Homework 9

Jeff Tilton

10/22/2018

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.

Response

There are often times in analytics when the required data are not readily available and it is necessary to collect it. The collection process may be costly so it is necessary to collect as much data as is required to answer the question and do it efficiently to keep costs low. This process is called "Design of Experiments," (DOE).

There are many situations when DOE would be appropriate. One such case is in marketing/web site design. For example, an online retailer might want to redisign their site to encourage users to click on a button for 'easy purchase'. A test can be set up for button placement to see where it is most clicked by users. Althogh this test does not seem like it would be expenxive to collect the data, there could be revenue lost because some of the users ended up not buying a particular product.

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.

Response

Factorial design is used to determine an appropriate sized test to determine how a set of independent variables (10 different yes/no features) o dependent variable (market value of a house). The R FrF2 library FrF2 function provides a regular Fractional Factorial 2-level designs. The input are the number of data points (16 fictitious houses) and the number of independent variables. The output is a boolean matrix comprised of 1, -1, which determines the set of features each house should have to have a statistically significant test.

Randomness is introduced to the test therefore, the same inputs will not guarantee the same output uless the seed is set.

```
##     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
```

```
##        A     B     C     D     E     F     G     H     J     K
## 1   TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## 2   TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
## 3  FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE
## 4  FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE
## 5   TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE
## 6  FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE
## 7  FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE
## 8   TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE
## 9   TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE
## 10 FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE
## 11  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE
## 12 FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE
## 13  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE
## 14 FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE
## 15  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE
## 16 FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE
```

The first matrix above demonstrates the output of a single design. The second matrix is a boolean matrix for design 1 == design 2 to demonstrate that randomness exists within the function and that 2 tests of equal inputs will not necessarily give equal outputs.

Question 13.1

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

- Binomial
- Geometric
- Poisson
- Exponential
- Weibull

Binomial

The binomial distribution is the probability of getting k successes out of n independent identically distributed ($i.i.d$) Bernoulli trials. The probability mass function (PMF) is seen below.

$$f(n,k,p) = \binom{n}{k} p^k (1-p)^{n-k}$$

A student guessing on multiple choice exam questions is an example of the binomial distribution. The questions can be thought of as the Bernoulli trials, if there are 4 answers for each question then each question has a .25 probability of success. If the student guesses on 5 of them then the chances that he gets 3 of them right are:

$$n = 5$$
$$k = 3$$
$$p = 0.25$$

not good or 0.09.

Geometric

This distribution was the hardest for me to comprehend, but I think I now understand it. THe geometric distribution is the probability that there are **exactly** x-1 failures before a success. I added the emphasis because when I heard it in lecture I thought it was describing at least x-1 failures before a success The PMF is seen below:

$$P(X = x) = (1 - p)^x p$$

An example of the geometric distribution is a get out the vote campaign trying to find likely voters. If there is a .4 chance of finding a likely voter then what is the chance of finding a voter after knocking on 5 doors?

so in our example:

$$x = 5$$
$$p = 0.4$$

There is a 3% chance that the campaigner would have to knock on 5 doors before finding a supporter.

Poisson

The Poisson distribution is a discrete probability distribution that expresses the probability of a given number of events occuring in a fixed interval of time or space if these events occur with a known constant rate and independently of the time since the last event. THe PMF is seen below:

$$f(x) = \frac{\lambda^x e^{-\lambda}}{x!}$$

An example of when the Poisson might be able to be used is traffic flow, cars arriving at a given stoplight.

Exponential

The exponential distribution describes time between events in a Poisson point process. The PMF is seen below:

$$f(x) = \lambda e^{-\lambda x}$$

Therefore taking the above example if 5 cars arrive at a stoplight a minute then the exponential distribution would describe the time in between the arrival of each car.

Weibull

The weibull distribution can model time in between failures. THe PMF is seen below:

$$f(x, \lambda, k) = \frac{k}{\lambda}(\frac{x}{\lambda})^{k-1}e^{-(\frac{x}{\lambda})^k}$$

An example of the Weibull distribution would be how long until a specific model of car breaks down, or perhaps how long until individual parts within the car break down.

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 lambda = 5 per minute (i.e., 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. [Hint: model them as one block that has more than one resource.] 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 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 lambda = 50 to simulate a busier airport.]

Response:

After running through Simpy I found that having a minimum of 7 checkers and 3 scanners was necessary to keep the average wait time under 15 minutes for an arrival rate of 5 per minute. With an arrival time of 50 people per minute there would need to be 54 checkers and 2 scanners. Thy Jupyter notebook is attached seperately.