

Supplementary material to a Bayesian framework for patient-level partitioned survival cost-utility analysis

A. Gabrio

1 Model Code

The **STAN** code for the model used in the analysis of TOPICAL is given below. We note that, when there are missing values, **STAN** requires these to be generated as random variables using appropriate distributions for each missingness patterns rather than passing the unobserved values as **NA** data to the model. Thus, although this may become cumbersome as the number of patterns increases, we impute the missing values under MAR by coding the model for each missingness pattern and generate imputations based on the parameter estimates informed from the observed data across all patterns.

We provide two versions of the code, the first being the one implemented to all the data from TOPICAL which handles missingness in each pattern under MAR (Section 1.1) and the second (less confusing to read) which is only fitted to the complete cases (Section 1.2).

1.1 STAN code - all cases under MAR

```
// data block contains observed data on each variable
// by treatment and pattern (8 in control and 9 in intervention)
data {
  int<lower=0> n1_pat1;
  int<lower=0> n1_pat2;
  int<lower=0> n1_pat4;
  int<lower=0> n1_pat6;
  int<lower=0> n1_pat7;
  int<lower=0> n1_pat8;
  int<lower=0> n1_pat9;
  int<lower=0> n1_pat11;
  int<lower=0> n2_pat1;
  int<lower=0> n2_pat2;
  int<lower=0> n2_pat3;
  int<lower=0> n2_pat4;
  int<lower=0> n2_pat5;
  int<lower=0> n2_pat8;
  int<lower=0> n2_pat9;
  int<lower=0> n2_pat10;
  int<lower=0> n2_pat12;
  int<lower=0> n_pred_pps;
  int<lower=0> n_pred_drug;
  int<lower=0> n_pred_hos;
  int<lower=0> n_pred_ae;
  vector[n1_pat1] qaly_PFS1_pat1;
  vector[n1_pat2] qaly_PFS1_pat2;
  vector[n1_pat4] qaly_PFS1_pat4;
  vector[n1_pat6] qaly_PFS1_pat6;
  vector[n1_pat7] qaly_PFS1_pat7;
  vector[n2_pat1] qaly_PFS2_pat1;
  vector[n2_pat2] qaly_PFS2_pat2;
  vector[n2_pat3] qaly_PFS2_pat3;
```

```

vector[n2_pat4] qaly_PFS2_pat4;
vector[n2_pat5] qaly_PFS2_pat5;
vector[n1_pat1] qaly_PPS1_pat1;
vector[n1_pat2] qaly_PPS1_pat2;
vector[n1_pat4] qaly_PPS1_pat4;
vector[n1_pat6] qaly_PPS1_pat6;
vector[n1_pat8] qaly_PPS1_pat8;
vector[n2_pat1] qaly_PPS2_pat1;
vector[n2_pat2] qaly_PPS2_pat2;
vector[n2_pat3] qaly_PPS2_pat3;
vector[n2_pat4] qaly_PPS2_pat4;
vector[n2_pat5] qaly_PPS2_pat5;
vector[n2_pat8] qaly_PPS2_pat8;
vector[n1_pat1] cost_drug1_pat1;
vector[n1_pat2] cost_drug1_pat2;
vector[n1_pat7] cost_drug1_pat7;
vector[n1_pat8] cost_drug1_pat8;
vector[n1_pat9] cost_drug1_pat9;
vector[n1_pat11] cost_drug1_pat11;
vector[n2_pat1] cost_drug2_pat1;
vector[n2_pat2] cost_drug2_pat2;
vector[n2_pat3] cost_drug2_pat3;
vector[n2_pat8] cost_drug2_pat8;
vector[n2_pat9] cost_drug2_pat9;
vector[n2_pat10] cost_drug2_pat10;
vector[n1_pat1] cost_hos1_pat1;
vector[n1_pat2] cost_hos1_pat2;
vector[n1_pat4] cost_hos1_pat4;
vector[n1_pat7] cost_hos1_pat7;
vector[n1_pat8] cost_hos1_pat8;
vector[n1_pat9] cost_hos1_pat9;
vector[n2_pat1] cost_hos2_pat1;
vector[n2_pat2] cost_hos2_pat2;
vector[n2_pat4] cost_hos2_pat4;
vector[n2_pat5] cost_hos2_pat5;
vector[n2_pat8] cost_hos2_pat8;
vector[n2_pat9] cost_hos2_pat9;
vector[n2_pat10] cost_hos2_pat10;
vector[n2_pat12] cost_hos2_pat12;
vector[n1_pat1] cost_ae1_pat1;
vector[n1_pat4] cost_ae1_pat4;
vector[n1_pat6] cost_ae1_pat6;
vector[n1_pat7] cost_ae1_pat7;
vector[n1_pat8] cost_ae1_pat8;
vector[n1_pat9] cost_ae1_pat9;
vector[n1_pat11] cost_ae1_pat11;
vector[n2_pat1] cost_ae2_pat1;
vector[n2_pat3] cost_ae2_pat3;
vector[n2_pat4] cost_ae2_pat4;
vector[n2_pat8] cost_ae2_pat8;
vector[n2_pat9] cost_ae2_pat9;
vector[n2_pat12] cost_ae2_pat12;
vector[n1_pat1] X1_PPS_pat1;
vector[n1_pat2] X1_PPS_pat2;
vector[n1_pat4] X1_PPS_pat4;
vector[n1_pat6] X1_PPS_pat6;
vector[n1_pat7] X1_PPS_pat7;
vector[n1_pat8] X1_PPS_pat8;
vector[n1_pat9] X1_PPS_pat9;
vector[n1_pat11] X1_PPS_pat11;
vector[n2_pat1] X2_PPS_pat1;
vector[n2_pat2] X2_PPS_pat2;
vector[n2_pat3] X2_PPS_pat3;
vector[n2_pat4] X2_PPS_pat4;
vector[n2_pat5] X2_PPS_pat5;
vector[n2_pat8] X2_PPS_pat8;
vector[n2_pat9] X2_PPS_pat9;
vector[n2_pat10] X2_PPS_pat10;
vector[n2_pat12] X2_PPS_pat12;
vector[n1_pat1] X1_drug_pat1;
vector[n1_pat2] X1_drug_pat2;

```

```

vector[n1_pat4] X1_drug_pat4;
vector[n1_pat6] X1_drug_pat6;
vector[n1_pat7] X1_drug_pat7;
vector[n1_pat8] X1_drug_pat8;
vector[n1_pat9] X1_drug_pat9;
vector[n1_pat11] X1_drug_pat11;
vector[n2_pat1] X2_drug_pat1;
vector[n2_pat2] X2_drug_pat2;
vector[n2_pat3] X2_drug_pat3;
vector[n2_pat4] X2_drug_pat4;
vector[n2_pat5] X2_drug_pat5;
vector[n2_pat8] X2_drug_pat8;
vector[n2_pat9] X2_drug_pat9;
vector[n2_pat10] X2_drug_pat10;
vector[n2_pat12] X2_drug_pat12;
vector[n1_pat1] X1_hos_pat1;
vector[n1_pat2] X1_hos_pat2;
vector[n1_pat4] X1_hos_pat4;
vector[n1_pat6] X1_hos_pat6;
vector[n1_pat7] X1_hos_pat7;
vector[n1_pat8] X1_hos_pat8;
vector[n1_pat9] X1_hos_pat9;
vector[n1_pat11] X1_hos_pat11;
vector[n2_pat1] X2_hos_pat1;
vector[n2_pat2] X2_hos_pat2;
vector[n2_pat3] X2_hos_pat3;
vector[n2_pat4] X2_hos_pat4;
vector[n2_pat5] X2_hos_pat5;
vector[n2_pat8] X2_hos_pat8;
vector[n2_pat9] X2_hos_pat9;
vector[n2_pat10] X2_hos_pat10;
vector[n2_pat12] X2_hos_pat12;
vector[n1_pat1] X1_ae_pat1;
vector[n1_pat2] X1_ae_pat2;
vector[n1_pat4] X1_ae_pat4;
vector[n1_pat6] X1_ae_pat6;
vector[n1_pat7] X1_ae_pat7;
vector[n1_pat8] X1_ae_pat8;
vector[n1_pat9] X1_ae_pat9;
vector[n1_pat11] X1_ae_pat11;
vector[n2_pat1] X2_ae_pat1;
vector[n2_pat2] X2_ae_pat2;
vector[n2_pat3] X2_ae_pat3;
vector[n2_pat4] X2_ae_pat4;
vector[n2_pat5] X2_ae_pat5;
vector[n2_pat8] X2_ae_pat8;
vector[n2_pat9] X2_ae_pat9;
vector[n2_pat10] X2_ae_pat10;
vector[n2_pat12] X2_ae_pat12;
}
// parameters block contains parameters shared across patterns
// and missing values by pattern for each variable
parameters {
real alpha0_PFS[2];
real<lower=0> sigma_e_PFS[2];
vector[n1_pat8] qaly_PFS1_pat8;
vector[n1_pat9] qaly_PFS1_pat9;
vector[n1_pat11] qaly_PFS1_pat11;
vector[n2_pat8] qaly_PFS2_pat8;
vector[n2_pat9] qaly_PFS2_pat9;
vector[n2_pat10] qaly_PFS2_pat10;
vector[n2_pat12] qaly_PFS2_pat12;
real alpha0_PPS[2];
real alpha1_PPS[2];
vector[n_pred_pps] gamma_PPS1;
vector[n_pred_pps] gamma_PPS2;
vector<lower=0>[n1_pat7] qaly_PPS1_pat7;
vector<lower=0>[n1_pat9] qaly_PPS1_pat9;
vector<lower=0>[n1_pat11] qaly_PPS1_pat11;
vector<lower=0>[n2_pat9] qaly_PPS2_pat9;
vector<lower=0>[n2_pat10] qaly_PPS2_pat10;

```

```

vector<lower=0>[n2_pat12] qaly_PPS2_pat12;
real beta0_drug[2];
real beta1_drug[2];
real beta2_drug[2];
real<lower=0> lsigma_c_drug[2];
vector[n_pred_drug] delta_drug1;
vector[n_pred_drug] delta_drug2;
vector<lower=0>[n1_pat4] cost_drug1_pat4;
vector<lower=0>[n1_pat6] cost_drug1_pat6;
vector<lower=0>[n2_pat4] cost_drug2_pat4;
vector<lower=0>[n2_pat5] cost_drug2_pat5;
vector<lower=0>[n2_pat12] cost_drug2_pat12;
real beta0_hos[2];
real beta1_hos[2];
real beta2_hos[2];
real beta3_hos[2];
real<lower=0> lsigma_c_hos[2];
vector[n_pred_hos] delta_hos1;
vector[n_pred_hos] delta_hos2;
vector<lower=0>[n1_pat6] cost_hos1_pat6;
vector<lower=0>[n1_pat11] cost_hos1_pat11;
vector<lower=0>[n2_pat3] cost_hos2_pat3;
real beta0_ae[2];
real beta1_ae[2];
real beta2_ae[2];
real beta3_ae[2];
real beta4_ae[2];
real<lower=0> lsigma_c_ae[2];
vector[n_pred_ae] delta_ae1;
vector[n_pred_ae] delta_ae2;
vector<lower=0>[n1_pat2] cost_ae1_pat2;
vector<lower=0>[n2_pat2] cost_ae2_pat2;
vector<lower=0>[n2_pat5] cost_ae2_pat5;
vector<lower=0>[n2_pat10] cost_ae2_pat10;
}
// transformed parameters block contains deterministic functions of parameters
// and the observed/missing values by pattern for each variable
transformed parameters {
real<lower=0> tau_e_PFS[2];
real eta_e_PFS[2];
vector<lower=0>[n1_pat1] eta_e_PPS1_pat1;
vector<lower=0>[n1_pat2] eta_e_PPS1_pat2;
vector<lower=0>[n1_pat4] eta_e_PPS1_pat4;
vector<lower=0>[n1_pat6] eta_e_PPS1_pat6;
vector<lower=0>[n1_pat7] eta_e_PPS1_pat7;
vector<lower=0>[n1_pat8] eta_e_PPS1_pat8;
vector<lower=0>[n1_pat9] eta_e_PPS1_pat9;
vector<lower=0>[n1_pat11] eta_e_PPS1_pat11;
vector<lower=0>[n2_pat1] eta_e_PPS2_pat1;
vector<lower=0>[n2_pat2] eta_e_PPS2_pat2;
vector<lower=0>[n2_pat3] eta_e_PPS2_pat3;
vector<lower=0>[n2_pat4] eta_e_PPS2_pat4;
vector<lower=0>[n2_pat5] eta_e_PPS2_pat5;
vector<lower=0>[n2_pat8] eta_e_PPS2_pat8;
vector<lower=0>[n2_pat9] eta_e_PPS2_pat9;
vector<lower=0>[n2_pat10] eta_e_PPS2_pat10;
vector<lower=0>[n2_pat12] eta_e_PPS2_pat12;
vector<lower=0>[n1_pat1] lambda_e_PPS1_pat1;
vector<lower=0>[n1_pat2] lambda_e_PPS1_pat2;
vector<lower=0>[n1_pat4] lambda_e_PPS1_pat4;
vector<lower=0>[n1_pat6] lambda_e_PPS1_pat6;
vector<lower=0>[n1_pat7] lambda_e_PPS1_pat7;
vector<lower=0>[n1_pat8] lambda_e_PPS1_pat8;
vector<lower=0>[n1_pat9] lambda_e_PPS1_pat9;
vector<lower=0>[n1_pat11] lambda_e_PPS1_pat11;
vector<lower=0>[n2_pat1] lambda_e_PPS2_pat1;
vector<lower=0>[n2_pat2] lambda_e_PPS2_pat2;
vector<lower=0>[n2_pat3] lambda_e_PPS2_pat3;
vector<lower=0>[n2_pat4] lambda_e_PPS2_pat4;
vector<lower=0>[n2_pat5] lambda_e_PPS2_pat5;
vector<lower=0>[n2_pat8] lambda_e_PPS2_pat8;

```

```

vector<lower=0>[n2_pat9] lambda_e_PPS2_pat9;
vector<lower=0>[n2_pat10] lambda_e_PPS2_pat10;
vector<lower=0>[n2_pat12] lambda_e_PPS2_pat12;
vector[n1_pat1] leta_c_drug1_pat1;
vector[n1_pat2] leta_c_drug1_pat2;
vector[n1_pat4] leta_c_drug1_pat4;
vector[n1_pat6] leta_c_drug1_pat6;
vector[n1_pat7] leta_c_drug1_pat7;
vector[n1_pat8] leta_c_drug1_pat8;
vector[n1_pat9] leta_c_drug1_pat9;
vector[n1_pat11] leta_c_drug1_pat11;
vector[n2_pat1] leta_c_drug2_pat1;
vector[n2_pat2] leta_c_drug2_pat2;
vector[n2_pat3] leta_c_drug2_pat3;
vector[n2_pat4] leta_c_drug2_pat4;
vector[n2_pat5] leta_c_drug2_pat5;
vector[n2_pat8] leta_c_drug2_pat8;
vector[n2_pat9] leta_c_drug2_pat9;
vector[n2_pat10] leta_c_drug2_pat10;
vector[n2_pat12] leta_c_drug2_pat12;
vector[n1_pat1] leta_c_hos1_pat1;
vector[n1_pat2] leta_c_hos1_pat2;
vector[n1_pat4] leta_c_hos1_pat4;
vector[n1_pat6] leta_c_hos1_pat6;
vector[n1_pat7] leta_c_hos1_pat7;
vector[n1_pat8] leta_c_hos1_pat8;
vector[n1_pat9] leta_c_hos1_pat9;
vector[n1_pat11] leta_c_hos1_pat11;
vector[n2_pat1] leta_c_hos2_pat1;
vector[n2_pat2] leta_c_hos2_pat2;
vector[n2_pat3] leta_c_hos2_pat3;
vector[n2_pat4] leta_c_hos2_pat4;
vector[n2_pat5] leta_c_hos2_pat5;
vector[n2_pat8] leta_c_hos2_pat8;
vector[n2_pat9] leta_c_hos2_pat9;
vector[n2_pat10] leta_c_hos2_pat10;
vector[n2_pat12] leta_c_hos2_pat12;
vector[n1_pat1] leta_c_ae1_pat1;
vector[n1_pat2] leta_c_ae1_pat2;
vector[n1_pat4] leta_c_ae1_pat4;
vector[n1_pat6] leta_c_ae1_pat6;
vector[n1_pat7] leta_c_ae1_pat7;
vector[n1_pat8] leta_c_ae1_pat8;
vector[n1_pat9] leta_c_ae1_pat9;
vector[n1_pat11] leta_c_ae1_pat11;
vector[n2_pat1] leta_c_ae2_pat1;
vector[n2_pat2] leta_c_ae2_pat2;
vector[n2_pat3] leta_c_ae2_pat3;
vector[n2_pat4] leta_c_ae2_pat4;
vector[n2_pat5] leta_c_ae2_pat5;
vector[n2_pat8] leta_c_ae2_pat8;
vector[n2_pat9] leta_c_ae2_pat9;
vector[n2_pat10] leta_c_ae2_pat10;
vector[n2_pat12] leta_c_ae2_pat12;
tau_e_PFS[1] = (sigma_e_PFS[1]*sqrt(6))/pi();
tau_e_PFS[2] = (sigma_e_PFS[2]*sqrt(6))/pi();
eta_e_PFS[1] = alpha0_PFS[1] - tau_e_PFS[1]*0.5722;
eta_e_PFS[2] = alpha0_PFS[2] - tau_e_PFS[2]*0.5722;
eta_e_PPS1_pat1 = exp(alpha0_PPS[1] + alpha1_PPS[1]*(qaly_PFS1_pat1));
eta_e_PPS1_pat2 = exp(alpha0_PPS[1] + alpha1_PPS[1]*(qaly_PFS1_pat2));
eta_e_PPS1_pat4 = exp(alpha0_PPS[1] + alpha1_PPS[1]*(qaly_PFS1_pat4));
eta_e_PPS1_pat6 = exp(alpha0_PPS[1] + alpha1_PPS[1]*(qaly_PFS1_pat6));
eta_e_PPS1_pat7 = exp(alpha0_PPS[1] + alpha1_PPS[1]*(qaly_PFS1_pat7));
eta_e_PPS1_pat8 = exp(alpha0_PPS[1] + alpha1_PPS[1]*(qaly_PFS1_pat8));
eta_e_PPS1_pat9 = exp(alpha0_PPS[1] + alpha1_PPS[1]*(qaly_PFS1_pat9));
eta_e_PPS1_pat11 = exp(alpha0_PPS[1] + alpha1_PPS[1]*(qaly_PFS1_pat11));
eta_e_PPS2_pat1 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2_pat1));
eta_e_PPS2_pat2 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2_pat2));
eta_e_PPS2_pat3 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2_pat3));
eta_e_PPS2_pat4 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2_pat4));
eta_e_PPS2_pat5 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2_pat5));

```

```

eta_e_PPS2_pat8 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2_pat8));
eta_e_PPS2_pat9 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2_pat9));
eta_e_PPS2_pat10 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2_pat10));
eta_e_PPS2_pat12 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2_pat12));
for (n in 1:n1_pat1) lambda_e_PPS1_pat1[n] = 1/eta_e_PPS1_pat1[n];
for (n in 1:n1_pat2) lambda_e_PPS1_pat2[n] = 1/eta_e_PPS1_pat2[n];
for (n in 1:n1_pat4) lambda_e_PPS1_pat4[n] = 1/eta_e_PPS1_pat4[n];
lambda_e_PPS1_pat6[1] = 1/eta_e_PPS1_pat6[1];
for (n in 1:n1_pat7) lambda_e_PPS1_pat7[n] = 1/eta_e_PPS1_pat7[n];
lambda_e_PPS1_pat8[1] = 1/eta_e_PPS1_pat8[1];
for (n in 1:n1_pat9) lambda_e_PPS1_pat9[n] = 1/eta_e_PPS1_pat9[n];
lambda_e_PPS1_pat11[1] = 1/eta_e_PPS1_pat11[1];
for (n in 1:n2_pat1) lambda_e_PPS2_pat1[n] = 1/eta_e_PPS2_pat1[n];
for (n in 1:n2_pat2) lambda_e_PPS2_pat2[n] = 1/eta_e_PPS2_pat2[n];
for (n in 1:n2_pat3) lambda_e_PPS2_pat3[n] = 1/eta_e_PPS2_pat3[n];
for (n in 1:n2_pat4) lambda_e_PPS2_pat4[n] = 1/eta_e_PPS2_pat4[n];
lambda_e_PPS2_pat5[1] = 1/eta_e_PPS2_pat5[1];
for (n in 1:n2_pat8) lambda_e_PPS2_pat8[n] = 1/eta_e_PPS2_pat8[n];
for (n in 1:n2_pat9) lambda_e_PPS2_pat9[n] = 1/eta_e_PPS2_pat9[n];
for (n in 1:n2_pat10) lambda_e_PPS2_pat10[n] = 1/eta_e_PPS2_pat10[n];
for (n in 1:n2_pat12) lambda_e_PPS2_pat12[n] = 1/eta_e_PPS2_pat12[n];
leta_c_drug1_pat1 = beta0_drug[1] + beta1_drug[1]*(qaly_PFS1_pat1) + beta2_drug[1]*(qaly_PPS1_pat1);
leta_c_drug1_pat2 = beta0_drug[1] + beta1_drug[1]*(qaly_PFS1_pat2) + beta2_drug[1]*(qaly_PPS1_pat2);
leta_c_drug1_pat4 = beta0_drug[1] + beta1_drug[1]*(qaly_PFS1_pat4) + beta2_drug[1]*(qaly_PPS1_pat4);
leta_c_drug1_pat6 = beta0_drug[1] + beta1_drug[1]*(qaly_PFS1_pat6) + beta2_drug[1]*(qaly_PPS1_pat6);
leta_c_drug1_pat7 = beta0_drug[1] + beta1_drug[1]*(qaly_PFS1_pat7) + beta2_drug[1]*(qaly_PPS1_pat7);
leta_c_drug1_pat8 = beta0_drug[1] + beta1_drug[1]*(qaly_PFS1_pat8) + beta2_drug[1]*(qaly_PPS1_pat8);
leta_c_drug1_pat9 = beta0_drug[1] + beta1_drug[1]*(qaly_PFS1_pat9) + beta2_drug[1]*(qaly_PPS1_pat9);
leta_c_drug1_pat11 = beta0_drug[1] + beta1_drug[1]*(qaly_PFS1_pat11) + beta2_drug[1]*(qaly_PPS1_pat11);
leta_c_drug2_pat1 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2_pat1) + beta2_drug[2]*(qaly_PPS2_pat1);
leta_c_drug2_pat2 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2_pat2) + beta2_drug[2]*(qaly_PPS2_pat2);
leta_c_drug2_pat3 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2_pat3) + beta2_drug[2]*(qaly_PPS2_pat3);
leta_c_drug2_pat4 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2_pat4) + beta2_drug[2]*(qaly_PPS2_pat4);
leta_c_drug2_pat5 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2_pat5) + beta2_drug[2]*(qaly_PPS2_pat5);
leta_c_drug2_pat8 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2_pat8) + beta2_drug[2]*(qaly_PPS2_pat8);
leta_c_drug2_pat9 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2_pat9) + beta2_drug[2]*(qaly_PPS2_pat9);
leta_c_drug2_pat10 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2_pat10) + beta2_drug[2]*(qaly_PPS2_pat10);
leta_c_drug2_pat12 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2_pat12) + beta2_drug[2]*(qaly_PPS2_pat12);
leta_c_hos1_pat1 = beta0_hos[1] + beta1_hos[1]*(qaly_PFS1_pat1) + beta2_hos[1]*(qaly_PPS1_pat1) +
    beta3_hos[1]*log(cost_drug1_pat1);
leta_c_hos1_pat2 = beta0_hos[1] + beta1_hos[1]*(qaly_PFS1_pat2) + beta2_hos[1]*(qaly_PPS1_pat2) +
    beta3_hos[1]*log(cost_drug1_pat2);
leta_c_hos1_pat4 = beta0_hos[1] + beta1_hos[1]*(qaly_PFS1_pat4) + beta2_hos[1]*(qaly_PPS1_pat4) +
    beta3_hos[1]*log(cost_drug1_pat4);
leta_c_hos1_pat6 = beta0_hos[1] + beta1_hos[1]*(qaly_PFS1_pat6) + beta2_hos[1]*(qaly_PPS1_pat6) +
    beta3_hos[1]*log(cost_drug1_pat6);
leta_c_hos1_pat7 = beta0_hos[1] + beta1_hos[1]*(qaly_PFS1_pat7) + beta2_hos[1]*(qaly_PPS1_pat7) +
    beta3_hos[1]*log(cost_drug1_pat7);
leta_c_hos1_pat8 = beta0_hos[1] + beta1_hos[1]*(qaly_PFS1_pat8) + beta2_hos[1]*(qaly_PPS1_pat8) +
    beta3_hos[1]*log(cost_drug1_pat8);
leta_c_hos1_pat9 = beta0_hos[1] + beta1_hos[1]*(qaly_PFS1_pat9) + beta2_hos[1]*(qaly_PPS1_pat9) +
    beta3_hos[1]*log(cost_drug1_pat9);
leta_c_hos1_pat11 = beta0_hos[1] + beta1_hos[1]*(qaly_PFS1_pat11) + beta2_hos[1]*(qaly_PPS1_pat11) +
    beta3_hos[1]*log(cost_drug1_pat11);
leta_c_hos2_pat1 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2_pat1) + beta2_hos[2]*(qaly_PPS2_pat1) +
    beta3_hos[2]*log(cost_drug2_pat1);
leta_c_hos2_pat2 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2_pat2) + beta2_hos[2]*(qaly_PPS2_pat2) +
    beta3_hos[2]*log(cost_drug2_pat2);
leta_c_hos2_pat3 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2_pat3) + beta2_hos[2]*(qaly_PPS2_pat3) +
    beta3_hos[2]*log(cost_drug2_pat3);
leta_c_hos2_pat4 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2_pat4) + beta2_hos[2]*(qaly_PPS2_pat4) +
    beta3_hos[2]*log(cost_drug2_pat4);
leta_c_hos2_pat5 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2_pat5) + beta2_hos[2]*(qaly_PPS2_pat5) +
    beta3_hos[2]*log(cost_drug2_pat5);
leta_c_hos2_pat8 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2_pat8) + beta2_hos[2]*(qaly_PPS2_pat8) +
    beta3_hos[2]*log(cost_drug2_pat8);
leta_c_hos2_pat9 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2_pat9) + beta2_hos[2]*(qaly_PPS2_pat9) +
    beta3_hos[2]*log(cost_drug2_pat9);
leta_c_hos2_pat10 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2_pat10) + beta2_hos[2]*(qaly_PPS2_pat10) +
    beta3_hos[2]*log(cost_drug2_pat10);
leta_c_hos2_pat12 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2_pat12) + beta2_hos[2]*(qaly_PPS2_pat12) +

```

```

      beta3_hos[2]*log(cost_drug2_pat12);
leta_c_ae1_pat1 = beta0_ae[1] + beta1_ae[1]*(qaly_PFS1_pat1) + beta2_ae[1]*(qaly_PPS1_pat1) +
      beta3_ae[1]*log(cost_drug1_pat1) + beta4_ae[1]*log(cost_hos1_pat1);
leta_c_ae1_pat2 = beta0_ae[1] + beta1_ae[1]*(qaly_PFS1_pat2) + beta2_ae[1]*(qaly_PPS1_pat2) +
      beta3_ae[1]*log(cost_drug1_pat2) + beta4_ae[1]*log(cost_hos1_pat2);
leta_c_ae1_pat4 = beta0_ae[1] + beta1_ae[1]*(qaly_PFS1_pat4) + beta2_ae[1]*(qaly_PPS1_pat4) +
      beta3_ae[1]*log(cost_drug1_pat4) + beta4_ae[1]*log(cost_hos1_pat4);
leta_c_ae1_pat6 = beta0_ae[1] + beta1_ae[1]*(qaly_PFS1_pat6) + beta2_ae[1]*(qaly_PPS1_pat6) +
      beta3_ae[1]*log(cost_drug1_pat6) + beta4_ae[1]*log(cost_hos1_pat6);
leta_c_ae1_pat7 = beta0_ae[1] + beta1_ae[1]*(qaly_PFS1_pat7) + beta2_ae[1]*(qaly_PPS1_pat7) +
      beta3_ae[1]*log(cost_drug1_pat7) + beta4_ae[1]*log(cost_hos1_pat7);
leta_c_ae1_pat8 = beta0_ae[1] + beta1_ae[1]*(qaly_PFS1_pat8) + beta2_ae[1]*(qaly_PPS1_pat8) +
      beta3_ae[1]*log(cost_drug1_pat8) + beta4_ae[1]*log(cost_hos1_pat8);
leta_c_ae1_pat9 = beta0_ae[1] + beta1_ae[1]*(qaly_PFS1_pat9) + beta2_ae[1]*(qaly_PPS1_pat9) +
      beta3_ae[1]*log(cost_drug1_pat9) + beta4_ae[1]*log(cost_hos1_pat9);
leta_c_ae1_pat11 = beta0_ae[1] + beta1_ae[1]*(qaly_PFS1_pat11) + beta2_ae[1]*(qaly_PPS1_pat11) +
      beta3_ae[1]*log(cost_drug1_pat11) + beta4_ae[1]*log(cost_hos1_pat11);
leta_c_ae2_pat1 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2_pat1) + beta2_ae[2]*(qaly_PPS2_pat1) +
      beta3_ae[2]*log(cost_drug2_pat1) + beta4_ae[2]*log(cost_hos2_pat1);
leta_c_ae2_pat2 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2_pat2) + beta2_ae[2]*(qaly_PPS2_pat2) +
      beta3_ae[2]*log(cost_drug2_pat2) + beta4_ae[2]*log(cost_hos2_pat2);
leta_c_ae2_pat3 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2_pat3) + beta2_ae[2]*(qaly_PPS2_pat3) +
      beta3_ae[2]*log(cost_drug2_pat3) + beta4_ae[2]*log(cost_hos2_pat3);
leta_c_ae2_pat4 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2_pat4) + beta2_ae[2]*(qaly_PPS2_pat4) +
      beta3_ae[2]*log(cost_drug2_pat4) + beta4_ae[2]*log(cost_hos2_pat4);
leta_c_ae2_pat5 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2_pat5) + beta2_ae[2]*(qaly_PPS2_pat5) +
      beta3_ae[2]*log(cost_drug2_pat5) + beta4_ae[2]*log(cost_hos2_pat5);
leta_c_ae2_pat8 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2_pat8) + beta2_ae[2]*(qaly_PPS2_pat8) +
      beta3_ae[2]*log(cost_drug2_pat8) + beta4_ae[2]*log(cost_hos2_pat8);
leta_c_ae2_pat9 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2_pat9) + beta2_ae[2]*(qaly_PPS2_pat9) +
      beta3_ae[2]*log(cost_drug2_pat9) + beta4_ae[2]*log(cost_hos2_pat9);
leta_c_ae2_pat10 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2_pat10) + beta2_ae[2]*(qaly_PPS2_pat10) +
      beta3_ae[2]*log(cost_drug2_pat10) + beta4_ae[2]*log(cost_hos2_pat10);
leta_c_ae2_pat12 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2_pat12) + beta2_ae[2]*(qaly_PPS2_pat12) +
      beta3_ae[2]*log(cost_drug2_pat12) + beta4_ae[2]*log(cost_hos2_pat12);
}
// model block contains distributions assigned to parameters
// and the observed/missing values by pattern for each variable
model {
sigma_e_PFS ~ uniform(0,100);
alpha0_PFS ~ normal(0,1000);
alpha0_PPS ~ normal(0,1000);
alpha1_PPS ~ normal(0,1000);
gamma_PPS1 ~ normal(0,100);
gamma_PPS2 ~ normal(0,100);
beta0_drug ~ normal(0,1000);
beta1_drug ~ normal(0,1000);
beta2_drug ~ normal(0,1000);
lsigma_c_drug ~ uniform(0,100);
delta_drug1 ~ normal(0,100);
delta_drug2 ~ normal(0,100);
beta0_hos ~ normal(0,1000);
beta1_hos ~ normal(0,1000);
beta2_hos ~ normal(0,1000);
beta3_hos ~ normal(0,1000);
lsigma_c_hos ~ uniform(0,100);
delta_hos1 ~ normal(0,100);
delta_hos2 ~ normal(0,100);
beta0_ae ~ normal(0,1000);
beta1_ae ~ normal(0,1000);
beta2_ae ~ normal(0,1000);
beta3_ae ~ normal(0,1000);
beta4_ae ~ normal(0,1000);
lsigma_c_ae ~ uniform(0,100);
delta_ae1 ~ normal(0,100);
delta_ae2 ~ normal(0,100);
qaly_PFS1_pat1 ~ gumbel(eta_e_PFS[1],tau_e_PFS[1]);
qaly_PFS1_pat2 ~ gumbel(eta_e_PFS[1],tau_e_PFS[1]);
qaly_PFS1_pat4 ~ gumbel(eta_e_PFS[1],tau_e_PFS[1]);
qaly_PFS1_pat6 ~ gumbel(eta_e_PFS[1],tau_e_PFS[1]);
qaly_PFS1_pat7 ~ gumbel(eta_e_PFS[1],tau_e_PFS[1]);

```

```

qaly_PFS1_pat8 ~ gumbel(eta_e_PFS[1],tau_e_PFS[1]);
qaly_PFS1_pat9 ~ gumbel(eta_e_PFS[1],tau_e_PFS[1]);
qaly_PFS1_pat11 ~ gumbel(eta_e_PFS[1],tau_e_PFS[1]);
qaly_PFS2_pat1 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
qaly_PFS2_pat2 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
qaly_PFS2_pat3 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
qaly_PFS2_pat4 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
qaly_PFS2_pat5 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
qaly_PFS2_pat8 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
qaly_PFS2_pat9 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
qaly_PFS2_pat10 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
qaly_PFS2_pat12 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
for (n in 1:n1_pat1) {
  if (qaly_PPS1_pat1[n] == 0)
    1 ~ bernoulli_logit(X1_PPS_pat1[n]*gamma_PPS1[1] + qaly_PFS1_pat1[n]*gamma_PPS1[2]);
  else {
    0 ~ bernoulli_logit(X1_PPS_pat1[n]*gamma_PPS1[1] + qaly_PFS1_pat1[n]*gamma_PPS1[2]);
    qaly_PPS1_pat1[n] ~ exponential(lambda_e_PPS1_pat1[n]);
  }

  if (cost_drug1_pat1[n] == 0)
    1 ~ bernoulli_logit(X1_drug_pat1[n]*delta_drug1[1] + qaly_PFS1_pat1[n]*delta_drug1[2] +
      qaly_PPS1_pat1[n]*delta_drug1[3]);
  else {
    0 ~ bernoulli_logit(X1_drug_pat1[n]*delta_drug1[1] + qaly_PFS1_pat1[n]*delta_drug1[2] +
      qaly_PPS1_pat1[n]*delta_drug1[3]);
    cost_drug1_pat1[n] ~ lognormal(leta_c_drug1_pat1[n], lsigma_c_drug1[1]);
  }

  if (cost_hos1_pat1[n] == 0)
    1 ~ bernoulli_logit(X1_hos_pat1[n]*delta_hos1[1] + qaly_PFS1_pat1[n]*delta_hos1[2] +
      qaly_PPS1_pat1[n]*delta_hos1[3] + log(cost_drug1_pat1[n])*delta_hos1[4]);
  else {
    0 ~ bernoulli_logit(X1_hos_pat1[n]*delta_hos1[1] + qaly_PFS1_pat1[n]*delta_hos1[2] +
      qaly_PPS1_pat1[n]*delta_hos1[3] + log(cost_drug1_pat1[n])*delta_hos1[4]);
    cost_hos1_pat1[n] ~ lognormal(leta_c_hos1_pat1[n], lsigma_c_hos1[1]);
  }

  if (cost_ae1_pat1[n] == 0)
    1 ~ bernoulli_logit(X1_ae_pat1[n]*delta_ae1[1] + qaly_PFS1_pat1[n]*delta_ae1[2] +
      qaly_PPS1_pat1[n]*delta_ae1[3] + log(cost_drug1_pat1[n])*delta_ae1[4] +
      log(cost_hos1_pat1[n])*delta_ae1[5]);
  else {
    0 ~ bernoulli_logit(X1_ae_pat1[n]*delta_ae1[1] + qaly_PFS1_pat1[n]*delta_ae1[2] +
      qaly_PPS1_pat1[n]*delta_ae1[3] + log(cost_drug1_pat1[n])*delta_ae1[4] +
      log(cost_hos1_pat1[n])*delta_ae1[5]);
    cost_ae1_pat1[n] ~ lognormal(leta_c_ae1_pat1[n], lsigma_c_ae1[1]);
  }
}
for (n in 1:n1_pat2) {
  if (qaly_PPS1_pat2[n] == 0)
    1 ~ bernoulli_logit(X1_PPS_pat2[n]*gamma_PPS1[1] + qaly_PFS1_pat2[n]*gamma_PPS1[2]);
  else {
    0 ~ bernoulli_logit(X1_PPS_pat2[n]*gamma_PPS1[1] + qaly_PFS1_pat2[n]*gamma_PPS1[2]);
    qaly_PPS1_pat2[n] ~ exponential(lambda_e_PPS1_pat2[n]);
  }

  if (cost_drug1_pat2[n] == 0)
    1 ~ bernoulli_logit(X1_drug_pat2[n]*delta_drug1[1] + qaly_PFS1_pat2[n]*delta_drug1[2] +
      qaly_PPS1_pat2[n]*delta_drug1[3]);
  else {
    0 ~ bernoulli_logit(X1_drug_pat2[n]*delta_drug1[1] + qaly_PFS1_pat2[n]*delta_drug1[2] +
      qaly_PPS1_pat2[n]*delta_drug1[3]);
    cost_drug1_pat2[n] ~ lognormal(leta_c_drug1_pat2[n], lsigma_c_drug1[1]);
  }

  if (cost_hos1_pat2[n] == 0)
    1 ~ bernoulli_logit(X1_hos_pat2[n]*delta_hos1[1] + qaly_PFS1_pat2[n]*delta_hos1[2] +
      qaly_PPS1_pat2[n]*delta_hos1[3] + log(cost_drug1_pat2[n])*delta_hos1[4]);
  else {
    0 ~ bernoulli_logit(X1_hos_pat2[n]*delta_hos1[1] + qaly_PFS1_pat2[n]*delta_hos1[2] +
      qaly_PPS1_pat2[n]*delta_hos1[3] + log(cost_drug1_pat2[n])*delta_hos1[4]);
    cost_hos1_pat2[n] ~ lognormal(leta_c_hos1_pat2[n], lsigma_c_hos1[1]);
  }

  if (cost_ae1_pat2[n] == 0)
    1 ~ bernoulli_logit(X1_ae_pat2[n]*delta_ae1[1] + qaly_PFS1_pat2[n]*delta_ae1[2] +

```



```

        qaly_PPS1_pat2[n]*delta_ae1[3] + log(cost_drug1_pat2[n])*delta_ae1[4] +
        log(cost_hos1_pat2[n])*delta_ae1[5]);
    else {
        0 ~ bernoulli_logit(X1_ae_pat2[n]*delta_ae1[1] + qaly_PFS1_pat2[n]*delta_ae1[2] +
        qaly_PPS1_pat2[n]*delta_ae1[3] + log(cost_drug1_pat2[n])*delta_ae1[4] +
        log(cost_hos1_pat2[n])*delta_ae1[5]);
        cost_ae1_pat2[n] ~ lognormal(leta_c_ae1_pat2[n], lsigma_c_ae[1]);
    }
}
for (n in 1:n1_pat4) {
    if (qaly_PPS1_pat4[n] == 0)
        1 ~ bernoulli_logit(X1_PPS_pat4[n]*gamma_PPS1[1] + qaly_PFS1_pat4[n]*gamma_PPS1[2]);
    else {
        0 ~ bernoulli_logit(X1_PPS_pat4[n]*gamma_PPS1[1] + qaly_PFS1_pat4[n]*gamma_PPS1[2]);
        qaly_PPS1_pat4[n] ~ exponential(lambda_e_PPS1_pat4[n]);
    }
    if (cost_drug1_pat4[n] == 0)
        1 ~ bernoulli_logit(X1_drug_pat4[n]*delta_drug1[1] + qaly_PFS1_pat4[n]*delta_drug1[2] +
        qaly_PPS1_pat4[n]*delta_drug1[3]);
    else {
        0 ~ bernoulli_logit(X1_drug_pat4[n]*delta_drug1[1] + qaly_PFS1_pat4[n]*delta_drug1[2] +
        qaly_PPS1_pat4[n]*delta_drug1[3]);
        cost_drug1_pat4[n] ~ lognormal(leta_c_drug1_pat4[n], lsigma_c_drug[1]);
    }
    if (cost_hos1_pat4[n] == 0)
        1 ~ bernoulli_logit(X1_hos_pat4[n]*delta_hos1[1] + qaly_PFS1_pat4[n]*delta_hos1[2] +
        qaly_PPS1_pat4[n]*delta_hos1[3] + log(cost_drug1_pat4[n])*delta_hos1[4]);
    else {
        0 ~ bernoulli_logit(X1_hos_pat4[n]*delta_hos1[1] + qaly_PFS1_pat4[n]*delta_hos1[2] +
        qaly_PPS1_pat4[n]*delta_hos1[3] + log(cost_drug1_pat4[n])*delta_hos1[4]);
        cost_hos1_pat4[n] ~ lognormal(leta_c_hos1_pat4[n], lsigma_c_hos[1]);
    }
    if (cost_ae1_pat4[n] == 0)
        1 ~ bernoulli_logit(X1_ae_pat4[n]*delta_ae1[1] + qaly_PFS1_pat4[n]*delta_ae1[2] +
        qaly_PPS1_pat4[n]*delta_ae1[3] + log(cost_drug1_pat4[n])*delta_ae1[4] +
        log(cost_hos1_pat4[n])*delta_ae1[5]);
    else {
        0 ~ bernoulli_logit(X1_ae_pat4[n]*delta_ae1[1] + qaly_PFS1_pat4[n]*delta_ae1[2] +
        qaly_PPS1_pat4[n]*delta_ae1[3] + log(cost_drug1_pat4[n])*delta_ae1[4] +
        log(cost_hos1_pat4[n])*delta_ae1[5]);
        cost_ae1_pat4[n] ~ lognormal(leta_c_ae1_pat4[n], lsigma_c_ae[1]);
    }
}

    if (qaly_PPS1_pat6[1] == 0)
        1 ~ bernoulli_logit(X1_PPS_pat6[1]*gamma_PPS1[1] + qaly_PFS1_pat6[1]*gamma_PPS1[2]);
    else {
        0 ~ bernoulli_logit(X1_PPS_pat6[1]*gamma_PPS1[1] + qaly_PFS1_pat6[1]*gamma_PPS1[2]);
        qaly_PPS1_pat6[1] ~ exponential(lambda_e_PPS1_pat6[1]);
    }
    if (cost_drug1_pat6[1] == 0)
        1 ~ bernoulli_logit(X1_drug_pat6[1]*delta_drug1[1] + qaly_PFS1_pat6[1]*delta_drug1[2] +
        qaly_PPS1_pat6[1]*delta_drug1[3]);
    else {
        0 ~ bernoulli_logit(X1_drug_pat6[1]*delta_drug1[1] + qaly_PFS1_pat6[1]*delta_drug1[2] +
        qaly_PPS1_pat6[1]*delta_drug1[3]);
        cost_drug1_pat6[1] ~ lognormal(leta_c_drug1_pat6[1], lsigma_c_drug[1]);
    }
    if (cost_hos1_pat6[1] == 0)
        1 ~ bernoulli_logit(X1_hos_pat6[1]*delta_hos1[1] + qaly_PFS1_pat6[1]*delta_hos1[2] +
        qaly_PPS1_pat6[1]*delta_hos1[3] + log(cost_drug1_pat6[1])*delta_hos1[4]);
    else {
        0 ~ bernoulli_logit(X1_hos_pat6[1]*delta_hos1[1] + qaly_PFS1_pat6[1]*delta_hos1[2] +
        qaly_PPS1_pat6[1]*delta_hos1[3] + log(cost_drug1_pat6[1])*delta_hos1[4]);
        cost_hos1_pat6[1] ~ lognormal(leta_c_hos1_pat6[1], lsigma_c_hos[1]);
    }
    if (cost_ae1_pat6[1] == 0)
        1 ~ bernoulli_logit(X1_ae_pat6[1]*delta_ae1[1] + qaly_PFS1_pat6[1]*delta_ae1[2] +
        qaly_PPS1_pat6[1]*delta_ae1[3] + log(cost_drug1_pat6[1])*delta_ae1[4] +
        log(cost_hos1_pat6[1])*delta_ae1[5]);
    else {

```

```

0 ~ bernoulli_logit(X1_ae_pat6[1]*delta_ae1[1] + qaly_PFS1_pat6[1]*delta_ae1[2] +
  qaly_PPS1_pat6[1]*delta_ae1[3] + log(cost_drug1_pat6[1])*delta_ae1[4] +
  log(cost_hos1_pat6[1])*delta_ae1[5]);
cost_ae1_pat6[1] ~ lognormal(leta_c_ae1_pat6[1], lsigma_c_ae1[1]);
}

for (n in 1:n1_pat7) {
  if (qaly_PPS1_pat7[n] == 0)
    1 ~ bernoulli_logit(X1_PPS_pat7[n]*gamma_PPS1[1] + qaly_PFS1_pat7[n]*gamma_PPS1[2]);
  else {
    0 ~ bernoulli_logit(X1_PPS_pat7[n]*gamma_PPS1[1] + qaly_PFS1_pat7[n]*gamma_PPS1[2]);
    qaly_PPS1_pat7[n] ~ exponential(lambda_e_PPS1_pat7[n]);
  }

  if (cost_drug1_pat7[n] == 0)
    1 ~ bernoulli_logit(X1_drug_pat7[n]*delta_drug1[1] + qaly_PFS1_pat7[n]*delta_drug1[2] +
      qaly_PPS1_pat7[n]*delta_drug1[3]);
  else {
    0 ~ bernoulli_logit(X1_drug_pat7[n]*delta_drug1[1] + qaly_PFS1_pat7[n]*delta_drug1[2] +
      qaly_PPS1_pat7[n]*delta_drug1[3]);
    cost_drug1_pat7[n] ~ lognormal(leta_c_drug1_pat7[n], lsigma_c_drug1[1]);
  }

  if (cost_hos1_pat7[n] == 0)
    1 ~ bernoulli_logit(X1_hos_pat7[n]*delta_hos1[1] + qaly_PFS1_pat7[n]*delta_hos1[2] +
      qaly_PPS1_pat7[n]*delta_hos1[3] + log(cost_drug1_pat7[n])*delta_hos1[4]);
  else {
    0 ~ bernoulli_logit(X1_hos_pat7[n]*delta_hos1[1] + qaly_PFS1_pat7[n]*delta_hos1[2] +
      qaly_PPS1_pat7[n]*delta_hos1[3] + log(cost_drug1_pat7[n])*delta_hos1[4]);
    cost_hos1_pat7[n] ~ lognormal(leta_c_hos1_pat7[n], lsigma_c_hos1[1]);
  }

  if (cost_ae1_pat7[n] == 0)
    1 ~ bernoulli_logit(X1_ae_pat7[n]*delta_ae1[1] + qaly_PFS1_pat7[n]*delta_ae1[2] +
      qaly_PPS1_pat7[n]*delta_ae1[3] + log(cost_drug1_pat7[n])*delta_ae1[4] +
      log(cost_hos1_pat7[n])*delta_ae1[5]);
  else {
    0 ~ bernoulli_logit(X1_ae_pat7[n]*delta_ae1[1] + qaly_PFS1_pat7[n]*delta_ae1[2] +
      qaly_PPS1_pat7[n]*delta_ae1[3] + log(cost_drug1_pat7[n])*delta_ae1[4] +
      log(cost_hos1_pat7[n])*delta_ae1[5]);
    cost_ae1_pat7[n] ~ lognormal(leta_c_ae1_pat7[n], lsigma_c_ae1[1]);
  }
}

  if (qaly_PPS1_pat8[1] == 0)
    1 ~ bernoulli_logit(X1_PPS_pat8[1]*gamma_PPS1[1] + qaly_PFS1_pat8[1]*gamma_PPS1[2]);
  else {
    0 ~ bernoulli_logit(X1_PPS_pat8[1]*gamma_PPS1[1] + qaly_PFS1_pat8[1]*gamma_PPS1[2]);
    qaly_PPS1_pat8[1] ~ exponential(lambda_e_PPS1_pat8[1]);
  }

  if (cost_drug1_pat8[1] == 0)
    1 ~ bernoulli_logit(X1_drug_pat8[1]*delta_drug1[1] + qaly_PFS1_pat8[1]*delta_drug1[2] +
      qaly_PPS1_pat8[1]*delta_drug1[3]);
  else {
    0 ~ bernoulli_logit(X1_drug_pat8[1]*delta_drug1[1] + qaly_PFS1_pat8[1]*delta_drug1[2] +
      qaly_PPS1_pat8[1]*delta_drug1[3]);
    cost_drug1_pat8[1] ~ lognormal(leta_c_drug1_pat8[1], lsigma_c_drug1[1]);
  }

  if (cost_hos1_pat8[1] == 0)
    1 ~ bernoulli_logit(X1_hos_pat8[1]*delta_hos1[1] + qaly_PFS1_pat8[1]*delta_hos1[2] +
      qaly_PPS1_pat8[1]*delta_hos1[3] + log(cost_drug1_pat8[1])*delta_hos1[4]);
  else {
    0 ~ bernoulli_logit(X1_hos_pat8[1]*delta_hos1[1] + qaly_PFS1_pat8[1]*delta_hos1[2] +
      qaly_PPS1_pat8[1]*delta_hos1[3] + log(cost_drug1_pat8[1])*delta_hos1[4]);
    cost_hos1_pat8[1] ~ lognormal(leta_c_hos1_pat8[1], lsigma_c_hos1[1]);
  }

  if (cost_ae1_pat8[1] == 0)
    1 ~ bernoulli_logit(X1_ae_pat8[1]*delta_ae1[1] + qaly_PFS1_pat8[1]*delta_ae1[2] +
      qaly_PPS1_pat8[1]*delta_ae1[3] + log(cost_drug1_pat8[1])*delta_ae1[4] +
      log(cost_hos1_pat8[1])*delta_ae1[5]);
  else {
    0 ~ bernoulli_logit(X1_ae_pat8[1]*delta_ae1[1] + qaly_PFS1_pat8[1]*delta_ae1[2] +
      qaly_PPS1_pat8[1]*delta_ae1[3] + log(cost_drug1_pat8[1])*delta_ae1[4] +
      log(cost_hos1_pat8[1])*delta_ae1[5]);
  }
}

```

```

cost_ae1_pat8[1] ~ lognormal(leta_c_ae1_pat8[1], lsigma_c_ae[1]);
}

for (n in 1:n1_pat9) {
  if (qaly_PPS1_pat9[n] == 0)
    1 ~ bernoulli_logit(X1_PPS_pat9[n]*gamma_PPS1[1] + qaly_PFS1_pat9[n]*gamma_PPS1[2]);
  else {
    0 ~ bernoulli_logit(X1_PPS_pat9[n]*gamma_PPS1[1] + qaly_PFS1_pat9[n]*gamma_PPS1[2]);
    qaly_PPS1_pat9[n] ~ exponential(lambda_e_PPS1_pat9[n]);
  }

  if (cost_drug1_pat9[n] == 0)
    1 ~ bernoulli_logit(X1_drug_pat9[n]*delta_drug1[1] + qaly_PFS1_pat9[n]*delta_drug1[2] +
      qaly_PPS1_pat9[n]*delta_drug1[3]);
  else {
    0 ~ bernoulli_logit(X1_drug_pat9[n]*delta_drug1[1] + qaly_PFS1_pat9[n]*delta_drug1[2] +
      qaly_PPS1_pat9[n]*delta_drug1[3]);
    cost_drug1_pat9[n] ~ lognormal(leta_c_drug1_pat9[n], lsigma_c_drug[1]);
  }

  if (cost_hos1_pat9[n] == 0)
    1 ~ bernoulli_logit(X1_hos_pat9[n]*delta_hos1[1] + qaly_PFS1_pat9[n]*delta_hos1[2] +
      qaly_PPS1_pat9[n]*delta_hos1[3] + log(cost_drug1_pat9[n])*delta_hos1[4]);
  else {
    0 ~ bernoulli_logit(X1_hos_pat9[n]*delta_hos1[1] + qaly_PFS1_pat9[n]*delta_hos1[2] +
      qaly_PPS1_pat9[n]*delta_hos1[3] + log(cost_drug1_pat9[n])*delta_hos1[4]);
    cost_hos1_pat9[n] ~ lognormal(leta_c_hos1_pat9[n], lsigma_c_hos[1]);
  }

  if (cost_ae1_pat9[n] == 0)
    1 ~ bernoulli_logit(X1_ae_pat9[n]*delta_ae1[1] + qaly_PFS1_pat9[n]*delta_ae1[2] +
      qaly_PPS1_pat9[n]*delta_ae1[3] + log(cost_drug1_pat9[n])*delta_ae1[4] +
      log(cost_hos1_pat9[n])*delta_ae1[5]);
  else {
    0 ~ bernoulli_logit(X1_ae_pat9[n]*delta_ae1[1] + qaly_PFS1_pat9[n]*delta_ae1[2] +
      qaly_PPS1_pat9[n]*delta_ae1[3] + log(cost_drug1_pat9[n])*delta_ae1[4] +
      log(cost_hos1_pat9[n])*delta_ae1[5]);
    cost_ae1_pat9[n] ~ lognormal(leta_c_ae1_pat9[n], lsigma_c_ae[1]);
  }
}

  if (qaly_PPS1_pat11[1] == 0)
    1 ~ bernoulli_logit(X1_PPS_pat11[1]*gamma_PPS1[1] + qaly_PFS1_pat11[1]*gamma_PPS1[2]);
  else {
    0 ~ bernoulli_logit(X1_PPS_pat11[1]*gamma_PPS1[1] + qaly_PFS1_pat11[1]*gamma_PPS1[2]);
    qaly_PPS1_pat11[1] ~ exponential(lambda_e_PPS1_pat11[1]);
  }

  if (cost_drug1_pat11[1] == 0)
    1 ~ bernoulli_logit(X1_drug_pat11[1]*delta_drug1[1] + qaly_PFS1_pat11[1]*delta_drug1[2] +
      qaly_PPS1_pat11[1]*delta_drug1[3]);
  else {
    0 ~ bernoulli_logit(X1_drug_pat11[1]*delta_drug1[1] + qaly_PFS1_pat11[1]*delta_drug1[2] +
      qaly_PPS1_pat11[1]*delta_drug1[3]);
    cost_drug1_pat11[1] ~ lognormal(leta_c_drug1_pat11[1], lsigma_c_drug[1]);
  }

  if (cost_hos1_pat11[1] == 0)
    1 ~ bernoulli_logit(X1_hos_pat11[1]*delta_hos1[1] + qaly_PFS1_pat11[1]*delta_hos1[2] +
      qaly_PPS1_pat11[1]*delta_hos1[3] + log(cost_drug1_pat11[1])*delta_hos1[4]);
  else {
    0 ~ bernoulli_logit(X1_hos_pat11[1]*delta_hos1[1] + qaly_PFS1_pat11[1]*delta_hos1[2] +
      qaly_PPS1_pat11[1]*delta_hos1[3] + log(cost_drug1_pat11[1])*delta_hos1[4]);
    cost_hos1_pat11[1] ~ lognormal(leta_c_hos1_pat11[1], lsigma_c_hos[1]);
  }

  if (cost_ae1_pat11[1] == 0)
    1 ~ bernoulli_logit(X1_ae_pat11[1]*delta_ae1[1] + qaly_PFS1_pat11[1]*delta_ae1[2] +
      qaly_PPS1_pat11[1]*delta_ae1[3] + log(cost_drug1_pat11[1])*delta_ae1[4] +
      log(cost_hos1_pat11[1])*delta_ae1[5]);
  else {
    0 ~ bernoulli_logit(X1_ae_pat11[1]*delta_ae1[1] + qaly_PFS1_pat11[1]*delta_ae1[2] +
      qaly_PPS1_pat11[1]*delta_ae1[3] + log(cost_drug1_pat11[1])*delta_ae1[4] +
      log(cost_hos1_pat11[1])*delta_ae1[5]);
    cost_ae1_pat11[1] ~ lognormal(leta_c_ae1_pat11[1], lsigma_c_ae[1]);
  }
}

```

```

for (n in 1:n2_pat1) {
  if (qaly_PPS2_pat1[n] == 0)
    1 ~ bernoulli_logit(X2_PPS_pat1[n]*gamma_PPS2[1] + qaly_PFS2_pat1[n]*gamma_PPS2[2]);
  else {
    0 ~ bernoulli_logit(X2_PPS_pat1[n]*gamma_PPS2[1] + qaly_PFS2_pat1[n]*gamma_PPS2[2]);
    qaly_PPS2_pat1[n] ~ exponential(lambda_e_PPS2_pat1[n]);
  }
  if (cost_drug2_pat1[n] == 0)
    1 ~ bernoulli_logit(X2_drug_pat1[n]*delta_drug2[1] + qaly_PFS2_pat1[n]*delta_drug2[2] +
      qaly_PPS2_pat1[n]*delta_drug2[3]);
  else {
    0 ~ bernoulli_logit(X2_drug_pat1[n]*delta_drug2[1] + qaly_PFS2_pat1[n]*delta_drug2[2] +
      qaly_PPS2_pat1[n]*delta_drug2[3]);
    cost_drug2_pat1[n] ~ lognormal(leta_c_drug2_pat1[n], lsigma_c_drug2[2]);
  }
  if (cost_hos2_pat1[n] == 0)
    1 ~ bernoulli_logit(X2_hos_pat1[n]*delta_hos2[1] + qaly_PFS2_pat1[n]*delta_hos2[2] +
      qaly_PPS2_pat1[n]*delta_hos2[3] + log(cost_drug2_pat1[n]*delta_hos2[4]));
  else {
    0 ~ bernoulli_logit(X2_hos_pat1[n]*delta_hos2[1] + qaly_PFS2_pat1[n]*delta_hos2[2] +
      qaly_PPS2_pat1[n]*delta_hos2[3] + log(cost_drug2_pat1[n]*delta_hos2[4]));
    cost_hos2_pat1[n] ~ lognormal(leta_c_hos2_pat1[n], lsigma_c_hos2[2]);
  }
  if (cost_ae2_pat1[n] == 0)
    1 ~ bernoulli_logit(X2_ae_pat1[n]*delta_ae2[1] + qaly_PFS2_pat1[n]*delta_ae2[2] +
      qaly_PPS2_pat1[n]*delta_ae2[3] + log(cost_drug2_pat1[n]*delta_ae2[4] +
      log(cost_hos2_pat1[n]*delta_ae2[5]));
  else {
    0 ~ bernoulli_logit(X2_ae_pat1[n]*delta_ae2[1] + qaly_PFS2_pat1[n]*delta_ae2[2] +
      qaly_PPS2_pat1[n]*delta_ae2[3] + log(cost_drug2_pat1[n]*delta_ae2[4] +
      log(cost_hos2_pat1[n]*delta_ae2[5]));
    cost_ae2_pat1[n] ~ lognormal(leta_c_ae2_pat1[n], lsigma_c_ae2[2]);
  }
}
for (n in 1:n2_pat2) {
  if (qaly_PPS2_pat2[n] == 0)
    1 ~ bernoulli_logit(X2_PPS_pat2[n]*gamma_PPS2[1] + qaly_PFS2_pat2[n]*gamma_PPS2[2]);
  else {
    0 ~ bernoulli_logit(X2_PPS_pat2[n]*gamma_PPS2[1] + qaly_PFS2_pat2[n]*gamma_PPS2[2]);
    qaly_PPS2_pat2[n] ~ exponential(lambda_e_PPS2_pat2[n]);
  }
  if (cost_drug2_pat2[n] == 0)
    1 ~ bernoulli_logit(X2_drug_pat2[n]*delta_drug2[1] + qaly_PFS2_pat2[n]*delta_drug2[2] +
      qaly_PPS2_pat2[n]*delta_drug2[3]);
  else {
    0 ~ bernoulli_logit(X2_drug_pat2[n]*delta_drug2[1] + qaly_PFS2_pat2[n]*delta_drug2[2] +
      qaly_PPS2_pat2[n]*delta_drug2[3]);
    cost_drug2_pat2[n] ~ lognormal(leta_c_drug2_pat2[n], lsigma_c_drug2[2]);
  }
  if (cost_hos2_pat2[n] == 0)
    1 ~ bernoulli_logit(X2_hos_pat2[n]*delta_hos2[1] + qaly_PFS2_pat2[n]*delta_hos2[2] +
      qaly_PPS2_pat2[n]*delta_hos2[3] + log(cost_drug2_pat2[n]*delta_hos2[4]));
  else {
    0 ~ bernoulli_logit(X2_hos_pat2[n]*delta_hos2[1] + qaly_PFS2_pat2[n]*delta_hos2[2] +
      qaly_PPS2_pat2[n]*delta_hos2[3] + log(cost_drug2_pat2[n]*delta_hos2[4]));
    cost_hos2_pat2[n] ~ lognormal(leta_c_hos2_pat2[n], lsigma_c_hos2[2]);
  }
  if (cost_ae2_pat2[n] == 0)
    1 ~ bernoulli_logit(X2_ae_pat2[n]*delta_ae2[1] + qaly_PFS2_pat2[n]*delta_ae2[2] +
      qaly_PPS2_pat2[n]*delta_ae2[3] + log(cost_drug2_pat2[n]*delta_ae2[4] +
      log(cost_hos2_pat2[n]*delta_ae2[5]));
  else {
    0 ~ bernoulli_logit(X2_ae_pat2[n]*delta_ae2[1] + qaly_PFS2_pat2[n]*delta_ae2[2] +
      qaly_PPS2_pat2[n]*delta_ae2[3] + log(cost_drug2_pat2[n]*delta_ae2[4] +
      log(cost_hos2_pat2[n]*delta_ae2[5]));
    cost_ae2_pat2[n] ~ lognormal(leta_c_ae2_pat2[n], lsigma_c_ae2[2]);
  }
}
for (n in 1:n2_pat3) {
  if (qaly_PPS2_pat3[n] == 0)
    1 ~ bernoulli_logit(X2_PPS_pat3[n]*gamma_PPS2[1] + qaly_PFS2_pat3[n]*gamma_PPS2[2]);

```

```

else {
  0 ~ bernoulli_logit(X2_PPS_pat3[n]*gamma_PPS2[1] + qaly_PFS2_pat3[n]*gamma_PPS2[2]);
  qaly_PPS2_pat3[n] ~ exponential(lambda_e_PPS2_pat3[n]);
}
if (cost_drug2_pat3[n] == 0)
  1 ~ bernoulli_logit(X2_drug_pat3[n]*delta_drug2[1] + qaly_PFS2_pat3[n]*delta_drug2[2] +
    qaly_PPS2_pat3[n]*delta_drug2[3]);
else {
  0 ~ bernoulli_logit(X2_drug_pat3[n]*delta_drug2[1] + qaly_PFS2_pat3[n]*delta_drug2[2] +
    qaly_PPS2_pat3[n]*delta_drug2[3]);
  cost_drug2_pat3[n] ~ lognormal(leta_c_drug2_pat3[n], lsigma_c_drug2[2]);
}
if (cost_hos2_pat3[n] == 0)
  1 ~ bernoulli_logit(X2_hos_pat3[n]*delta_hos2[1] + qaly_PFS2_pat3[n]*delta_hos2[2] +
    qaly_PPS2_pat3[n]*delta_hos2[3] + log(cost_drug2_pat3[n]*delta_hos2[4]));
else {
  0 ~ bernoulli_logit(X2_hos_pat3[n]*delta_hos2[1] + qaly_PFS2_pat3[n]*delta_hos2[2] +
    qaly_PPS2_pat3[n]*delta_hos2[3] + log(cost_drug2_pat3[n]*delta_hos2[4]));
  cost_hos2_pat3[n] ~ lognormal(leta_c_hos2_pat3[n], lsigma_c_hos2[2]);
}
if (cost_ae2_pat3[n] == 0)
  1 ~ bernoulli_logit(X2_ae_pat3[n]*delta_ae2[1] + qaly_PFS2_pat3[n]*delta_ae2[2] +
    qaly_PPS2_pat3[n]*delta_ae2[3] + log(cost_drug2_pat3[n]*delta_ae2[4] +
    log(cost_hos2_pat3[n]*delta_ae2[5]));
else {
  0 ~ bernoulli_logit(X2_ae_pat3[n]*delta_ae2[1] + qaly_PFS2_pat3[n]*delta_ae2[2] +
    qaly_PPS2_pat3[n]*delta_ae2[3] + log(cost_drug2_pat3[n]*delta_ae2[4] +
    log(cost_hos2_pat3[n]*delta_ae2[5]));
  cost_ae2_pat3[n] ~ lognormal(leta_c_ae2_pat3[n], lsigma_c_ae2[2]);
}
}
for (n in 1:n2_pat4) {
  if (qaly_PPS2_pat4[n] == 0)
    1 ~ bernoulli_logit(X2_PPS_pat4[n]*gamma_PPS2[1] + qaly_PFS2_pat4[n]*gamma_PPS2[2]);
  else {
    0 ~ bernoulli_logit(X2_PPS_pat4[n]*gamma_PPS2[1] + qaly_PFS2_pat4[n]*gamma_PPS2[2]);
    qaly_PPS2_pat4[n] ~ exponential(lambda_e_PPS2_pat4[n]);
  }
  if (cost_drug2_pat4[n] == 0)
    1 ~ bernoulli_logit(X2_drug_pat4[n]*delta_drug2[1] + qaly_PFS2_pat4[n]*delta_drug2[2] +
      qaly_PPS2_pat4[n]*delta_drug2[3]);
  else {
    0 ~ bernoulli_logit(X2_drug_pat4[n]*delta_drug2[1] + qaly_PFS2_pat4[n]*delta_drug2[2] +
      qaly_PPS2_pat4[n]*delta_drug2[3]);
    cost_drug2_pat4[n] ~ lognormal(leta_c_drug2_pat4[n], lsigma_c_drug2[2]);
  }
  if (cost_hos2_pat4[n] == 0)
    1 ~ bernoulli_logit(X2_hos_pat4[n]*delta_hos2[1] + qaly_PFS2_pat4[n]*delta_hos2[2] +
      qaly_PPS2_pat4[n]*delta_hos2[3] + log(cost_drug2_pat4[n]*delta_hos2[4]));
  else {
    0 ~ bernoulli_logit(X2_hos_pat4[n]*delta_hos2[1] + qaly_PFS2_pat4[n]*delta_hos2[2] +
      qaly_PPS2_pat4[n]*delta_hos2[3] + log(cost_drug2_pat4[n]*delta_hos2[4]));
    cost_hos2_pat4[n] ~ lognormal(leta_c_hos2_pat4[n], lsigma_c_hos2[2]);
  }
  if (cost_ae2_pat4[n] == 0)
    1 ~ bernoulli_logit(X2_ae_pat4[n]*delta_ae2[1] + qaly_PFS2_pat4[n]*delta_ae2[2] +
      qaly_PPS2_pat4[n]*delta_ae2[3] + log(cost_drug2_pat4[n]*delta_ae2[4] +
      log(cost_hos2_pat4[n]*delta_ae2[5]));
  else {
    0 ~ bernoulli_logit(X2_ae_pat4[n]*delta_ae2[1] + qaly_PFS2_pat4[n]*delta_ae2[2] +
      qaly_PPS2_pat4[n]*delta_ae2[3] + log(cost_drug2_pat4[n]*delta_ae2[4] +
      log(cost_hos2_pat4[n]*delta_ae2[5]));
    cost_ae2_pat4[n] ~ lognormal(leta_c_ae2_pat4[n], lsigma_c_ae2[2]);
  }
}
}
if (qaly_PPS2_pat5[1] == 0)
  1 ~ bernoulli_logit(X2_PPS_pat5[1]*gamma_PPS2[1] + qaly_PFS2_pat5[1]*gamma_PPS2[2]);
else {
  0 ~ bernoulli_logit(X2_PPS_pat5[1]*gamma_PPS2[1] + qaly_PFS2_pat5[1]*gamma_PPS2[2]);
  qaly_PPS2_pat5[1] ~ exponential(lambda_e_PPS2_pat5[1]);
}

```

```

}
if (cost_drug2_pat5[1] == 0)
  1 ~ bernoulli_logit(X2_drug_pat5[1]*delta_drug2[1] + qaly_PFS2_pat5[1]*delta_drug2[2] +
    qaly_PPS2_pat5[1]*delta_drug2[3]);
else {
  0 ~ bernoulli_logit(X2_drug_pat5[1]*delta_drug2[1] + qaly_PFS2_pat5[1]*delta_drug2[2] +
    qaly_PPS2_pat5[1]*delta_drug2[3]);
  cost_drug2_pat5[1] ~ lognormal(leta_c_drug2_pat5[1], lsigma_c_drug2[2]);
}

if (cost_hos2_pat5[1] == 0)
  1 ~ bernoulli_logit(X2_hos_pat5[1]*delta_hos2[1] + qaly_PFS2_pat5[1]*delta_hos2[2] +
    qaly_PPS2_pat5[1]*delta_hos2[3] + log(cost_drug2_pat5[1])*delta_hos2[4]);
else {
  0 ~ bernoulli_logit(X2_hos_pat5[1]*delta_hos2[1] + qaly_PFS2_pat5[1]*delta_hos2[2] +
    qaly_PPS2_pat5[1]*delta_hos2[3] + log(cost_drug2_pat5[1])*delta_hos2[4]);
  cost_hos2_pat5[1] ~ lognormal(leta_c_hos2_pat5[1], lsigma_c_hos2[2]);
}

if (cost_ae2_pat5[1] == 0)
  1 ~ bernoulli_logit(X2_ae_pat5[1]*delta_ae2[1] + qaly_PFS2_pat5[1]*delta_ae2[2] +
    qaly_PPS2_pat5[1]*delta_ae2[3] + log(cost_drug2_pat5[1])*delta_ae2[4] +
    log(cost_hos2_pat5[1])*delta_ae2[5]);
else {
  0 ~ bernoulli_logit(X2_ae_pat5[1]*delta_ae2[1] + qaly_PFS2_pat5[1]*delta_ae2[2] +
    qaly_PPS2_pat5[1]*delta_ae2[3] + log(cost_drug2_pat5[1])*delta_ae2[4] +
    log(cost_hos2_pat5[1])*delta_ae2[5]);
  cost_ae2_pat5[1] ~ lognormal(leta_c_ae2_pat5[1], lsigma_c_ae2[2]);
}

for (n in 1:n2_pat8) {
  if (qaly_PPS2_pat8[n] == 0)
    1 ~ bernoulli_logit(X2_PPS_pat8[n]*gamma_PPS2[1] + qaly_PFS2_pat8[n]*gamma_PPS2[2]);
  else {
    0 ~ bernoulli_logit(X2_PPS_pat8[n]*gamma_PPS2[1] + qaly_PFS2_pat8[n]*gamma_PPS2[2]);
    qaly_PPS2_pat8[n] ~ exponential(lambda_e_PPS2_pat8[n]);
  }

  if (cost_drug2_pat8[n] == 0)
    1 ~ bernoulli_logit(X2_drug_pat8[n]*delta_drug2[1] + qaly_PFS2_pat8[n]*delta_drug2[2] +
      qaly_PPS2_pat8[n]*delta_drug2[3]);
  else {
    0 ~ bernoulli_logit(X2_drug_pat8[n]*delta_drug2[1] + qaly_PFS2_pat8[n]*delta_drug2[2] +
      qaly_PPS2_pat8[n]*delta_drug2[3]);
    cost_drug2_pat8[n] ~ lognormal(leta_c_drug2_pat8[n], lsigma_c_drug2[2]);
  }

  if (cost_hos2_pat8[n] == 0)
    1 ~ bernoulli_logit(X2_hos_pat8[n]*delta_hos2[1] + qaly_PFS2_pat8[n]*delta_hos2[2] +
      qaly_PPS2_pat8[n]*delta_hos2[3] + log(cost_drug2_pat8[n])*delta_hos2[4]);
  else {
    0 ~ bernoulli_logit(X2_hos_pat8[n]*delta_hos2[1] + qaly_PFS2_pat8[n]*delta_hos2[2] +
      qaly_PPS2_pat8[n]*delta_hos2[3] + log(cost_drug2_pat8[n])*delta_hos2[4]);
    cost_hos2_pat8[n] ~ lognormal(leta_c_hos2_pat8[n], lsigma_c_hos2[2]);
  }

  if (cost_ae2_pat8[n] == 0)
    1 ~ bernoulli_logit(X2_ae_pat8[n]*delta_ae2[1] + qaly_PFS2_pat8[n]*delta_ae2[2] +
      qaly_PPS2_pat8[n]*delta_ae2[3] + log(cost_drug2_pat8[n])*delta_ae2[4] +
      log(cost_hos2_pat8[n])*delta_ae2[5]);
  else {
    0 ~ bernoulli_logit(X2_ae_pat8[n]*delta_ae2[1] + qaly_PFS2_pat8[n]*delta_ae2[2] +
      qaly_PPS2_pat8[n]*delta_ae2[3] + log(cost_drug2_pat8[n])*delta_ae2[4] +
      log(cost_hos2_pat8[n])*delta_ae2[5]);
    cost_ae2_pat8[n] ~ lognormal(leta_c_ae2_pat8[n], lsigma_c_ae2[2]);
  }
}

for (n in 1:n2_pat9) {
  if (qaly_PPS2_pat9[n] == 0)
    1 ~ bernoulli_logit(X2_PPS_pat9[n]*gamma_PPS2[1] + qaly_PFS2_pat9[n]*gamma_PPS2[2]);
  else {
    0 ~ bernoulli_logit(X2_PPS_pat9[n]*gamma_PPS2[1] + qaly_PFS2_pat9[n]*gamma_PPS2[2]);
    qaly_PPS2_pat9[n] ~ exponential(lambda_e_PPS2_pat9[n]);
  }

  if (cost_drug2_pat9[n] == 0)
    1 ~ bernoulli_logit(X2_drug_pat9[n]*delta_drug2[1] + qaly_PFS2_pat9[n]*delta_drug2[2] +

```

```

        qaly_PPS2_pat9[n]*delta_drug2[3]);
else {
  0 ~ bernoulli_logit(X2_drug_pat9[n]*delta_drug2[1] + qaly_PFS2_pat9[n]*delta_drug2[2] +
    qaly_PPS2_pat9[n]*delta_drug2[3]);
  cost_drug2_pat9[n] ~ lognormal(leta_c_drug2_pat9[n], lsigma_c_drug[2]);
}

  if (cost_hos2_pat9[n] == 0)
  1 ~ bernoulli_logit(X2_hos_pat9[n]*delta_hos2[1] + qaly_PFS2_pat9[n]*delta_hos2[2] +
    qaly_PPS2_pat9[n]*delta_hos2[3] + log(cost_drug2_pat9[n]*delta_hos2[4]));
else {
  0 ~ bernoulli_logit(X2_hos_pat9[n]*delta_hos2[1] + qaly_PFS2_pat9[n]*delta_hos2[2] +
    qaly_PPS2_pat9[n]*delta_hos2[3] + log(cost_drug2_pat9[n]*delta_hos2[4]));
  cost_hos2_pat9[n] ~ lognormal(leta_c_hos2_pat9[n], lsigma_c_hos[2]);
}

  if (cost_ae2_pat9[n] == 0)
  1 ~ bernoulli_logit(X2_ae_pat9[n]*delta_ae2[1] + qaly_PFS2_pat9[n]*delta_ae2[2] +
    qaly_PPS2_pat9[n]*delta_ae2[3] + log(cost_drug2_pat9[n]*delta_ae2[4] +
    log(cost_hos2_pat9[n]*delta_ae2[5]));
else {
  0 ~ bernoulli_logit(X2_ae_pat9[n]*delta_ae2[1] + qaly_PFS2_pat9[n]*delta_ae2[2] +
    qaly_PPS2_pat9[n]*delta_ae2[3] + log(cost_drug2_pat9[n]*delta_ae2[4] +
    log(cost_hos2_pat9[n]*delta_ae2[5]));
  cost_ae2_pat9[n] ~ lognormal(leta_c_ae2_pat9[n], lsigma_c_ae[2]);
}
}
for (n in 1:n2_pat10) {
  if (qaly_PPS2_pat10[n] == 0)
  1 ~ bernoulli_logit(X2_PPS_pat10[n]*gamma_PPS2[1] + qaly_PFS2_pat10[n]*gamma_PPS2[2]);
else {
  0 ~ bernoulli_logit(X2_PPS_pat10[n]*gamma_PPS2[1] + qaly_PFS2_pat10[n]*gamma_PPS2[2]);
  qaly_PPS2_pat10[n] ~ exponential(lambda_e_PPS2_pat10[n]);
}

  if (cost_drug2_pat10[n] == 0)
  1 ~ bernoulli_logit(X2_drug_pat10[n]*delta_drug2[1] + qaly_PFS2_pat10[n]*delta_drug2[2] +
    qaly_PPS2_pat10[n]*delta_drug2[3]);
else {
  0 ~ bernoulli_logit(X2_drug_pat10[n]*delta_drug2[1] + qaly_PFS2_pat10[n]*delta_drug2[2] +
    qaly_PPS2_pat10[n]*delta_drug2[3]);
  cost_drug2_pat10[n] ~ lognormal(leta_c_drug2_pat10[n], lsigma_c_drug[2]);
}

  if (cost_hos2_pat10[n] == 0)
  1 ~ bernoulli_logit(X2_hos_pat10[n]*delta_hos2[1] + qaly_PFS2_pat10[n]*delta_hos2[2] +
    qaly_PPS2_pat10[n]*delta_hos2[3] + log(cost_drug2_pat10[n]*delta_hos2[4]));
else {
  0 ~ bernoulli_logit(X2_hos_pat10[n]*delta_hos2[1] + qaly_PFS2_pat10[n]*delta_hos2[2] +
    qaly_PPS2_pat10[n]*delta_hos2[3] + log(cost_drug2_pat10[n]*delta_hos2[4]));
  cost_hos2_pat10[n] ~ lognormal(leta_c_hos2_pat10[n], lsigma_c_hos[2]);
}

  if (cost_ae2_pat10[n] == 0)
  1 ~ bernoulli_logit(X2_ae_pat10[n]*delta_ae2[1] + qaly_PFS2_pat10[n]*delta_ae2[2] +
    qaly_PPS2_pat10[n]*delta_ae2[3] + log(cost_drug2_pat10[n]*delta_ae2[4] +
    log(cost_hos2_pat10[n]*delta_ae2[5]));
else {
  0 ~ bernoulli_logit(X2_ae_pat10[n]*delta_ae2[1] + qaly_PFS2_pat10[n]*delta_ae2[2] +
    qaly_PPS2_pat10[n]*delta_ae2[3] + log(cost_drug2_pat10[n]*delta_ae2[4] +
    log(cost_hos2_pat10[n]*delta_ae2[5]));
  cost_ae2_pat10[n] ~ lognormal(leta_c_ae2_pat10[n], lsigma_c_ae[2]);
}
}
for (n in 1:n2_pat12) {
  if (qaly_PPS2_pat12[n] == 0)
  1 ~ bernoulli_logit(X2_PPS_pat12[n]*gamma_PPS2[1] + qaly_PFS2_pat12[n]*gamma_PPS2[2]);
else {
  0 ~ bernoulli_logit(X2_PPS_pat12[n]*gamma_PPS2[1] + qaly_PFS2_pat12[n]*gamma_PPS2[2]);
  qaly_PPS2_pat12[n] ~ exponential(lambda_e_PPS2_pat12[n]);
}

  if (cost_drug2_pat12[n] == 0)
  1 ~ bernoulli_logit(X2_drug_pat12[n]*delta_drug2[1] + qaly_PFS2_pat12[n]*delta_drug2[2] +
    qaly_PPS2_pat12[n]*delta_drug2[3]);
else {
  0 ~ bernoulli_logit(X2_drug_pat12[n]*delta_drug2[1] + qaly_PFS2_pat12[n]*delta_drug2[2] +

```

```

        qaly_PPS2_pat12[n]*delta_drug2[3]);
cost_drug2_pat12[n] ~ lognormal(leta_c_drug2_pat12[n], lsigma_c_drug[2]);
}
    if (cost_hos2_pat12[n] == 0)
    1 ~ bernoulli_logit(X2_hos_pat12[n]*delta_hos2[1] + qaly_PFS2_pat12[n]*delta_hos2[2] +
        qaly_PPS2_pat12[n]*delta_hos2[3] + log(cost_drug2_pat12[n])*delta_hos2[4]);
else {
    0 ~ bernoulli_logit(X2_hos_pat12[n]*delta_hos2[1] + qaly_PFS2_pat12[n]*delta_hos2[2] +
        qaly_PPS2_pat12[n]*delta_hos2[3] + log(cost_drug2_pat12[n])*delta_hos2[4]);
    cost_hos2_pat12[n] ~ lognormal(leta_c_hos2_pat12[n], lsigma_c_hos[2]);
}
    if (cost_ae2_pat12[n] == 0)
    1 ~ bernoulli_logit(X2_ae_pat12[n]*delta_ae2[1] + qaly_PFS2_pat12[n]*delta_ae2[2] +
        qaly_PPS2_pat12[n]*delta_ae2[3] + log(cost_drug2_pat12[n])*delta_ae2[4] +
        log(cost_hos2_pat12[n])*delta_ae2[5]);
else {
    0 ~ bernoulli_logit(X2_ae_pat12[n]*delta_ae2[1] + qaly_PFS2_pat12[n]*delta_ae2[2] +
        qaly_PPS2_pat12[n]*delta_ae2[3] + log(cost_drug2_pat12[n])*delta_ae2[4] +
        log(cost_hos2_pat12[n])*delta_ae2[5]);
    cost_ae2_pat12[n] ~ lognormal(leta_c_ae2_pat12[n], lsigma_c_ae[2]);
}
}
}
}

```


1.2 STAN code - complete cases

```
data {
  int<lower=0> n1;
  int<lower=0> n2;
  int<lower=0> n_pred_pps1;
  int<lower=0> n_pred_pps2;
  int<lower=0> n_pred_drug1;
  int<lower=0> n_pred_drug2;
  int<lower=0> n_pred_hos1;
  int<lower=0> n_pred_hos2;
  int<lower=0> n_pred_ae1;
  int<lower=0> n_pred_ae2;
  vector[n1] qaly_PFS1;
  vector[n2] qaly_PFS2;
  vector[n1] qaly_PPS1;
  vector[n2] qaly_PPS2;
  vector[n1] cost_drug1;
  vector[n2] cost_drug2;
  vector[n1] cost_hos1;
  vector[n2] cost_hos2;
  vector[n1] cost_ae1;
  vector[n2] cost_ae2;
  matrix[n1,n_pred_pps1] X1_PPS;
  matrix[n2,n_pred_pps2] X2_PPS;
  matrix[n1,n_pred_drug1] X1_drug;
  matrix[n2,n_pred_drug2] X2_drug;
  matrix[n1,n_pred_hos1] X1_hos;
  matrix[n2,n_pred_hos2] X2_hos;
  matrix[n1,n_pred_ae1] X1_ae;
  matrix[n2,n_pred_ae2] X2_ae;
}
parameters {
  real alpha0_PFS[2];
  real alpha0_PPS[2];
  real alpha1_PPS[2];
  real<lower=0> sigma_e_PFS[2];
  real beta0_drug[2];
  real beta1_drug[2];
  real beta2_drug[2];
  real<lower=0> lsigma_c_drug[2];
  real beta0_hos[2];
  real beta1_hos[2];
  real beta2_hos[2];
  real beta3_hos[2];
  real<lower=0> lsigma_c_hos[2];
  real beta0_ae[2];
  real beta1_ae[2];
  real beta2_ae[2];
  real beta3_ae[2];
  real beta4_ae[2];
  real<lower=0> lsigma_c_ae[2];
  vector[n_pred_pps1] gamma_PPS1;
  vector[n_pred_pps2] gamma_PPS2;
  vector[n_pred_drug1] delta_drug1;
  vector[n_pred_drug2] delta_drug2;
  vector[n_pred_hos1] delta_hos1;
  vector[n_pred_hos2] delta_hos2;
  vector[n_pred_ae1] delta_ae1;
  vector[n_pred_ae2] delta_ae2;
}
transformed parameters {
  real<lower=0> tau_e_PFS[2];
  real eta_e_PFS[2];
  vector<lower=0>[n1] eta_e_PPS1;
  vector<lower=0>[n2] eta_e_PPS2;
  vector<lower=0>[n1] lambda_e_PPS1;
  vector<lower=0>[n2] lambda_e_PPS2;
  vector[n1] leta_c_drug1;
  vector[n2] leta_c_drug2;
  vector[n1] leta_c_hos1;
```

```

vector[n2] leta_c_hos2;
vector[n1] leta_c_ae1;
vector[n2] leta_c_ae2;
tau_e_PFS[1] = (sigma_e_PFS[1]*sqrt(6))/pi();
tau_e_PFS[2] = (sigma_e_PFS[2]*sqrt(6))/pi();
eta_e_PFS[1] = alpha0_PFS[1] - tau_e_PFS[1]*0.5722;
eta_e_PFS[2] = alpha0_PFS[2] - tau_e_PFS[2]*0.5722;
eta_e_PPS1 = exp(alpha0_PPS[1] + alpha1_PPS[1]*(qaly_PFS1));
eta_e_PPS2 = exp(alpha0_PPS[2] + alpha1_PPS[2]*(qaly_PFS2));
  for (n in 1:n1) {
    lambda_e_PPS1[n] = 1/eta_e_PPS1[n];
  }
  for (n in 1:n2) {
    lambda_e_PPS2[n] = 1/eta_e_PPS2[n];
  }
leta_c_drug1 = beta0_drug[1] + beta1_drug[1]*(qaly_PFS1) + beta2_drug[1]*(qaly_PPS1);
leta_c_drug2 = beta0_drug[2] + beta1_drug[2]*(qaly_PFS2) + beta2_drug[2]*(qaly_PPS2);
leta_c_hos1 = beta0_hos[1] + beta1_hos[1]*(qaly_PFS1) + beta2_hos[1]*(qaly_PPS1) + beta3_hos[1]*log(cost_drug1+0.01);
leta_c_hos2 = beta0_hos[2] + beta1_hos[2]*(qaly_PFS2) + beta2_hos[2]*(qaly_PPS2) + beta3_hos[2]*log(cost_drug2+0.01);
leta_c_ae1 = beta0_ae[1] + beta1_ae[1]*(qaly_PFS1) + beta2_ae[1]*(qaly_PPS1) + beta3_ae[1]*log(cost_drug1+0.01) + beta4_ae[1]*log(cost_
leta_c_ae2 = beta0_ae[2] + beta1_ae[2]*(qaly_PFS2) + beta2_ae[2]*(qaly_PPS2) + beta3_ae[2]*log(cost_drug2+0.01) + beta4_ae[2]*log(cost_
)
model {
sigma_e_PFS ~ uniform(0,100);
alpha0_PFS ~ normal(0,1000);
alpha0_PPS ~ normal(0,1000);
alpha1_PPS ~ normal(0,1000);
beta0_drug ~ normal(0,1000);
beta1_drug ~ normal(0,1000);
beta2_drug ~ normal(0,1000);
beta0_hos ~ normal(0,1000);
beta1_hos ~ normal(0,1000);
beta2_hos ~ normal(0,1000);
beta3_hos ~ normal(0,1000);
beta0_ae ~ normal(0,1000);
beta1_ae ~ normal(0,1000);
beta2_ae ~ normal(0,1000);
beta3_ae ~ normal(0,1000);
beta4_ae ~ normal(0,1000);
gamma_PPS1 ~ normal(0,100);
gamma_PPS2 ~ normal(0,100);
delta_drug1 ~ normal(0,100);
delta_hos1 ~ normal(0,100);
delta_ae1 ~ normal(0,100);
delta_drug2 ~ normal(0,100);
delta_hos2 ~ normal(0,100);
delta_ae2 ~ normal(0,100);
lsigma_c_drug ~ uniform(0,100);
lsigma_c_hos ~ uniform(0,100);
lsigma_c_ae ~ uniform(0,100);
qaly_PFS1 ~ gumbel(eta_e_PFS[1],tau_e_PFS[1]);
qaly_PFS2 ~ gumbel(eta_e_PFS[2],tau_e_PFS[2]);
  for (n in 1:n1) {
    if (qaly_PPS1[n] == 0)
      1 ~ bernoulli_logit(X1_PPS[n]*gamma_PPS1);
    else {
      0 ~ bernoulli_logit(X1_PPS[n]*gamma_PPS1);
      qaly_PPS1[n] ~ exponential(lambda_e_PPS1[n]);
    }
    if (cost_drug1[n] == 0)
      1 ~ bernoulli_logit(X1_drug[n]*delta_drug1);
    else {
      0 ~ bernoulli_logit(X1_drug[n]*delta_drug1);
      cost_drug1[n] ~ lognormal(leta_c_drug1[n], lsigma_c_drug[1]);
    }
    if (cost_hos1[n] == 0)
      1 ~ bernoulli_logit(X1_hos[n]*delta_hos1);
    else {
      0 ~ bernoulli_logit(X1_hos[n]*delta_hos1);
      cost_hos1[n] ~ lognormal(leta_c_hos1[n], lsigma_c_hos[1]);
    }
  }
}

```

```

    if (cost_ae1[n] == 0)
      1 ~ bernoulli_logit(X1_ae[n]*delta_ae1);
    else {
      0 ~ bernoulli_logit(X1_ae[n]*delta_ae1);
      cost_ae1[n] ~ lognormal(leta_c_ae1[n], lsigma_c_ae1[1]);
    }
  }
}
for (n in 1:n2) {
  if (qaly_PPS2[n] == 0)
    1 ~ bernoulli_logit(X2_PPS[n]*gamma_PPS2);
  else {
    0 ~ bernoulli_logit(X2_PPS[n]*gamma_PPS2);
    qaly_PPS2[n] ~ exponential(lambda_e_PPS2[n]);
  }
  if (cost_drug2[n] == 0)
    1 ~ bernoulli_logit(X2_drug[n]*delta_drug2);
  else {
    0 ~ bernoulli_logit(X2_drug[n]*delta_drug2);
    cost_drug2[n] ~ lognormal(leta_c_drug2[n], lsigma_c_drug[2]);
  }
  if (cost_hos2[n] == 0)
    1 ~ bernoulli_logit(X2_hos[n]*delta_hos2);
  else {
    0 ~ bernoulli_logit(X2_hos[n]*delta_hos2);
    cost_hos2[n] ~ lognormal(leta_c_hos2[n], lsigma_c_hos[2]);
  }
  if (cost_ae2[n] == 0)
    1 ~ bernoulli_logit(X2_ae[n]*delta_ae2);
  else {
    0 ~ bernoulli_logit(X2_ae[n]*delta_ae2);
    cost_ae2[n] ~ lognormal(leta_c_ae2[n], lsigma_c_ae[2]);
  }
}
}
generated quantities {
vector[n1] log_lik_e_PFS1;
vector[n2] log_lik_e_PFS2;
vector[n1] log_lik_e_PPS1;
vector[n2] log_lik_e_PPS2;
vector[n1] log_lik_c_drug1;
vector[n2] log_lik_c_drug2;
vector[n1] log_lik_c_hos1;
vector[n2] log_lik_c_hos2;
vector[n1] log_lik_c_ae1;
vector[n2] log_lik_c_ae2;
for (n in 1:n1) {
  log_lik_e_PFS1[n] = gumbel_lpdf(qaly_PFS1[n] | eta_e_PFS[1], tau_e_PFS[1]);
  if (qaly_PPS1[n] == 0)
    log_lik_e_PPS1[n] = bernoulli_logit_lpmf(1 | X1_PPS[n]*gamma_PPS1);
  else {
    log_lik_e_PPS1[n] = exponential_lpdf(qaly_PPS1[n] | lambda_e_PPS1[n]);
  }
  if (cost_drug1[n] == 0)
    log_lik_c_drug1[n] = bernoulli_logit_lpmf(1 | X1_drug[n]*delta_drug1);
  else {
    log_lik_c_drug1[n] = lognormal_lpdf(cost_drug1[n] | leta_c_drug1[n], lsigma_c_drug[1]);
  }
  if (cost_hos1[n] == 0)
    log_lik_c_hos1[n] = bernoulli_logit_lpmf(1 | X1_hos[n]*delta_hos1);
  else {
    log_lik_c_hos1[n] = lognormal_lpdf(cost_hos1[n] | leta_c_hos1[n], lsigma_c_hos[1]);
  }
  if (cost_ae1[n] == 0)
    log_lik_c_ae1[n] = bernoulli_logit_lpmf(1 | X1_ae[n]*delta_ae1);
  else {
    log_lik_c_ae1[n] = lognormal_lpdf(cost_ae1[n] | leta_c_ae1[n], lsigma_c_ae[1]);
  }
}
}
for (n in 1:n2) {
  log_lik_e_PFS2[n] = gumbel_lpdf(qaly_PFS2[n] | eta_e_PFS[2], tau_e_PFS[2]);
  if (qaly_PPS2[n] == 0)

```

```

    log_lik_e_PPS2[n] = bernoulli_logit_lpmf(1 | X2_PPS[n]*gamma_PPS2);
else {
    log_lik_e_PPS2[n] = exponential_lpdf(qaly_PPS2[n] | lambda_e_PPS2[n]);
}
    if (cost_drug2[n] == 0)
        log_lik_c_drug2[n] = bernoulli_logit_lpmf(1 | X2_drug[n]*delta_drug2);
    else {
        log_lik_c_drug2[n] = lognormal_lpdf(cost_drug2[n] | leta_c_drug2[n], lsigma_c_drug2[2]);
    }
    if (cost_hos2[n] == 0)
        log_lik_c_hos2[n] = bernoulli_logit_lpmf(1 | X2_hos[n]*delta_hos2);
    else {
        log_lik_c_hos2[n] = lognormal_lpdf(cost_hos2[n] | leta_c_hos2[n], lsigma_c_hos2[2]);
    }
    if (cost_ae2[n] == 0)
        log_lik_c_ae2[n] = bernoulli_logit_lpmf(1 | X2_ae[n]*delta_ae2);
    else {
        log_lik_c_ae2[n] = lognormal_lpdf(cost_ae2[n] | leta_c_ae2[n], lsigma_c_ae2[2]);
    }
}
}
}

```