Sample size and precision in the App study

SDC / MaEJ
http://bendixcarstensen.com/SDC/App
August 2015
1

Compiled Tuesday 1st September, 2015, 21:32
from: /home/bendix/sdc/coll/maej/appCompliance/pwrprc.tex

Bendix Carstensen Steno Diabetes Center, Gentofte, Denmark
& Department of Biostatistics, University of Copenhagen
bxc@steno.dk
http://BendixCarstensen.com

Contents

1 The study

The study intends to randomize 1500 patients evenly in two groups, one receiving conventional therapy and care at SDC and the other being additionally provided with a smart-phone app to remind the patient about taking medication.

In order to assess the effect we have some data from the study of patients at Steno indicating how compliant patients are, or more precisely, how the compliance to different drugs are distributed in the patient population:

```
> library( Epi )
> cmp <- read.csv("mboks.csv")
> str( cmp )
'data.frame':        100 obs. of  11 variables:
 $ atc               : Factor w/ 5 levels "A10BA02","C03AB",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ ATC_name          : Factor w/ 5 levels "A10BA02 metformin",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ antal_person_PDC  : int  194 28 25 33 44 48 57 65 79 85 ...
 $ antal_opfdage_PDC : int  104696 37101 24942 25284 38295 45721 34860 50323 59856 50665 ...
 $ opfdage_median_PDC: num  96 1048 727 377 260 ...
 $ antal_person_PDP  : int  185 20 16 19 21 18 27 33 34 43 ...
 $ antal_opfdage_PDP : int  86904 33619 19531 18493 18997 22498 17469 20775 33957 31956 ...
 $ opfdage_median_PDP: num  85 1936 1048 648 727 ...
 $ total_person      : int  3366 3366 3366 3366 3366 3366 3366 3366 3366 3366 ...
 $ total_opfdage     : int  3059830 3059830 3059830 3059830 3059830 3059830 3059830 3059830 3059830
 $ frak_5pct         : int  1 2 3 4 5 6 7 8 9 10 ...
```

```
> head( cmp )
      atc          ATC_name antal_person_PDC antal_opfdage_PDC opfdage_median_PDC antal_person_PDP
1 A10BA02 A10BA02 metformin              194            104696                96.0              185
2 A10BA02 A10BA02 metformin               28             37101              1048.5               20
3 A10BA02 A10BA02 metformin               25             24942               727.0               16
4 A10BA02 A10BA02 metformin               33             25284               377.0               19
5 A10BA02 A10BA02 metformin               44             38295               259.5               21
6 A10BA02 A10BA02 metformin               48             45721               461.0               18
  antal_opfdage_PDP opfdage_median_PDP total_person total_opfdage frak_5pct
1             86904               85.0         3366       3059830         1
2             33619             1935.5         3366       3059830         2
3             19531             1048.5         3366       3059830         3
4             18493              648.0         3366       3059830         4
5             18997              727.0         3366       3059830         5
6             22498              644.5         3366       3059830         6
```

```
> with( cmp, table(frak_5pct,ATC_name) )
         ATC_name
frak_5pct A10BA02 metformin C03ABxx thiazide diuretics C09AACA ACEi + ARB C10AA01 simvastatin
       1                  1                          1                  1                    1
       2                  1                          1                  1                    1
       3                  1                          1                  1                    1
       4                  1                          1                  1                    1
       5                  1                          1                  1                    1
       6                  1                          1                  1                    1
       7                  1                          1                  1                    1
       8                  1                          1                  1                    1
       9                  1                          1                  1                    1
       10                 1                          1                  1                    1
       11                 1                          1                  1                    1
       12                 1                          1                  1                    1
       13                 1                          1                  1                    1
       14                 1                          1                  1                    1
       15                 1                          1                  1                    1
       16                 1                          1                  1                    1
       17                 1                          1                  1                    1
       18                 1                          1                  1                    1
```

```
       19                 1                          1                  1                    1
       20                 1                          1                  1                    1
         ATC_name
frak_5pct C10AAxx statins
       1                 1
       2                 1
       3                 1
       4                 1
       5                 1
       6                 1
       7                 1
       8                 1
       9                 1
       10                1
       11                1
       12                1
       13                1
       14                1
       15                1
       16                1
       17                1
       18                1
       19                1
       20                1
```

```
> par( mfrow=c(1,5) )
> for( tp in levels(cmp$ATC_name) )
+    {
+   with( subset(cmp, ATC_name==tp),
+         plot( 1:20, antal_person_PDC,
+               type="h", lwd=3 ) )
+    }
```

Since the compliance figures are broadly similar between the 5 drugs, we average across them:

```
> tt <- with( cmp, xtabs( antal_person_PDC ~ frak_5pct + ATC_name) )
> tt <- sweep( tt, 1, apply( tt, 2, sum ), "/" )
> ( tt <- apply( tt, 1, mean ) )
         1          2          3          4          5          6          7          8          9
0.07950089 0.02896305 0.01518917 0.01282730 0.01168402 0.01699346 0.03647199 0.01833748 0.02359518
        10         11         12         13         14         15         16         17         18
0.02199644 0.02448010 0.05494636 0.02894228 0.04054134 0.04074168 0.05407011 0.12991657 0.07616680
        19         20
0.10556046 0.19700279
```

Now the vector `tt` contains the mean of the compliance distributions for the 5 drugs of interest, what we could call an overall compliance distribution in 5% intervals, this is shown in figure 2:

```
> plot(1:20,tt,type="h",lwd=5)
```

2 Intervention effect

If we assume 2 to be the compliance distribution in the placebo group, we can make a hypothesis about how the distribution is going to look among persons allocated to App-exposure.

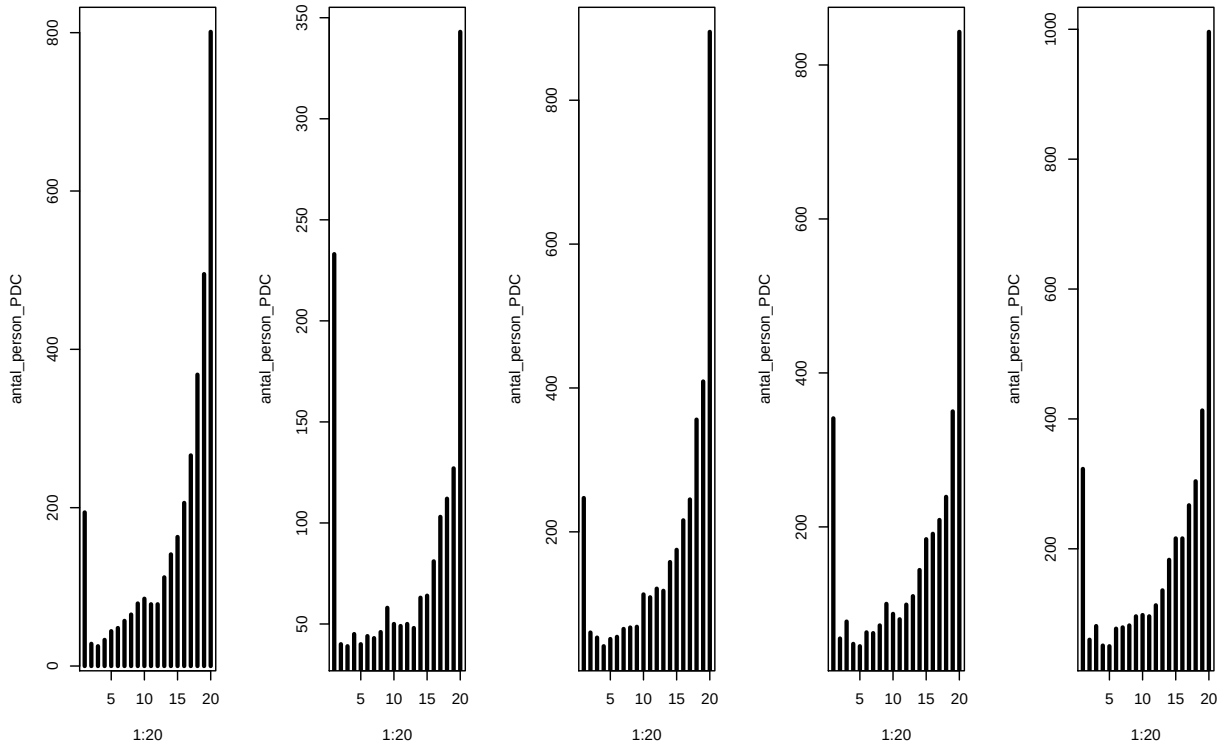

Figure 1: *The distribution of compliance (fraction of person-days covered) for 5 different drugs.*

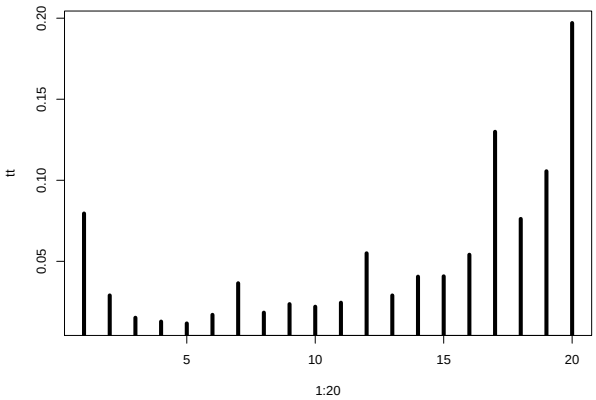

Figure 2: *Overall distribution of compliance, averaged over 5 drugs.*

2.1 Procedure

We assume that the compliance OR between the intervention group and placebo group is, say, e^{θ}; and we use the empirical distribution of compliances as base for the simulations:

- Fix θ at some value.

- Sample N twice from the empirical distribution of compliances to represent the compliances in the two groups.
- Make a logit transform of one sample representing the logits of the compliances in the placebo group.
- Make a logit transform of the other sample and multiply by θ representing the logits of the compliances in the app group.
- Compute:
  - the difference of the average logits, transformed by the exp to give the estimated OR
  - the standard deviation of the average logits transformed by $\exp(2 \times \text{s.d.})$ to give the average error-factor for the precision of the OR.
  - the p-value of the unpaired t-test of the differences between the two samples of logits

  This is easiest done by fitting a linear model to the combined logits.
- Store the three numbers in a row of a 1000×3 array.
- Repeat 1000 times to fill all rows.
- Compute mean of the mean and sds, and fraction of p-values under 0.05.
- Repeat for a different value of θ.

2.2 Implementation in R

The array we use to collect data is only classified by θ (the OR) and (mean, sd, power)

```
> theta <- seq(1.1,5,0.1)
> res <- NArray( list( OR = theta,
+                      what = c("mean","sd","power") ) )
```

Then we devise a function to simulate `N` replicates of the experiment, but first we make a sample of 10,000 compliances from which we sample the experiments:

```
> obs <- numeric(0)
> for( i in 1:length(tt) )
+    obs <- c(obs,runif( floor(tt[i]*10000), (i-1)/20, i/20 ) )
```

Thus the vector `obs` will be the sampling base for our repeat experiments each mimicking the trial under different assumptions about treatment effect.

Then we devise a function to simulate and analyze one experiment (but first the logit and its inverse):

```
> logit <- function(p) log(p/(1-p))
> tigol <- function(x) 1/(1+exp(-x))
> ex <-
+ function( OR, np=700, na=700 )
+ {
+ sp <- logit( sample( obs, np ) )
+ sa <- logit( sample( obs, na ) ) + log(OR)
+ yy <- c( sp, sa )
+ xx <- rep(0:1,c(np,na))
+ mm <- lm( yy ~ xx )
+ summary(mm)$coef[2,]
+ }
> sm <-
+ function( OR, N, np=700, na=700 )
+ {
+ res <- matrix(NA,N,4)
+ for( i in 1:N ) res[i,] <- ex( OR, np=np, na=na )
+ res
+ }
```

With these two functions we can now devise results for a range of ORs:

```
> N <- c(300,500,700)
> OR <- seq(1.05,2,0.05)
> res <- NArray( list( N = N,
+                      OR = OR,
+                    what = c("OR","erf","power") ) )
> for( nn in paste(N) )
+ for( or in paste(OR) )
+ {
+ zz <- sm( as.numeric(or), 1000, np=as.numeric(nn),
+                                  na=as.numeric(nn) )
+ ee <- apply( zz[,1:2], 2, mean )
+ res[nn,or,1] <- exp( ee[1] )
+ res[nn,or,2] <- exp( ee[2]*2 )
+ res[nn,or,3] <- mean( zz[,4]<0.05 )
+ }
> round( ftable(res, col.vars=c(1,3) ), 3 )
     N      300               500               700
     what    OR   erf power    OR   erf power    OR   erf power
OR
1.05      1.048 1.451 0.052 1.049 1.333 0.072 1.050 1.275 0.058
1.1       1.092 1.451 0.070 1.110 1.333 0.098 1.101 1.275 0.104
1.15      1.150 1.450 0.108 1.147 1.334 0.149 1.152 1.276 0.208
1.2       1.209 1.450 0.173 1.201 1.333 0.232 1.203 1.275 0.329
1.25      1.266 1.450 0.252 1.252 1.333 0.329 1.250 1.276 0.458
1.3       1.302 1.450 0.288 1.301 1.334 0.435 1.304 1.275 0.590
1.35      1.353 1.450 0.364 1.344 1.333 0.543 1.343 1.275 0.682
1.4       1.385 1.450 0.410 1.407 1.333 0.669 1.413 1.275 0.815
1.45      1.446 1.449 0.498 1.456 1.334 0.740 1.446 1.275 0.866
1.5       1.488 1.451 0.571 1.504 1.334 0.808 1.498 1.276 0.921
1.55      1.552 1.450 0.661 1.543 1.334 0.854 1.562 1.275 0.954
1.6       1.591 1.449 0.721 1.598 1.334 0.900 1.601 1.276 0.981
1.65      1.669 1.450 0.801 1.652 1.334 0.937 1.654 1.276 0.986
1.7       1.706 1.450 0.834 1.704 1.334 0.959 1.703 1.276 0.994
1.75      1.749 1.449 0.851 1.754 1.333 0.979 1.765 1.275 0.997
1.8       1.795 1.450 0.887 1.803 1.333 0.987 1.800 1.276 0.998
1.85      1.851 1.449 0.914 1.847 1.334 0.990 1.842 1.276 0.999
1.9       1.904 1.450 0.942 1.895 1.334 0.998 1.914 1.276 0.999
1.95      1.963 1.449 0.957 1.947 1.334 0.994 1.942 1.275 1.000
2         2.011 1.449 0.967 2.002 1.333 0.999 1.998 1.275 1.000
```

Thus we see that if we succeed in randomizing 700 in each group we have more than 80% power to see an improvement in compliance corresponding to an OR of 1.4, but if we only succeed in getting 500, we have 80% power to see an OR of 1.5 and for 300 and OR of 1.7.

3 The meaning of compliance OR

In order to illustrate what an OR of 1.4, 1.5 and 1.7 means visually in the context of the distribution of compliances used, we show the distribution change for these values *given* the compliance distribution we have:

```
> par( mfrow=c(4,1), mar=c(3,3,1,1), mgp=c(3,1,0)/1.6 )
> hist( obs, breaks=0:50/50, col="black", ylim=c(0,1200), xlab="", main="" )
> for( or in c(1.4,1.5,1.6) )
+    {
+ oo <- tigol( logit(obs)+log(or) )
+ hist( oo, breaks=0:50/50, col="red", border="red", ylim=c(0,1200),
+      xlab="", main="" )
+ text( 0.5, 1100, paste("OR =", or), font=2, col="red" )
+    }
```

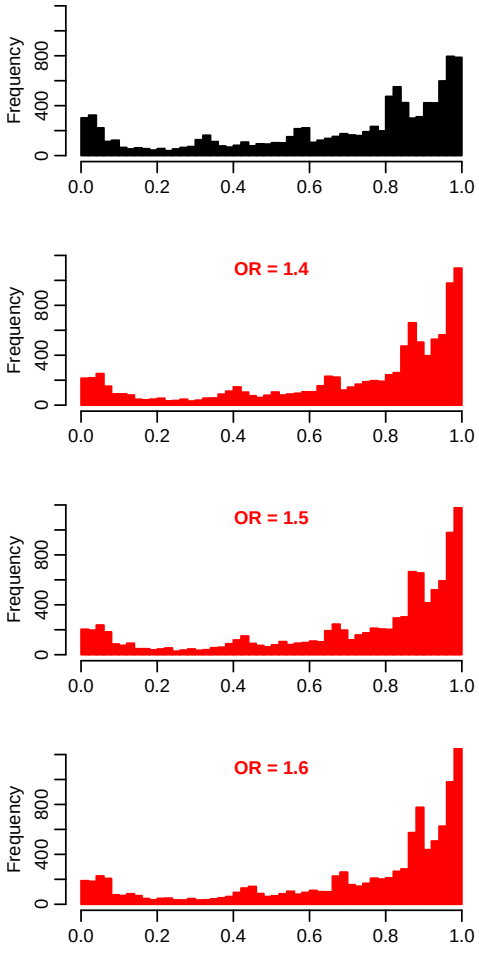

Figure 3: *Compliance distribution in the Steno patient population (black), and how it would look by an improvement of OR=1.4, 1.5 and 1.7, respectively (red).*