Assignment: Real Elliptic Curves, Due Monday, 11:59pm

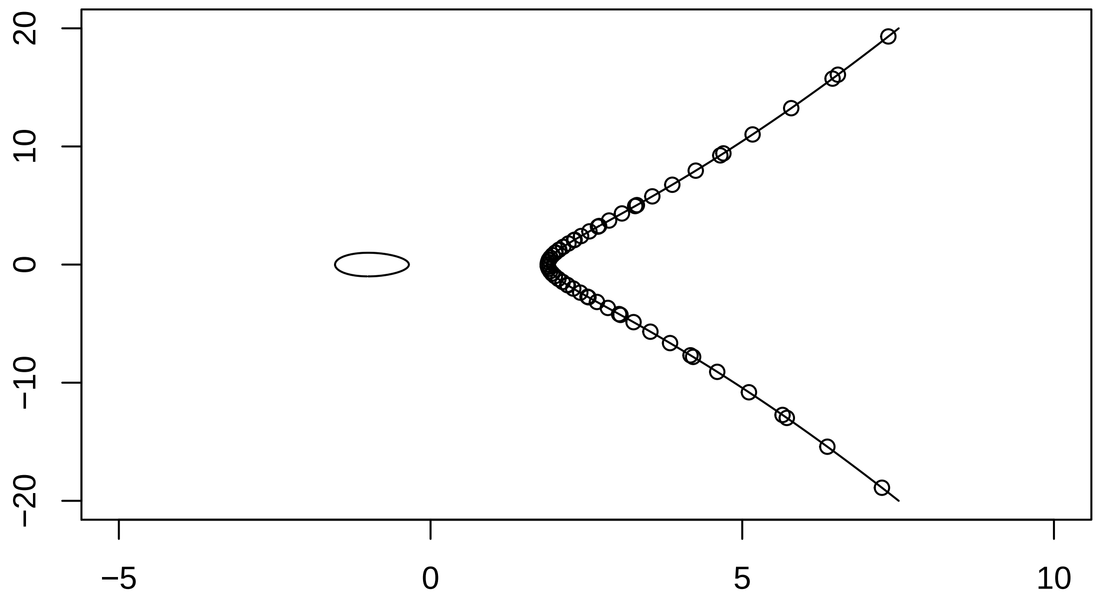
Chris Betsill

October 31, 2016

1. Repeated "exponentiation"

Consider the point $P = (2, 1)$ on the real elliptic curve $y^2 = x^3 - 3x - 1$. Plot the points nP for $n = 1, 2, \ldots, 100$. Is there a pattern? (The first point is plotted for you. Note that the first two arguments to the `points` command are a vector of x-coordinates and a vector of y-coordinates.)

```
x<-seq(-5,10,length=1000)
y<-seq(-20,20,length=1000)
z<-outer(x,y,function(x,y) -y^2  + x^3 - 3*x - 1)
contour(x,y,z,levels=0, labels="", labcex=0.1)
xlist <- vector()
ylist  <- vector()
for(i in 1:100){
  point <- ecPowReal(-3, -1, c(2,1), i)
  xlist <- c(xlist, point[1])
  ylist <- c(ylist, point[2])
}
points(xlist, ylist)
```



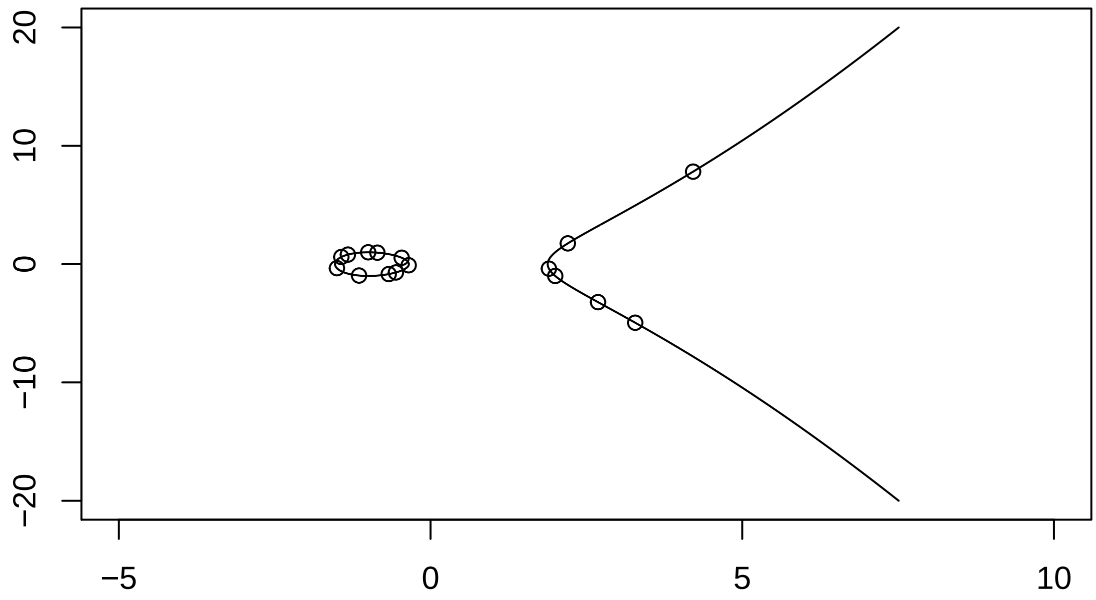
Notice that the points nP lie on only one component of this curve. Find a point Q on this curve so the the points nQ span both components. Illustrate the points nQ below for $n = 1, 2, \ldots, 20$. For which values of n is nQ on the infinite component?

```
x<-seq(-5,10,length=1000)
y<-seq(-20,20,length=1000)
z<-outer(x,y,function(x,y) -y^2  + x^3 - 3*x - 1)
contour(x,y,z,levels=0, labels="", labcex=0.1)
xlist <- vector()
ylist  <- vector()
for(i in 1:20){
  point <- ecPowReal(-3, -1, c(-1,1), i)
  xlist <- c(xlist, point[1])
  ylist <- c(ylist, point[2])
  if(point[1] > 0)
    print(i)
  }
```

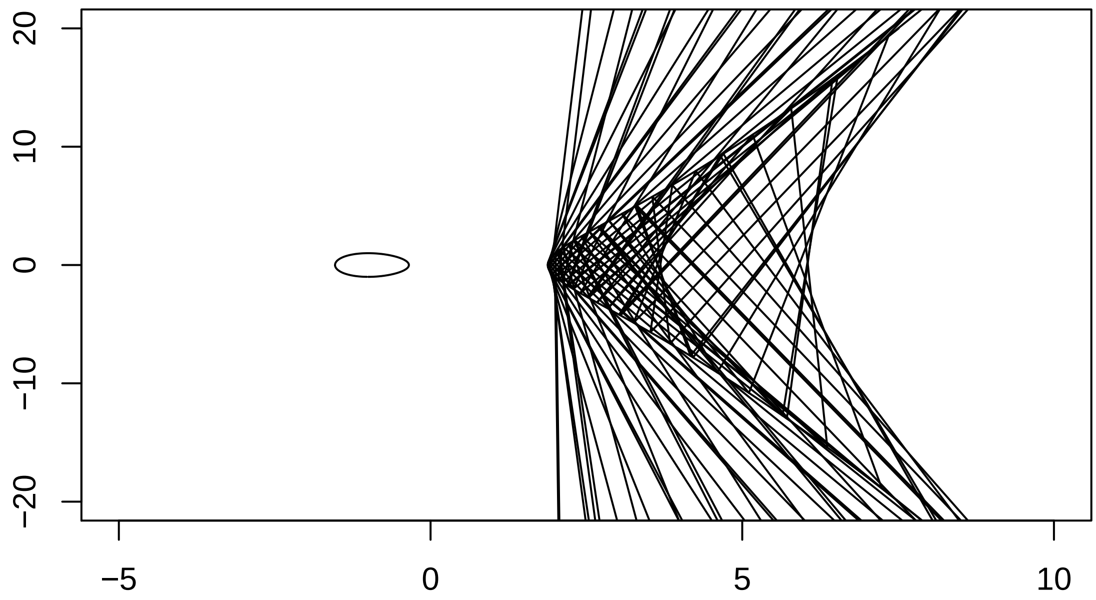
```
[1] 2
[1] 4
[1] 6
[1] 8
[1] 10
[1] 12
[1] 14
[1] 16
```

```
[1] 18
[1] 20
```

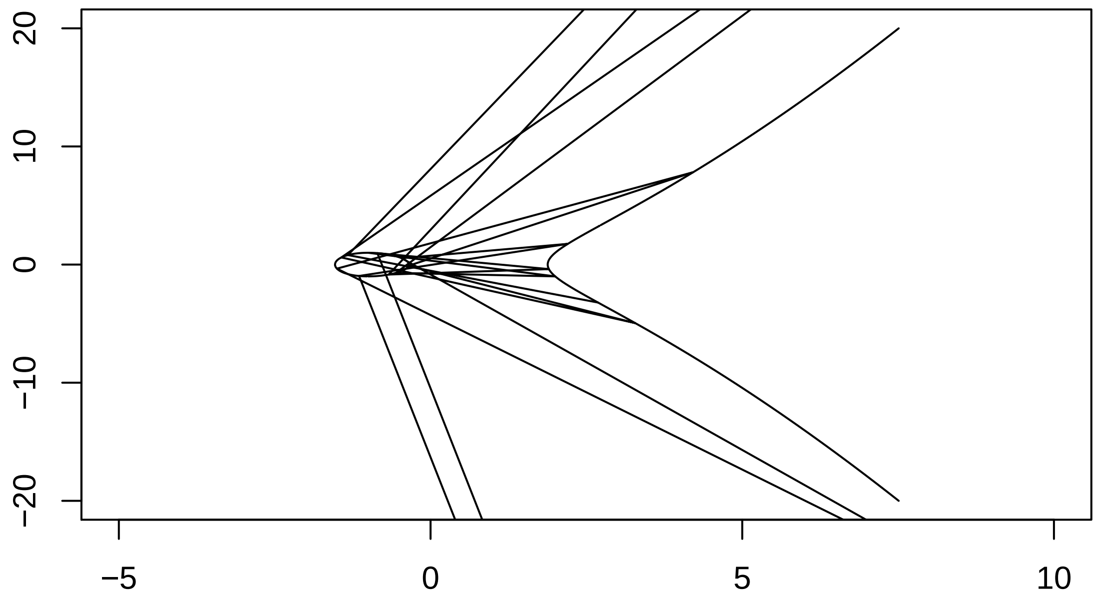
```r
points(xlist, ylist)
```




```r
contour(x,y,z,levels=0, labels="", labcex=0.1)
xlist <- vector()
ylist  <- vector()
for(i in 1:100){
  point <- ecPowReal(-3, -1, c(2,1), i)
  xlist <- c(xlist, point[1])
  ylist <- c(ylist, point[2])
}
lines(xlist, ylist)
```




```
contour(x,y,z,levels=0, labels="", labcex=0.1)
xlist <- vector()
ylist  <- vector()
for(i in 1:20){
  point <- ecPowReal(-3, -1, c(-1,1), i)
  xlist <- c(xlist, point[1])
  ylist <- c(ylist, point[2])
}
lines(xlist, ylist)
```



Now make two new graphs by repeating the code that made the two graphs above, except replace the `points` command with the `lines` command. What do you observe?

The lines appear to be much more uniform in the first graph opposed to the second, where points seem to be more randomly distributed.

2. A point such that $2P = \infty$.

Find a point $P \neq \infty$ on the real elliptic curve $y^2 = x^3 - 10x + 24$ such that $2P = \infty$. Explain how you know that your answer is correct.

Find x when $y = 0$. The line tangent to the curve at point $(-4, 0)$ is verticle, thus $2P$ is infinite.

```
ecPowReal(-10, 24, c(-4,0), 2)
```

```
[1] NA NA
```

3. Order of a point on an elliptic curve

The *order* of a point P on an elliptic curve is the smallest positive integer n such that $nP = \infty$. (The order is infinite if no such integer exists.) For each given point P and real elliptic curve, find the order of P. Show how you found your answers.

```r
findOrder <- function(b, c, point){ #not sure how to do this mathematically, and this function will only
  i <- 1
  while(1){
    if(all(is.na(ecPowReal(b, c, point, i))))
      return(i)
    i <- i + 1
    if(i > 10^8)
      return("There probably isn't an order")
  }
}

findOrder(0, 256, c(0,16))
```

```
[1] 3
```

```r
findOrder(.25, 0, c(.5, .5))
```

```
[1] 4
```

```r
findOrder(-43, 166, c(3,8))
```

```
[1] 7
```

1. $P = (0, 16)$ on the curve $y^2 = x^3 + 256$.
2. $P = \left(\frac{1}{2}, \frac{1}{2}\right)$ on the curve $y^2 = x^3 + \frac{1}{4}x$.
3. $P = (3, 8)$ on the curve $y^2 = x^3 - 43x + 166$.

4. An elliptic curve recurrence relation

Consider the real elliptic curve $y^2 = x^3 - 2x + 10$. Let P_1 and P_2 be points on this curve in the first quadrant, with $P_1 = (1, y_1)$ and $P_2 = (2, y_2)$.

1. Compute y_1 and y_2.

```r
y1 <- sqrt((1^3 - 2*1 + 10))
y2 <- sqrt((2^3 - 2*2 + 10))
y1
```

```
[1] 3
```

```r
y2
```

```
[1] 3.741657
```

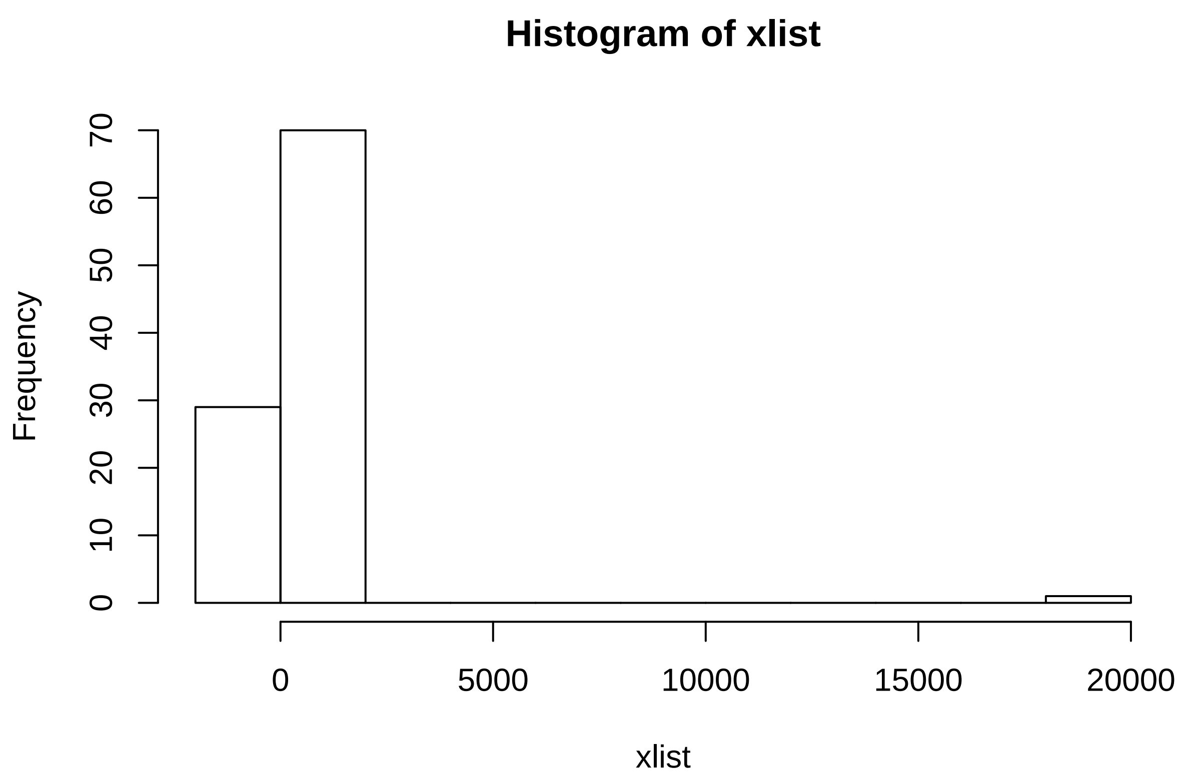
2. We can define a sequence of points on this elliptic curve using the recurrence relation $P_{i+2} = P_{i+1} + P_i$, for $i = 1, 2, 3, \ldots$, where P_1 and P_2 are defined as above. Compute $P_1, P_2, \ldots P_{100}$, and let $x_1, x_2, \ldots, x_{100}$ be the x-coordinates of these points. List x_1, x_2, x_3, x_4, x_5 below.

```
p1 <- c(1, y1)
p2 <- c(2, y2)
xlist <- c(1, 2)
ylist <- c(y1,y2)
for(i in 3:100){
  p3 <- ecAddReal(-2, 10, p1, p2)
  p1 <- p2
  p2 <- p3
  xlist <- c(xlist, p3[1])
  ylist <- c(ylist, p3[2])
}
for(i in 1:5){
  print(xlist[i])
}
```

```
[1] 1
[1] 2
[1] -2.449944
[1] 1.333558
[1] 1.615838
```

3. Let `xpts` be a vector containing the x-coordinates $x_1, x_2, \ldots, x_{100}$. Use the `hist(xpts)` command to create a histogram. Do these values appear randomly distributed?

```
hist(xlist)
```



These values do not appear to be distributed randomly - There is a heavy grouping in the beginning.