic %>% slice(1:400)
m2 <- glm(Survived~Pclass, titanic2, family = binomial)
emmeans(m2, ~Pclass, infer = T, type = "response")

 Pclass  prob     SE  df asymp.LCL asymp.UCL null z.ratio p.value
 1      0.576 0.0515 Inf     0.473     0.673  0.5   1.454  0.1460
 2      0.434 0.0544 Inf     0.332     0.542  0.5  -1.204  0.2287
 3      0.302 0.0306 Inf     0.246     0.365  0.5  -5.764  <.0001

Confidence level used: 0.95
Intervals are back-transformed from the logit scale
Tests are performed on the logit scale
```

Reverse odds ratios if needed

```
emmeans(m1, ~Pclass, type = "response") %>%
  pairs(reverse = T, infer = T)

 contrast          odds.ratio     SE  df asymp.LCL asymp.UCL null z.ratio
 Pclass2 / Pclass1      0.535 0.1090 Inf     0.332     0.864    1  -3.056
 Pclass3 / Pclass1      0.191 0.0337 Inf     0.126     0.289    1  -9.395
 Pclass3 / Pclass2      0.357 0.0647 Inf     0.233     0.546    1  -5.684
 p.value
  0.0063
  <.0001
  <.0001

Confidence level used: 0.95
Conf-level adjustment: tukey method for comparing a family of 3 estimates
Intervals are back-transformed from the log odds ratio scale
P value adjustment: tukey method for comparing a family of 3 estimates
Tests are performed on the log odds ratio scale
```

```
• Reverse = T gives us odds ratios below one

• Passengers in 2nd class were 0.535 times as likely to
  survive compared to passengers in 1st class

• Passengers in 3rd class were 0.191 times as likely to
  survive compared to passengers in 1st class

• Passengers in 3rd class were 0.357 times as likely to
  survive compared to passengers in 2nd class
```

```
Get publication ready table
library(gtsummary)

fancy_table <- tbl_regression(
  m1, exponentiate = T, add_pairwise_contrasts = T) %>%
  add_significance_stars(
    hide_p = F, hide_se = T, hide_ci = F) %>%
  bold_p()

fancy_table
```

Characteristic	OR[1,2]	95% CI[2]	p-value
Pclass			
Pclass2 / Pclass1	0.54**	0.33, 0.86	**0.006**
Pclass3 / Pclass1	0.19***	0.13, 0.29	**<0.001**
Pclass3 / Pclass2	0.36***	0.23, 0.55	**<0.001**

[1] *p<0.05; **p<0.01; ***p<0.001

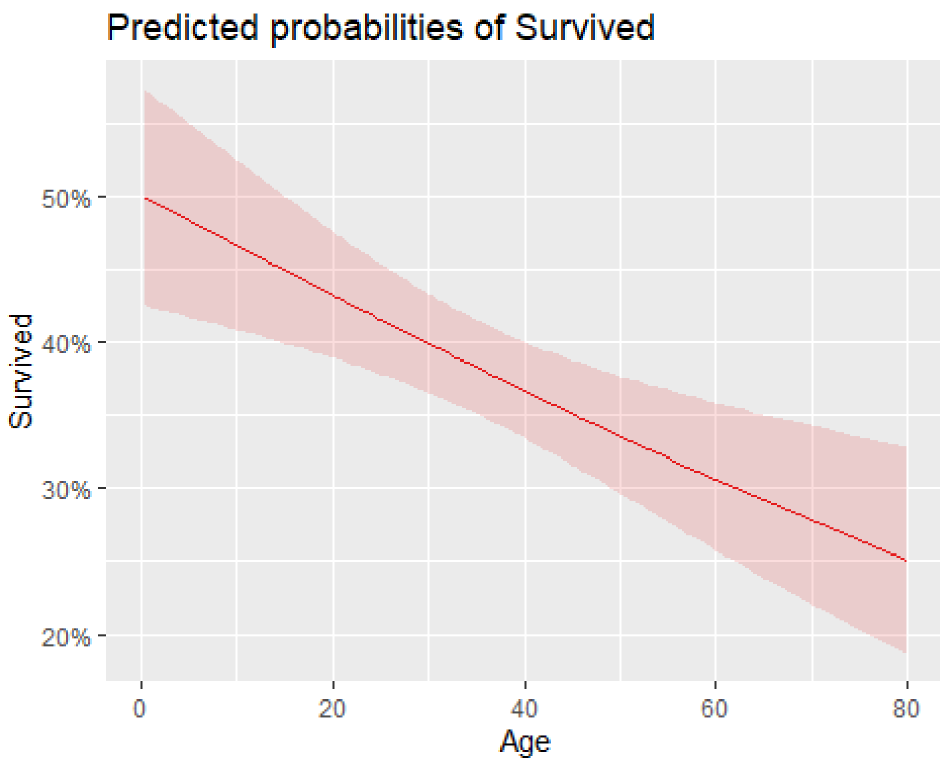
```
How to produce model equations
library(equatiomatic)

extract_eq(m1)
```

$$\log\left[\frac{P(\text{Survived}=1)}{1-P(\text{Survived}=1)}\right]=\alpha+\beta_1(\text{Pclass}_2)+\beta_2(\text{Pclass}_3)$$

```
How to get odds for categories
emmeans(m1, ~Pclass, infer = T) %>%
  as_tibble() %>%
  dplyr::select(Pclass, emmean) %>%
  mutate(odds = exp(emmean)) %>%
  mutate_if(is.numeric, ~round(., 2))

   # A tibble: 3 × 3

   Pclass emmean  odds
   <fct>   <dbl> <dbl>
 1 1        0.52  1.67
 2 2       -0.11  0.9
 3 3       -1.14  0.32
```

- The odds of survival for 1st class passengers is 67% higher than those of dying
- 2nd class passengers have 10% lower odds of survival compared to their odds of dying
- 3rd class passengers face a significant 68% lower chance of survival relative to their odds of dying.

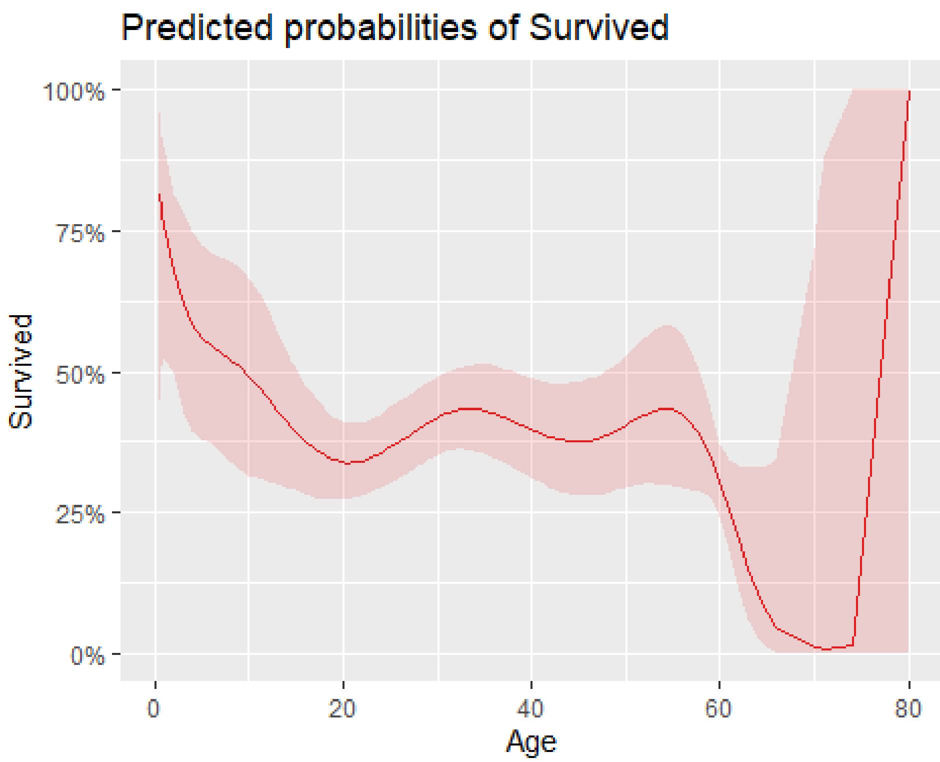
```
Contact logistic regression with numeric predictors
modela <- glm(Survived~Age, titanic, family = binomial)

Visualize predictions
library(sjPlot)
summary(modela)
 Call:
 glm(formula = Survived ~ Age, family = binomial, data = titanic)

 Coefficients:
              Estimate Std. Error z value Pr(>|z|)
 (Intercept)  0.002310   0.153782   0.015 0.988018
 Age         -0.013688   0.003958  -3.458 0.000544 ***

 Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

 (Dispersion parameter for binomial family taken to be 1)

     Null deviance: 1182.8  on 888  degrees of freedom
 Residual deviance: 1170.7  on 887  degrees of freedom
 AIC: 1174.7

 Number of Fisher Scoring iterations: 4

plot_model(modela, type = "pred", terms = "Age[all]")
```



- The probability of survival decreases with an increase in ag
e

- Higher proportion of young people survived during the titani
c accident as compared to older ones.

```
How to check for non-linearity
Use polynomial degrees based on age
modela1 <- glm(Survived~poly(Age, 2), data = titanic,
               family = binomial)
modela2 <- glm(Survived~poly(Age, 3), data = titanic,
               family = binomial)
modela3 <- glm(Survived~poly(Age, 4), data = titanic,
               family = binomial)
Dont overdo
modela4 <- glm(Survived~poly(Age, 10), data = titanic,
               family = binomial)
plot_model(modela4, type = "pred", terms = "Age")
```



- It is difficult to interpret the survival rates of passengers due to complexity of the model.

```
summary(modela4)
```

```
Call:
glm(formula = Survived ~ poly(Age, 10), family = binomial, data = titanic)

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)     -0.50462    0.07648  -6.598 4.16e-11 ***
poly(Age, 10)1  -8.28223    2.87775  -2.878    0.004 **
poly(Age, 10)2   0.96533    3.83039   0.252    0.801
poly(Age, 10)3  -7.61989    7.14851  -1.066    0.286
poly(Age, 10)4   4.66868    8.65548   0.539    0.590
poly(Age, 10)5   3.66760    6.66164   0.551    0.582
poly(Age, 10)6   6.87432    7.17499   0.958    0.338
poly(Age, 10)7   7.23940    6.56334   1.103    0.270
poly(Age, 10)8   3.12351    4.19513   0.745    0.457
poly(Age, 10)9  -1.97385    3.24969  -0.607    0.544
poly(Age, 10)10 -0.29224    3.01064  -0.097    0.923

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1182.8  on 888  degrees of freedom
Residual deviance: 1146.5  on 878  degrees of freedom
AIC: 1168.5

Number of Fisher Scoring iterations: 7
```

```
•  Degrees more than 3 and 4 are rarely used due to the risk of
   overfitting, which makes the model overly complex and less
   interpretable as of the example above
```

```
Compare all the models we have created so far
AIC(modela1, modela2, modela3, modela4)
```

```
        df      AIC
modela1  3 1175.682
modela2  4 1167.686
modela3  5 1165.135
modela4 11 1168.470
```

```
•  Select the model with the lowest aic. aic measures the
   relative quality of statistical models

•  Here we will select the model with 3rd polynomial degree
   and move on
```

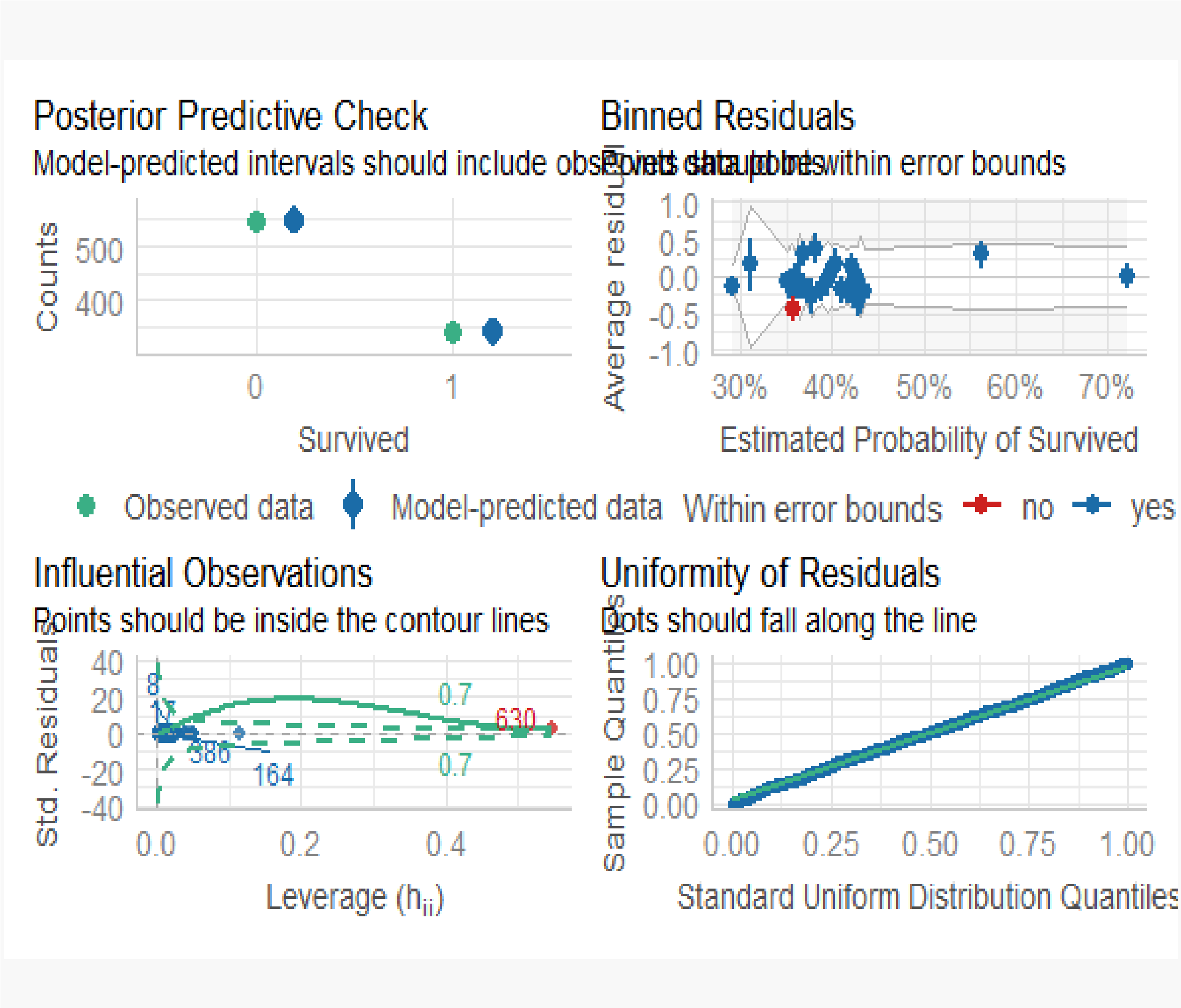
```
Another method used to choose polynomial degrees
tab_model(modela4)
```

	Survived		
Predictors	*Odds Ratios*	*CI*	*p*
(Intercept)	0.60	0.51 – 0.74	**<0.201**
Age [1st degree]	0.00	0.00 – 63.93	**0.304**
Age [2nd degree]	2.63	0.00 – NA	0.801
Age [3rd degree]	0.00	0.00 – NA	0.001
Age [4th degree]	106.56	0.00 – NA	0.590
Age [5th degree]	39.16	0.15 – NA	0.582

Age [6th degree]	967.12	0.90 – NA	0.338
Age [7th degree]	1393.25	1.24 – NA	0.270
Age [8th degree]	22.73	0.09 – NA	0.457
Age [9th degree]	0.14	0.00 – 219448.52	0.544
Age [10th degree]	0.75	0.00 – 2800.19	0.923
Observations	889		
R^2 Tjur	0.038		

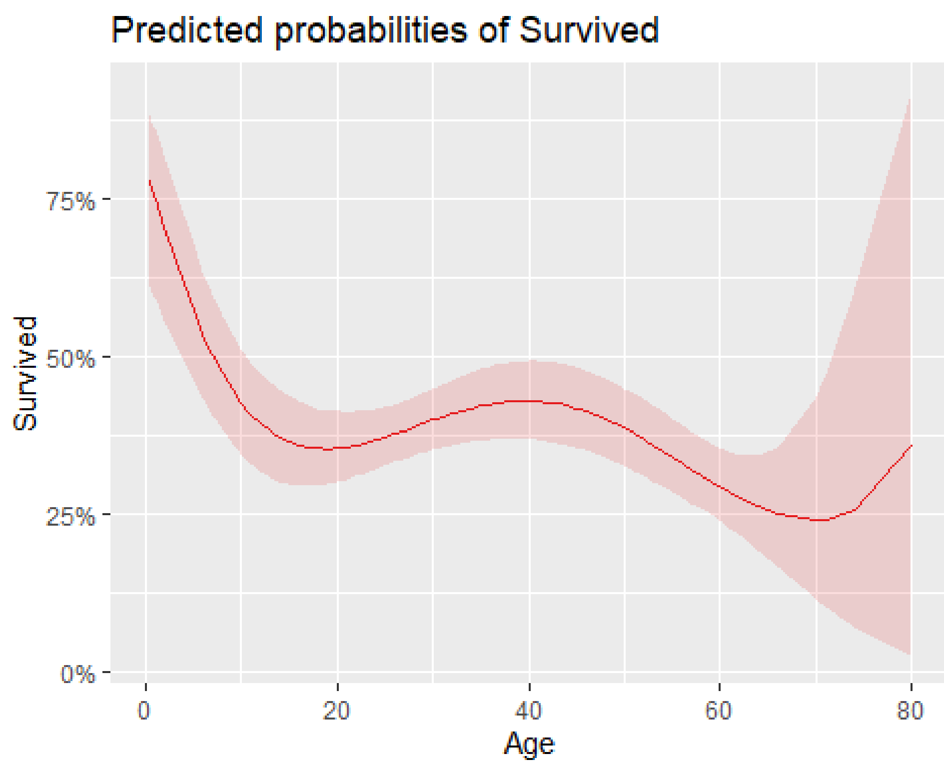
```
•  Here we choose the polynomial degree that is statistically
   significant.
```

```
Check model assumptions
library(performance)

check_model(modela3)
```



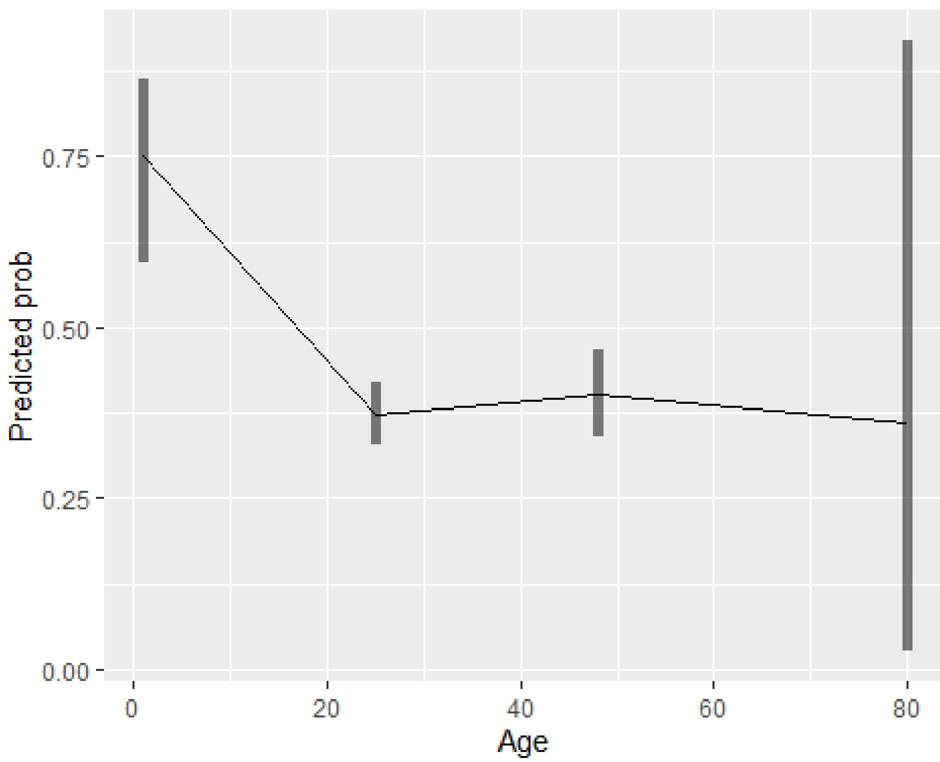
- The posterior predictive check involves comparing the models predicted intervals with the actual observed values. It helps assess how well the model aligns with the data
- Most residuals fall within the error bands with only one potential outlier
- The influential observational plot tells us that this outlier do not appear to be influential.
- The residuals exhibit a uniform distribution
- The model seems to be okay and we are good to go

```
Visualize predictions
plot_model(modela3, type = "eff", terms = "Age[all]")
```




```
•  The plot clearly shows that babies and young children have
   the highest survival rate.

•  Survival probability then decreases until around age 25 before
   gradually increasing to a peak at approximately 48 years old.

•  After this, it declines again. This pattern indicates two turning
   points and essentially divides the data into three distinct areas.
```

```
Visualize particular predictions
library(emmeans)
b <- emmip(modela3, ~Age, CIs = T, type = "response",
           at = list(Age = c(1, 25, 48, 80)))+
  scale_y_continuous()
b
```



Predicted prob
0.00
0.25
0.50
0.75
0
20
40
60
80
Age