STAT 453 - Exam Notes

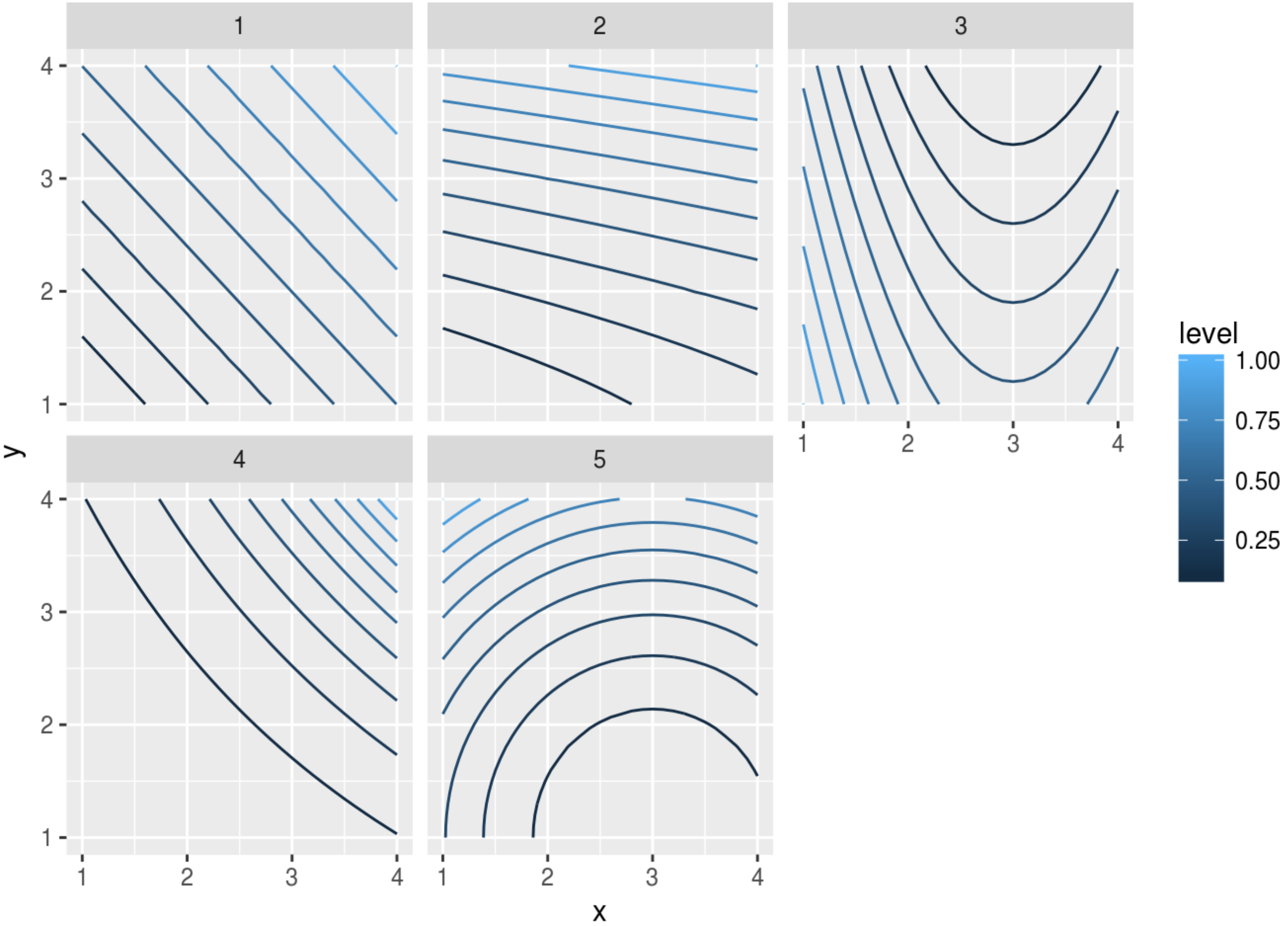
By: Jordan Poles

http://odin.mdacc.tmc.edu/~yyuan/index_code.html (http://odin.mdacc.tmc.edu/~yyuan/index_code.html) ##Question 1 *Use a stoping size equal to cohort size!*

Create your own toxicity scenarios. ####Part 1

Here, I simulate the operating characteristics of a BOIN trial design given 5 different meaningful toxicity scenarios of my own design. I decided to take a slightly novel approach, and use procedural generation to describe several different toxicity scenarios using simple multivariate equations.

I've created a set of functions in R which make it simple to functionally compose a toxicity scenario using any bivariate equation (the results of which are automatically scaled from 0-1), and receive back both a plot of the toxicity scenario in 3D as well as a set of data points (a toxicity level for each level of treatment) which may be input directly into the BOIN R/GUI software in order to discretely simulate the scenario with a given set of drug concentrations. Additionally, the levels for each of the two drugs in question may be set, such that one increase in dosage is not necessarily as significant as a larger increase (or the relationship could be linear for each dosage step).



A summary of our different toxicity scenarios:

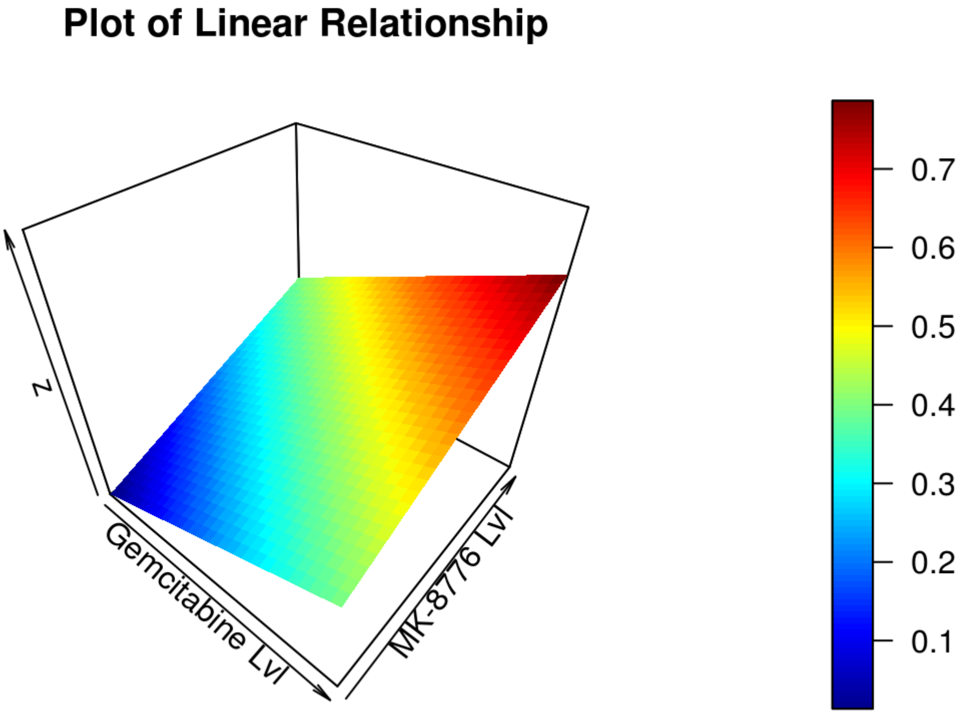
```
xside=c(400, 600, 800, 1000)
yside = c(20,40,80,112)
```

1. A linear relationship with both drugs. Maxmimum toxicity when both drugs are at maximum. Here, and in the following scenarios, we assume that dosage levels are set such that 100% toxicity never occurs (we estimate max 80% toxicity).

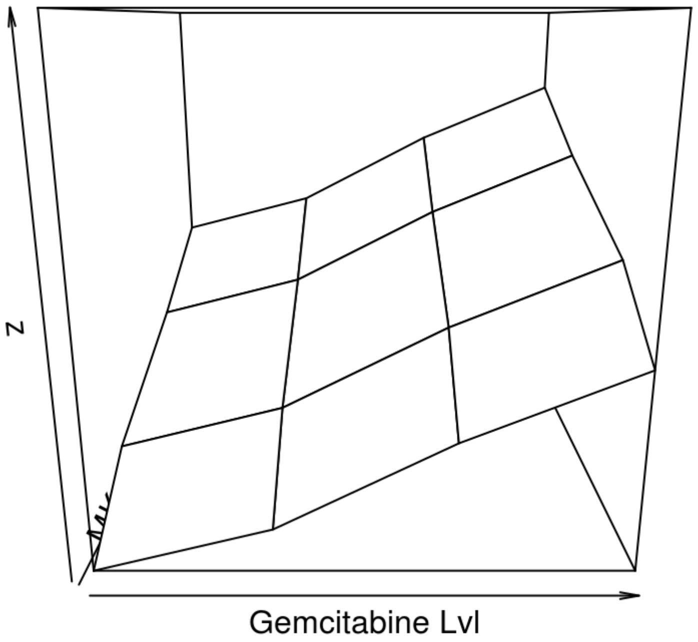
```
f1
```

```
## function(i){(i["x"]+i["y"])}
```

```
maxtox=.8
ds = plot_sim(f1, maxtox, "Linear Relationship", 1)
```



```
scenario_sim(f1, maxtox, xside, yside, 1)
```



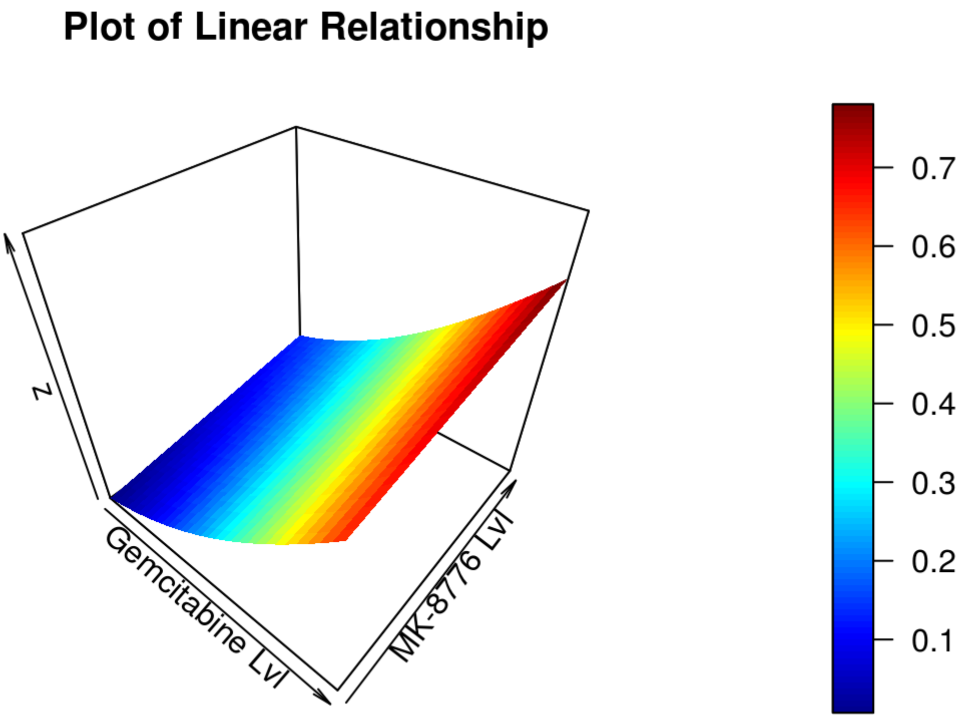
```
##             [,1]      [,2]      [,3]      [,4]
## [1,] 0.00000000 0.1333333 0.2666667 0.4000000
## [2,] 0.08695652 0.2202899 0.3536232 0.4869565
## [3,] 0.26086957 0.3942029 0.5275362 0.6608696
## [4,] 0.40000000 0.5333333 0.6666667 0.8000000
```

2. A linear relationship with Gemcitabine; quadratic growth with MK-8776 level. Maxmimum toxicity when both drugs are at maximum.

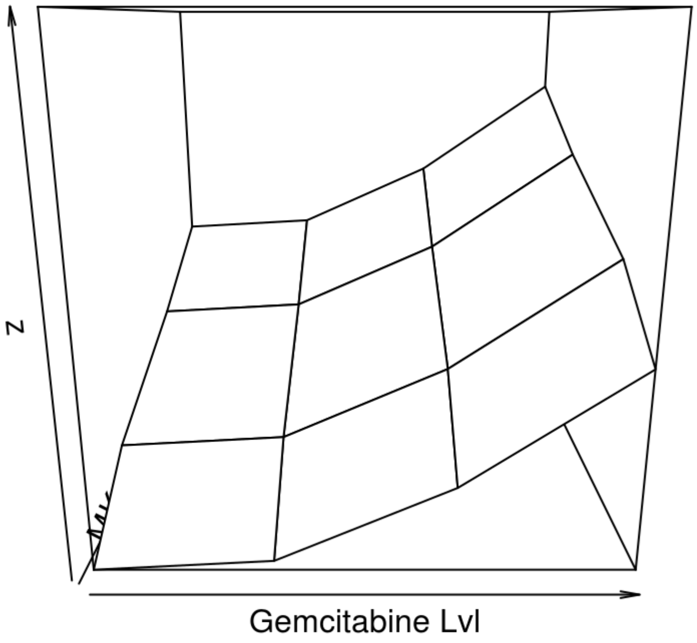
```
f2
```

```
## function(i){(i["x"]+i["y"]**2)}
```

```
ds = plot_sim(f2, maxtox, "Linear Relationship", output=1)
```



```
scenario_sim(f2, maxtox, xside, yside, 1)
```



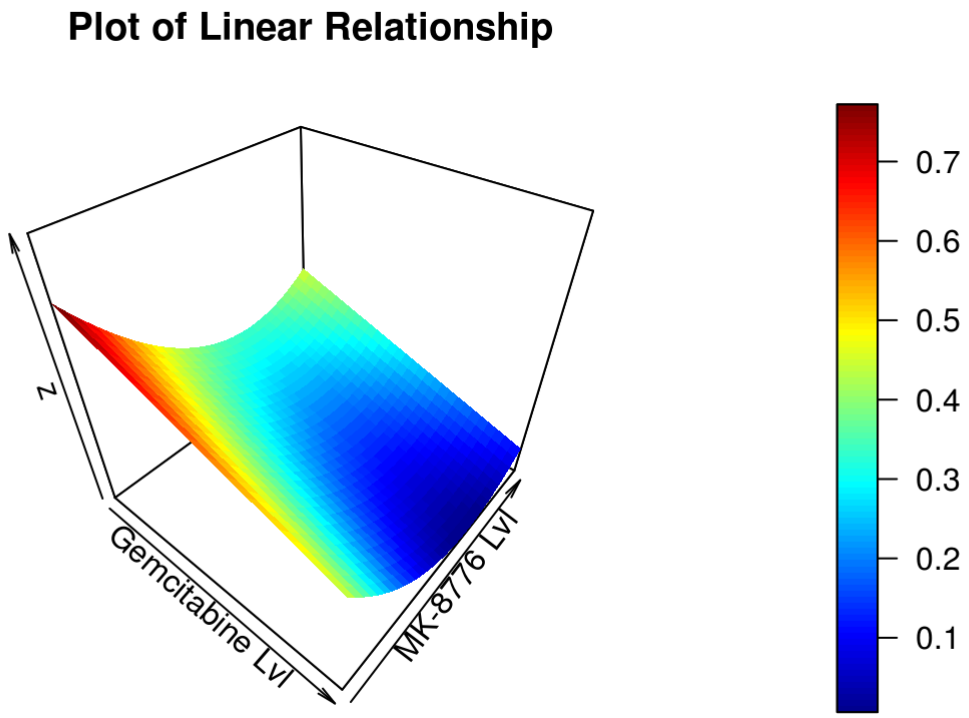
```
##             [,1]      [,2]      [,3]      [,4]
## [1,] 0.00000000 0.1333333 0.2666667 0.4000000
## [2,] 0.01890359 0.1522369 0.2855703 0.4189036
## [3,] 0.17013233 0.3034657 0.4367990 0.5701323
## [4,] 0.40000000 0.5333333 0.6666667 0.8000000
```

3.

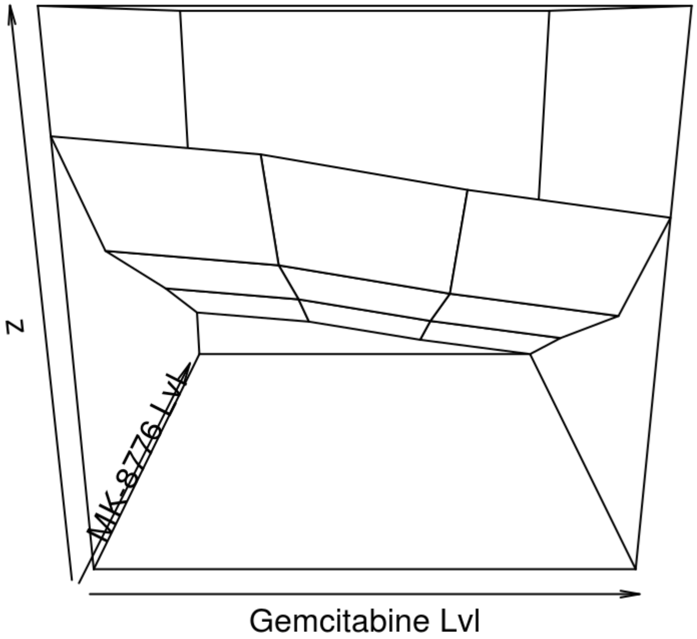
```
f3
```

```
## function(i){((i["x"]-3)**2-i["y"])}
```

```
ds = plot_sim(f3, maxtox, "Linear Relationship", output=1)
```



```
scenario_sim(f3, maxtox, xside, yside, 1)
```



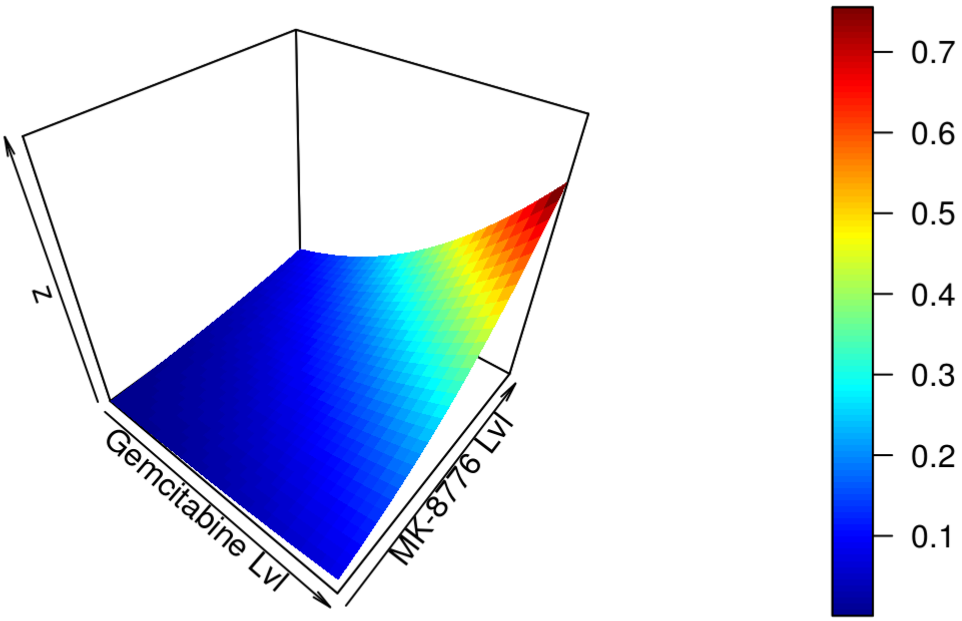
```
##           [,1]      [,2]      [,3]       [,4]
## [1,] 0.8000000 0.5481481 0.3259259 0.13333333
## [2,] 0.7710145 0.5191626 0.2969404 0.10434783
## [3,] 0.7130435 0.4611916 0.2389694 0.04637681
## [4,] 0.6666667 0.4148148 0.1925926 0.00000000
```

4.

```
f4
```

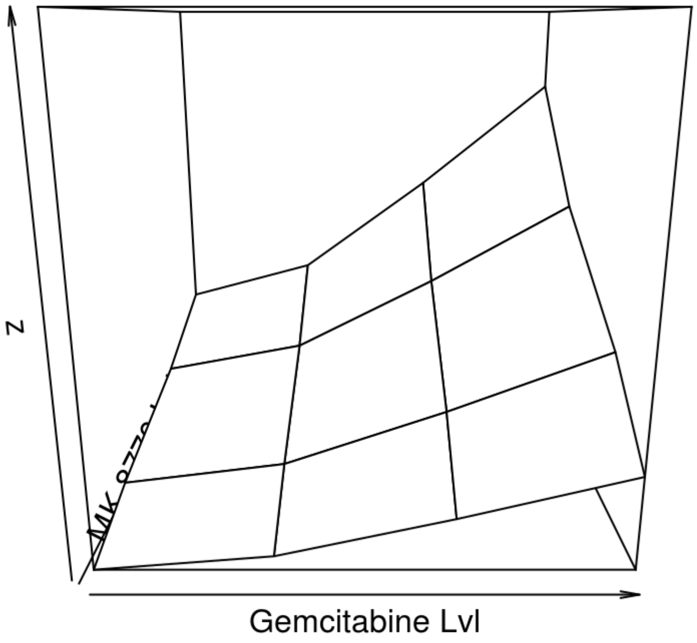
```
## function(i){((i["x"]+3)**4*(i["y"]+3)**4)}
```

```
ds = plot_sim(f4, maxtox, "Linear Relationship", output=1)
```



Plot of Linear Relationship

```
scenario_sim(f4, maxtox, xside, yside, 1)
```



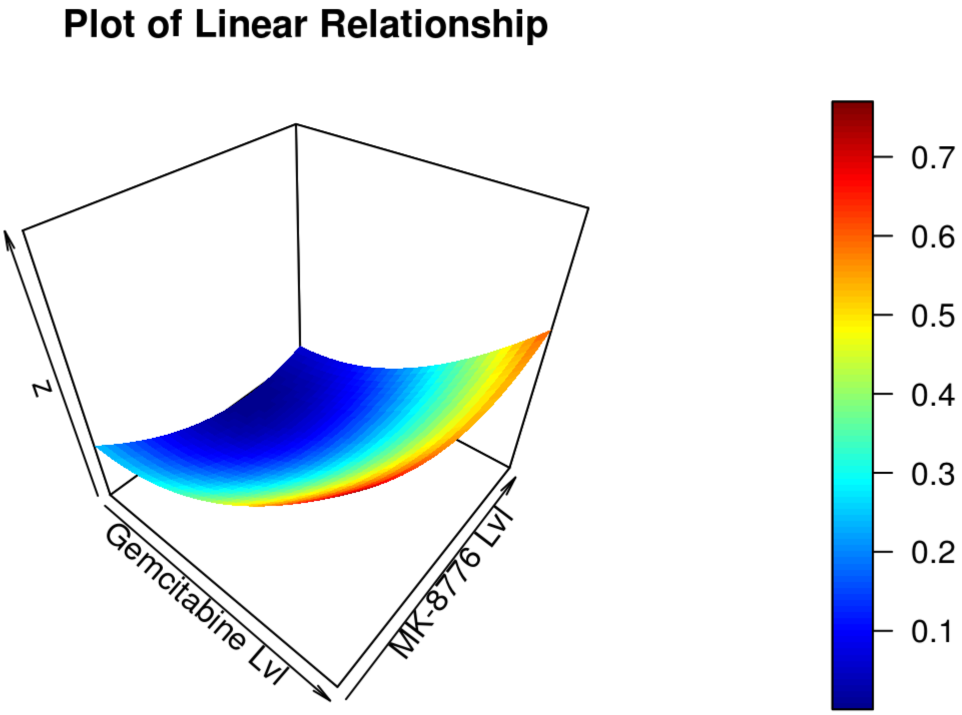
```
##              [,1]       [,2]      [,3]      [,4]
## [1,] 0.00000000 0.04665028 0.1096058 0.1922849
## [2,] 0.02873919 0.09045333 0.1737378 0.2831149
## [3,] 0.10648431 0.20894918 0.3472276 0.5288279
## [4,] 0.19228487 0.33972277 0.5386932 0.8000000
```

5.

```
f5
```

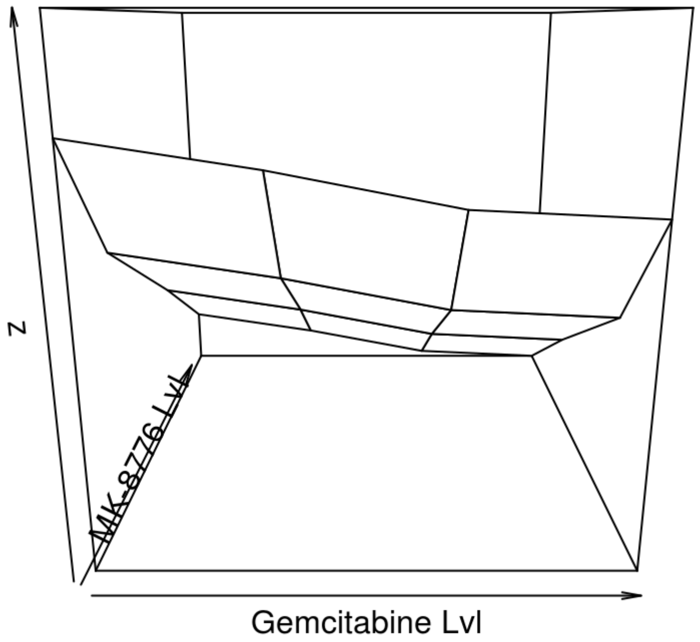
```
## function(i){((i["x"]-3)**2+(i["y"]-1)**2)}
```

```
ds = plot_sim(f5, maxtox, "Linear Relationship", output=1)
```



Plot of Linear Relationship

```
scenario_sim(f5, maxtox, xside, yside, 1)
```



```
##           [,1]      [,2]      [,3]       [,4]
## [1,] 0.8000000 0.5481481 0.3259259 0.13333333
## [2,] 0.7483302 0.4964783 0.2742561 0.08166352
## [3,] 0.6827977 0.4309459 0.2087237 0.01613106
## [4,] 0.6666667 0.4148148 0.1925926 0.00000000
```

Design

```
get.oc.comb(.3, scenario_sim(f2, .8, xside, yside, 0), 42, 42, n.earlystop=NULL, startdose=c(1,
1), ntrial=1000, MTD.contour=FALSE)
```

```
## True toxicity rate of dose combinations:
##  0.00   0.13   0.27   0.40
##  0.02   0.15   0.29   0.42
##  0.17   0.30   0.44   0.57
##  0.40   0.53   0.67   0.80
##
## selection percentage at each dose combination (%):
##  0.00   0.80  19.00   0.80
##  0.00   2.60  28.80   0.70
##  6.90  36.10   0.80   0.00
##  3.50   0.00   0.00   0.00
##
## number of patients treated at each dose combination:
## 42.00  23.27  28.27   4.28
## 21.21  27.51  38.09   3.11
## 20.96  48.09   5.80   0.00
##  9.83   1.81   0.00   0.00
##
## number of toxicity observed at each dose combination:
##  0.00   3.09   7.63   1.76
##  0.41   4.16  10.86   1.28
##  3.61  14.47   2.57   0.00
##  3.98   0.95   0.00   0.00
##
## average number of toxicities: 54.8
## average number of patients: 274.2
## selection percentage of MTD: 0.0
## percentage of patients treated at MTD: 0.0
## percentage of early stopping due to toxicity: 0.00
```

Part 2

Here we create a realistic phase 1 clinical trial scenario and

We want to know if there are any dose transitions recommended by the software which look a bit fishy? ##Question 2 *Set target toxicity probability (under target probability) to .25*

Use the sample size stuff that's in there by default

Question 3

- You should be able to assign a drug quanity for each dosage level.
- You should be able to import/export curves as a CSV list.