Resolución

1.1.

```
> y <- c(2, 1, 4, 2, 3, 6, 2, 5, 4, 6)
> x <- c(7, 6, 2, 8, 4, 1, 7, 3, 4, 2)
> model_poiss <- glm(y ~ x, family = poisson)
> summary(model_poiss)

Call:
glm(formula = y ~ x, family = poisson)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.9667  -0.1942   0.0818   0.3374   0.3946

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.02283    0.32046   6.312 2.75e-10 ***
x           -0.19830    0.08007  -2.476   0.0133 *
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 8.4933  on 9  degrees of freedom
Residual deviance: 1.7834  on 8  degrees of freedom
AIC: 35.943

Number of Fisher Scoring iterations: 4
```

1.2.

```
> library(rstan)
> rstan_options(auto_write = TRUE)
> options(mc.cores = parallel::detectCores())
> set.seed(1)
> datos <- list(n = length(y), x = x, y = y)
> codigo <- "
+   data {
+     int<lower=0> n;
+     int<lower=0> y[n];
+     int<lower=0> x[n];
+   }
```

```
+
+   parameters {
+     real tau;
+     real omega;
+   }
+   model {
+           real lambda;
+     tau ~ normal(0, 0.5);
+     omega ~ normal(0, 0.5);
+     for (i in 1:n) {
+             lambda = exp(tau + omega*x[i]);
+       y[i] ~ poisson(lambda);
+     }
+   }
+ "
> fit <- stan(model_code = codigo, data = datos, iter = 1000)
> print(fit)
```

```
Inference for Stan model: anon_model.
4 chains, each with iter=1000; warmup=500; thin=1;
post-warmup draws per chain=500, total post-warmup draws=2000.

       mean se_mean   sd  2.5%   25%   50%   75% 97.5%
tau    1.35    0.01 0.30  0.77  1.14  1.36  1.57  1.94
omega -0.07    0.00 0.07 -0.21 -0.12 -0.07 -0.02  0.06
lp__   5.56    0.04 0.95  3.15  5.20  5.85  6.25  6.52
      n_eff Rhat
tau     620    1
omega   600    1
lp__    721    1

Samples were drawn using NUTS(diag_e) at Sat Apr  8 18:00:55 2023.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

1.3.

```
> set.seed(1)
> datos <- list(n = length(y), x = x, y = y)
> codigo <- "
+   data {
+     int<lower=0> n;
+     int<lower=0> y[n];
+     int<lower=0> x[n];
+   }
```

```
+
+   parameters {
+     real tau;
+     real omega;
+   }
+   model {
+           real lambda;
+     tau ~ normal(0, 5);
+     omega ~ normal(0, 5);
+     for (i in 1:n) {
+             lambda = exp(tau + omega*x[i]);
+       y[i] ~ poisson(lambda);
+     }
+   }
+ "
> fit <- stan(model_code = codigo, data = datos, iter = 1000)
> print(fit)
```

```
Inference for Stan model: anon_model.
4 chains, each with iter=1000; warmup=500; thin=1;
post-warmup draws per chain=500, total post-warmup draws=2000.

       mean se_mean   sd  2.5%   25%   50%   75% 97.5%
tau    1.99    0.01 0.33  1.32  1.78  2.00  2.21  2.60
omega -0.20    0.00 0.08 -0.36 -0.26 -0.20 -0.14 -0.04
lp__  11.08    0.04 1.01  8.31 10.67 11.39 11.83 12.09
      n_eff Rhat
tau     560 1.00
omega   550 1.01
lp__    691 1.01

Samples were drawn using NUTS(diag_e) at Sat Apr  8 18:01:53 2023.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

1.4.

Hemos de tener en cuenta que el primer modelo es frecuentista y parte de diferentes supuestos y trata con la estimación puntual de los coeficientes; no considero que sus resultados sean comparables a los 2 modelos bayesianos siguientes, que tratan con distribuciones a posteriori.
Entre estos últimos, calculamos $AIC = 2k - 2logverosimilitud$:

```
> AIC_2 <- 2*2 - 2*5.56
> AIC_3 <- 2*2 - 2*11.08
> AIC_2
```

```
[1] -7.12

> AIC_3

[1] -18.16
```

Por lo tanto, determinamos que el modelo 3 presenta mejor ajuste teniendo en cuenta las penalizaciones de complejidad de dichos modelos.
Este a su vez nos indica que por cada hora de formación previa recibida, se cometen 0.2 errores menos de media, frente a los 0.07 del anterior.

2.1.

```
> set.seed(1)
> x1 = c(28, 43, 19, 30, 25, 32, 45, 41, 26, 22, 32, 41, 36, 20, 29)
> y = c(1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0)
> datos <- list(n = length(y), x = x1, y = y)
> codigo <- "
+ data {
+   int<lower=0> n;
+   int<lower=0,upper=1> y[n];
+   real x[n];
+ }
+
+ parameters {
+   real tau;
+   real omega;
+ }
+
+ model {
+         real pi;
+   tau ~ normal(0,5);
+   omega ~ normal(0,5);
+   for (i in 1:n) {
+           pi = (exp(tau+omega*x[i]))/(1+exp(tau+omega*x[i])) ;
+     y[i] ~ bernoulli(pi);
+   }
+ }
+ "
> fit <- stan(model_code = codigo, data = datos, iter = 1000)
> print(fit)

Inference for Stan model: anon_model.
4 chains, each with iter=1000; warmup=500; thin=1;
post-warmup draws per chain=500, total post-warmup draws=2000.
```

```
       mean se_mean   sd   2.5%   25%   50%   75% 97.5%
tau    5.60    0.14 2.58   1.14  3.91  5.38  7.15 11.41
omega -0.18    0.00 0.08  -0.37 -0.23 -0.17 -0.12 -0.04
lp__  -8.81    0.05 1.08 -11.85 -9.15 -8.48 -8.05 -7.79
      n_eff Rhat
tau     356 1.01
omega   368 1.01
lp__    412 1.02

Samples were drawn using NUTS(diag_e) at Sat Apr  8 18:02:59 2023.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).

>
```

2.2.
Dado que el intervalo posterior de omega no incluye el 0, determinamos que la edad contribuye a la recuperación de forma significativa. Concretamente, disminuye la probabilidad de recuperación.
2.3.

```
> set.seed(1)
> x1 = c(28, 43, 19, 30, 25, 32, 45, 41, 26, 22, 32, 41, 36, 20, 29)
> x2 = c(11, 17, 14, 12, 18, 12, 14, 16, 12, 13, 17, 15, 11, 13, 20)
> y = c(1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0)
> datos <- list(n = length(y), x1 = x1, x2 = x2, y = y)
> codigo <- "
+ data {
+   int<lower=0> n;
+   int<lower=0,upper=1> y[n];
+   real x1[n];
+   real x2[n];
+ }
+
+ parameters {
+   real tau;
+   real omega1;
+   real omega2;
+ }
+
+ model {
+         real pi;
+   tau ~ normal(0, 5);
+   omega1 ~ normal(0, 5);
+   omega2 ~ normal(0, 5);
```

```
+
+   for (i in 1:n) {
+           pi = (exp(tau+omega1*x1[i]+omega2*x2[i]))/(1+exp(tau+omega1*x1[i]+omega2*x2[i]))
+     y[i] ~ bernoulli(pi);
+   }
+ }
+ "
> fit <- stan(model_code = codigo, data = datos, iter = 1000)
> print(fit)
```

```
Inference for Stan model: anon_model.
4 chains, each with iter=1000; warmup=500; thin=1;
post-warmup draws per chain=500, total post-warmup draws=2000.

        mean se_mean   sd   2.5%   25%   50%   75% 97.5%
tau    11.17    0.13 3.53   4.25  8.84 10.98 13.47 18.31
omega1 -0.11    0.00 0.09  -0.29 -0.16 -0.11 -0.05  0.05
omega2 -0.54    0.01 0.25  -1.07 -0.71 -0.54 -0.36 -0.10
lp__   -6.67    0.05 1.35 -10.31 -7.30 -6.30 -5.70 -5.16
       n_eff Rhat
tau      790 1.00
omega1   911 1.00
omega2   835 1.00
lp__     654 1.01

Samples were drawn using NUTS(diag_e) at Sat Apr  8 18:04:06 2023.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

2.4.

Asumiendo que no existe interacción entre edad y puntuación en depresión, concluimos que, al incluir ambos intervalos posteriores el 0, contamos con incertidumbre en lo que a la contribución de cada variable predictora a la predicción de recuperación se refiere.

Interpretando los efectos principales, encontramos que el aumento en la edad disminuye las probabilidades de recuperación, al igual que ocurre con el aumento de la puntuación en depresión.