

S1 Appendix:

Slope of CD4 count models:

Slope model 1: primary cumulative log viral load model

```
model{
  for( i in 1 : N ) {
    for( j in 1 : M ) {
      CD4[i , j] ~ dnorm(mu[i , j], tau.int)
      mu[i , j] <- beta[1] + beta[2] * sex.f[i] +
beta[3] * age_count[i] + beta[4] * base.cd4[i] + beta[5] *
base.logvl[i] + beta[6] * (yrs.fu[j] - 2.5) + beta[7] *
pow(yrs.fu[j] - 2.5, 2) + beta[8] * cumvl[i , j] *
cumvlFlag[i , j] + U[i , 1] + U[i , 2] * yrs.fu[j]
    }
    U[i , 1:2] ~ dmnorm(U0[], tau[ , ])
    based4.imp[i] <- cut(base.cd4[i])
    baselogvl.imp[i] <- cut(base.logvl[i])
    base.cd4[i] ~ dnorm(mu1[i], tau1.int)
    mu1[i] <- beta1[1] + beta1[2] * sex.f[i] +
beta1[3] * age_count[i]
    base.logvl[i] ~ dnorm(mu2[i], tau2.int)
    mu2[i] <- beta2[1] + beta2[2] * sex.f[i] +
beta2[3] * age_count[i]
    sex.f[i] ~ dbern(p[i])
    p[i] <- max(1.0E-5, min(0.99999, ptemp[i]))
    logit(ptemp[i]) <- beta3[1] +
beta3[2] * age_count[i] +
beta3[3] * baselogvl[i] + beta3[4] * based4[i]
    baselogvl[i]~dnorm(mua[i],taua.int)
      mua[i] <- betaa[1] + betaa[2]*sex.f[i] +
betaa[3]*age.count[i]
    based4[i]~dnorm(mub[i],taub.int)
      mub[i] <- betab[1] +betab[2]*sex.f[i] +
betab[3]*age.count[i]
    ###End of missing data imputation section
    for( j in 1 : M ) {
      cd4.pl[i , j] ~ dnorm(mu[i , j], tau.int)
      r12[i , j] <- CD4[i , j] - cd4.pl[i , j]
      sqr[i , j] <- r12[i , j] * r12[i , j]
    }
  }
  mspe <- mean(sqr[ , ])
  sigmau0 ~ dunif(0, 100)
  sigmau1 ~ dunif(0, 100)
  cor ~ dunif(-1, 1)
  sigmaU[1 , 1] <- pow(sigmau0, 2)
  sigmaU[2 , 2] <- pow(sigmau1, 2)
  sigmaU[1 , 2] <- sigmau0 * sigmau1 * cor
  sigmaU[2 , 1] <- sigmaU[1 , 2]
  tau[1:2 , 1:2] <- inverse(sigmaU[ , ])
  sigma1 <- pow(tau.int,-2)
  tau.int ~ dgamma(0.001,0.001)
  sigma1.imp <- pow(tau1.int,-2)
```

```

tau1.int ~ dgamma(0.001,0.001)
sigma2.imp <- pow(tau2.int,-2)
tau2.int ~ dgamma(0.001,0.001)
sigmaa.imp <- pow(taua.int,-2)
taua.int ~ dgamma(0.001,0.001)
sigmab.imp <- pow(taub.int,-2)
taub.int ~ dgamma(0.001,0.001)

for( k in 1 : 8 ) {
  beta[k] ~ dnorm(0, 0.001)
}
for( k in 1 : 3 ) {
  beta1[k] ~ dnorm(0, 0.001)
}
for( k in 1 : 3 ) {
  beta2[k] ~ dnorm(0, 0.001)
}
for( k in 1 : 4 ) {
  beta3[k] ~ dnorm(0, 0.001)
}
for (k in 1:3) { ##imputations of baselogVL
betaa[k]~dnorm(0,0.001)
}
for (k in 1:3) { ##imputations of based4
betab[k]~dnorm(0,0.001)
}
}

```

Model 1 priors:

In this model, we used vague priors on all parameters as we wanted the prior to be roughly constant in the interval where the likelihood is not zero (18). In this section we use the variance and not precision notation, i.e. $x \sim N(\mu, s^2)$ not $x \sim N(\mu, \Psi)$ with the precision $\Psi = s^{-2}$, for all normally distributed priors even though we used precision notation in the OpenBUGS model (18). The following priors were implemented: For fixed effects vectors β for the main and ϕ_i for $i = (1, \dots, 5)$ imputation sub models, we specified an independent Gaussian distribution i.e. $N(0, 1000)$; scale parameter for CD4 count response σ_ε had a conditional conjugate prior 'tau.int' with an Inverse-gamma distribution, i.e. $IG(0.001, 0.001)$ and the same distribution for scale parameters $\tau_\varepsilon = (\tau_{1\varepsilon}, \tau_{2\varepsilon}, \tau_{3\varepsilon}, \tau_{4\varepsilon})$ denoted as 'tau1.int', 'tau2.int', 'taua.int' and 'taub.int', respectively, for each imputation model; for random-effects covariance matrix Σ_b denoted as 'tau', we specified uniform distributions for the standard deviations σ_0 denoted 'sigmau0' and σ_1 denoted 'sigmau1' for random intercept and slope i.e. $U(0, 100)$; and for the correlation ρ denoted as 'cor' a Uniform distribution $U(1, -1)$.

Slope model with cumulative log viral load – model 2: cubic b-splines with 3 inner knots

```

model{
  for( i in 1 : N ) {
    for( j in 1 : M ) {
      CD4[i , j] ~ dnorm(mu[i , j], tau.int)
      mu[i , j] <- beta[1] * sex.f[i] + beta[2] *
age_count[i] + beta[3] * base.cd4[i] + beta[4] * base.logvl[i]
+ beta[5] * cumvl[i , j] * cumvlFlag[i , j] + U[i , 1] + U[i ,
2] * yrs.fu[j] + bspline[1] * time1[i , j] + bspline[2] *
time2[i , j] + bspline[3] * time3[i , j] + bspline[4] *
time4[i , j] + bspline[5] * time5[i , j] + bspline[6] *
time6[i , j] + bspline[7] * time7[i , j] + bspline[8] *
time8[i , j]
    }
    U[i , 1:2] ~ dnmnorm(U0[], tau[ , ])
    basecd4.imp[i] <- cut(base.cd4[i])
    baselogvl.imp[i] <- cut(base.logvl[i])
    base.cd4[i] ~ dnorm(mu1[i], tau1.int)
    mu1[i] <- beta1[1] + beta1[2] * sex.f[i] +
beta1[3] * age_count[i]
    base.logvl[i] ~ dnorm(mu2[i], tau2.int)
    mu2[i] <- beta2[1] + beta2[2] * sex.f[i] +
beta2[3] * age_count[i]
    sex.f[i] ~ dbern(p[i])
    p[i] <- max(1.0E-5, min(0.99999, ptemp[i]))
    logit(ptemp[i]) <- beta3[1] + beta3[2] * age_count[i] +
beta3[3] * baselogvl[i] + beta3[4] * basecd4[i]
    baselogvl[i]~dnorm(mua[i],taua.int)
      mua[i] <- betaa[1] + betaa[2]*sex.f[i] +
      betaa[3]*age.count[i]
    basecd4[i]~dnorm(mub[i],taub.int)
      mub[i] <- betab[1] +betab[2]*sex.f[i] +
      betab[3]*age.count[i]
    ###End of missing data imputation section

    for( j in 1 : M ) {
      cd4.p1[i , j] ~ dnorm(mu[i , j], tau.int)
      r12[i , j] <- CD4[i , j] - cd4.p1[i , j]
      sqr[i , j] <- r12[i , j] * r12[i , j]
    }
  }
  mspe <- mean(sqr[ , ])
  sigmau0 ~ dunif(0, 100)
  sigmau1 ~ dunif(0, 100)
  cor ~ dunif(-1, 1)
  sigmaU[1 , 1] <- pow(sigmau0, 2)
  sigmaU[2 , 2] <- pow(sigmau1, 2)
  sigmaU[1 , 2] <- sigmau0 * sigmau1 * cor
  sigmaU[2 , 1] <- sigmaU[1 , 2]
  tau[1:2 , 1:2] <- inverse(sigmaU[ , ])
  signal <- pow(tau.int,-2)
  tau.int ~ dgamma(0.001,0.001)

```

```

sigma1.imp <- pow(tau1.int,-2)
tau1.int ~ dgamma(0.001,0.001)
sigma2.imp <- pow(tau2.int,-2)
tau2.int ~ dgamma(0.001,0.001)
sigmaa.imp <- pow(taua.int,-2)
taua.int ~ dgamma(0.001,0.001)
sigmab.imp <- pow(taub.int,-2)
taub.int ~ dgamma(0.001,0.001)

for( k in 1 : 5 ) {
  beta[k] ~ dnorm(0, 0.001)
}
for( k in 1 : 3 ) {
  beta1[k] ~ dnorm(0, 0.001)
}
for( k in 1 : 3 ) {
  beta2[k] ~ dnorm(0, 0.001)
}
for( k in 1 : 4 ) {
  beta3[k] ~ dnorm(0, 0.001)
}
for( d in 1 : 8 ) {
  bspline[d] ~ dnorm(0, 1.0E-6)
}
for (k in 1:3) { ##imputations of baselogVL
betaa[k]~dnorm(0,0.001)
}
for (k in 1:3) { ##imputations of basecd4
betab[k]~dnorm(0,0.001)
}
}

```

Model 2 priors

In this model, we used vague priors on all parameters as we wanted the prior to be roughly constant in the interval where the likelihood is not zero (18). In this section we use the variance and not precision notation, i.e. $x \sim N(\mu, s^2)$ not $x \sim N(\mu, \Psi)$ with the precision $\Psi = s^{-2}$, for all normally distributed priors even though we used precision notation in the OpenBUGS model (18). The following priors were implemented: Fixed effects vector β for the main and ϕ_i for $i = (1, \dots, 5)$ imputation sub models had an independent Gaussian distribution i.e. $N(0, 1000)$; spline terms θ denoted 'bspline' were also assumed to have an independent Gaussian distribution $N(0, 10^6)$ because this provides a small precision and would vary less compared to fixed effects; scale parameter for CD4 count response σ_ε had a conditional conjugate prior 'tau.int' with an Inverse-gamma distribution, i.e. $IG(0.001, 0.001)$ and the same distribution for scale parameters $\tau_\varepsilon = (\tau_{1\varepsilon}, \tau_{2\varepsilon}, \tau_{3\varepsilon}, \tau_{4\varepsilon})$ denoted as 'tau1.int', 'tau2.int', 'taua.int' and 'taub.int', respectively, for each imputation model; for random-effects covariance matrix Σ_b

denoted as '*tau*', we specified uniform distributions for the standard deviations σ_0 denoted '*sigmau0*' and σ_1 denoted '*sigmau1*' for random intercept and slope i.e. $U(0, 100)$; and for the correlation ρ denoted as '*cor*' a Uniform distribution $U(1, -1)$.

Slope model with cumulative log viral load – Model 3: cubic B-splines with 5 inner knots

```

model{
  for( i in 1 : N ) {
    for( j in 1 : M ) {
      CD4[i , j] ~ dnorm(mu[i , j], tau.int)
      mu[i , j] <- beta[1] * sex.f[i] + beta[2] *
age_count[i] + beta[3] * base.cd4[i] + beta[4] * base.logvl[i]
+ beta[5] * cumvl[i , j] * cumvlFlag[i , j] + U[i , 1] + U[i ,
2] * yrs.fu[j] + bspline[1] * time1[i , j] + bspline[2] *
time2[i , j] + bspline[3] * time3[i , j] + bspline[4] *
time4[i , j] + bspline[5] * time5[i , j] + bspline[6] *
time6[i , j] + bspline[7] * time7[i , j] + bspline[8] *
time8[i , j] + bspline[9] * time9[i , j] + bspline[10] *
time10[i , j]
    }
    U[i , 1:2] ~ dnorm(U0[], tau[ , ])
    basecd4.imp[i] <- cut(base.cd4[i])
    baselogvl.imp[i] <- cut(base.logvl[i])
    base.cd4[i] ~ dnorm(mu1[i], tau1.int)
    mu1[i] <- beta1[1] + beta1[2] * sex.f[i] +
beta1[3] * age_count[i]
    base.logvl[i] ~ dnorm(mu2[i], tau2.int)
    mu2[i] <- beta2[1] + beta2[2] * sex.f[i] +
beta2[3] * age_count[i]
    sex.f[i] ~ dbern(p[i])
    p[i] <- max(1.0E-5, min(0.99999, ptemp[i]))
    logit(ptemp[i]) <- beta3[1] +
beta3[2] * age_count[i] +
beta3[3] * baselogvl[i] + beta3[4] * basecd4[i]
    baselogvl[i] ~ dnorm(mua[i], taua.int)
    mua[i] <- betaa[1] + betaa[2]*sex.f[i] +
betaaa[3]*age.count[i]
    basecd4[i] ~ dnorm(mub[i], taub.int)
    mub[i] <- betab[1] + betab[2]*sex.f[i] +
betab[3]*age.count[i]
    ###End of missing data imputation section

    for( j in 1 : M ) {
      cd4.p1[i , j] ~ dnorm(mu[i , j], tau.int)
      r12[i , j] <- CD4[i , j] - cd4.p1[i , j]
      sqr[i , j] <- r12[i , j] * r12[i , j]
    }
  }
  mspe <- mean(sqr[ , ])
  sigmau0 ~ dunif(0, 100)
  sigmau1 ~ dunif(0, 100)
  cor ~ dunif(-1, 1)

```

```

sigmaU[1 , 1] <- pow(sigmau0, 2)
sigmaU[2 , 2] <- pow(sigmau1, 2)
sigmaU[1 , 2] <- sigmau0 * sigmau1 * cor
sigmaU[2 , 1] <- sigmaU[1 , 2]
tau[1:2 , 1:2] <- inverse(sigmaU[ , ])
sigma1 <- pow(tau.int,-2)
tau.int ~ dgamma(0.001,0.001)
sigma1.imp <- pow(tau1.int,-2)
tau1.int ~ dgamma(0.001,0.001)
sigma2.imp <- pow(tau2.int,-2)
tau2.int ~ dgamma(0.001,0.001)
sigmaa.imp <- pow(taua.int,-2)
taua.int ~ dgamma(0.001,0.001)
sigmab.imp <- pow(taub.int,-2)
taub.int ~ dgamma(0.001,0.001)
for( k in 1 : 5 ) {
  beta[k] ~ dnorm(0, 0.001)
}
for( k in 1 : 3 ) {
  beta1[k] ~ dnorm(0, 0.001)
}
for( k in 1 : 3 ) {
  beta2[k] ~ dnorm(0, 0.001)
}
for( k in 1 : 4 ) {
  beta3[k] ~ dnorm(0, 0.001)
}
for( d in 1 : 10 ) {
  bspline[d] ~ dnorm(0, 1.0E-6)
}
for (k in 1:3) { ##imputations of baselogVL
betaa[k]~dnorm(0,0.001)
}
for (k in 1:3) { ##imputations of basecd4
betab[k]~dnorm(0,0.001)
}
}

```

Model 3 priors

In this model, we used vague priors on all parameters as we wanted the prior to be roughly constant in the interval where the likelihood is not zero (18). In this section we use the variance and not precision notation, i.e. $x \sim N(\mu, s^2)$ not $x \sim N(\mu, \Psi)$ with the precision $\Psi = s^{-2}$, for all normally distributed priors even though we used precision notation in the OpenBUGS model (18). The following priors were implemented: fixed effects vector β for the main and ϕ_i for $i = (1, \dots, 5)$ imputation sub models had an independent Gaussian distribution i.e. $N(0, 1000)$; spline terms θ denoted 'bspline' were also assumed to have an independent Gaussian distribution, i.e. $N(0, 10^6)$ because this provides a small precision and would vary less compared to fixed effects; scale parameter for CD4 count response σ_ε had a conditional conjugate prior 'tau.int'

with an Inverse-gamma distribution, i.e. $IG(0.001, 0.001)$ and the same distribution for scale parameters $\tau_{\varepsilon} = (\tau_{1\varepsilon}, \tau_{2\varepsilon}, \tau_{3\varepsilon}, \tau_{4\varepsilon})$ denoted as 'tau1.int', 'tau2.int', 'tau3.int' and 'tau4.int', respectively, for each imputation model; for random-effects covariance matrix Σ_b denoted as 'tau', we specified uniform distributions for the standard deviations σ_0 denoted 'sigmau0' and σ_1 denoted 'sigmau1' for random intercept and slope i.e. $U(0, 100)$; and for the correlation ρ denoted as 'cor' a Uniform distribution $U(1, -1)$.

Slope model with cumulative log viral load – Model 4: Skew-normal random-effects

```

model{
  for( i in 1 : N ) {
    for( j in 1 : M ) {
      CD4[i , j] ~ dnorm(mu[i , j], tau.int)
      mu[i , j] <- beta[1] + beta[2] * sex.f[i] +
beta[3] * age_count[i] + beta[4] * base.cd4[i] + beta[5] *
base.logvl[i] + beta[6] * (yrs.fu[j] - 2.5) + beta[7] *
pow(yrs.fu[j] - 2.5, 2) + beta[8] * cumvl[i , j] *
cumvlFlag[i , j] + U[i , 1] + U[i , 2] * yrs.fu[j]
    }
    U0.t[i , 1] ~ dnorm(0, 1)
    U0.t[i , 2] ~ dnorm(0, 1)
    U0[i , 1] <- deltab.abs[1] * U0.t[i , 1]
    U0[i , 2] <- deltab.abs[2] * U0.t[i , 2]
    U1[i , 1:2] ~ dnorm(meanb[1:2], tau[1:2 , 1:2])
    for( ii in 1 : 2 ) {
      U[i , ii] <- U0[i , ii] + U1[i , ii]
    }
    based4.imp[i] <- cut(base.cd4[i])
    baselogvl.imp[i] <- cut(base.logvl[i])
    base.cd4[i] ~ dnorm(mu1[i], tau1.int)
    mu1[i] <- beta1[1] + beta1[2] * sex.f[i] +
beta1[3] * age_count[i]
    base.logvl[i] ~ dnorm(mu2[i], tau2.int)
    mu2[i] <- beta2[1] + beta2[2] * sex.f[i] +
beta2[3] * age_count[i]
    sex.f[i] ~ dbern(p[i])
    p[i] <- max(1.0E-5, min(0.99999, ptemp[i]))
    logit(ptemp[i]) <- beta3[1] +
beta3[2] * age_count[i] +
beta3[3] * baselogvl[i] + beta3[4] * based4[i]
    baselogvl[i] ~ dnorm(mua[i], taua.int)
    mua[i] <- betaa[1] + betaa[2]*sex.f[i] +
betaa[3]*age.count[i]
    based4[i] ~ dnorm(mub[i], taub.int)
    mub[i] <- betab[1] +betab[2]*sex.f[i] +
betab[3]*age.count[i]
    ###End of missing data imputation section
  }
}

```

```

        for( j in 1 : M ) {
            cd4.p1[i , j] ~ dnorm(mu[i , j], tau.int)
            r12[i , j] <- CD4[i , j] - cd4.p1[i , j]
            sqr[i , j] <- r12[i , j] * r12[i , j]
        }
    }
    mspe <- mean(sqr[ , ])
    Om[1 , 1] <- a * Omega[1 , 1]
    Om[1 , 2] <- a * Omega[1 , 2]
    Om[2 , 1] <- a * Omega[2 , 1]
    Om[2 , 2] <- a * Omega[2 , 2]
    Q[1:2 , 1:2] ~ dwish(Om[1:2 , 1:2], 3)
    tau[1 , 1] <- pow(D1, -2) * Q[1 , 1]
    tau[1 , 2] <- pow(D1, -1) * pow(D2, -1) * Q[1 , 2]
    tau[2 , 1] <- pow(D2, -1) * pow(D1, -1) * Q[2 , 1]
    tau[2 , 2] <- pow(D2, -2) * Q[2 , 2]
    sigmab2[1:2, 1:2] <- inverse(tau[ , ])
    sigmab[1] <- sqrt(sigmab2[1 , 1])
    sigmab[2] <- sqrt(sigmab2[2 , 2])
    corrb <- sigmab2[1 , 2] / sigmab[1] * sigmab[2]
    D1 ~ dunif(0, 100)
    D2 ~ dunif(0, 100)
    deltab[1] ~ dnorm(0.0, 1.0E-6)
    deltab[2] ~ dnorm(0.0, 1.0E-6)
    deltab.abs[1] <- abs(deltab[1])
    deltab.abs[2] <- abs(deltab[2])
    sigma1 <- pow(tau.int,-2)
    tau.int ~ dgamma(0.001,0.001)
    sigma1.imp <- pow(tau.int,-2)
    tau1.int ~ dgamma(0.001,0.001)
    sigma2.imp <- pow(tau2.int,-2)
    tau2.int ~ dgamma(0.001,0.001)
    sigmaa.imp <- pow(taua.int,-2)
    taua.int ~ dgamma(0.001,0.001)
    sigmab.imp <- pow(taub.int,-2)
    taub.int ~ dgamma(0.001,0.001)

    for( k in 1 : 8 ) {
        beta[k] ~ dnorm(0, 0.001)
    }
    for( k in 1 : 3 ) {
        beta1[k] ~ dnorm(0, 0.001)
    }
    for( k in 1 : 3 ) {
        beta2[k] ~ dnorm(0, 0.001)
    }
    for( k in 1 : 4 ) {
        beta3[k] ~ dnorm(0, 0.001)
    }
    for (k in 1:3) { ##imputations of baselogVL
    betaa[k]~dnorm(0,0.001)
    }
    for (k in 1:3) { ##imputations of based4
    betab[k]~dnorm(0,0.001)
    }
}
}

```

Model 4 priors

In this model, we used vague priors on all parameters as we wanted the prior to be roughly constant in the interval where the likelihood is not zero (18). In this section we use the variance and not precision notation, i.e. $x \sim N(\mu, s^2)$ not $x \sim N(\mu, \Psi)$ with the precision $\Psi = s^{-2}$, for all normally distributed priors even though we used precision notation in the OpenBUGS model (18). The following priors were implemented: fixed effects vector β for the main and ϕ_i for $i = (1, \dots, 5)$ imputation sub models had an independent Gaussian distribution i.e. $N(0, 1000)$; the scale parameter for CD4 count response σ_ε , had a conditional conjugate prior ‘tau.int’ with an Inverse-gamma distribution, i.e. $IG(0.001, 0.001)$ and the same distribution for scale parameters $\tau_\varepsilon = (\tau_{1\varepsilon}, \tau_{2\varepsilon}, \tau_{3\varepsilon}, \tau_{4\varepsilon})$ denoted as ‘tau1.int’, ‘tau2.int’, ‘tau3.int’ and ‘tau4.int’, respectively for each imputation model. The coefficients of the random effects $\alpha = (\alpha_{i,1}, \alpha_{i,2})$ for each i^{th} patient denoted $U_{i,1}$ and $U_{i,2}$ for intercept and slope, respectively, were multivariate normally distributed and had a dispersion parameter denoted ‘U0’ for each intercept and slope, which is a product of a normally distributed i.e. $N(0, 10^6)$ absolute parameter $\Delta_b = (\Delta_{1b}, \Delta_{2b})$ denoted as ‘deltab.abs’, and the respective $\vartheta_b = (\vartheta_{1b}, \vartheta_{2b})$ denoted ‘U0.t’—absolute values from a normal distribution $N(0, 1)$. The product makes a distribution of ‘U0’ peaked around zero, which forms a favourable distribution for random-effects. The random-effects covariance matrix Σ_b denoted as ‘tau’ follows an Inverse-Wishart distribution, where Wishart distribution i.e. $W(R, 3)$ (R is a 2-by-2 diagonal matrix of 0.1) is multiplied by uniformly distributed i.e. $U(0, 100)$ shape parameters $\gamma_b = (\gamma_{1b}, \gamma_{2b})$ denoted as ‘D1’ and ‘D2’ for intercept and slope, respectively.

Slope model with cumulative log viral load – Model 5: Skew-normal measurement error and random-effects

```
model {
for(i in 1:N){
  for (j in 1:M) {
    c[i,j] ~ dnorm(0,1)
    w[i,j] <- abs(c[i,j])
    CD4[i,j] ~ dnorm(mu[i,j], tau.int)

    mu[i,j] <- beta[1] +
      beta[2]*sex.f[i] +
      beta[3]*age_count[i]+
      beta[4]*base.cd4[i]+
      beta[5]*base.logvl[i] +
```

```

        beta[6]*(yrs.fu[j]-2.5)+
        beta[7]*pow((yrs.fu[j]-2.5),2)+
        beta[8]*(cumvl[i,j]*cumvlFlag[i,j])+
        alpha[1]*U[i,1] +
        alpha[2]*U[i,2]*(yrs.fu[j]-2.5) +
        delta.abs*w[i,j]
    }

    for(k in 1:2){
        ub[i,k] ~ dnorm(0,1)
        wb[i,k] <- abs(ub[i,k])
        ab[i,k] <- deltab.abs[k]*wb[i,k]
    }
    U[i,1:2] ~ dnorm(ab[i,1:2], taub[1:2,1:2])

##Missing data imputation section for 59 patients
basecd4.imp[i] <- cut(base.cd4[i])
baselogvl.imp[i] <- cut(base.logvl[i])
#imputation model for baseline CD4
base.cd4[i]~dnorm(mu1[i],taul.int)
mu1[i] <- beta1[1] +
beta1[2]*sex.f[i] +
beta1[3]*age_count[i]
#imputation model for baseline log VL
base.logvl[i]~dnorm(mu2[i],tau2.int)
mu2[i] <- beta2[1] +
beta2[2]*sex.f[i] +
beta2[3]*age_count[i]
#imputation model for gender
sex.f[i] ~ dbern(p[i])
p[i]<-max(0.00001,min(0.99999, ptemp[i]))
logit(ptemp[i]) <- beta3[1] +
beta3[2]*age_count[i]+
beta3[3]*baselogvl[i]+
beta3[4]*basecd4[i]
baselogvl[i]~dnorm(mua[i],taua.int)
mua[i] <- betaa[1] + betaa[2]*sex.f[i] +
betaa[3]*age.count[i]
basecd4[i]~dnorm(mub[i],taub.int)
mub[i] <- betab[1] +betab[2]*sex.f[i] +
betab[3]*age.count[i]
###End of missing data imputation section

for (j in 1: M) {
    cd4.p1[i,j] ~ dnorm(mu[i, j], tau.int)
    r12[i,j] <- CD4[i,j]-cd4.p1[i,j]
    sqr[i,j] <- r12[i,j]*r12[i,j]
}
}
mspe<-mean(sqr[,])
##Changing the cov.-matrix of the Wishart from identity
to a multiplication of ##the identity

Om[1,1]<-a*Omega[1,1]
Om[1,2]<-a*Omega[1,2]
Om[2,1]<-a*Omega[2,1]

```

```

    Om[2,2]<-a*Omega[2,2]

    taub[1:2,1:2]~dwish(Om[1:2,1:2],3)
    Q[1:2,1:2]<-inverse(taub[,])
    sigmab2[1,1] <- Q[1,1]*pow(alpha[1],2)
    sigmab2[1,2] <- Q[1,1]*alpha[1]*alpha[2]
    sigmab2[2,1] <- Q[1,1]*alpha[1]*alpha[2]
    sigmab2[2,2] <- Q[1,1]*pow(alpha[2],2)
    for(l in 1:2){
    sigmabb[l] <- sqrt(sigmab2[l,1])
    }
    corrb<-sigmab2[1,2]/(sigmabb[1]*sigmabb[2])
    for(ll in 1:2){alpha[ll] ~dnorm(0.0,1.0E-6)}

    deltab[1] ~ dnorm(0.0,1.0E-6)
    deltab[2] ~ dnorm(0.0,1.0E-6)
    deltab.abs[1] <- abs(deltab[1])
    deltab.abs[2] <- abs(deltab[2])
    delta~dnorm(0,1.0E-6)
    delta.abs<-abs(delta)
    signal <- pow(tau.int,-2)
    tau.int ~ dgamma(0.001,0.001)
    signal.imp <- pow(tau1.int,-2)
    tau1.int ~ dgamma(0.001,0.001)
    sigma2.imp <- pow(tau2.int,-2)
    tau2.int ~ dgamma(0.001,0.001)
    sigmaa.imp <- pow(taua.int,-2)
    taua.int ~ dgamma(0.001,0.001)
    sigmab.imp <- pow(taub.int,-2)
    taub.int ~ dgamma(0.001,0.001)
##Vage normal priors on the betas
    for (k in 1:8) {
    beta[k]~dnorm(0,0.001)
    }

    for (k in 1:3) { #imputations of base.CD4
    beta1[k]~dnorm(0,0.001)
    }
    for (k in 1:3) { ##imputations of base.logVL
    beta2[k]~dnorm(0,0.001)
    }
    for (k in 1:4) { ##imputations of sex
    beta3[k]~dnorm(0,0.001)
    }
    for (k in 1:3) { ##imputations of baselogVL
    betaa[k]~dnorm(0,0.001)
    }
    for (k in 1:3) { ##imputations of basecd4
    betab[k]~dnorm(0,0.001)
    }

}

```

Model 5 priors

In this model, we used vague priors on all parameters as we wanted the prior to be roughly constant in the interval where the likelihood is not zero (18). In this section we use the variance and not precision notation, i.e. $x \sim N(\mu, s^2)$ not $x \sim N(\mu, \Psi)$ with the precision $\Psi = s^{-2}$, for all normally distributed priors even though we used precision notation in the OpenBUGS model (18). The following priors were implemented: fixed effects vector β for the main and ϕ_i for $i = (1, \dots, 5)$ imputation sub models had an independent Gaussian distribution i.e. $N(0, 1000)$; the scale parameter for CD4 count response σ_ε had a conditional conjugate prior ‘*tau.int*’ with an Inverse-gamma distribution, i.e. $IG(0.001, 0.001)$ and likewise the scale parameters $\tau_\varepsilon = (\tau_{1\varepsilon}, \tau_{2\varepsilon}, \tau_{3\varepsilon}, \tau_{4\varepsilon})$ denoted as ‘*tau1.int*’, ‘*tau2.int*’, ‘*tau3.int*’ and ‘*tau4.int*’, respectively, for each imputation model. Random effects with normally distributed coefficient α denoted ‘*alpha*’, $N(0, 10^6)$. The mean for the multivariate normally distributed random effects ‘*ab*’ was estimated from the product of the dispersion parameters $\Delta_b = (\Delta_{1b}, \Delta_{2b})$ denoted as ‘*deltab.abs*’—absolute values from a normal distribution $N(0, 10^6)$ —and the respective $\vartheta_b = (\vartheta_{1b}, \vartheta_{2b})$ denoted ‘*wb*’—absolute values from a normal distribution $N(0, 1)$. The product makes a distribution of ‘*ab*’ peaked, which forms a favourable multivariate distribution for the mean of the random-effects. The random-effects covariance matrix Σ_b denoted ‘*taub*’ follows Inverse-Wishart distribution obtained from the product of a Wishart distribution $W(R, 3)$ (R is a 2-by-2 matrix is a diagonal matrix of 0.1). The measurement error σ_ε^2 denoted as ‘*w*’ is always positive and is normally distributed such that $N(0, 1)$. Further, we assumed a normal distribution $N(0, 10^6)$ for the dispersion parameter for measurement error Δ_{ε_i} denoted ‘*delta.abs*’.

Asymptote models:

Asymptote model with cumulative log viral load – Model 6: primary model

```

model{
  for( i in 1 : N ) {
    for( j in 1 : M ) {
      response[i , j] ~ dbern(p[i , j])
      p[i , j] <- max(1.0E-5, min(0.99999, ptemp[i ,
j]))
      logit(ptemp[i , j]) <- beta[1] + beta[2] *
sex.f[i] + beta[3] * age.count[i] + beta[4] * base.cd4[i] +
beta[5] * base.logvl[i] + beta[6] * (yrs.fu[j] - 2.5) +
beta[7] * pow(yrs.fu[j] - 2.5, 2) + beta[8] * cumvl[i , j] *
cumvflflag[i , j] + b[id[i] , 1] + b[id[i] , 2] * yrs.fu[j]
    }
    b[i , 1:2] ~ dnorm(meanb[], tau[ , ])
    basecd4.imp[i] <- cut(base.cd4[i])
    baselogvl.imp[i] <- cut(base.logvl[i])
    base.cd4[i] ~ dnorm(mul[i], tau1.int)
  }
}

```

```

    mul[i] <- beta1[1] + beta1[2] * sex.f[i] +
    beta1[3] * age.count[i]
    base.logvl[i] ~ dnorm(mu2[i], tau2.int)
    mu2[i] <- beta2[1] + beta2[2] * sex.f[i] +
    beta2[3] * age.count[i]
    sex.f[i] ~ dbern(phi[i])
    phi[i] <- max(1.0E-5, min(0.99999, phitemp[i]))
    logit(phitemp[i]) <- beta3[1] +
    beta3[2] * age.count[i] + beta3[3] * baselogvl[i] +
    beta3[4] * basecd4[i]
  }
  sigmau0 ~ dunif(0, 100)
  sigmau1 ~ dunif(0, 100)
  cor ~ dunif(-1, 1)
  sigmaU[1, 1] <- pow(sigmau0, 2)
  sigmaU[2, 2] <- pow(sigmau1, 2)
  sigmaU[1, 2] <- sigmau0 * sigmau1 * cor
  sigmaU[2, 1] <- sigmaU[1, 2]
  tau[1:2, 1:2] <- inverse(sigmaU[, ])
  sigma1 <- pow(tau.int, -2)
  tau.int ~ dgamma(0.001, 0.001)
  sigma1.imp <- pow(tau1.int, -2)
  tau1.int ~ dgamma(0.001, 0.001)
  sigma2.imp <- pow(tau2.int, -2)
  tau2.int ~ dgamma(0.001, 0.001)
  sigmaa.imp <- pow(taua.int, -2)
  taua.int ~ dgamma(0.001, 0.001)
  sigmab.imp <- pow(taub.int, -2)
  taub.int ~ dgamma(0.001, 0.001)
  for( k in 1 : 8 ) {
    beta[k] ~ dnorm(0, 0.001)
  }
  for( k in 1 : 3 ) {
    beta1[k] ~ dnorm(0, 0.001)
  }
  for( k in 1 : 3 ) {
    beta2[k] ~ dnorm(0, 0.001)
  }
  for( k in 1 : 4 ) {
    beta3[k] ~ dnorm(0, 0.001)
  }
  for( k in 1:3) { ##imputations of baselogVL
    betaa[k]~dnorm(0,0.001)
  }
  for( k in 1:3) { ##imputations of basecd4
    betab[k]~dnorm(0,0.001)
  }
}

```

Model 6 priors

In this model, we used vague priors on all parameters as we wanted the prior to be roughly constant in the interval where the likelihood is not zero (18). In this section we

use the variance and not precision notation, i.e. $x \sim N(\mu, s^2)$ not $x \sim N(\mu, \Psi)$ with the precision $\Psi = s^{-2}$, for all normally distributed priors even though we used precision notation in the OpenBUGS model (18). The following priors were implemented: fixed effects vector β for the main and ϕ_i for $i = (1, \dots, 5)$ imputation sub models had an independent Gaussian distribution i.e. $N(0, 1000)$; the scale parameters of the imputation sub models $\tau_\varepsilon = (\tau_{1\varepsilon}, \tau_{2\varepsilon}, \tau_{3\varepsilon}, \tau_{4\varepsilon})$ denoted as 'tau1.int', 'tau2.int', 'tau3.int' and 'tau4.int', respectively, had conditional conjugate priors with an Inverse-gamma distribution, i.e. $IG(0.001, 0.001)$; for random-effects covariance matrix Σ_b denoted as 'tau', we specified uniform distributions for the standard deviations σ_0 denoted 'sigmau0' and σ_1 denoted 'sigmau1' for random intercept and slope i.e. $U(0, 100)$; and for the correlation ρ denoted as 'cor' a Uniform distribution $U(1, -1)$.

Asymptote model with cumulative log viral load – Model 7: skew-normal random-effects distribution model

```

model
{
  for(i in 1:N) {
    for(j in 1:M) {
      response[i,j] ~ dbern(p[i,j])
      p[i,j] <- max(0.00001, min(0.99999, ptemp[i,j]))
      logit(ptemp[i,j]) <- beta[1] + beta[2]*sex.f[i] +
        beta[3]*age.count[i] + beta[4]*base.cd4[i] +
        beta[5]*base.logvl[i] +
        beta[6]*(yrs.fu[j]-2.5) +
        beta[7]*pow((yrs.fu[j]-2.5), 2) +
        beta[8]*(cumvl[i,j]*cumvlflag[i,j]) +
        U[id[i],1] + U[id[i],2]*yrs.fu[j]
    }

    U0.t[i,1] ~ dnorm(0,1)
    U0.t[i,2] ~ dnorm(0,1)
    U0[i,1] <- deltab.abs[1]*U0.t[i,1]
    U0[i,2] <- deltab.abs[2]*U0.t[i,2]
    U1[i,1:2] ~dmnorm(meanb[1:2], tau[1:2,1:2])
    for(ii in 1:2){
      U[i,ii] <- U0[i,ii] + U1[i,ii]
    }
  }

  ##Missing data imputation section for 59 patients
  basecd4.imp[i] <- cut(base.cd4[i])
  baselogvl.imp[i] <- cut(base.logvl[i])
  #imputation model for baseline CD4
  base.cd4[i] ~ dnorm(mu1[i], tau1.int)
  mu1[i] <- beta1[1] +
    beta1[2]*sex.f[i] +
    beta1[3]*age_count[i]
  #imputation model for baseline log VL
  base.logvl[i] ~ dnorm(mu2[i], tau2.int)
  mu2[i] <- beta2[1] +
    beta2[2]*sex.f[i] +

```

```

        beta2[3]*age_count[i]
#imputation model for gender
sex.f[i] ~ dbern(phi[i])
phi[i]<-max(0.00001,min(0.99999, phitemp[i]))
        logit(phitemp[i]) <- beta3[1] +
        beta3[2]*age_count[i]+
        beta3[3]*baselogvl[i]+
        beta3[4]*basedcd4[i]
###End of missing data imputation section
    }
##Changing the cov.-matrix of the Wishart from identity to a
multiplication of the ##identity
    Om[1,1]<-a*Omega[1,1]
    Om[1,2]<-a*Omega[1,2]
    Om[2,1]<-a*Omega[2,1]
    Om[2,2]<-a*Omega[2,2]
    Q[1:2,1:2]~dwish(Om[1:2,1:2],3)
    tau[1,1]<-pow(D1,-2)*Q[1,1]
    tau[1,2]<-pow(D1,-1)*pow(D2,-1)*Q[1,2]
    tau[2,1]<-pow(D2,-1)*pow(D1,-1)*Q[2,1]
    tau[2,2]<-pow(D2,-2)*Q[2,2]
    sigmab2[1:2,1:2]<-inverse(tau[,])
    sigmab[1]<-sqrt(sigmab2[1,1])
    sigmab[2]<-sqrt(sigmab2[2,2])
    corrb<-sigmab2[1,2]/(sigmab[1]*sigmab[2])
    #####Priors for the scale parameters
    D1~dunif(0,100)
    D2~dunif(0,100)
    deltab[1] ~ dnorm(0.0,1.0E-6)
    deltab[2] ~ dnorm(0.0,1.0E-6)
    deltab.abs[1] <- abs(deltab[1])
    deltab.abs[2] <- abs(deltab[2])
    signal <- pow(tau.int,-2)
    tau.int ~ dgamma(0.001,0.001)
    signal.imp <- pow(tau1.int,-2)
    tau1.int ~ dgamma(0.001,0.001)
    sigma2.imp <- pow(tau2.int,-2)
    tau2.int ~ dgamma(0.001,0.001)
    sigmaa.imp <- pow(taua.int,-2)
    taua.int ~ dgamma(0.001,0.001)
    sigmab.imp <- pow(taub.int,-2)
    taub.int ~ dgamma(0.001,0.001)
##Vage normal priors on the betas
for (k in 1:8) {
    beta[k]~dnorm(0,0.001)
}
for (k in 1:3) { #imputations of base.CD4
    betal[k]~dnorm(0,0.001)
}
for (k in 1:3) { ##imputations of base.logVL
    beta2[k]~dnorm(0,0.001)
}
for (k in 1:4) { ##imputations of sex
    beta3[k]~dnorm(0,0.001)
}
}

```

Model 7 priors

In this model, we used vague priors on all parameters as we wanted the prior to be roughly constant in the interval where the likelihood is not zero (18). In this section we use the variance and not precision notation, i.e. $x \sim N(\mu, s^2)$ not $x \sim N(\mu, \Psi)$ with the precision $\Psi = s^{-2}$, for all normally distributed priors even though we used precision notation in the OpenBUGS model (18). The following priors were implemented: fixed effects vector β for the main and ϕ_i for $i = (1, \dots, 5)$ imputation sub models had an independent Gaussian distribution i.e. $N(0, 1000)$; the scale parameters of the imputation sub models $\tau_\varepsilon = (\tau_{1\varepsilon}, \tau_{2\varepsilon}, \tau_{3\varepsilon}, \tau_{4\varepsilon})$ denoted as 'tau1.int', 'tau2.int', 'tau3.int' and 'tau4.int', respectively, had conditional conjugate priors with an Inverse-gamma distribution, i.e. $IG(0.001, 0.001)$. The coefficients of the random effects $\alpha = (\alpha_{i,1}, \alpha_{i,2})$ for each i^{th} patient denoted $U_{i,1}$ and $U_{i,2}$ for intercept and slope, respectively, were multivariate normally distributed and had a dispersion parameter denoted 'UO' for the intercept and slope, which is a product of a normally distributed absolute parameter $\Delta_b = (\Delta_{1b}, \Delta_{2b})$ denoted as 'deltab.abs' $N(0, 10^6)$, and the respective $\vartheta_b = (\vartheta_{1b}, \vartheta_{2b})$ denoted 'UO.t' also absolute and normally distributed $N(0, 1)$. The product makes a distribution of 'UO' peaked around zero, which forms a favourable distribution for random-effects. The random-effects covariance matrix Σ_b denoted as 'tau' follows an Inverse-Wishart distribution, where Wishart distribution i.e. $W(R, 3)$ (R is a 2-by-2 diagonal matrix of 0.1) is multiplied by shape parameters $\gamma_b = (\gamma_{1b}, \gamma_{2b})$ denoted as 'D1' and 'D2' for intercept and slope, respectively, and are uniformly distributed $U(0, 100)$.

Asymptote model with cumulative log viral load – Model 8: cubic B-splines with 3 inner knots

```
model{
  for( i in 1 : N ) {
    for( j in 1 : M ) {
      response[i , j] ~ dbern(p[i , j])
      p[i , j] <- max(1.0E-5, min(0.99999, ptemp[i , j]))
      logit(ptemp[i , j]) <- beta[1] * sex.f[i] +
beta[2] * age.count[i] + beta[3] * base.cd4[i] + beta[4] *
base.logvl[i] + beta[5] * cumvl[i , j] * cumvlflag[i , j] +
b[id[i] , 1] + b[id[i] , 2] * yrs.fu[j] + bspline[1] *
time1[i , j] + bspline[2] * time2[i , j] + bspline[3] *
time3[i , j] + bspline[4] * time4[i , j] + bspline[5] *
time5[i , j] + bspline[6] * time6[i , j] + bspline[7] *
time7[i , j] + bspline[8] * time8[i , j]
```

```

    }
    b[i , 1:2] ~ dnorm(meanb[], tau[ , ])
    based4.imp[i] <- cut(base.cd4[i])
    baselogvl.imp[i] <- cut(base.logvl[i])
    base.cd4[i] ~ dnorm(mu1[i], tau1.int)
    mu1[i] <- beta1[1] + beta1[2] * sex.f[i] +
    beta1[3] * age.count[i]
    base.logvl[i] ~ dnorm(mu2[i], tau2.int)
    mu2[i] <- beta2[1] + beta2[2] * sex.f[i] +
    beta2[3] * age.count[i]
    sex.f[i] ~ dbern(phi[i])
    phi[i] <- max(1.0E-5, min(0.99999, phitemp[i]))
    logit(phitemp[i]) <- beta3[1] +
    beta3[2] * age.count[i] + beta3[3] * baselogvl[i] +
    beta3[4] * based4[i]
  }
  sigmau0 ~ dunif(0, 100)
  sigmau1 ~ dunif(0, 100)
  corrb ~ dunif(-1, 1)
  sigmaU[1 , 1] <- pow(sigmau0, 2)
  sigmaU[2 , 2] <- pow(sigmau1, 2)
  sigmaU[1 , 2] <- sigmau0 * sigmau1 * cor
  sigmaU[2 , 1] <- sigmaU[1 , 2]
  tau[1:2 , 1:2] <- inverse(sigmaU[ , ])
  sigma1 <- pow(tau.int,-2)
  tau.int ~ dgamma(0.001,0.001)
  sigma1.imp <- pow(tau1.int,-2)
  tau1.int ~ dgamma(0.001,0.001)
  sigma2.imp <- pow(tau2.int,-2)
  tau2.int ~ dgamma(0.001,0.001)
  sigmaa.imp <- pow(taua.int,-2)
  taua.int ~ dgamma(0.001,0.001)
  sigmab.imp <- pow(taub.int,-2)
  taub.int ~ dgamma(0.001,0.001)
  for( k in 1 : 5 ) {
    beta[k] ~ dnorm(0, 0.001)
  }
  for( k in 1 : 3 ) {
    beta1[k] ~ dnorm(0, 0.001)
  }
  for( k in 1 : 3 ) {
    beta2[k] ~ dnorm(0, 0.001)
  }
  for( k in 1 : 4 ) {
    beta3[k] ~ dnorm(0, 0.001)
  }
  for( d in 1 : 8 ) {
    bspline[d] ~ dnorm(0, 1.0E-6)
  }
}

```

Model 8 priors

In this model, we used vague priors on all parameters as we wanted the prior to be roughly constant in the interval where the likelihood is not zero (18). In this section we use the variance and not precision notation, i.e. $x \sim N(\mu, s^2)$ not $x \sim N(\mu, \Psi)$ with the precision $\Psi = s^{-2}$, for all normally distributed priors even though we used precision notation in the OpenBUGS model (18). The following priors were implemented: fixed effects vector β for the main and ϕ_i for $i = (1, \dots, 5)$ imputation sub models had an independent Gaussian distribution i.e. $N(0, 1000)$; spline terms 'bspline' denoted θ were also assumed to have an independent Gaussian distribution, i.e. $N(0, 10^6)$ because this provides a small precision and would vary less compared to fixed effects; the scale parameters of the imputation sub models $\tau_\varepsilon = (\tau_{1\varepsilon}, \tau_{2\varepsilon}, \tau_{3\varepsilon}, \tau_{4\varepsilon})$ denoted as 'tau1.int', 'tau2.int', 'tau3.int' and 'tau4.int', respectively, had conditional conjugate priors with an Inverse-gamma distribution, i.e. $IG(0.001, 0.001)$; for random-effects covariance matrix Σ_b denoted as 'tau', we specified uniform distributions for the standard deviations σ_0 denoted 'sigmau0' and σ_1 denoted 'sigmau1' for random intercept and slope i.e. $U(0, 100)$; and for the correlation ρ denoted as 'cor' a Uniform distribution $U(1, -1)$.

Asymptote model with cumulative log viral load – Model 9: cubic B-splines with 5 inner knots

```

model{
  for( i in 1 : N ) {
    for( j in 1 : M ) {
      response[i , j] ~ dbern(p[i , j])
      p[i , j] <- max(1.0E-5, min(0.99999, ptemp[i , j]))
      logit(ptemp[i , j]) <- beta[1] * sex.f[i] +
beta[2] * age.count[i] + beta[3] * base.cd4[i] + beta[4] *
base.logvl[i] + beta[5] * cumvl[i , j] * cumvlflag[i , j] +
b[id[i] , 1] + b[id[i] , 2] * yrs.fu[j] + bspline[1] *
time1[i , j] + bspline[2] * time2[i , j] + bspline[3] *
time3[i , j] + bspline[4] * time4[i , j] + bspline[5] *
time5[i , j] + bspline[6] * time6[i , j] + bspline[7] *
time7[i , j] + bspline[8] * time8[i , j] + bspline[9] *
time9[i , j] + bspline[10] * time10[i , j]
    }
    b[i , 1:2] ~ dnorm(meanb[], tau[ , ])
    basecd4.imp[i] <- cut(base.cd4[i])
    baselogvl.imp[i] <- cut(base.logvl[i])
    base.cd4[i] ~ dnorm(mu1[i], tau1.int)
    mu1[i] <- beta1[1] + beta1[2] * sex.f[i] +
beta1[3] * age.count[i]
    base.logvl[i] ~ dnorm(mu2[i], tau2.int)
    mu2[i] <- beta2[1] + beta2[2] * sex.f[i] +
beta2[3] * age.count[i]
    sex.f[i] ~ dbern(phi[i])
    phi[i] <- max(1.0E-5, min(0.99999, phitemp[i]))
    logit(phitemp[i]) <- beta3[1] +

```

```

        beta3[2] * age.count[i] + beta3[3] * baselogv1[i] +
        beta3[4] * basecd4[i]
    }
    sigmau0 ~ dunif(0, 100)
    sigmau1 ~ dunif(0, 100)
    corrb ~ dunif(-1, 1)
    sigmaU[1 , 1] <- pow(sigmau0, 2)
    sigmaU[2 , 2] <- pow(sigmau1, 2)
    sigmaU[1 , 2] <- sigmau0 * sigmau1 * cor
    sigmaU[2 , 1] <- sigmaU[1 , 2]
    tau[1:2 , 1:2] <- inverse(sigmaU[ , ])
    sigma1 <- pow(tau.int,-2)
    tau.int ~ dgamma(0.001,0.001)
    sigma1.imp <- pow(tau1.int,-2)
    tau1.int ~ dgamma(0.001,0.001)
    sigma2.imp <- pow(tau2.int,-2)
    tau2.int ~ dgamma(0.001,0.001)
    sigmaa.imp <- pow(taua.int,-2)
    taua.int ~ dgamma(0.001,0.001)
    sigmab.imp <- pow(taub.int,-2)
    taub.int ~ dgamma(0.001,0.001)
    for( k in 1 : 5 ) {
        beta[k] ~ dnorm(0, 0.001)
    }
    for( k in 1 : 3 ) {
        beta1[k] ~ dnorm(0, 0.001)
    }
    for( k in 1 : 3 ) {
        beta2[k] ~ dnorm(0, 0.001)
    }
    for( k in 1 : 4 ) {
        beta3[k] ~ dnorm(0, 0.001)
    }
    for( d in 1 : 10 ) {
        bspline[d] ~ dnorm(0, 1.0E-6)
    }
}

```

Model 9 priors

In this model, we used vague priors on all parameters as we wanted the prior to be roughly constant in the interval where the likelihood is not zero (18). In this section we use the variance and not precision notation, i.e. $x \sim N(\mu, s^2)$ not $x \sim N(\mu, \Psi)$ with the precision $\Psi = s^{-2}$, for all normally distributed priors even though we used precision notation in the OpenBUGS model (18). The following priors were implemented: fixed effects vector β for the main and ϕ_i for $i = (1, \dots, 5)$ imputation sub models had an independent Gaussian distribution i.e. $N(0, 1000)$; spline terms 'bspline' denoted θ were also assumed to have an independent Gaussian distribution, i.e. $N(0, 10^6)$ because this provides a small precision and would vary less compared to fixed effects; the scale parameters of the imputation sub models $\tau_\varepsilon = (\tau_{1\varepsilon}, \tau_{2\varepsilon}, \tau_{3\varepsilon}, \tau_{4\varepsilon})$ denoted as

'*tau1.int*', '*tau2.int*', '*taua.int*' and '*taub.int*', respectively, had conditional conjugate priors with an Inverse-gamma distribution, i.e. $IG(0.001, 0.001)$; for random-effects covariance matrix Σ_b denoted as '*tau*', we specified uniform distributions for the standard deviations σ_0 denoted '*sigmau0*' and σ_1 denoted '*sigmau1*' for random intercept and slope i.e. $U(0, 100)$; and for the correlation ρ denoted as '*cor*' a Uniform distribution $U(1, -1)$.