UKRI MRC Biostatistics Unit
UNIVERSITY OF CAMBRIDGE

Dose-escalation designs for combination & dose-schedule studies

MRC Biostatistics Unit

Practical 3: Dose-escalation in a trial with unknown ordering

You are asked to design a Phase I dose-escalation clinical trial of a combined induction chemotherapy and immunotherapy for patients with high-risk neuroblastoma. The clinical team would like to evaluate the following schedules

- 2 days immunotherapy AFTER chemotherapy (S_1)
- 3 days immunotherapy AFTER chemotherapy (S_2)
- 4 days immunotherapy OVERLAP with chemotherapy for 1 days (S_3)
- 4 days immunotherapy OVERLAP with chemotherapy for 2 days (S_4)

that will be administered as one of the following six treatment regimens: It is known that, R_1 and R_2 are the least toxic regimens, and R_6 is the most toxic regimens. However, the clinicians do not know what is most and least toxic among R_3, R_4, R_5.

Regimen	R_1	R_2	R_3	R_4	R_5	R_6
Cycle 1		S_1	S_2	S_3	S_3	S_4
Cycle 2	S_1	S_2	S_2	S_3	S_4	S_4

The escalation will be driven by the safety of the regimens only. The objective of the trial is to study safety and tolerability of the combination given under various schedules and to identify the maximum tolerated combination-schedule (MTC) associated with 30% risk of a DLT. The acceptable toxicity range is 25%-35%. The safety outcome of interest is a dose-limiting toxicity (DLT) within the two cycles of the treatment (6 weeks). Up to 30 patients are planned to be enrolled in the trial and no early stopping rules are currently imposed. If no DLTs are observed in the trial, the clinical team would like to escalate from R_1 to R_6 in the order the regimens are numbered. Given the uncertainty in the toxicity ordering for some of the combination, you asked to explore the two-stage POCRM design implemented in the R-package `pocrm` for the considered trial.

You can use `P3-template.R` as a starting point.

(a) Specify all possible orderings that can occur in this trial. Using skeleton

$$(0.10, 0.21, 0.24, 0.30, 0.35, 0.40)$$

and function getwm() construct ordering-specific working models (skeletons). Examine the output to check that working models correctly reflect the assumptions on the known (and unknown) ordering of toxicity.

```
orders<-matrix(nrow=6,ncol=6)
orders[1,]<-c(1, 2, 3, 4, 5, 6)
orders[2,]<-c(1, 2, 3, 5, 4, 6)
orders[3,]<-c(1, 2, 4, 3, 5, 6)
orders[4,]<-c(1, 2, 4, 5, 3, 6)
orders[5,]<-c(1, 2, 5, 3, 4, 6)
orders[6,]<-c(1, 2, 5, 4, 3, 6)
skeleton<-c(0.10,0.21,0.24,0.30,0.35,0.40)
alpha<-getwm(orders,skeleton)
```

(b) The clinical team believes that any of the specified ordering is equally likely. Specify the corresponding prior probability of each ordering prior.o

```
prior.o<-rep(1/nrow(orders),nrow(orders))
```

(c) You now need to evaluate the performance of the proposed form of the POCRM design under various scenarios of true toxicity orderings. As a starting point, use the scenarios from L6 (Slide 9). Run the simulation under the 3 scenarios. Compare the performance of the POCRM to the single-agent designs. Does it perform better or worse? Why? What metrics did you consider for the conclusion?

```
Scenarios from L7:
r1.1<-c(0.05,0.10,0.20,0.30,0.45,0.70)
r1.2<-c(0.05,0.10,0.30,0.20,0.45,0.70)
r1.3<-c(0.05,0.10,0.20,0.45,0.30,0.70)
Escalation strategy for Stage 1:
x0<-c(1,2,3,4,5,6) (from the text above)
Simulations:
pocrm.sim(r=r1.3, alpha=alpha, prior.o=prior.o, x0=x0, stop=100, n=30,
theta=0.30, nsim=2000, tox.range=0.05)
Results:
Scenario 1.1:
[1] 0.00 0.09 0.33 0.30 0.25 0.03
Scenario 1.2:
[1] 0.00 0.08 0.32 0.31 0.26 0.03
Scenario 1.3:
[1] 0.00 0.10 0.30 0.25 0.32 0.04
```

POCRM performs similarly in all considered scenario, 30%-32% compared to very uneven performance of the CRM and mTPI. However, the performance is 18% lower under the monotonic ordering for the benefit of nearly 14% under Scenario 1.2 and 16% under Scenario 1.3 (compared to the CRM). This is expected as the POCRM considers the monotonic ordering as only one possibility.

(d) A comprehensive simulation study could include scenarios covering all possible toxicity orderings. Do the scenarios above cover all the possibilities? If yes, explain why. If no, please add variants of the Scenario 1.1 covering the missing cases/

```
A short cut to produce all possible scenarios is
getwm(orders,r1.1)
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,] 0.05  0.1 0.20 0.30 0.45  0.7
[2,] 0.05  0.1 0.20 0.45 0.30  0.7
[3,] 0.05  0.1 0.30 0.20 0.45  0.7
[4,] 0.05  0.1 0.45 0.20 0.30  0.7
[5,] 0.05  0.1 0.30 0.45 0.20  0.7
[6,] 0.05  0.1 0.45 0.30 0.20  0.7
Orderings 4:6 are missing from the simulations.
Simulation for these scenarios:
Scenario 1.4:
[1] 0.00 0.08 0.25 0.31 0.32 0.03
Scenario 1.5:
[1] 0.00 0.09 0.30 0.26 0.31 0.04
Scenario 1.6:
[1] 0.00 0.09 0.25 0.31 0.31 0.04
```

Similar performance in all scenarios as all orderings are equally likely, 30%-32%.

(e) After some discussions, the clinical team agreed that it is less likely that R_5 will be less toxic than R_3. Specifically, it is thought to be half as likely. Refine the POCRM design to reflect this change in the prior beliefs. Run the simulation for the refined POCRM. Do you need to reconsider the scenarios for the simulation study? Their interpretation? Does the POCRM perform better in one scenarios than in others? Why?

Orderings 4, 5, and 6 assume that R_5 is less toxic than R_3. This change can be reflected in the prior probability of orderings

```
> one<-1/(2*3 + 3)
> prior.o<-c(2*one,2*one,2*one,one,one,one)
> sum(prior.o)
[1] 1
```

We still should consider all six scenario from the above but as scenarios 4, 5, 6 are thought to be less likely, we should appropriately account for this when interpreting the results.

```
Scenario 1.1:
[1] 0.00 0.07 0.35 0.30 0.26 0.02
Scenario 1.2:
[1] 0.00 0.08 0.38 0.29 0.24 0.02
Scenario 1.3:
[1] 0.00 0.08 0.28 0.25 0.36 0.03
Scenario 1.4:
[1] 0.00 0.10 0.32 0.37 0.17 0.04
Scenario 1.5:
[1] 0.00 0.12 0.24 0.26 0.33 0.05
Scenario 1.6:
[1] 0.00 0.14 0.23 0.35 0.22 0.06
```

The performance under scenario 1 is the same or improved by 4%-6%. The scenario appears to be harder for the POCRM this might be due to the (i) chosen skeleton (not equally spaced) or (ii) scenario 1.1 being more difficult as the neighbouring regimens (to the target one) has the toxicity rate closest to the target. The performance under scenarios 4 and 5 is worsen as these are less likely scenarios now.

(f) Right after you complete the simulation study and write up of the result, you are invited to an online meeting at which you are notified that the clinical team decided to refine R_5 such that it is not clinically plausible at all that R_5 will be less toxic than R_3. How will this change the POCRM design. Please make the appropriate changes. Would this change the scenarios under which you evaluate the design. How does this performance compare to the single-agent designs and to the first obtained results. Why?

Now we exclude orderings 4, 5, and 6 at all and exclude the corresponding scenarios from the evaluation (as these are not possible at all).

```
Scenario 1.1:
[1] 0.00 0.07 0.34 0.32 0.25 0.02
Scenario 1.2:
[1] 0.00 0.09 0.38 0.29 0.22 0.02
Scenario 1.3:
[1] 0.00 0.07 0.27 0.24 0.39 0.03
```

Compared to the first output, the performance improved by 2%-7%. This is due to less uncertainty in the ordering under the new construction of the design. However, the performance under the monotonic scenario is still worse than under the original CRM.

(g) Fortunately, there were no further changes to the design and the time to conduct the analysis has come.

- Assume that the 0/3 DLTs were observed on R_1 and then 1/3 DLTs on R_2. Fit the POCRM using `pocrm.imp`. What is the probability of each ordering? How does it compare to the prior probability of orderings? What is the next recommended dose? Why? Would you suggest to follow this recommendation?

```
> combos<-c(1,1,1,2,2,2)
> y<-     c(0,0,0,1,0,0)
> fit<-pocrm.imp(alpha=alpha,prior.o=prior.o,theta=0.30,y=y,combos=combos)
> fit$ord.prob
[1] 0.333 0.333 0.333
> fit$order.est
[1] 2
> fit$dose.rec
[1] 5
> fit$ptox.est
[1] 0.098 0.207 0.237 0.346 0.296 0.396
```

  The design does not incorporate escalation constraint, hence, follows the point estimate. The probability of ordering is not updated because 2 os the second toxic in all orderings → provides no information about ordering. The decision is to stay at R_2.

- Assume that given 1 DLT at R_2, the decision is to stay at R_2 where 0/3 DLTs is observed. What are the estimates now? How the probability of each orderings are updated? Why? What would be model's recommendation? Yours?

```
> combos<-c(1,1,1,2,2,2,2,2,2)
> y<-     c(0,0,0,1,0,0,0,0,0)
> fit<-pocrm.imp(alpha=alpha,prior.o=prior.o,theta=0.30,y=y,combos=combos)
> fit$ord.prob
[1] 0.333 0.333 0.333
> fit$order.est
[1] 3
> fit$ptox.est
[1] 0.051 0.132 0.210 0.158 0.257 0.305
> fit$dose.rec
[1] 6
```

  Again, the probability of ordering is not updated. The ordering is chosen at random, hence the decision is to escalate to R_3.

- Assume that the decision is to escalate to R_3 where 1/3 DLTs is observed. What are the estimates now? How the probability of each orderings are updated? Why? What would be model's recommendation? Yours?

```
> combos<-c(1,1,1,2,2,2,2,2,2,3,3,3)
> y<-     c(0,0,0,1,0,0,0,0,0,1,0,0)
> fit<-pocrm.imp(alpha=alpha,prior.o=prior.o,theta=0.30,y=y,combos=combos)
> fit$ord.prob
[1] 0.321 0.321 0.358
> fit$order.est
[1] 3
> fit$ptox.est
[1] 0.067 0.160 0.244 0.188 0.292 0.341
> fit$dose.rec
[1] 5
```

The probabilities of the orderings are updated for the first time. The ordering under which 3 is more toxic is high (slightly) more likely.

- Since 1 DLT is observed, the clinical team decided to overrule the model and de-escalate to a less toxic dose. What would be the model's recommendation?

  Under ordering 3, R_4 is less toxic. Hence, if the decision is to de-escalate, then the recommendation should be R_4 (while the index is increasing, it is actually de-escalation.

(h) Could you recommend the developed design for the use in the actual study? If yes, why? If not, what is missing? Why is it important to align all escalation constraints with the implementation of the design?

Things to mention: (i) escalation constraints; (ii) investigate the uneven performance under scenarios; (iii) aligning the design with the real-life constraint will help to optimize the design.