Technical appendix to "Could Graded Repetitive Arm Strengthening Programme (GRASP) benefit stroke survivors in a community setting? Results of a pilot randomised controlled trial"

The code for inputting the raw, anonymous data and running the model in BUGS are given below.

We used the WinBUGS software, with a burn-in of 1000 iterations to achieve convergence, then evaluated over 200,000 iterations thinned to every tenth to reduce autocorrelation.

Diffuse conjugate priors were used throughout. Times taken in each task were logarithmically transformed to an approximate normal distribution (after adding one second to avoid very large negative values), and those that were not completed were incorporated by regarding the times as a mixture of two normal distributions: one containing the log-times to completion and the other containing the incomplete tasks with a shared large mean and very small standard deviation. To achieve this we replaced the 120s with values of 901, sufficiently far removed from the completed times to clearly be from a different distribution. As the completion of the task was known, this allowed participants who could not complete a task before treatment, but could afterwards, to 'switch' from one distribution to the other without artificially large changes in WMFT time unduly dominating the data.

The patients' impairment and tasks' difficulty are represented as crossed random effects. So, although this model included all results on all tasks at both time points, it does not regard these as independent observations but as arising from a specific patient and a specific task.

BUGS model:

```
model {
      for (i in 1:10) {
            for (j in 1:2) {
                  impair[i,j] <- (beta[1]*(1-ngrasp[i])) +
                                 (beta[2]*ngrasp[i]) +
                                 (beta[3]*period[j]) +
                                 (beta[4]*ngrasp[i]*period[j]) + ui[i]
            }
      }
      for (i in 1:N) {
            mu[i] <- impair[id[i],(1+nperiod[i])] + diff[ntask[i]]
            logit(p[i]) <- beta[5] + (beta[6] * mu[i])
            d[i] ~ dbern(p[i])
            mu.logt[i] <-(d[i] * mu[i])+ ((1-d[i]) * 6.803505)
```

```
            logt[i] ~ dnorm(mu.logt[i],tau[i])
            tau[i] <- (d[i] * tau.logt) + ((1-d[i]) * 1000)
#           mu.fas[i] <- (beta[7] * d[i]) +
                         (beta[8] * (1-d[i])) +
                         (beta[9] * mu[i])
            mu.fas[i] <- beta[7] + (beta[8] * mu[i])
            y.fas[i] ~ dnorm(mu.fas[i],tau.fas)T(thr[fas[i]+1],thr[fas[i]+2])
      }
      for (i in 1:10) {
            for (j in 1:2) {
                  mu.dyna[i,j] <- beta[9] + (beta[10] * impair[i,j])
                  dyna[i,j] ~ dnorm(mu.dyna[i,j],tau.dyna)
            }
      }
      for (j in 1:15) {
            diff[j] ~ dnorm(0,tau.diff)
      }
      for (i in 1:10) {
            ui[i] ~ dnorm(0,tau.i)
      }
      tau.diff ~ dgamma(0.01,0.01)
      tau.i ~ dgamma(0.01,0.01)
      tau.logt ~ dgamma(0.01,0.01)
      tau.fas ~ dgamma(0.01,0.01)
      tau.dyna ~ dgamma(0.01,0.01)
      for (i in 1:10) {
            beta[i] ~ dnorm(0,0.001)
      }
      sd[1] <- pow(tau.logt,-0.5)
      sd[2] <- pow(tau.i,-0.5)
      sd[3] <- pow(tau.diff,-0.5)
      sd[4] <- pow(tau.fas,-0.5)
      sd[5] <- pow(tau.dyna,-0.5)
}
```

BUGS data:

```
graspdata<-list( N=300,
logt = c( 6.803505, 1.481605, 1.435085, 1.609438, 1.22083, 1.386294,
1.871802, 1.410987, 0.8754687, 1.252763, 6.803505, 1.526056, 1.667707,
1.774952, 1.512927, 1.386294, 1.88707, 1.547563, 1.223775, 6.803505, 4.624973,
6.803505, 1.589235, 1.824549, 3.188417, 1.386294, 1.871802, 1.824549, 1.722767,
6.803505, 6.803505, 6.803505, 1.686399, 1.740466, 6.803505, 1.386294, 2.424803,
1.704748, 1.791759, 6.803505, 6.803505, 2.04122, 1.435085, 1.704748, 2.596001,
1.386294, 1.88707, 1.435085, 1.308333, 3.044522, 6.803505, 1.824549, 1.629241,
1.774952, 2.601207, 1.386294, 1.88707, 1.629241, 1.386294, 3.374169, 6.803505,
3.044522, 1.526056, 2.85647, 2.420368, 1.360977, 1.88707, 3.005683, 1.064711,
6.803505, 6.803505, 6.803505, 1.526056, 3.854394, 3.095578, 1.252763, 1.871802,
6.803505, 1.704748, 6.803505, 6.803505, 3.569533, 1.458615, 6.803505, 3.454106,
1.609438, 2.61007, 6.803505, 1.252763, 6.803505, 4.727388, 6.803505, 1.435085,
6.803505, 3.261935, 1.568616, 2.80336, 6.803505, 1.871802, 6.803505, 6.803505,
6.803505, 1.547563, 6.803505, 3.205182, 1.856298, 2.928524, 6.803505, 1.871802,
6.803505, 6.803505, 6.803505, 1.856298, 6.803505, 4.058026, 1.568616, 2.747271,
6.803505, 2.727853, 6.803505, 6.803505, 6.803505, 1.223775, 6.803505, 1.870263,
1.589235, 2.747271, 6.803505, 1.131402, 6.803505, 6.803505, 6.803505, 2.24071,
3.921973, 3.058707, 1.609438, 2.76001, 6.803505, 2.501436, 6.803505, 6.803505,
3.433987, 1.435085, 6.803505, 6.803505, 1.609438, 2.459589, 6.803505, 1.435085,
6.803505, 6.803505, 0.9932518, 1.458615, 1.131402, 2.140066, 0.5877867,
0.07696104, 1.386294, 1.098612, 0.8329091, 6.803505, 2.128232, 1.667707,
1.435085, 3.74242, 0.5877867, 0.6931472, 1.504077, 0.6931472, 0.7419373,
6.803505, 6.803505, 1.504077, 2.302585, 2.766319, 0.5306283, 0.6931472,
1.88707, 1.223775, 2.60269, 6.803505, 6.803505, 1.686399, 1.704748, 3.025291,
0.8329091, 1.458615, 1.774952, 1.223775, 1.722767, 6.803505, 0.7419373,
1.386294, 1.098612, 2.251292, 0.9162907, 0.07696104, 1.435085, 1.193922,
```

```
0.8754687, 6.803505, 1.589235, 1.722767, 1.504077, 2.451005, 0.5877867,
1.458615, 1.568616, 0.9162907, 0.9555114, 6.803505, 2.70805, 1.547563,
2.933857, 3.325036, 0.5306283, 1.84055, 2.451005, 1.568616, 2.965273, 6.803505,
4.067316, 1.547563, 2.624669, 3.230804, 0.5306283, 0.8754687, 6.803505,
1.481605, 6.803505, 6.803505, 3.054001, 0.5877867, 2.116256, 3.756538,
0.6931472, 0.8754687, 6.803505, 0.8329091, 6.803505, 6.803505, 6.803505,
1.667707, 6.803505, 6.803505, 0.4700036, 0.6418539, 6.803505, 1.667707,
6.803505, 6.803505, 4.046554, 1.335001, 6.803505, 4.582925, 1.386294, 1.504077,
6.803505, 1.871802, 6.803505, 6.803505, 6.803505, 2.197225, 6.803505, 6.803505,
1.686399, 1.791759, 6.803505, 2.174752, 6.803505, 6.803505, 6.803505, 1.098612,
2.939162, 6.803505, 1.308333, 1.193922, 6.803505, 1.131402, 6.803505, 6.803505,
3.718438, 2.197225, 3.632309, 4.165114, 1.131402, 2.351375, 2.379546, 2.646175,
2.747271, 6.803505, 4.75359, 1.526056, 3.703768, 6.803505, 0.6418539, 2.186051,
2.379546, 1.335001, 2.360854) ,  id = c( 1, 2, 3,
4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9,
10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4,
5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9,
10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4,
5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9,
10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4,
5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9,
10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4,
5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9,
10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4,
5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9,
10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10) , ntask = c(
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 8, 8, 8, 8, 8, 8, 8, 8, 8, 8, 9, 9, 9, 9, 9, 9,
9, 9, 9, 9, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 11, 11, 11, 11, 11, 11, 11,
11, 11, 11, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 13, 13, 13, 13, 13, 13, 13,
13, 13, 13, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 15, 15, 15, 15, 15, 15, 15,
15, 15, 15, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 6, 6, 6,
6, 6, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 8, 8, 8, 8,
8, 8, 8, 8, 8, 8, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 10, 10, 10, 10, 10, 10, 10, 10,
10, 10, 11, 11, 11, 11, 11, 11, 11, 11, 11, 11, 12, 12, 12, 12, 12, 12, 12, 12,
12, 12, 13, 13, 13, 13, 13, 13, 13, 13, 13, 13, 14, 14, 14, 14, 14, 14, 14, 14,
14, 14, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 5, 5, 5, 5, 5, 5,
5, 5, 5, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7) , d
= c( 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1,
1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0,
1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1,
0, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 1,
0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1,
0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1,
1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0,
0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1,
1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1,
1, 1, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 1,
0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1,
1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1) ,
fas = c( 1, 3, 4, 3, 3, 5, 4, 3, 3, 2, 1, 3, 4, 3, 3, 5, 4, 3, 4, 1, 3,
1, 4, 3, 3, 5, 4, 3, 4, 1, 1, 0, 4, 3, 1, 5, 4, 2, 3, 1, 1, 3, 4, 3, 3, 4, 4,
3, 3, 1, 1, 3, 4, 3, 3, 4, 4, 3, 3, 2, 1, 2, 4, 2, 3, 4, 4, 2, 4, 1, 1, 1, 4,
2, 3, 5, 4, 0, 4, 1, 1, 2, 4, 1, 3, 4, 4, 0, 4, 1, 2, 1, 4, 1, 3, 4, 4, 0, 3,
1, 0, 1, 4, 1, 3, 4, 4, 0, 3, 1, 0, 1, 3, 1, 3, 3, 4, 0, 3, 1, 0, 1, 4, 1, 3,
3, 3, 0, 4, 1, 0, 1, 3, 1, 3, 4, 3, 0, 4, 1, 1, 2, 4, 1, 1, 4, 3, 0, 4, 1, 1,
3, 4, 3, 3, 5, 5, 3, 4, 3, 1, 3, 4, 3, 3, 5, 5, 3, 4, 3, 1, 1, 4, 3, 3, 5, 5,
3, 5, 3, 1, 0, 4, 3, 3, 5, 5, 2, 5, 3, 1, 3, 4, 3, 3, 5, 5, 3, 5, 3, 1, 3, 4,
3, 3, 5, 5, 3, 4, 4, 1, 3, 4, 3, 2, 5, 5, 3, 4, 4, 1, 3, 4, 3, 3, 5, 5, 1, 4,
1, 1, 3, 4, 3, 3, 5, 5, 1, 4, 1, 1, 3, 4, 1, 3, 5, 5, 1, 4, 1, 1, 3, 4, 2, 2,
5, 5, 2, 4, 1, 1, 3, 4, 1, 1, 5, 5, 2, 4, 1, 1, 1, 4, 3, 1, 5, 5, 1, 4, 2, 1,
2, 4, 3, 2, 5, 5, 2, 4, 2, 1, 2, 4, 3, 1, 5, 5, 2, 4, 3),
dyna=structure(.Data=c(0.7, 6.3, 7.0, 2.7, 6.0, 18.3, 8.0, 1.7,
21.3, 8.0, 1.0, 8.0, 11.3, 2.7, 3.3, 19.0, 9.0, 3.0, 14.7, 8.7),
.Dim=c(10,2)),
```

```
period=c(0,1),
thr=c(-200, -1.645, -0.524, -0.385, 0.385, 1.282, 200),
nperiod = c( 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1),
ngrasp=c(0,0,0,0,0,1,1,1,1,1))
```