ISYE-6501
Fall 2024

Homework 9

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.

As a data analyst in a hospital, a situation in which I can use a design of experiments would be determining the effect of a plan for reducing readmissions. I would compare patients who received this new plan compared to patients who have not and assess the difference. The factors I will test are the type of intervention (old vs new plan), follow-up method (physician appointment or follow up phone call), and several patient demographics (age, comorbidities, etc). I would assign random patients to the control group (old plan) vs intervention group (new reduction plan) and collect their 30-day readmission rates.

Question 12.2

To determine the value of 10 different yes/no features to the market value of a house (large yard, solar, roof, etc.), a real estate agent plans to survey 50 potential buyers, showing a fictitious house with different combinations of features. To reduce the survey size, the agent wants to show just 16 fictitious houses. Use R's FrF2 function (in the FrF2 package) to find a fractional factorial design for this experiment: what set of features should each of the 16 fictitious houses have? Note: the output of FrF2 is 1 (include) or -1 (don't include) for each feature.

Below is the code/ R markdown, find the input in black, the comments in green, and the output in blue.

<u>CODE:</u>

```
> #load the correct library
> library(FrF2)
> #apply the FrF2 function on 16 fictitious houses with 10 features.
> FrF2(16, 10)
##     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
```

Question 13.1

For each of the following distributions, give an example of data that would expect to follow this distribution (besides the examples already discussed in class).

a. **Binomial:** The number of successful surgeries out of a fixed number of surgeries performed in a hospital. Let's say a surgeon performs 10 surgeries and you want to find his success rate.
b. **Geometric:** The number of days until a patient returns to the hospital after discharge for a specific reason or diagnosis. Let's say you are interested in how many days until the first readmission occurs.
c. **Poisson:** The number of ER visits at a hospital during a given hour. Let's say you are interested in seeing the number of patients arriving in the ER in one hour.
d. **Exponential:** The time until the next patient arrives for a scheduled appointment at a clinic. Let's say you want to measure the time intervals between patient arrivals and these intervals are memoryless.
e. **Weibull:** The time until failure of a piece of equipment. Let's say you want to model the lifespan of a ventilator where rate of failure can change over time. The older the equipment the higher the rate of failure.

Question 13.2

In this problem, you can simulate a simplified airport security system at a busy airport. Passengers arrive according to a Poisson distribution with lambda1 = 5 per minute (mean interarrival rate = 0.2 minutes) to the ID/boarding-pass check queue, where there are several servers who each have exponential service time with mean rate = 0.75 minutes. After that, the passengers are assigned to the shortest of the several personal-check queues, where they go through the personal scanner (time is uniformly distributed between 0.5 minutes and 1 minute).

Use the Arena software (PC users) or Python with SimPy (PC or MAC) 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 time below 15 minutes.

Had too many troubles trying to download Arena software so I decided to use SimPy instead.

For each of these I used 100 replications of simulations. I started the simulation with 30 checkers and scanners and got an average wait time of 74.39 minutes. I was able to then keep reducing the amount of checkers and scanners to find the lowest amount while still having a wait time under 15 minutes.

I found that running the simulation with 36 scanners and checkers gave me an average wait time of 16.64 minutes. **After running the simulation with 37 scanners and checkers I got an average wait time of 8.04 minutes. Adding more scanners and checkers just reduces the wait times increasingly but loses a lot of value so the optimal amount of checkers and scanners is 37.**