Lab2-Part2-a-solution

Isabel Sassoon

Part 1 Distributions

(1) The evidence shows that 25% of people exposed to a specific virus will show symptoms. If we have a group of 5 people all exposed to this virus.

(a) Use the dbinom function to compute the probabilities that 0,1,2,3,4,5 of the people who are exposed will show symptoms.

```
dbinom(0:5, size = 5, prob = 0.25)
```

```
## [1] 0.2373046875 0.3955078125 0.2636718750 0.0878906250 0.0146484375
## [6] 0.0009765625
```

To specify the **binomial** distribution you need parameters **size** (in this case 5) and **prob** the porbability of infection.

`(b) What is the probability that at least one will show symptoms? (hint: use pbinom)`

The probability that at least one will show symptoms is 1- the probability that none will show symptoms. The latter can be obtained with pbinom(0,5,0.25)

```
1-pbinom(0,5,0.25)
```

```
## [1] 0.7626953
```

Therefore the probability is 0.76 that at least someone will show symptoms out of the 5.

---

(2) Assume that among diabetics the fasting blood level of glucose is approximately normally distributed with a mean of 105 mg per 100 ml and a standard deviation of 9mg per 100ml.

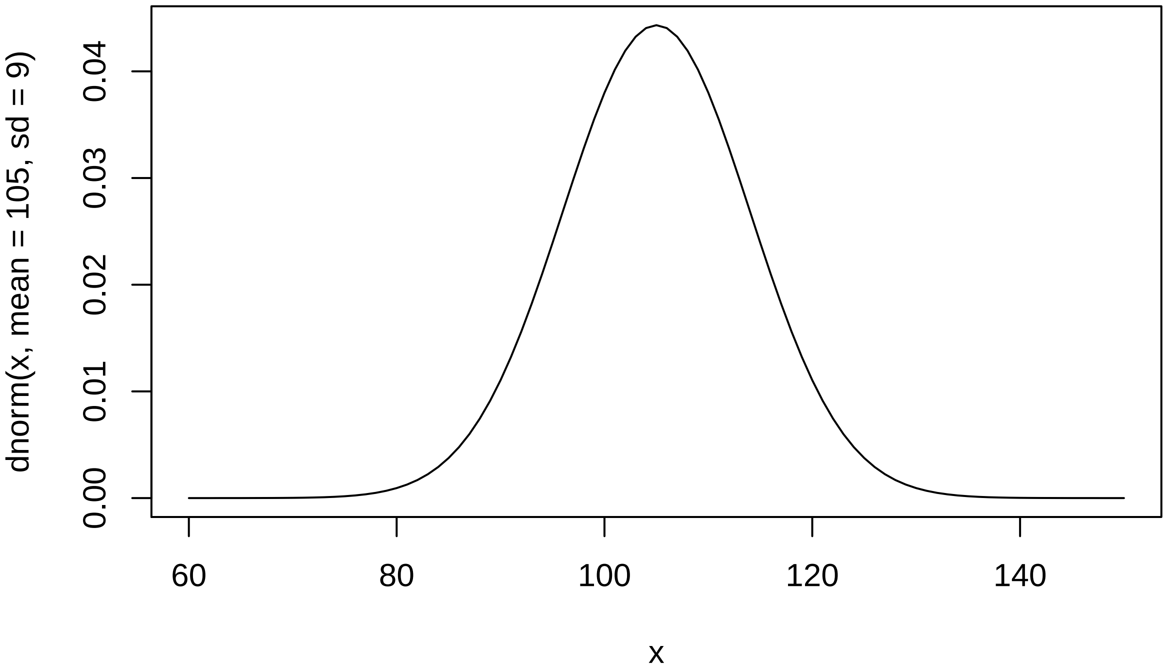
plot the density function for this distribution.

In order to do this first we need to generate the x values that are suitable for a normal distribution with the parameters above

```
x<-seq(60,150)
y<-dnorm(x,mean = 105, sd=9)
```

Then we can plot this

```
plot(x, dnorm(x,mean = 105, sd=9), type = "l")
```



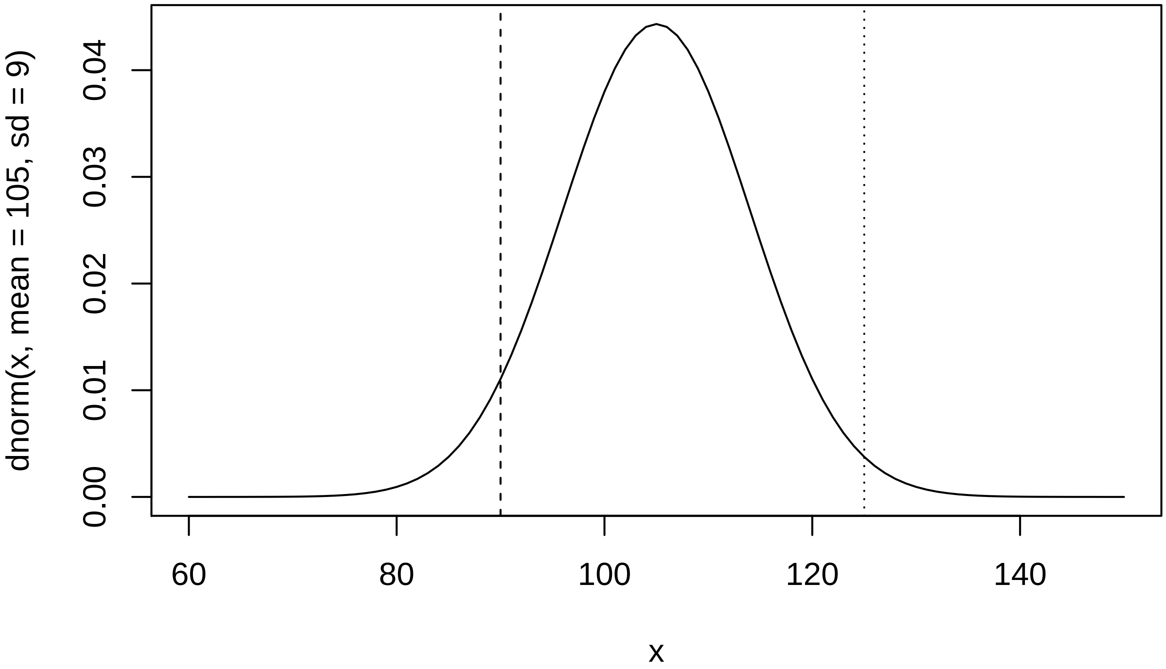
–note–

it is also possible to use ggplot to do this.

what proportion of diabetics have levels between 90 and 125 mg per 100ml? (This quantity is represented by the area under the normal curve between 90 and 125)

We can visualise this using

```r
plot(x, dnorm(x,mean = 105, sd=9), type = "l")
abline(v=90, lty=2)
abline(v=125, lty=3)
```



you should see from the plot that most of the area under the normal curve is contained between the two vertical lines. Therefore we should expect that the proportion of diabetics with levels between 90 and 125 mg per 100 ml to be high.

Use the **pnorm** function to calculate this proportion. (you may need to use it twice)

```
pnorm(125, mean = 105, sd=9)-pnorm(90, mean = 105, sd=9)
```

```
## [1] 0.9390755
```

94% of the diabetics have values between 90 and 125

what level cuts off the lower 10% of diabetics? (hint: use **qnorm**)

```
qnorm(0.1, mean = 105, sd=9)
```

```
## [1] 93.46604
```

```
This cut off value is 93ml
```

Part 2 Confidence Intervals

(2) Read in the auto-mpg.csv data set into a new R notebook

```
mpg<-read.csv("data/auto-mpg.csv")
```

(3) Visualise the data and check which columns from the continuous variables look normally distributed

First lets look at the numerical summaries of the variables

```
summary(mpg)
```

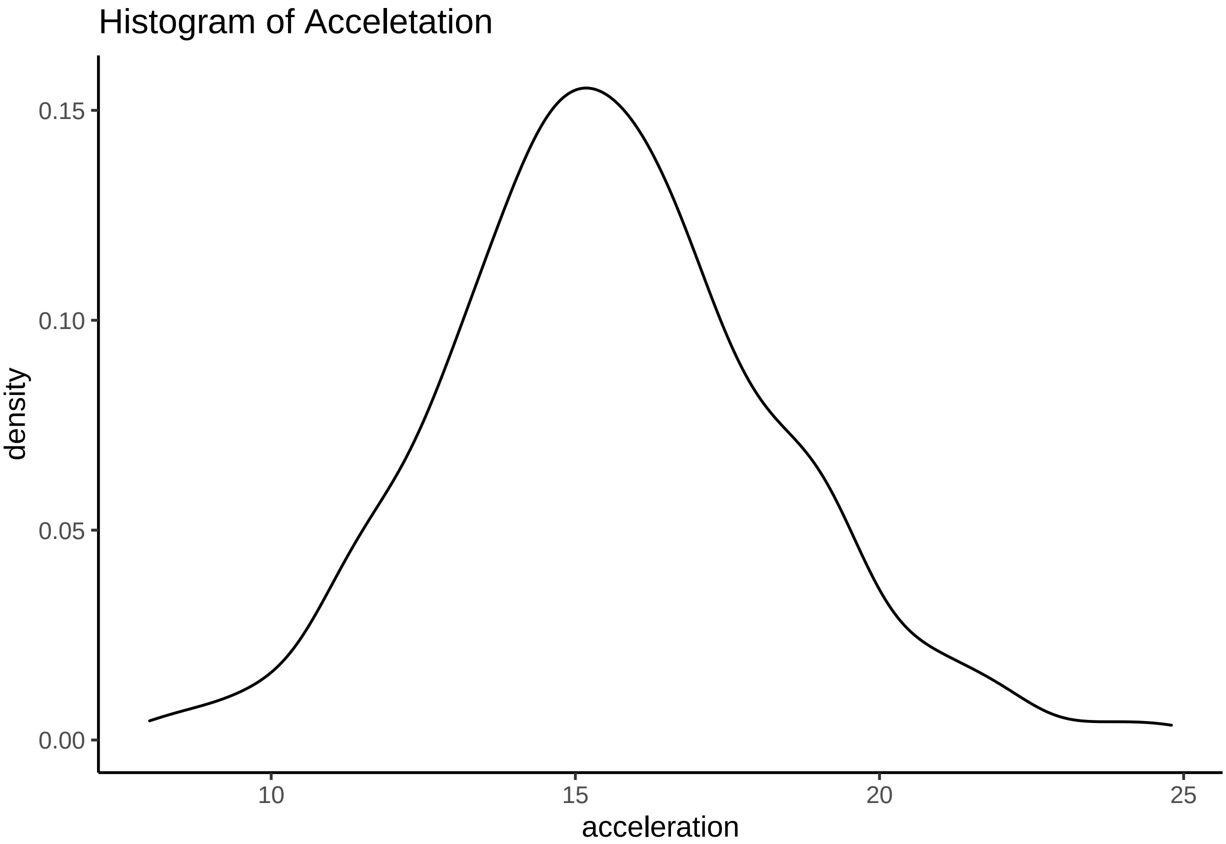
```
##       mpg          cylinders      displacement    horsepower       
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Length:398        
##  1st Qu.:17.50   1st Qu.:4.000   1st Qu.:104.2   Class :character  
##  Median :23.00   Median :4.000   Median :148.5   Mode  :character  
##  Mean   :23.51   Mean   :5.455   Mean   :193.4                     
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:262.0                     
##  Max.   :46.60   Max.   :8.000   Max.   :455.0                     
##      weight       acceleration     model.year        origin     
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000  
##  1st Qu.:2224   1st Qu.:13.82   1st Qu.:73.00   1st Qu.:1.000  
##  Median :2804   Median :15.50   Median :76.00   Median :1.000  
##  Mean   :2970   Mean   :15.57   Mean   :76.01   Mean   :1.573  
##  3rd Qu.:3608   3rd Qu.:17.18   3rd Qu.:79.00   3rd Qu.:2.000  
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000  
##    car.name        
##  Length:398        
##  Class :character  
##  Mode  :character  
##                    
##                    
##                    
```

Acceleration looks to have a similar mean and median, so I am picking this one to explore further

```
library(ggplot2)
```

```
## 
## Attaching package: 'ggplot2'
## The following object is masked _by_ '.GlobalEnv':
##
##     mpg
```

```
ggplot(data=mpg, aes(x=acceleration)) + geom_density() +
  theme_classic() + ggtitle("Histogram of Acceletation")
```



This plot gives the distribution.

(4) Compute a 95% confidence interval for the mean for one of the variable that appears to be normally distributed

```
mean.acceleration<-mean(mpg$acceleration)
sd.acceleration<-sd(mpg$acceleration)
```

The value for Z at 95% is (remember its two sided)

```
qnorm(0.975)
```

```
## [1] 1.959964
```

```
ucl<-mean.acceleration-qnorm(0.975)*sd.acceleration/sqrt(398)
lcl<-mean.acceleration+qnorm(0.975)*sd.acceleration/sqrt(398)
```

So the confidence intervals are:

```
ucl
```

```
## [1] 15.29716
```

```
lcl
```

```
## [1] 15.83902
```

The sample is very large so the confidence interval is small.

(50) The model years range from the 70s and the 80s. What proportion of cars are from the 80s?

```
table(mpg$model.year)
```

```
##
## 70 71 72 73 74 75 76 77 78 79 80 81 82
## 29 28 28 40 27 30 34 28 36 29 29 29 31
```

```
sum(mpg$model.year>79)
```

```
## [1] 89
```

There are 89 cars from the years 80,81 and 82

```
prop.80<-sum(mpg$model.year>79)/nrow(mpg)
prop.80
```

```
## [1] 0.2236181
```

About 22% of the cars are from the 80s.

Compute a 90% confidence interval for the proportion of cars from the 80s?

```
vr<-prop.80*(1-prop.80)/nrow(mpg)

prop.ucl<-prop.80+ qnorm(0.975)*sqrt(vr)
prop.lcl<-prop.80- qnorm(0.975)*sqrt(vr)
```

So the confidence intervals are

```
prop.ucl
```

```
## [1] 0.2645534
```

```
prop.lcl
```

```
## [1] 0.1826828
```

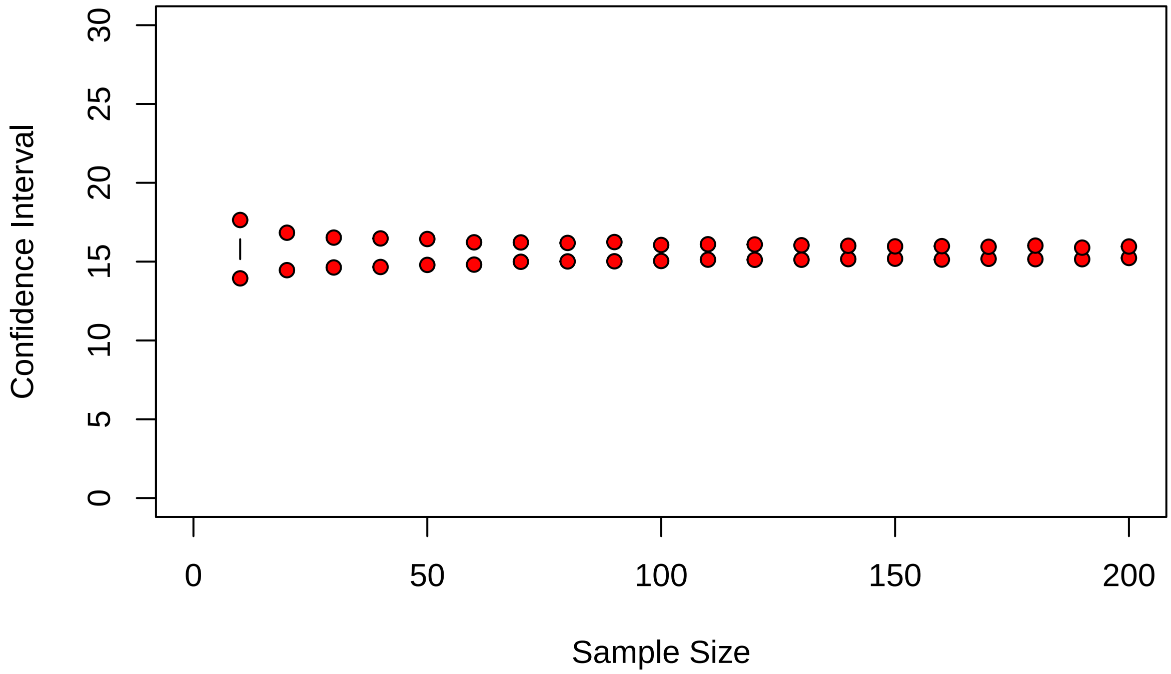
(6) [OPTIONAL] Compute the confidence intervals for the mean using the "bootstrap method".

This code loops over the different sample sizes and samples a size of 200 with replacement, returning the mean every time. The distribution of these values is then used to plot the confidence interval.

```
a<-sample(mpg$acceleration ,100, replace=T)

plot(c(0,200), c(0,30), type="n", xlab="Sample Size", ylab="Confidence Interval")

for (k in seq(10,200,10)){
  a<-numeric(200)
  for (i in 1:200){
  a[i]<-mean(sample(mpg$acceleration, k , replace=T)     )
  }
  points(c(k,k),quantile(a,c(0.025, 0.975)), type="b", pch=21, bg="red")
}
```


Confidence Interval
0
5
10
15
20
25
30
0
50
100
150
200
Sample Size