

# Low Default Portfolios - Conservative Estimation of Default Probabilities

Benjamin-Cathcart-Ryan Approach

Andrija Djurovic

[www.linkedin.com/in/andrija-djurovic](http://www.linkedin.com/in/andrija-djurovic)

# Low Default Portfolio

- Qualitative descriptions of an LDP leave ample room for interpretation, as different individuals may have varying opinions on whether a given portfolio qualifies as an LDP.
- The low number of defaults within a low default portfolio undermines estimates' reliability and statistical validity for quantitative risk parameters based on historical default experience.
- Low default portfolios are not necessarily low-data portfolios. The scarcity of defaults needs to be considered in relation to the size of the portfolio producing them.
- Regulators may be concerned that PD estimates based solely on simple historical averages or judgmental considerations may underestimate the bank's capital requirements due to default scarcity.
- The BCR approach is one way to address the estimation of PD for the low default portfolio.
- Alternative methods are also available and warrant examination and comparison with the BCR approach.

# BCR Proposal for the PDs Adjustment

- 1 Identify the low default portfolio.
- 2 Identify the historical sample for the LDP (multi-year period) using internal data. The sample should specify each obligor's grade at the start of each year of the historical period.
- 3 Determine the PD for each grade, derived subjectively, using an analytical method or combining these two methods.
- 4 Calculate the portfolio PD as a weighted average of the grade-level PDs.
- 5 Compare the portfolio PD with the upper bound of the PD obtained from the BCR method (explained in the following slides).
- 6 If the upper bound PD is higher than the portfolio PD, rescale the grade-level PD to match the upper bound PD.

# BCR Method - Upper Bound of the Portfolio PD

- 1 The change in the value of obligor  $i$ 's assets over a year  $t$  is given by:

$$y_{i,t} = \sqrt{\rho} * z_t + \sqrt{1 - \rho} * \epsilon_{i,t}$$

where  $\rho$  is asset correlation,  $z$  and  $\epsilon_i$  systemic and idiosyncratic factor, respectively.

The obligor defaults if the asset value is lower than the value of  $c$  for the specific year. Given the above assumptions, we are interested in the probability ( $PD$ ):

$$P(y_{i,t} \leq c) = PD_{i,t} = PD$$

which describes the long-term average 1-year probability of default among the obligors that have not defaulted before and may be considered a through-the-cycle PD.

- 2 Given the number of years ( $T$ ) and year-to-year correlation ( $\theta$ ) we simulate the systemic factor for each year as follows:

$$z_{t,t \leq T} = \theta * z_{t-1} + \sqrt{(1 - \theta^2)} * \epsilon_t$$

- 3 Using the  $z_{t,t \leq T}$  we define the conditional PD ( $PD_c$ ) for each year  $T$  as follows:

$$PD_c = N \left[ \frac{N^{-1}(PD) - \rho * z_t}{\sqrt{1 - \rho}} \right]$$

## BCR Method - Upper Bound of the Portfolio PD cont.

- 4 For the multi-year period, we calculate the cumulative  $PD$  as follows:

$$PD = 1 - \prod_{t=1}^T (1 - PD_{ct})$$

- 5 Given the average number of obligors per year  $n$ , the total number of defaults  $k$ , and the cumulative probability of default  $PD$ , we can calculate the likelihood of observing no more than  $k$  defaults out of  $n$  obligors as:

$$1 - \gamma = \sum_{i=0}^k \binom{n}{k} PD^i * (1 - PD)^{n-i}$$

where  $\gamma$  denotes the selected confidence level.

- 6 Finally, we determine the upper limit of the portfolio PD through numerical optimization for  $PD$  using Monte Carlo simulations to minimize the difference between the average simulated and the chosen confidence level ( $\gamma$ ).

# Simulation Datasets

Number of obligors in each grade per year:

##	GRADE	Y1	Y2	Y3	Y4	Y5
## 1	A	8	7	6	4	1
## 2	B	24	25	25	24	24
## 3	C	37	37	36	37	35
## 4	D	23	24	25	26	25
## 5	E	5	6	4	4	5
## 6	F	1	3	3	2	5
## 7	G	1	1	2	2	3

Aggregated dataset with the PD values:

##	GRADE	PD	OBLIGORS
## 1	A	0.0003	26
## 2	B	0.0010	122
## 3	C	0.0030	182
## 4	D	0.0100	123
## 5	E	0.0300	24
## 6	F	0.1000	14
## 7	G	0.3000	9

Defaults in each grade during the year:

##	GRADE	Y1	Y2	Y3	Y4	Y5
## 1	A	0	0	0	0	0
## 2	B	0	0	0	0	0
## 3	C	0	0	0	0	0
## 4	D	0	0	0	0	0
## 5	E	1	0	0	0	0
## 6	F	0	1	0	0	0
## 7	G	0	0	0	1	1

Summary:

Number of obligor years: 500

Average number of obligors per year: 100

Number of defaults (Y1-Y5): 4

Portfolio PD: 1.35%

# R Code

```
#source r script (data & functions)
source("https://raw.githubusercontent.com/andrija-djurovic/adsfcr/main/model_dev_and_vld/bcr.R")
#inputs
n = 100      #number of obligors
k = 4        #number of defaults
theta = 0.30 #year-to-year correlation
rho = 0.12   #asset correlation
T = 5        #number of years
cl = 0.75    #confidence level
N <- 1e4     #number of simulations
#random seed
set.seed(1)
#optimize the function (opt.f)
res <- uniroot(f = opt.f,
              interval = c(0,1),
              n = n,
              k = k,
              theta = theta,
              rho = rho,
              T = T,
              cl = cl,
              N = N)
#upper bound of the pd
pd.ub <- res[[1]]
pd.ub

## [1] 0.01677385
```

## R Code cont.

```
#portfolio pd
pd.wa <- weighted.mean(x = tbl.3$PD,
                       w = tbl.3$OBLIGORS)
pd.wa
```

```
## [1] 0.0134516
```

```
#scaling factor
sf <- pd.ub / pd.wa
sf
```

```
## [1] 1.246978
```

```
#scaled pds
tbl.3$PD_SCALED <- sf * tbl.3$PD
tbl.3
```

```
##   GRADE    PD OBLIGORS   PD_SCALED
## 1    A 0.0003         26 0.0003740935
## 2    B 0.0010        122 0.0012469782
## 3    C 0.0030        182 0.0037409347
## 4    D 0.0100        123 0.0124697824
## 5    E 0.0300         24 0.0374093472
## 6    F 0.1000         14 0.1246978241
## 7    G 0.3000          9 0.3740934723
```



# Python Code

```
#source python script (data & functions)
import requests
url = "https://raw.githubusercontent.com/andrija-djurovic/adsfcr/main/model_dev_and_vld/bcr.py"
r = requests.get(url)
exec(r.text)
#inputs
n = 100      #number of obligors
k = 4        #number of defaults
theta = 0.30 #year-to-year correlation
rho = 0.12   #asset correlation
T = 5        #number of years
cl = 0.75    #confidence level
N = int(1e4) #number of simulations
#random seed
np.random.seed(321)
#optimize the function (opt_f)
res = root_scalar(f = opt_f,
                  args = (n, k, theta, rho, T, cl, N),
                  bracket = [0, 1])
#upper bound of the pd
pd_ub = res.root
pd_ub

## 0.016866226993434416
```

# Python Code cont.

```
#portfolio pd
pd_wa = np.average(a = tbl_3["PD"],
                  weights = tbl_3["OBLIGORS"])
pd_wa
```

```
## 0.0134516
```

```
#scaling factor
sf = pd_ub / pd_wa
sf
```

```
## 1.2538454156705832
```

```
#scaled pds
tbl_3["PD_SCALED"] = sf * tbl_3["PD"]
tbl_3
```

##	GRADE	PD	OBLIGORS	PD_SCALED
## 0	A	0.0003	26	0.000376
## 1	B	0.0010	122	0.001254
## 2	C	0.0030	182	0.003762
## 3	D	0.0100	123	0.012538
## 4	E	0.0300	24	0.037615
## 5	F	0.1000	14	0.125385
## 6	G	0.3000	9	0.376154