STAT590: Assignment 1

Nick Graetz

January 27, 2019

1.

Given this data, we have no way of knowing how quickly these individuals would have recovered had they not taken Vitamin C. This represents the potential outcome of no treatment (treatment being Vitamin C). Indeed, it could be the case that 100% of people who don't take Vitamin C get over their cold within a week and this treatment is actually harmful. This is of course unlikely in this example, but we cannot show that isn't the case without a control group who do not receive the treatment.

2.(a)

In a randomized experiment, we want to make sure the only thing varying between the treatment and control groups is the treatment variable under study (seeding). Flying the plane through the clouds during non-seeding days ensures that there is no additional difference introduced between the two groups simply related to the plane moving through the clouds. If the pilots knew the result of the randomization (whether or not seeding was to occur), it may have also introduced an additional difference by them altering their behavior based on that knowledge (e.g. flying in a different pattern while seeding vs. not seeding).

2.(b)

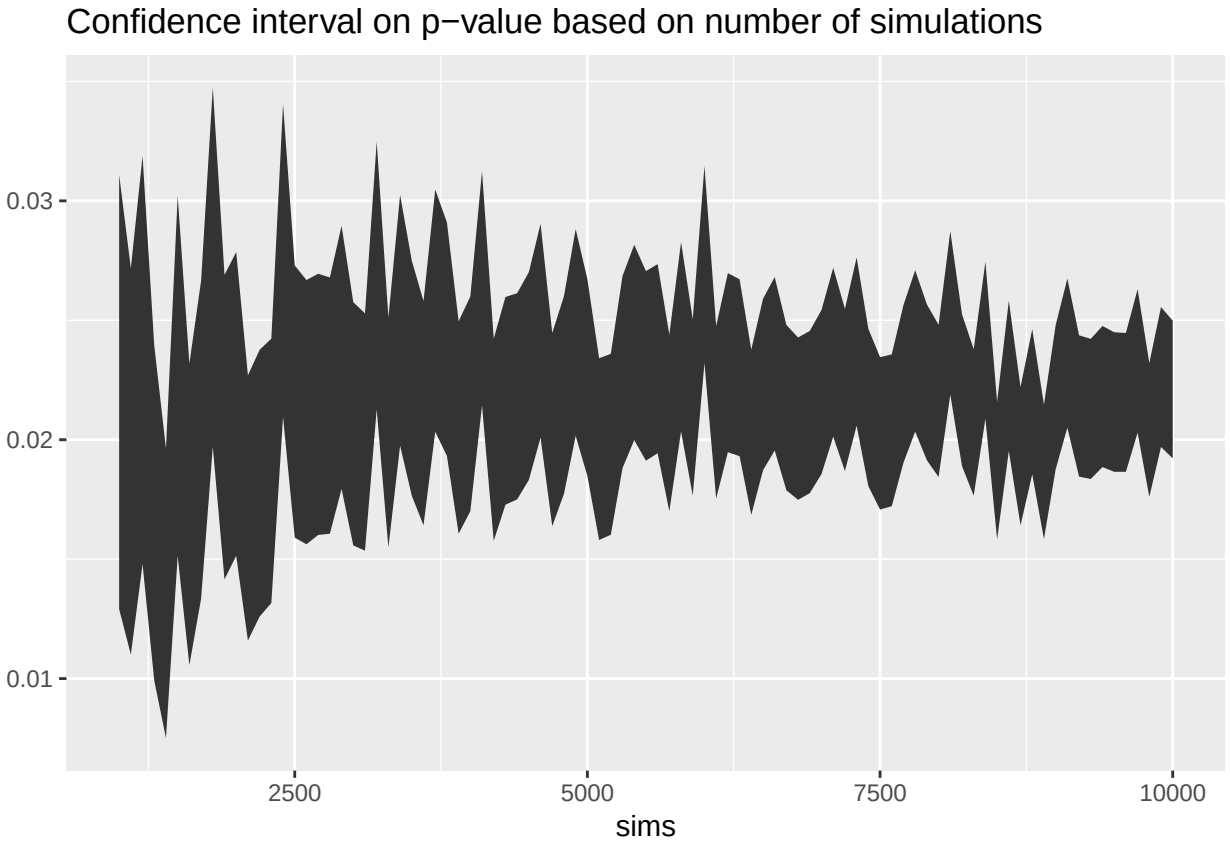
```
treat.effect.samplemean.montecarlo.test.func(10000,treatment,control)
```

```
##     pval    lowerci    upperci
## 1 0.0217 0.01884424 0.02455576
```

2.(c)

Based on the plot below running our MC function with different numbers of simulations, the p-value seems to converge fairly well by the time we get up to running with 10,000 simulations. We would want to iterate further out to make sure.

```
p_cis <- rbindlist(lapply(seq(1000,10000,100),
                          treat.effect.samplemean.montecarlo.test.func,
                          treated.r = treatment, control.r = control))
p_cis$sims <- seq(1000,10000,100)
ggplot(data=p_cis) +
  geom_ribbon(aes(x=sims,
                  ymin=lowerci,
                  ymax=upperci)) +
  labs(title = 'Confidence interval on p-value based on number of simulations')
```



2.(d)

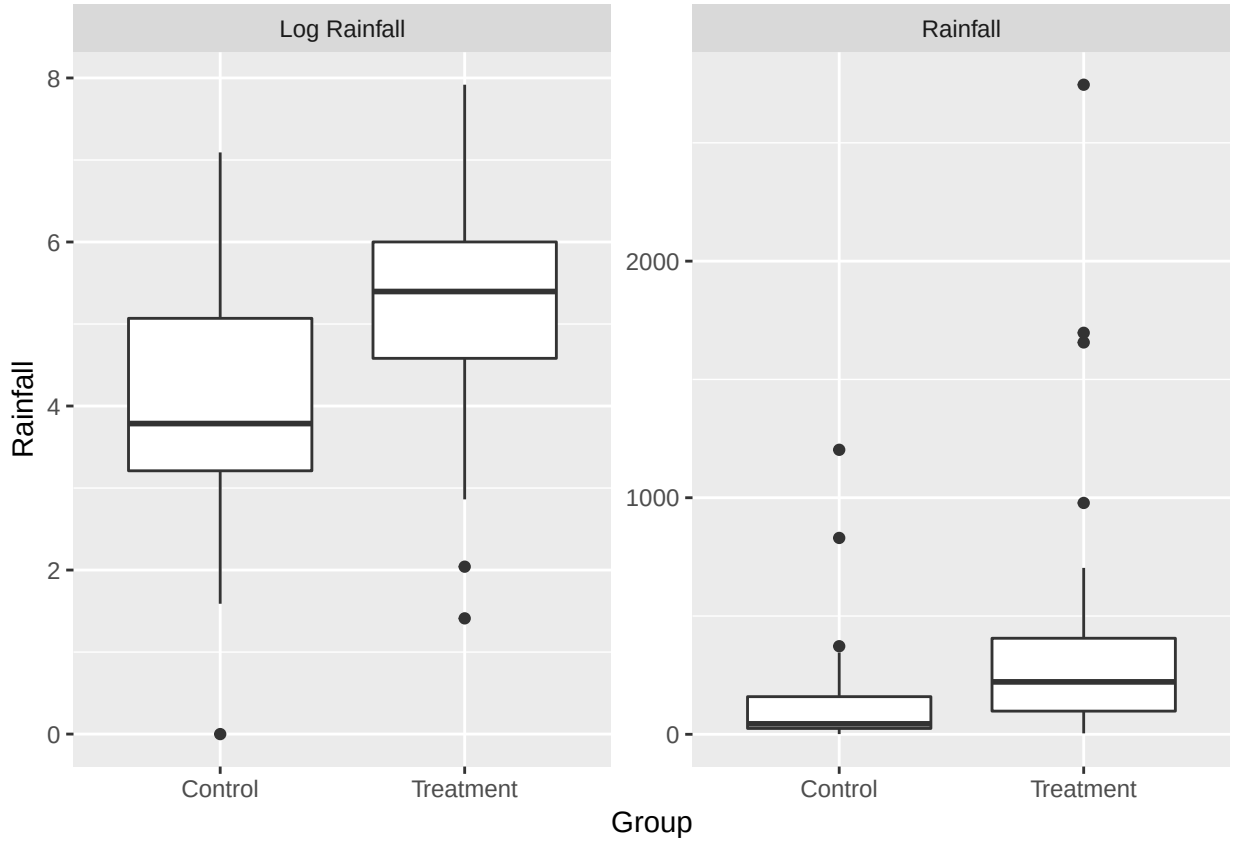
```
treat.effect.samplevariance.montecarlo.test.func=function(treated.r,control.r,K){
  # Create vectors for r and Z, and find total number in
  # experiment and number of treated subjects
  r=c(treated.r,control.r);
  Z=c(rep(1,length(treated.r)),rep(0,length(control.r)));
  N=length(r);
  m=length(treated.r);

  # Observed test statistic
  obs.test.stat=var(r[Z==1])-var(r[Z==0]);

  # Monte Carlo simulatoin
  montecarlo.test.stat=rep(0,K);
  for(i in 1:K){
    treatedgroup=sample(1:N,m);  # Draw random assignment
    controlgroup=(1:N)[-treatedgroup];
    # Compute test statistic for random assignment
    montecarlo.test.stat[i]=var(r[treatedgroup])-var(r[controlgroup]);

  }
```

```
  # Monte Carlo p-value is proportion of randomly drawn
  # test statistics that are >= observed test statistic
  pval=sum(montecarlo.test.stat>=obs.test.stat)/K;
  # 95% CI for true p-value based on Monte Carlo p-value
  lowerci=pval-1.96*sqrt(pval*(1-pval)/K);
  upperci=pval+1.96*sqrt(pval*(1-pval)/K);
  list(pval=pval,lowerci=lowerci,upperci=upperci);
}
treat.effect.samplevariance.montecarlo.test.func(treatment,control,10000)
```

```
## $pval
## [1] 0.0927
##
## $lowerci
## [1] 0.08701577
##
## $upperci
## [1] 0.09838423
```

2.(e)

```
plot <- data.table(Rainfall=c(treatment,control),
                   Group=c(rep('Treatment',length(treatment)),
                           rep('Control',length(control))),
                   Type=rep('Rainfall', length(c(treatment,control))))
plot_log <- copy(plot)
plot_log[, Rainfall := log(Rainfall)]
plot_log[, Type := 'Log Rainfall']
plot <- rbind(plot, plot_log)
boxes <- ggplot() +
  geom_boxplot(data=plot,
               aes(x=Group,
                   y=Rainfall)) +
  facet_wrap(~Type, scales='free_y')
print(boxes)
```



Additive treatment effect model implies that the distribution of observed outcomes among treated is the same as among control. Our boxplot of rainfall in normal space suggests this is not the case (much more dispersion and extreme outliers among treated). In log space, the disperions are roughly equal between treated and control groups.

2.(f)

```
treat.effect.samplemean.montecarlo.test.func(10000,log(treatment),log(control))
```

```
##     pval     lowerci     upperci
## 1 0.0075 0.005808967 0.009191033
```

2.(g)

```
wilcox.test(log(treatment),log(control),conf.int=TRUE)$conf.int
```

```
## [1] 0.2816254 2.0967816
## attr(,"conf.level")
## [1] 0.95
```

```
# Compare point estimate of treatment effect under Fisher and Wilcoxon.
# Mean difference
mean(log(treatment) - log(control))
```

```
## [1] 1.144458
```

```
# Median of the difference between a sample from x and a sample from y.
wilcox.test(log(treatment),log(control),conf.int=TRUE)$estimate
```

```
## difference in location
##                1.26038
```

2.(h)

```
multiplicative_treatment_effect <- exp(mean(log(treatment) - log(control)))
```

We can conclude with a high degree of confidence ($p < 0.05$) that the seeding treatment resulted in 3.14 times higher rainfall ($p = 0.006$).

3.

There are many problematic assumptions in Mr. X's argument. First, their conceptualization of potential outcomes is flawed because money is probably being spent on public health research in all years. As we don't observe what would have unfolded in those years if vaccinations, peniciliin, etc. had not been introduced, it is a hard comparison to make. There are also many other variables changing over time that may affect the crude death rate. Additionally, many of these public health innovations may primarily affect infants while the crude mortality rate captures changes at all ages. For example, an aging population drives up the crude mortality rate because older age is associated with higher mortality rates, which may be offsetting huge improvements to infant and child mortality.