Exercise 3 - Question 1 Solution

Question 1

Data: 'process.csv'

The data are from an experiment designed to assess the effect of temperature on the output of a manufacturing process. For this experiment temperatures of 50 or 100 degrees were randomly assigned to process runs and yield was recorded from each run.

1.1

```
d <- read.csv("process.csv")
xbar<-tapply(d$output,d$temp,mean) #sample means
n<-tapply(d$output,d$temp,length) #sample sizes
sd<-tapply(d$output,d$temp,sd) #sample SDs
se<-sqrt(sum(sd^2/n)) # SE of difference between sample means
Z<-diff(xbar)/se #Z statistic
Z
```

```
##      100
## 3.340701
```

```
#Rejection rule: reject if |Z|>97.5% quantile of N(0,1)
abs(Z)>qnorm(0.975)
```

```
##  100
## TRUE
```

```
#P-value: 2-sided tail probability
p<-2*(1-pnorm(Z))
p
```

```
##          100
## 0.0008356714
```

```
# Confidence interval
# Lower limit:
diff(xbar) - qnorm(0.975)*se
```

```
##      100
## 55.51298
```

```
# Upper limit:
diff(xbar) + qnorm(0.975)*se
```

```
##     100
## 213.115
```

1.2

```
N<-5000 #number of samples
set.seed(79)
```

```r
n<-nrow(d)
sim_result<-replicate(N,{
    sim_data<-d # Use the data set as a template for your simulated data sets
    sim_data$output<-rnorm(n,mean=1000,sd=100)
    xbar<-with(sim_data,tapply(output,temp,mean))
    samplevar<-with(sim_data,tapply(output,temp,var))
    samplesize<-c(n/2,n/2)
    Z<-diff(xbar)/sqrt(sum(samplevar/samplesize))
    abs(Z)>qnorm(0.975)
})
mean(sim_result)
```

```
## [1] 0.0568
```

1.3

```r
N<-5000 #number of samples
set.seed(80)
n<-nrow(d)
sim_result<-replicate(N,{
    sim_data<-d
    sim_data$output<-rnorm(n,mean=10,sd=20)# Changed mean and sd of population
    xbar<-with(sim_data,tapply(output,temp,mean))
    samplevar<-with(sim_data,tapply(output,temp,var))
    samplesize<-c(n/2,n/2)
    Z<-diff(xbar)/sqrt(sum(samplevar/samplesize))
    abs(Z)>qnorm(0.975)
})
mean(sim_result)
```

```
## [1] 0.053
```

Result is very similar because the Z-statistic is invariant to changes in the mean and SD, because it is based on the difference between means and is scaled by the SD. In fact, if you use the same random seed, you will get the exact same answer.

1.4

```r
N<-5000 #number of samples
set.seed(100)
n<-nrow(d)
sim_result<-replicate(N,{
    sim_data<-d
    sim_data$output<-rt(n,df=3)# Used a t distribution with 3 df
    xbar<-with(sim_data,tapply(output,temp,mean))
    samplevar<-with(sim_data,tapply(output,temp,var))
    samplesize<-c(n/2,n/2)
    Z<-diff(xbar)/sqrt(sum(samplevar/samplesize))
    abs(Z)>qnorm(0.975)
})
mean(sim_result)
```

```
## [1] 0.0492
```

```r
# Try with uniform distribution
sim_result<-replicate(N,{
    sim_data<-d
    sim_data$output<-runif(n,0,1)# Used U(0,1) distribution
    xbar<-with(sim_data,tapply(output,temp,mean))
    samplevar<-with(sim_data,tapply(output,temp,var))
    samplesize<-c(n/2,n/2)
    Z<-diff(xbar)/sqrt(sum(samplevar/samplesize))
    abs(Z)>qnorm(0.975)
})
mean(sim_result)
```

```
## [1] 0.0586
```

1.5

```r
N<-5000 #number of samples
set.seed(102)
n<-10
# Create a template for the simulated data sets
sim_data<-data.frame(temp=c(rep(50,n/2),rep(100,n/2)),output=rep(NA,n))
sim_result<-replicate(N,{
    sim_data$output<-rnorm(n,0,1)# Used a N(0,1) distribution
    xbar<-with(sim_data,tapply(output,temp,mean))
    samplevar<-with(sim_data,tapply(output,temp,var))
    samplesize<-c(n/2,n/2)
    Z<-diff(xbar)/sqrt(sum(samplevar/samplesize))
    abs(Z)>qnorm(0.975)
})
mean(sim_result)
```

```
## [1] 0.0822
```

```r
set.seed(111)
# Create a template for the simulated data sets
sim_data<-data.frame(temp=c(rep(50,n/2),rep(100,n/2)),output=rep(NA,n))
sim_result<-replicate(N,{
    sim_data$output<-rt(n,df=3)# t distribution with 3 df
    xbar<-with(sim_data,tapply(output,temp,mean))
    samplevar<-with(sim_data,tapply(output,temp,var))
    samplesize<-c(n/2,n/2)
    Z<-diff(xbar)/sqrt(sum(samplevar/samplesize))
    abs(Z)>qnorm(0.975)
})
mean(sim_result)
```

```
## [1] 0.0786
```

```r
set.seed(111)
# Create a template for the simulated data sets
sim_data<-data.frame(temp=c(rep(50,n/2),rep(100,n/2)),output=rep(NA,n))
sim_result<-replicate(N,{
    sim_data$output<-runif(n)# U(0,1)
```

```r
    xbar<-with(sim_data,tapply(output,temp,mean))
    samplevar<-with(sim_data,tapply(output,temp,var))
    samplesize<-c(n/2,n/2)
    Z<-diff(xbar)/sqrt(sum(samplevar/samplesize))
    abs(Z)>qnorm(0.975)
})
mean(sim_result)
```

```
## [1] 0.0858
```

How to obtain the test statistic from the 't.test' function

```r
result<-t.test(d$output[d$temp==50],d$output[d$temp==100])
result
```

```
##
##  Welch Two Sample t-test
##
## data:  d$output[d$temp == 50] and d$output[d$temp == 100]
## t = -3.3407, df = 56.917, p-value = 0.00148
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -214.82630  -53.80164
## sample estimates:
## mean of x mean of y
##  899.8219 1034.1358
```

```r
names(result)
```

```
## [1] "statistic"   "parameter"   "p.value"     "conf.int"    "estimate"
## [6] "null.value"  "alternative" "method"      "data.name"
```

```r
result$statistic
```

```
##         t
## -3.340701
```