1 Introduction

This document describes the model as found in Broekhuizen et al, PharmacoEconomics 2016. The document is divided into three parts. First, the model inputs (patient preferences and clinical evidence) are presented. Secondly, the model is run. Finally, the model outputs are given.

1.1 Load in packages

The model uses functions from some R packages, these are loaded in first. For readers wanting to run the R scripts themselves, please see ?install.packages in R to find out how to install packages.

```r
require("MASS")
## Loading required package: MASS
require("plyr")
## Loading required package: plyr
```

Furthermore, we will here define a function that will be used later on for calculating rank probabilities.

```r
# required function for rank probabilities
getRankMat <- function(values, doExclude=FALSE, excludeA=NA){
  if (doExclude==TRUE){
    values[, excludeA] <- min(values) - 1e6
  }
  ranks <- aaply(-values,1,rank,.progress = "none")
  return(ranks)
}
```

1.2 Model parameters

And we will define some model options here:nIter is the amount of Monte Carlo simulations, paUncPref denotes is parameter uncertainty in patient preferences should be taken into account, paSpPeV denotes if patient-specific preference variation is taken into account, and paUncPerf denotes if parameter uncertainty in clinical performances is taken into account. In this file we will replicate the paper’s base case results, and therefore we include all types of uncertainty.

```r
nIter <- 1e5
paUncPref <- TRUE
paSpPeV <- TRUE
paUncPerf <- TRUE
```

2 Model inputs

2.1 Patient preferences

Preferences for attributes of highly active anti-retroviral treatments (HAART) were obtained from a study by Hauber et al (reference 30 in the paper). These are the data:
mdata <- structure(list(par=c("S_VIROCO","S_ALLECO","S_BONECO","S_BONE_D", "S_BONE_N","S_KIDCON","S_KID_DK","S_KID_NT"),mean=structure(c(-0.0458, -0.0604,-0.1609,-0.2235,-0.0457,-0.1022,-0.217),.Names=c("S_VIROCO", "S_ALLECO","S_BONECO","S_BONE_D","S_BONE_N","S_KIDCON","S_KID_DK","S_KID_NT")),meancov=structure(c(0.00017161,-4.78674e-05, 4.305708e-05,-5.2269e-06,0.0001348858,-6.224465e-05,0.0001626024, -1.380456e-05,0.00015876,-1.380456e-05,0.00015876, -1.380456e-05,0.00015876,0.00062001,-0.00045488565, 0.00031586148,0.0001651973,-5.2269e-06,4.74012e-05,-0.00045488565,0.00081225, 0.0004944579,0.0002512275,-0.0004286585,-0.0003228585, 0.0001343798,4.664268e-05,-0.00030650904,0.0004944579,0.0198916, 0.0002579441,-0.00015056514,-0.00054819644,-0.2235,0.0198916, -0.0000416514,-0.0000461651,-0.0001056514,-0.0000461651,0.01116281,0.00067728056, 5.758236e-05,7.69234e-06,0.00041651973,-0.00032862585,-0.00054819644),.Dim=c(8L,8L),.Dimnames=list( c("S_VIROCO","S_ALLECO","S_BONECO","S_BONE_D","S_BONE_N", "S_KIDCON","S_KID_DK","S_KID_NT")),sd=structure(c(0.0501,0.0572,0.0552,0.0619, 0.223,0.1152,0.0294,0.205),.Names=c("S_VIROCO","S_ALLECO","S_BONECO","S_BONE_D","S_BONE_N", "S_KIDCON","S_KID_DK","S_KID_NT")),sdcov=structure(c(0.00043264,-6.80264e-05,-5.67216e-05,-0.00014586,0.00016689504, 8.577088e-05,-0.00058792e-05,0.00028446912,-6.80264e-05,0.00044521,0.00022888014,0.00020889,-0.00033729194, -0.00017453076,0.0002237022,-0.00021037122,-5.67216e-05,0.00022888014, 0.00367236,0.00019089,-0.00065586168,-7.245336e-05,0.00038191938, -0.00053143776,-0.00014586,0.00020899,0.00019089,0.00140625, -0.00047583,-0.000240645,0.000167865,-0.00048546,0.00016689504, -0.00033729194,-0.00065586168,-0.00047583,0.000386884,0.00022461664, -0.00038101232,0.00071755164,8.577088e-05,-0.00017453076, -7.245336e-05,-0.000240645,0.00022461664,0.00059536,-0.00021844832, 0.00034899382,-0.805792e-05,0.0002237022,0.00038191938,0.000167865, -0.00038101232,-0.00021844832,0.00346921,-0.00025215565,0.00028446912, -0.00021037122,-0.00053143776,-0.00048546,0.00071755164,0.00034899382, -0.00025215565,0.000272484),.Dim=c(8L,8L),.Dimnames=list( c("S_VIROCO","S_ALLECO","S_BONECO","S_BONE_D","S_BONE_N", "S_KIDCON","S_KID_DK","S_KID_NT")),betaSE=c(0.0131,0.0126,0.0249,0.0285, 0.0446,0.0215,0.0341,0.0407)),.Names=c("par","mean","meancov", "sd","sdcov","betaSE"))

And printed here in human-readable shape

print(mdata)

## $par
## [1] "S_VIROCO" "S_ALLECO" "S_BONECO" "S_BONE_D" "S_BONE_N" "S_KIDCON"
## [7] "S_KID_DK" "S_KID_NT"
##
## $mean
Here, VIROCO is the preference for virological failure, ALLECO is the preference for allergic reaction, BONECO is the preference for treatable bone damage, BONE_D is the preference for bone damage with unknown treatability, BONE_N is the preference for non-treatable bone damage, KIDCON is the preference for treatable kidney damage, KID_DK is the preference for kidney damage with unknown treatability, and
KID_N is the preference for non-treatable kidney damage. That these are the data from the study by Hauber et al. becomes clear when we plot them (applying the same normalization as was done in the Hauber et al study):

```r
barplot(
  -mdata$mean / (min(mdata$mean) - max(mdata$mean)),
  main="Preference weights from Hauber et al study (blue=included)",
  names.arg=mdata$par,
  col=rainbow(2)[c(2,2,2,1,1,1,1,2)]
)
```

And the colors here indicate whether or not these were included in the model (see the publication for details).

2.2 Clinical evidence

We now load in the clinical data. The references to all used clinical trials can be found in the supplementary material as the suppl_citedTrials.docx file. These are the clinical trials used by the National Institute of Health guideline (reference 29 in the paper).

```r
pdata <- structure(list(par=c("S_VIROCO","S_ALLECO","S_BONECO","S_BONE_D",
  "S_BONE_N","S_KIDCON","S_KID_DK","S_KID_NT"),p1=structure(c(121L,
  8L,27L,0L,0L,0L,18L,152L,4L,0L,0L,0L,0L,0L,0L,28L,81L,1L,0L,0L,
  0L,0L,1L,0L,0L,0L,0L,0L,0L,0L,0L,0L,0L,0L,304L,25L,27L,0L,0L,
  0L,0L,6L,70L,0L,0L,0L,0L,0L,0L,0L,0L,47L,6L,6L,0L,0L,0L,0L,
  0L,0L,2L,47L,0L,7L,0L,0L,0L,0L,0L,0L,0L,0L,3L,79L,1L,0L,
  0L,0L,0L,0L,0L),.Names=c("S_VIROCO_a1_e","S_ALLECO_a1_e",
  "S_BONECO_a1_e","S_BONE_D_a1_e","S_BONE_N_a1_e","S_KIDCON_a1_e",
  "S_KID_DK_a1_e","S_KID_NT_a1_e","S_VIROCO_a2_e","S_ALLECO_a2_e",
  "S_BONECO_a2_e","S_BONE_D_a2_e","S_BONE_N_a2_e","S_KIDCON_a2_e",
  "S_KID_DK_a2_e","S_KID_NT_a2_e")),row.names=c(1,2),.Names=structure(c("S_VIROCO","S_ALLECO",
  "S_BONECO","S_BONE_D","S_BONE_N","S_KIDCON","S_KID_DK",
  "S_KID_NT"),.Names="la1","la2"))
```
print(pdata)

## $par
## [1] "S_VIROCO" "S_ALLECO" "S_BONECO" "S_BONE_D" "S_BONE_N" "S_KIDCON"
## [7] "S_KID_DK" "S_KID_NT"

## $p1
## S_VIROCO_a1_e S_ALLECO_a1_e S_BONECO_a1_e S_BONE_D_a1_e S_BONE_N_a1_e
## 121 8 27 0 0
## S_KIDCON_a1_e S_KID_DK_a1_e S_KID_NT_a1_e S_VIROCO_a2_e S_ALLECO_a2_e
## 0 0 18 152 4
## S_BONECO_a2_e S_BONE_D_a2_e S_BONE_N_a2_e S_KIDCON_a2_e S_KID_DK_a2_e
## 13 0 0 0 0
## S_KID_NT_a2_e S_VIROCO_a3_e S_ALLECO_a3_e S_BONECO_a3_e S_BONE_D_a3_e
## 28 81 1 0 0
## S_BONE_N_a3_e S_KIDCON_a3_e S_KID_DK_a3_e S_KID_NT_a3_e S_VIROCO_a4_e
## 0 0 0 1 304
Here, $e$ refers to the number of events and $ne$ refers to the number of non-events. These are the two parameters that will be used as parameters for the beta distributions, as per the recommendation of the ISPOR task force on good modelling practices. $ai$ refers to the $i$th HAART. The list of HAART is printed below. Due to evidence constraints described in the paper, we excluded Raltegravir and Darunavir.

```r
drugNames <- c("Abacavir/Lamivudine", "Tenofovir/Emtricitabine", "Dolutegravir+TE/AL", "Efavirenz+AL", "Lipodipine+ATV/RTV")
```

---

```r
drugNames <- c("Abacavir/Lamivudine", "Tenofovir/Emtricitabine", "Dolutegravir+TE/AL", "Efavirenz+AL", "Lipodipine+ATV/RTV")
```
"Raltegravir (EXCLUDED)",
"Atazanavir/Ritonavir+EL",
"Elvitegravir/Cobicistat+EL",
"Darunavir/Ritonavir (EXCLUDED)"

included=c(T,T,T,F,T,T,F)
print(paste("a", 1:6, ":", drugNames[included=c(T,T,T,F,T,T,F)]))

## [1] "a 1 : Abacavir/Lamivudine" "a 2 : Tenofovir/Emtricitabine"
## [3] "a 3 : Dolutegravir+TE/AL" "a 4 : Efavirenz+AL"
## [5] "a 5 : Atazanavir/Ritonavir+EL" "a 6 : Elvitegravir/Cobicistat+EL"

### Running the simulations

#### 3.1 Include parameter uncertainty in patient preferences

If we opt to include parameter uncertainty in preferences, we draw a sample from the appropriate multivariate normal distribution. Otherwise, we just get the mean values.

```r
if (paUncPref){
  popweights <- mvrnorm(nIter, mdata$mean, mdata$meancov)
} else {
  popweights <- matrix(rep(mdata$mean, 8 * nIter), nrow=nIter, ncol=8, byrow = T)
}
```

#### 3.2 Include patient-specific preference variation

We take a similar approach for patient-specific preference variation:

```r
ptspecificvariation <- matrix(0, nrow=nIter, ncol=8)
if (paSpPeV){
  parUncVarEstimator <- mvrnorm(nIter, mdata$sd, mdata$sdcov)
  for(r in 1:nIter){
    ptspecificvariation[r, ]=mvrnorm(1, rep(0,8), diag(parUncVarEstimator[r, ] ^ 2))
  }
}
```

#### 3.3 Obtain preference samples

This is formula 3 in the paper. We print the top rows of this matrix.

```r
prefsamples <- popweights + ptspecificvariation
colnames(prefsamples) <- mdata$par
boxplot(prefsamples)
```
3.4 Obtain performance samples

The performances are assumed to be distributed according to beta distributions. If parameter uncertainty in clinical performances is not taken into account, we just take the mean performance.

```r
perfsamples <- matrix(0, nrow=nIter, ncol=length(pdata$p1))
colnames(perfsamples) <- paste(rep(pdata$par,(length(pdata$p1)/length(pdata$par)))," ",rep(1:(length(pdata$par)/length(pdata$par)),each=length(pdata$par)),sep=")
if (paUncPerf){
  for(i in 1:ncol(perfsamples)){
    if(pdata$p2[i]>pdata$p1[i]){ # assumption : all distributions are beta
      perfsamples[,i] <- rbeta(nIter,pdata$p1[i],pdata$p2[i])
    }
  }
} else {
  for(i in 1:ncol(perfsamples)){
    if (pdata$p1[i]>0 & pdata$p2[i]>0){
      perfsamples[,i] <- pdata$p1[i] / (pdata$p1[i]+pdata$p2[i])
    }
  }
}
```

# 4.2 rescale because preferences are over percentages [1;100], not over probabilities [0;1]
perfsamples <- perfsamples * 100

And a boxplot of the performance samples:

```r
boxplot(perfsamples[, which(pdata$p2>0)], horizontal = TRUE)
```
3.5 Calculate (partial) values

The partial values are obtained by simply multiplying the weights with the performances for each criterion/treatment combination. As explained in the paper there is no need for an intermediate partial value function step as the clinical performances are all probabilities which are already naturally constrained between zero and one.

```r
predPrefWeights <- matrix(nrow=nIter, ncol=length(pdata$p1))
rownames(predPrefWeights) <- paste("sim",1:nIter)
colnames(predPrefWeights) <- paste(colnames(perfsamples),"_pw",sep="")
n <- length(pdata$par)
for(i in 1:(length(pdata$p1) / length(pdata$par))){
  predPrefWeights[, ((i-1)*n+1):(i * n)] <- prefsamples * perfsamples[, ((i-1)*n+1):(i*n)]
}
print(head(predPrefWeights))
```

```
## S_VIROCO_1_pw S_ALLECO_1_pw S_BONECO_1_pw S_BONE_D_1_pw
## sim 1 0.5111166 -0.28633292 0.04775016 0
## sim 2 -0.5739654 -0.41559548 -0.1857167 0
## sim 3 -0.8510468 0.01043367 0.54187102 0
## sim 4 -0.9109286 -0.25836193 -0.24681184 0
## sim 5 -1.0449863 0.26692790 0.57624309 0
## sim 6 -1.1499375 0.26692790 0.57624309 0
## S_BONE_N_1_pw S_KIDCON_1_pw S_KID_DK_1_pw S_KID_NT_1_pw
## sim 1 0 0 0 -0.3280561
## sim 2 0 0 0 -2.4468412
## sim 3 0 0 0 -1.5942345
## sim 4 0 0 0 -0.1046937
## sim 5 0 0 0 -0.6506555
## sim 6 0 0 0 -0.7769621
## S_VIROCO_2_pw S_ALLECO_2_pw S_BONECO_2_pw S_BONE_D_2_pw
## sim 1 0.6202402 -0.112301980 0.01537830 0
## sim 2 -0.7370310 -0.117538082 -0.07749365 0
## sim 3 -1.0424163 0.003188909 0.32652486 0
## sim 4 -0.9542178 -0.397163340 -0.13934011 0
## sim 5 -1.3889309 0.006912136 0.23135456 0
## sim 6 -1.5967929 0.117475876 0.26456527 0
## S_BONE_N_2_pw S_KIDCON_2_pw S_KID_DK_2_pw S_KID_NT_2_pw
## sim 1 0 0 0 -0.4104068
## sim 2 0 0 0 -4.3581106
## sim 3 0 0 0 -2.5070071
## sim 4 0 0 0 -0.2113034
## sim 5 0 0 0 -0.9979599
## sim 6 0 0 0 -1.0401847
## S_VIROCO_3_pw S_ALLECO_3_pw S_BONECO_3_pw S_BONE_D_3_pw
## sim 1 0.2349649 -0.003260910 0 0
## sim 2 -0.2952951 -0.004779330 0 0
## sim 3 -0.3792135 0.001049721 0 0
## sim 4 -0.3955020 -0.026977278 0 0
## sim 5 -0.5768729 0.007764632 0 0
## sim 6 -0.4852708 0.012709097 0 0
## S_BONE_N_3_pw S_KIDCON_3_pw S_KID_DK_3_pw S_KID_NT_3_pw
## sim 1 0 0 0 -0.0373162965
```

| Sim | S_VIROCO_4_pw | S_ALLECO_4_pw | S_BONECO_4_pw | S_BONE_D_4_pw |
|-----|---------------|---------------|---------------|---------------|
| 1   | 0.4025335     | -0.122818699  | 0.01762462    | 0             |
| 2   | -0.4923454    | -0.18093345   | -0.14847258   | 0             |
| 3   | -0.7279917    | 0.007309845   | 0.26280833    | 0             |
| 4   | -0.6702997    | -0.554760174  | -0.10490111   | 0             |
| 5   | -1.0642051    | 0.033099088   | 0.16005200    | 0             |
| 6   | -1.2304820    | 0.151640560   | 0.20463627    | 0             |

| Sim | S_BONE_N_4_pw | S_KIDCON_4_pw | S_KID_DK_4_pw | S_KID_NT_4_pw |
|-----|---------------|---------------|---------------|---------------|
| 1   | 0             | 0             | 0             | -0.02265699   |
| 2   | 0             | 0             | 0             | -0.13830829   |
| 3   | 0             | 0             | 0             | -0.14589601   |
| 4   | 0             | 0             | 0             | -0.01342888   |
| 5   | 0             | 0             | 0             | -0.10128500   |
| 6   | 0             | 0             | 0             | -0.10100511   |

| Sim | S_VIROCO_5_pw | S_ALLECO_5_pw | S_BONECO_5_pw | S_BONE_D_5_pw |
|-----|---------------|---------------|---------------|---------------|
| 1   | 0.2761834     | 0             | 0             | 0             |
| 2   | -0.3386080    | 0             | 0             | 0             |
| 3   | -0.5518521    | 0             | 0             | 0             |
| 4   | -0.4935682    | 0             | 0             | 0             |
| 5   | -0.5780296    | 0             | 0             | 0             |
| 6   | -0.7908583    | 0             | 0             | 0             |

| Sim | S_BONE_N_5_pw | S_KIDCON_5_pw | S_KID_DK_5_pw | S_KID_NT_5_pw |
|-----|---------------|---------------|---------------|---------------|
| 1   | 0             | 0             | 0             | 0             |
| 2   | 0             | 0             | 0             | 0             |
| 3   | 0             | 0             | 0             | 0             |
| 4   | 0             | 0             | 0             | 0             |
| 5   | 0             | 0             | 0             | 0             |
| 6   | 0             | 0             | 0             | 0             |

| Sim | S_VIROCO_6_pw | S_ALLECO_6_pw | S_BONECO_6_pw | S_BONE_D_6_pw |
|-----|---------------|---------------|---------------|---------------|
| 1   | 0.1476312     | -0.160997078  | 0.008198196   | 0             |
| 2   | -0.1790431    | -0.107158090  | -0.054832057  | 0             |
| 3   | -0.2860936    | 0.006545414   | 0.137147433   | 0             |
| 4   | -0.2218707    | -0.398914239  | -0.071430265  | 0             |
| 5   | -0.3388950    | 0.030224241   | 0.074302408   | 0             |
| 6   | -0.3503293    | 0.104361306   | 0.177691186   | 0             |

| Sim | S_BONE_N_6_pw | S_KIDCON_6_pw | S_KID_DK_6_pw | S_KID_NT_6_pw |
|-----|---------------|---------------|---------------|---------------|
| 1   | 0             | 0             | 0             | -0.05103969   |
| 2   | 0             | 0             | 0             | -0.30576285   |
| 3   | 0             | 0             | 0             | -0.08505409   |
| 4   | 0             | 0             | 0             | -0.01176872   |
| 5   | 0             | 0             | 0             | -0.04854393   |
| 6   | 0             | 0             | 0             | -0.04338097   |

| Sim | S_VIROCO_7_pw | S_ALLECO_7_pw | S_BONECO_7_pw | S_BONE_D_7_pw |
|-----|---------------|---------------|---------------|---------------|
| 1   | 0.4100696     | 0.01653465    | 0             | 0             |
| 2   | -0.5886333    | 0.010324663   | 0             | 0             |
| 3   | -0.6275281    | 0.13644962    | 0             | 0             |
| 4   | -0.7010103    | 0.03877581    | 0             | 0             |
| 5   | -0.9294719    | 0.15641004    | 0             | 0             |
| 6   | -1.1035777    | 0.14709623    | 0             | 0             |
For the overall values, we apply formula 10 in the paper.

```r
overallValues <- matrix(nrow=nIter, ncol=length(pdata$p1) / length(pdata$par))
colnames(overallValues) <- drugNames
rownames(overallValues) <- paste("sim",1:nIter)
n <- length(pdata$par)
for(i in 1:(length(pdata$p1) / length(pdata$par))){
  overallValues[,i] <- rowSums(predPrefWeights[, ((i-1)*n+1):(i*n)])
}
```

Which gives us these mean values for each included treatment:

```r
print(head(overallValues[, included]))
```
print(head(overallValues[, included]))

## Abacavir/Lamivudine Tenofovir/Emtricitabine Dolutegravir+TE/AL
## sim 1   -0.05552228   0.1129097   0.1943877
## sim 2   -3.62197373  -5.2901734  -0.3605364
## sim 3   -1.89297654  -3.2197097  -0.4236072
## sim 4   -1.52079609  -1.7020247  -0.4243442
## sim 5   -1.41641725  -2.1486241  -0.6057583
## sim 6   -1.08372866  -2.2549364  -0.4735280

## Efavirenz+AL Atazanavir/Ritonavir+EL Elvitegravir/Cobicistat+EL
## sim 1     0.2746824  -0.05620742  0.3184268
## sim 2    -0.9592196  -0.64679614 -1.2704776
## sim 3    -0.6037695  -0.22745486  0.8676750
## sim 4    -1.3433899  -0.70398389 -0.8109622
## sim 5    -0.9723390  -0.28291415  1.0337483
## sim 6    -0.9752103  -0.11165781 -1.3775613

barplot(colMeans(overallValues[,included]),col=rainbow(6),names.arg = drugNames[included])

3.6 Ranking probabilities

Finally, we calculate the ranking probabilities per treatment:

r <- getRankMat(overallValues, T, !included)
colnames(r) <- drugNames
rownames(r) <- paste("rank in sim", 1:nrow(r))
print(head(r[, included]))
Further information

This file showed in detail how the model works. To run the model itself in R, one needs to either 1) copy the pieces of code in this file and run them in R, or 2) use the function rScriptBroekhuizen.R contained in the supplementary material. For any questions about the model, please contact c.g.m.oudshoorn@utwente.nl