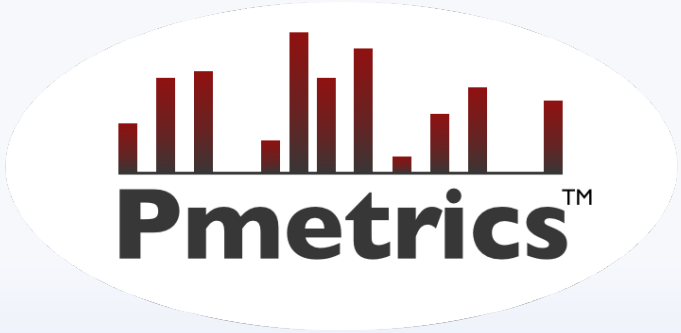


Probability of Target Attainment

Michael Neely, MD
Professor of Pediatrics and Clinical Scholar
Director, Laboratory of Applied Pharmacokinetics and Bioinformatics
University of Southern California, Children's Hospital Los Angeles

Children's Hospital LOS ANGELES
We Treat Kids Better

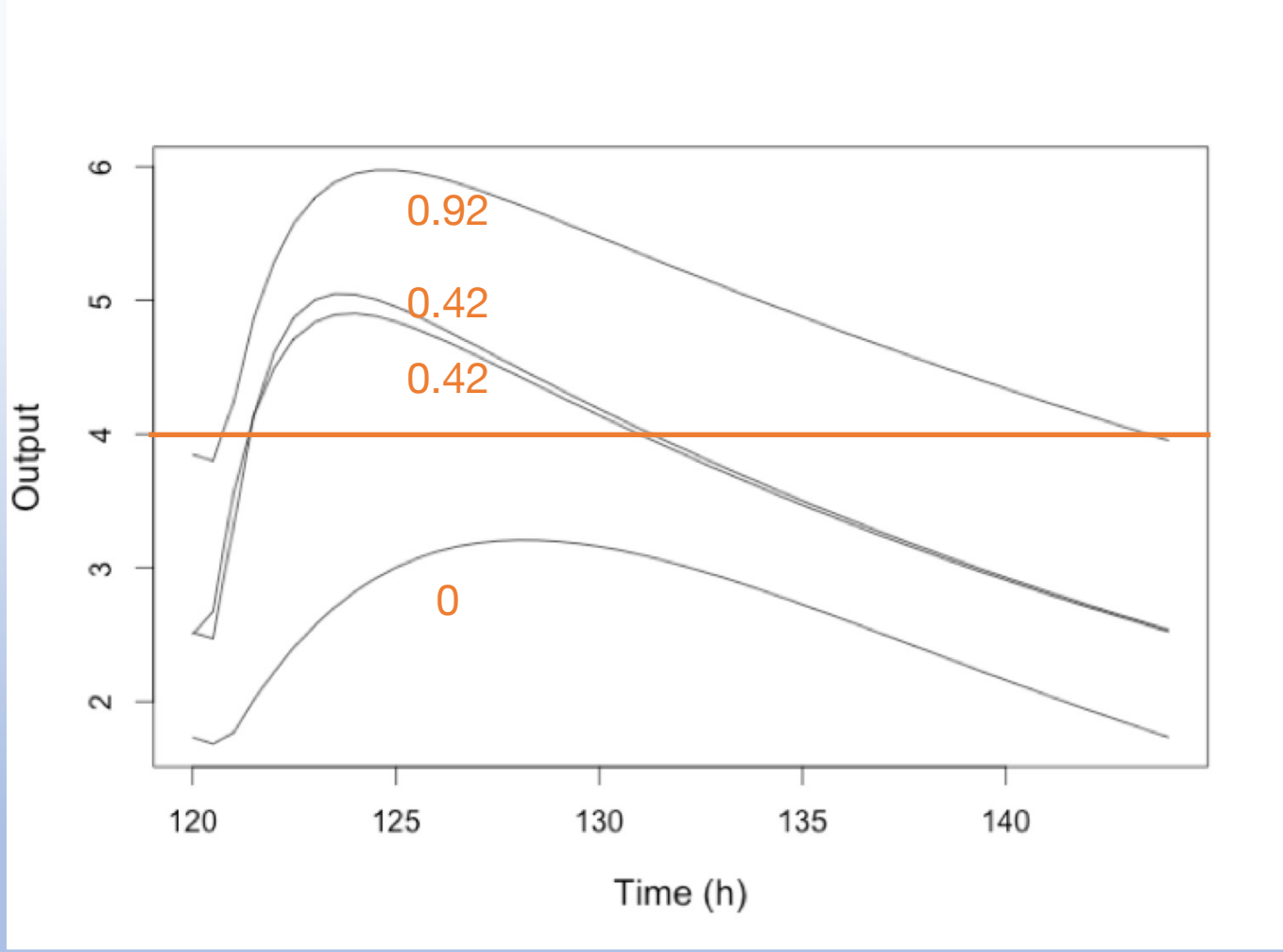
USC University of Southern California

LAPKB
Laboratory of Applied Pharmacokinetics & Bioinformatics

Probability of Target Attainment (PTA)

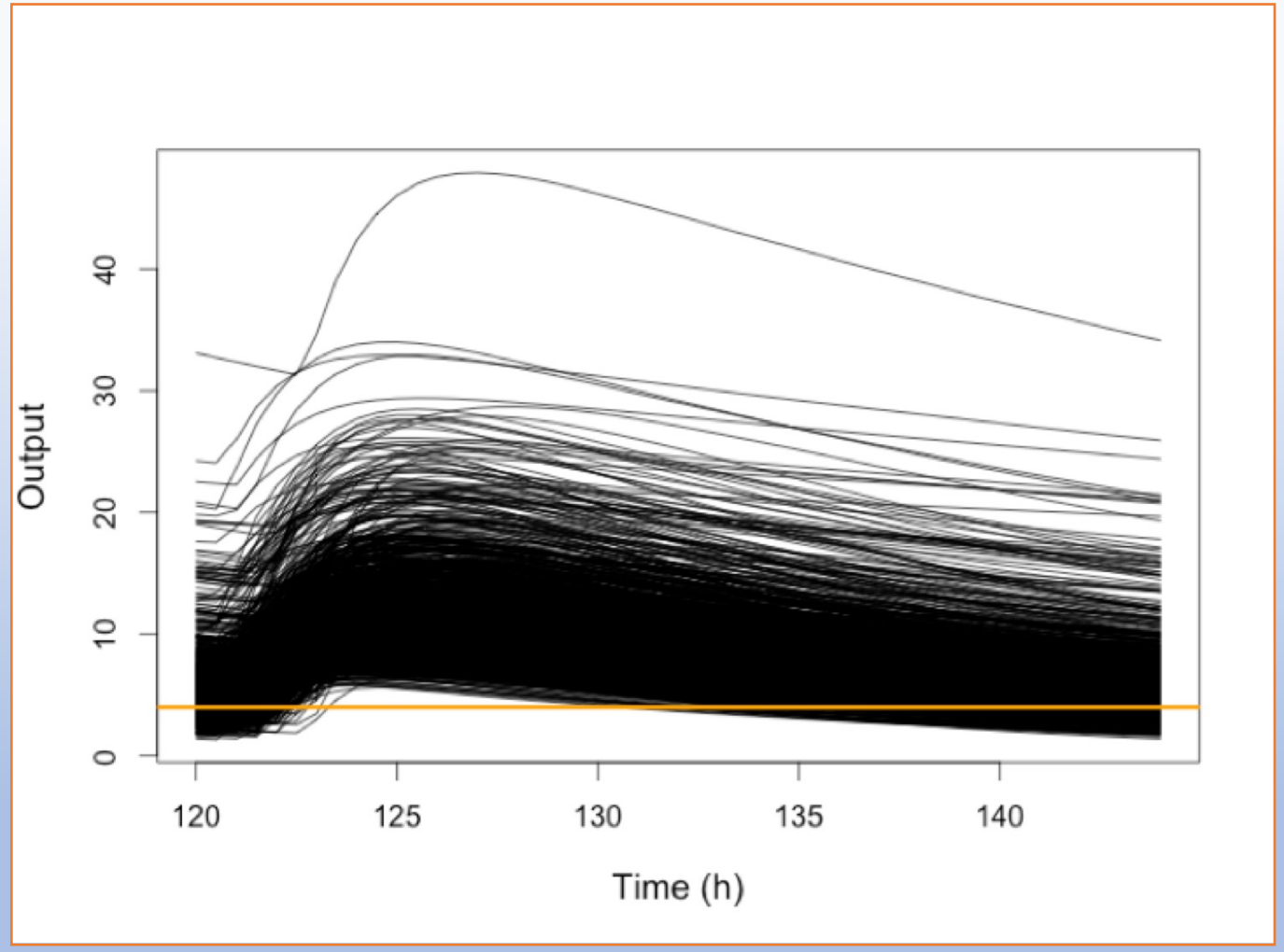
- Simulation technique
- Calculate the proportion of simulated profiles which meet a predefined success threshold for a given dosage regimen
- E.g. AUC:MIC > 10, or %dosage interval (time) > 0.6

PTA



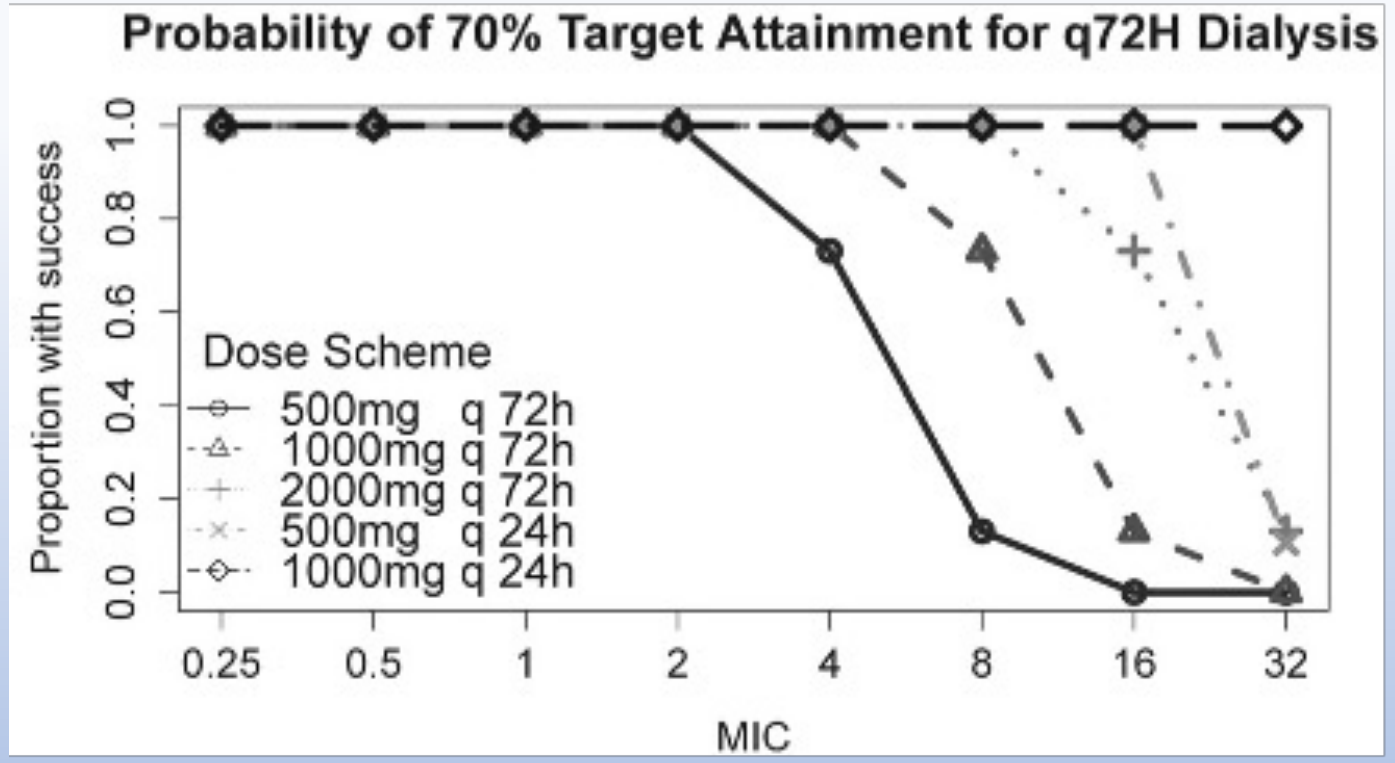
N=4
Mean: 0.44
SD: 0.39
PTA: 25%

PTA



N=1000
Mean: 0.91
SD: 0.14
PTA: 95%

PTA



Loo AS, Neely MN, Anderson EJ, Ghossein C, McLaughlin M, Scheetz MH. Pharmacodynamic target attainment for various ceftazidime dosing schemes in high-flux hemodialysis. Antimicrob Agents Chemother. 2013;57(12):5854–5859.

PM_pta

- Contains all information from a simulation to perform PTA analysis
- class: PMpta, list

```
PM_pta$new(simdata, simlabels, targets,
target.type, success, outeq = 1,  free.fraction
= 1, start, end)
```

Or

```
PM_sim$pta(…)
```

PM_pta$new(): simdata

- A list of simulation outputs. Do not use combine.
- Eg. `simdata <- PM_sim$run(…, combine = F)`

PM_pta$new(): simlabels

- An optional character vector of labels for each simulation in simdata, in order, to be used in plots.
- E.g. `simlabels = c("1200 mg daily", "600 mg bid", "300 mg tie")`
- Default is `simlabels = c("Regimen 1", "Regimen 2", "Regimen 3",…)`

PM_pta$new():targets

- A vector of targets to be compared to exposure
- E.g. `targets=c(0.25, 0.5, 1, 2, 4, 8, 16, 32, 64)` for doubling minimum inhibitory concentrations (MICs) or `targets=4` for a specific concentration which might be a trough

PM_pta$new(): targets

- Can also be a simulated distribution of targets
- Use `makePTAtarget(x)`, where x is a data frame with two columns: targets, frequency

X	Y
0.002	0
0.004	0
0.008	0
0.016	0
0.032	0
0.064	1
0.125	69
0.25	539
0.5	13997
1	68601
2	8445
4	218
8	3
16	0
32	1
64	1
128	1
256	0
512	0

S. aureus vancomycin MIC (Eucast)

PM_pta$new(): target.type

- Specifies which exposure to compare to targets
  - **time**…Will compute the proportion of specified `start` to `end` dosing interval that output is above each target
  - **auc**…Will compute the ratio of AUC for specified `start` to `end` dosing interval to each target
  - **peak**…Will compute the ratio of maximum concentration in specified `start` to `end` dosing interval to each target
  - **min**…Will compute the ratio of minimum concentration in specified `start` to `end` dosing interval to each target
  - **n**…Will compute the ratio of concentration at time n to each target; n must be a simulated time point within the specified `start` to `end` dosing interval

PM_pta$new(): success

- Single value that defines the threshold *proportion* (`target.type="time"`) or *ratio* (all other target.types) to be defined as success
- e.g. `success=10` for AUC:target ≥ 10, or `success=0.8` for 80% of dosing interval above target

PM_pta$new(): outeq

- Define the output equation values to use for the PTA

PM_pta$new():free.fraction

- Multiply output values by this amount prior to PTA
- Useful to simulate free, active fraction of total concentration

PM_pta$new(): start and end

- **start**…The beginning of the interval whose cocentrations will be used to calculate the PTA
- **end**…The end of the interval

PM_pta structure

- **results**…data frame with the following columns:
  - simnum - regimen
  - id - simulated subject (1 to nsim)
  - target - targets
  - pdi - ratio/proportion, as specified by `target.type`
- **outcome**…data frame with the following columns
  - simnum
  - target (if `targets` are discrete)
  - prop.success
  - pdi.mean

Summarize PM_pta

- When targets are discrete:

$pta

	simnum	target	prop.success	pdi.mean	pdi.sd
1	1	0.25	1.000	27.8001331	25.3544073
2	1	0.50	1.000	13.9000666	12.6772037
3	1	1.00	1.000	6.9500333	6.3386018
4	1	2.00	0.956	3.4750166	3.1693009
5	1	4.00	0.647	1.7375083	1.5846505
6	1	8.00	0.251	0.8687542	0.7923252
7	1	16.00	0.063	0.4343771	0.3961626
8	1	32.00	0.009	0.2171885	0.1980813
9	2	0.25	1.000	55.6002668	50.7088175
10	2	0.50	1.000	27.8001334	25.3544088
11	2	1.00	1.000	13.9000667	12.6772044

$pdi

	target	simnum	lowerCI	median	upperCI
1	0.25	1	6.87164250	20.6004900	90.4035400
2	0.25	2	13.74328490	41.2009800	180.8070790
3	0.25	3	10.71711530	27.3798100	121.3775870
4	0.25	4	21.43423050	54.7596200	242.7551350
5	0.50	1	3.43582125	10.3002450	45.2017700
6	0.50	2	6.87164245	20.6004900	90.4035395
7	0.50	3	5.35855765	13.6899050	60.6887935
8	0.50	4	10.71711525	27.3798100	121.3775675
9	1.00	1	1.71791063	5.1501225	22.6008850
10	1.00	2	3.43582123	10.3002450	45.2017697
11	1.00	3	2.67927883	6.8449525	30.3443967
12	1.00	4	5.35855763	13.6899050	60.6887837

pta = probability of target attainment

pdi = pharmacodynamic index

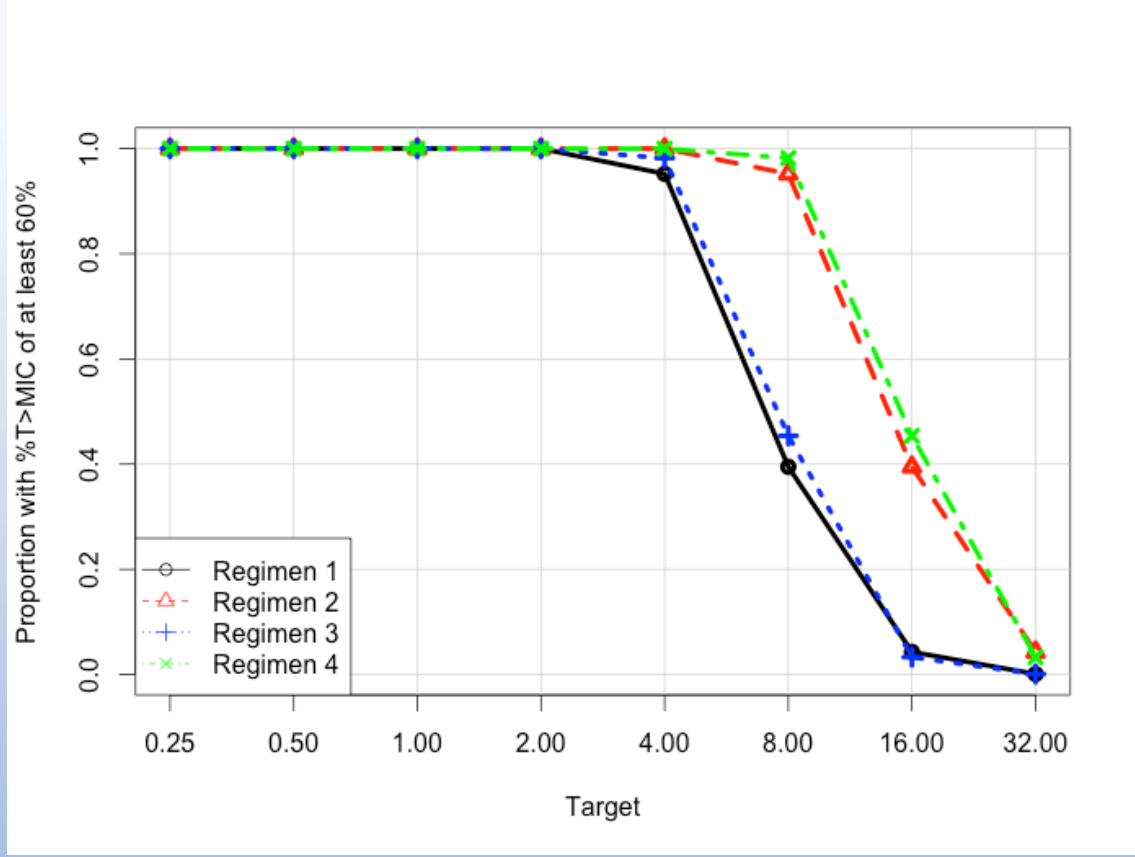
target.type="min", success=1

Summarize PM_pta

- When targets are continuous:

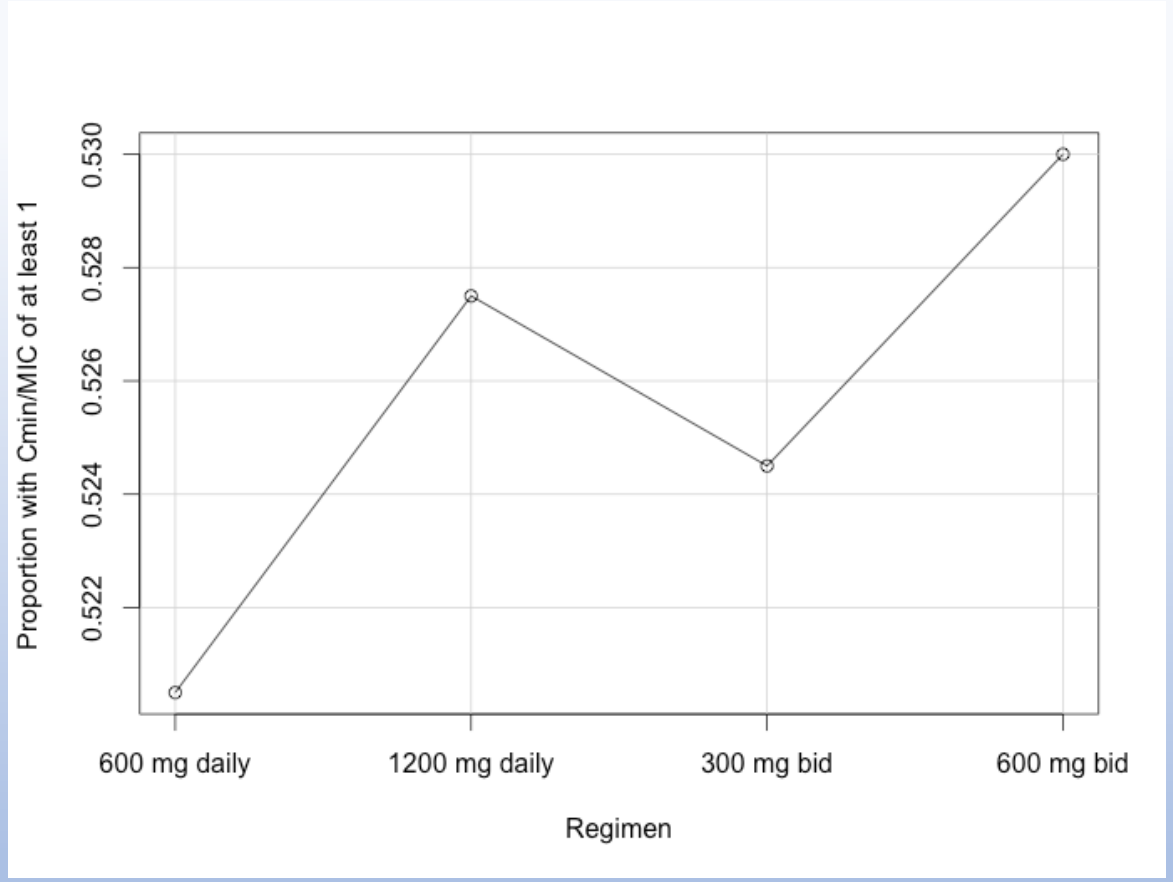
```
$pta
  simnum prop.success  pdi.mean   pdi.sd
1      1       0.5205  83.66498 210.8501
2      2       0.5275 164.57114 421.4380
3      3       0.5245 105.44602 235.1845
4      4       0.5300 208.13321 470.3897

$pdi
  simnum lowerCI   median   upperCI
1      1   0.016 2.172168  586.0183
2      2   0.016 4.000000 1172.0363
3      3   0.016 2.907910  689.3476
4      4   0.016 4.000000 1378.6958
```

PM_pta$plot()



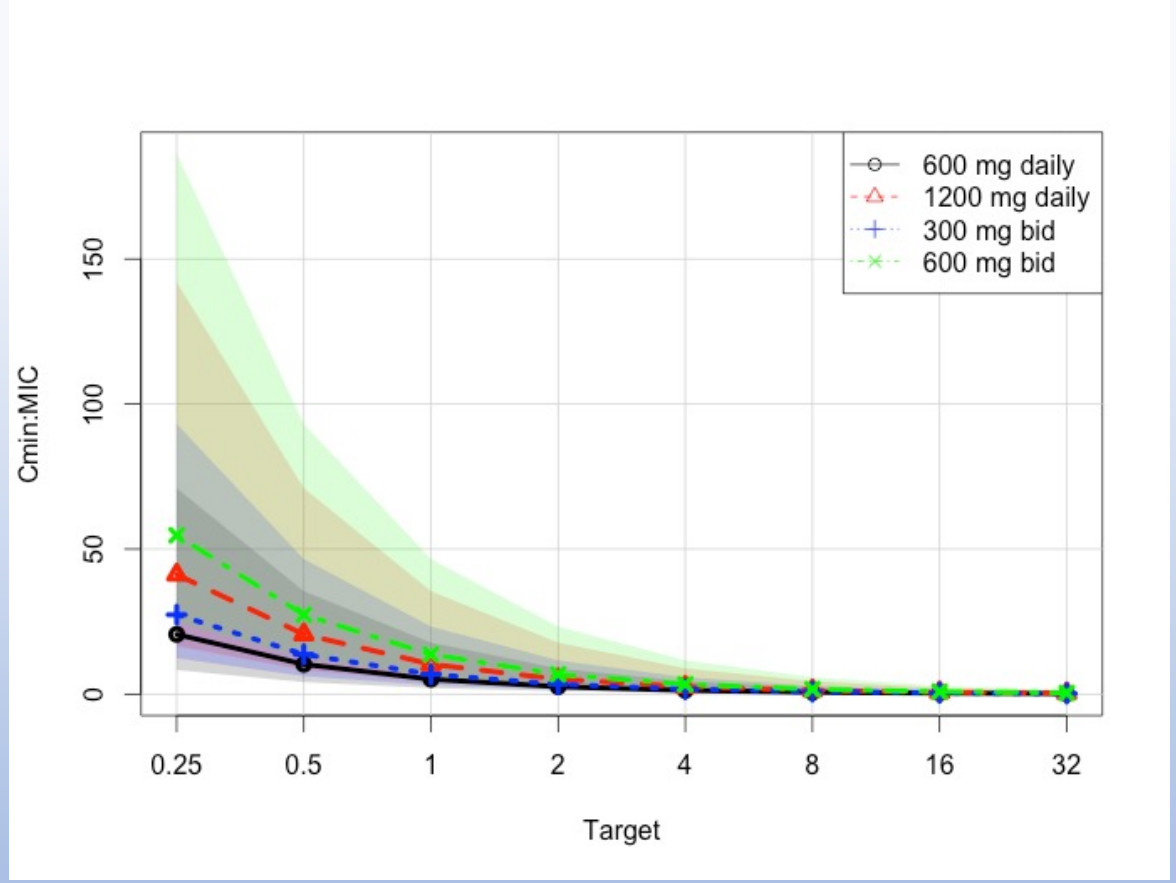

```
pta1$plot(,ylab="Proportion with %T>MIC of at least 60%", grid=T,legend = list(x = "bottomleft"))
```

Targets are continuous



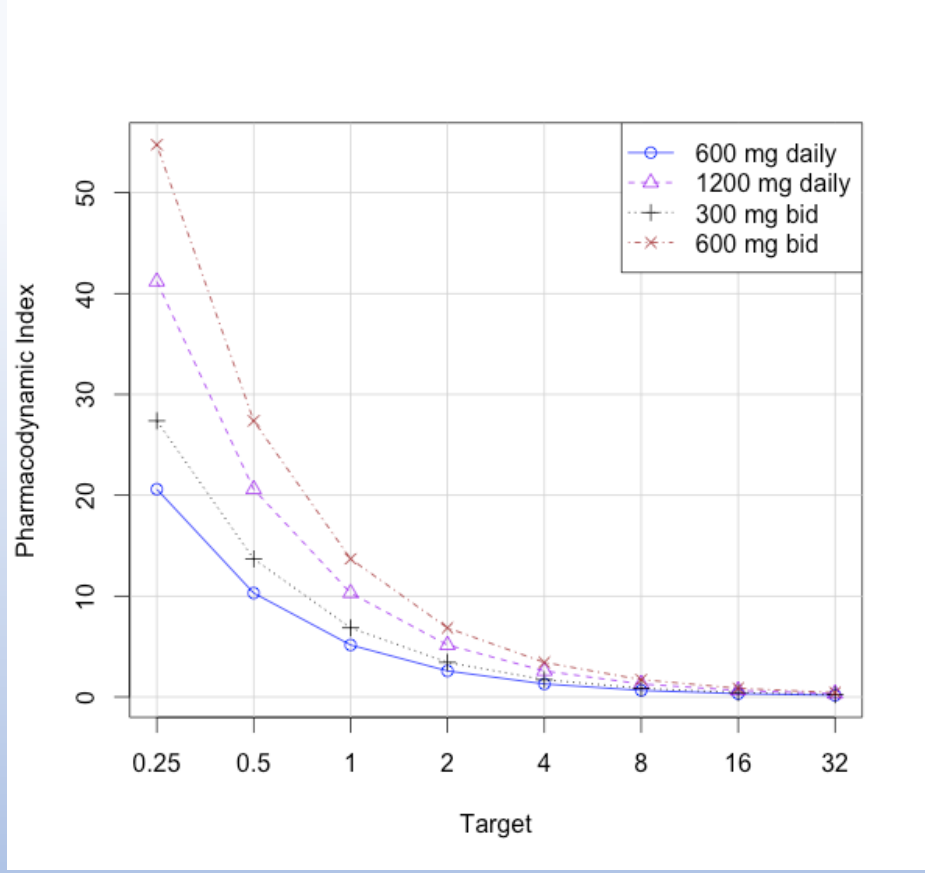

```
pta2$plot(grid = T, ylab = "Proportion with Cmin/MIC of at least 1")
```

PDI plot

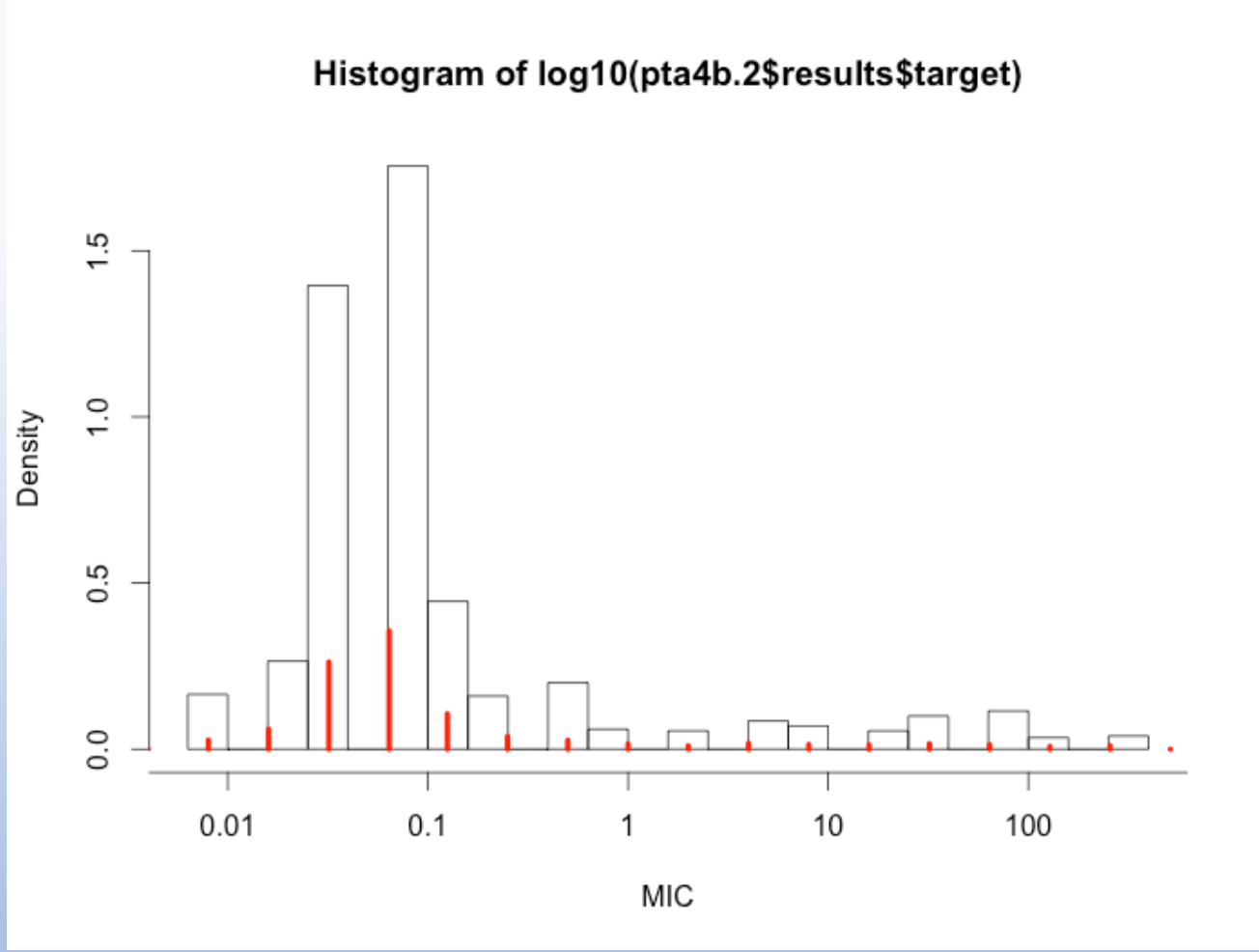



```
Pta2$plot(plot.type = "pdi", ylab = "Cmin:MIC", grid = T)
```

PDI plot




```
pta2$plot(plot.type = "pdi", ci = 0, grid = T, col = c("blue", "purple", "black", "brown"), lwd = 1)
```




```
hist(log10(pta4b.2$results$target),breaks=20,freq=F,xlab="MIC",xaxt="n")
axis(side=1,at=pretty(log10(pta4b.2$results$target)),labels=10**pretty(log10(pta4b.2$results$target)))
lines(x=log10(mic1$mic),y=mic1$n/sum(mic1$n),type="h",col="red",lwd=4)
```