Homework9

Pablo

3/10/2019

Contents

Question 12.1 **1**

Question 12.2 **1**

Question 13.1 **2**

Binomial . . . 2

Geometric . . . 2

Poisson . . . 3

Expontential . . . 3

Weibull . . . 3

Question 13.2 **3**

Results . . . 3

Busy airport . . . 3

Question 12.1

Describe a situation or problem from your job, everyday life, current events, etc., for which a design of experiments approach would be appropriate.

I work at a telephone company, so here it is a possible application. There are multiple combinations of plans for the postpaid numbers, each has a data plan, an sms plan and a voice plan. Each of this combinations results in a different price. There are too many combinations to test all of the possible combinations of plans to test on the market, so a design of experiments can help our company to reduce the possibilities by collecting data and make decisions.

Question 12.2

Find a fractional factorial design for this experiment: what set of features should each of the 16 fictitious houses have?

```r
library(FrF2)
```

```
## Loading required package: DoE.base

## Loading required package: grid

## Loading required package: conf.design

##
## Attaching package: 'DoE.base'

## The following objects are masked from 'package:stats':
##
##     aov, lm
```

```
## The following object is masked from 'package:graphics':
##
##     plot.design
```

```
## The following object is masked from 'package:base':
##
##     lengths
```

```r
set.seed(42)
# 10 features and 16 ficticious houses
experiment <- FrF2(nruns= 16,nfactors = 10)
experiment
```

```
##     A  B  C  D  E  F  G  H  J  K
## 1  -1  1  1  1 -1 -1  1 -1  1 -1
## 2   1  1  1  1  1  1  1  1  1  1
## 3  -1 -1  1 -1  1 -1 -1  1  1 -1
## 4  -1  1 -1  1 -1  1 -1 -1 -1  1
## 5   1  1  1 -1  1  1  1 -1 -1 -1
## 6   1 -1  1 -1 -1  1 -1 -1  1  1
## 7   1  1 -1  1  1 -1 -1  1 -1 -1
## 8   1 -1 -1 -1 -1 -1  1 -1 -1 -1
## 9  -1 -1  1  1  1 -1 -1 -1 -1  1
## 10  1 -1  1  1 -1  1 -1  1 -1 -1
## 11 -1  1 -1 -1 -1  1 -1  1  1 -1
## 12  1  1 -1 -1  1 -1 -1 -1  1  1
## 13 -1  1  1 -1 -1 -1  1  1 -1  1
## 14 -1 -1 -1 -1  1  1  1  1 -1  1
## 15  1 -1 -1  1 -1 -1  1  1  1  1
## 16 -1 -1 -1  1  1  1  1 -1  1 -1
## class=design, type= FrF2
```

The result of this experiment is determine which features should the 16 houses should the real state agent use to accomplish the fractional factorial design. The result is above and 1 means to include the feature and -1 means to not include it. For each house, there are different features to show to the survey takers, this way the combinations are reduced and the real state agent can collect significant data.

Question 13.1

For each of the following distributions, give an example of data that you would expect to follow this distribution.

Binomial

For a job hiring process, a technical test is required for all the applicants. For the n applicants, the number of successful test notes might yield a binomial distribution.

Geometric

I play on a soccer team every Sunday. My team usually wins the loses, so the number of weeks that my team loses before we actually win, might yield a geometric distribution.

Poisson

The expected number of requests/hits on a popular blog (server) in a day might yield a poisson distribution.

Expontential

The time between the requests on a popular blog (server) in a day might yield a exponential distribution.

Weibull

In a extracting natural resources factory, the time the extracting machine works fine before having to stop might yield a weidbull distribution. Since it is a machine it is expected that failure rate will increase over time, so a k > 1 is expected.

Question 13.2

Use the Arena software (PC users) or Python with SimPy (PC or Mac users) to build a simulation of the system, and then vary the number of ID/boarding-pass checkers and personal-check queues to determine how many are needed to keep average wait times below 15 minutes. [If you're using SimPy, or if you have access to a non-student version of Arena, you can use 1 = 50 to simulate a busier airport.]

For this problem I used simpy v2. I used FIFO queues for the board checker queues and for the scanning queues, after I designed the experiment I made changes to the number of boarding checkers and the number of scanners to find when the wait time was less than 15 minutes. Here are the results.

Here are the results from the python. I made 5 simulations each one with 50 iterations to find that the best combination is:

4 boarding check assitants 4 scan assistants

to have less than 15 minute wait time. This combination resulted in 3.42 minutes of wait time with 50 iterations. Hiring more assitants might not be necessary because the goal is already accomplished.

Results

What else could be done?

Remember what Dr. Sokol said about simulations, these results should be compared with real life data to see if the results are reasonable (out of scope of this homework).

Busy airport

Now what will happen when $lambda = 50$ arrivals per minute

After some test a good range to test was from 30-45 checkers/scanners The best result was:

38 board checkers 38 scanners with 1.46 minutes of wait time.

```
----------------Results #1:   1 checker and 1 scanner--------------------
mean all time: 301.241175409
mean check time: 0.742379258194
mean scan time: 0.750675111619
mean wait time: 299.748121039

----------------Results# 2:  2 checker and 2 scanner--------------------
mean all time: 192.977745828
mean check time: 0.751846684337
mean scan time: 0.749708118659
mean wait time: 191.476191025

----------------Results # 3:  4 checker and 4 scanner--------------------
mean all time: 4.91652445277
mean check time: 0.740936847763
mean scan time: 0.750846308592
mean wait time: 3.42474129642

----------------Results #4: 3 checker and 3 scanner--------------------
mean all time: 29.8594343708
mean check time: 0.746534378458
mean scan time: 0.750424529178
mean wait time: 28.3624754632

----------------Results #5: 4 checker and 3 scanner--------------------
mean all time: 34.7001243767
mean check time: 0.745714093182
mean scan time: 0.750324004228
mean wait time: 33.2040862793
```

Figure 1: results regular airport

```
----------------Results #1:  30 checker and 30 scanner--------------------
mean all time: 81.9009418386
mean check time: 0.747599468753
mean scan time: 0.749732300825
mean wait time: 80.403610069

----------------Results# 2:  35 checker and 35 scanner--------------------
mean all time: 29.4880065284
mean check time: 0.747383562653
mean scan time: 0.750364924861
mean wait time: 27.9902580409

----------------Results # 3:  38 checker and 38 scanner--------------------
mean all time: 2.95398527246
mean check time: 0.748054765029
mean scan time: 0.749677517499
mean wait time: 1.45625298993

----------------Results #4: 40 checker and 40 scanner--------------------
mean all time: 2.58773289098
mean check time: 0.747793619664
mean scan time: 0.749701418523
mean wait time: 1.09023785279

----------------Results #5: 45 checker and 45 scanner--------------------
mean all time: 2.1958284752
mean check time: 0.747697186468
mean scan time: 0.749747399911
mean wait time: 0.698383888816
```

Figure 2: results of busy airport