Data 609 Week 1

Jason Joseph

September 5, 2016

```
##
## Attaching package: 'dplyr'
##
## The following objects are masked from 'package:stats':
##
##     filter, lag
##
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

PG 8 #10

Your grandparents have an annuity. The value of the annuity increases each month by an automatic deposit of 1% interest on the previous month's balance.Your grandparents withdraw $1000 at the beginning of each month for living expenses. Currently, they have $50,000 in the annuity. Model the annuity with a dynamical system. Will the annuity run out of money? When? Hint: What value will a_n have when the annuity is depleted?

Annuity balance: $50,000

Monthly interest: 1%

Monthly withdrawal: $1,000

Model

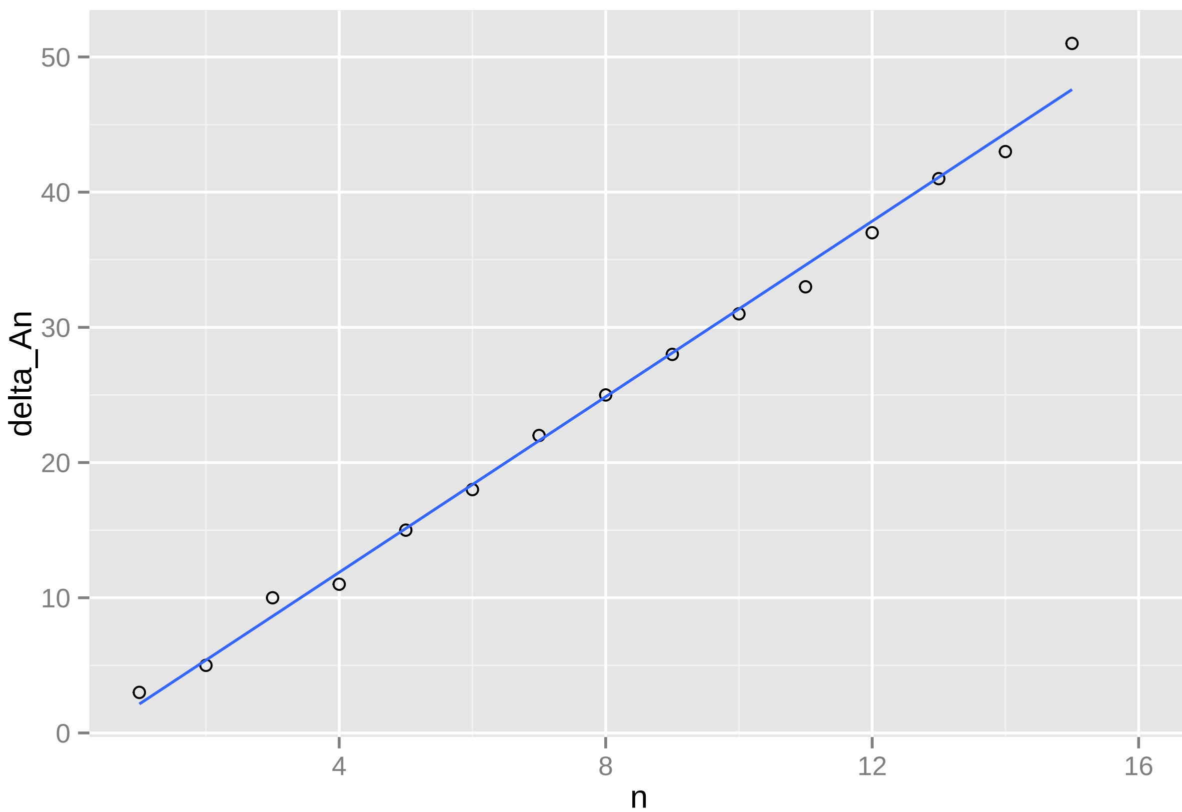
$A_{n+1} = (1.01A_n)$ - 1000

```r
n <- c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)
speed_mph <- n*5
stop_distance_ft <- c(3,6,11,21,32,47,65,87,112,140,171,204,241,282,325,376)

data <- data.frame(n,speed_mph,stop_distance_ft,stringsAsFactors = FALSE)

# using Delta AN = An+1 - An
data$delta_An <- lead(stop_distance_ft) - stop_distance_ft
data
```

```
##    n speed_mph stop_distance_ft delta_An
## 1  1         5                3        3
## 2  2        10                6        5
## 3  3        15               11       10
## 4  4        20               21       11
## 5  5        25               32       15
## 6  6        30               47       18
## 7  7        35               65       22
## 8  8        40               87       25
## 9  9        45              112       28
```

```
## 10 10        50             140        31
## 11 11        55             171        33
## 12 12        60             204        37
## 13 13        65             241        41
## 14 14        70             282        43
## 15 15        75             325        51
## 16 16        80             376        NA
```

```
ggplot(data,aes(x=n,y=delta_An)) + geom_point(shape=1)+ geom_smooth(method =lm,se=FALSE)
```



The graph resionalby approximates a linerar relationship

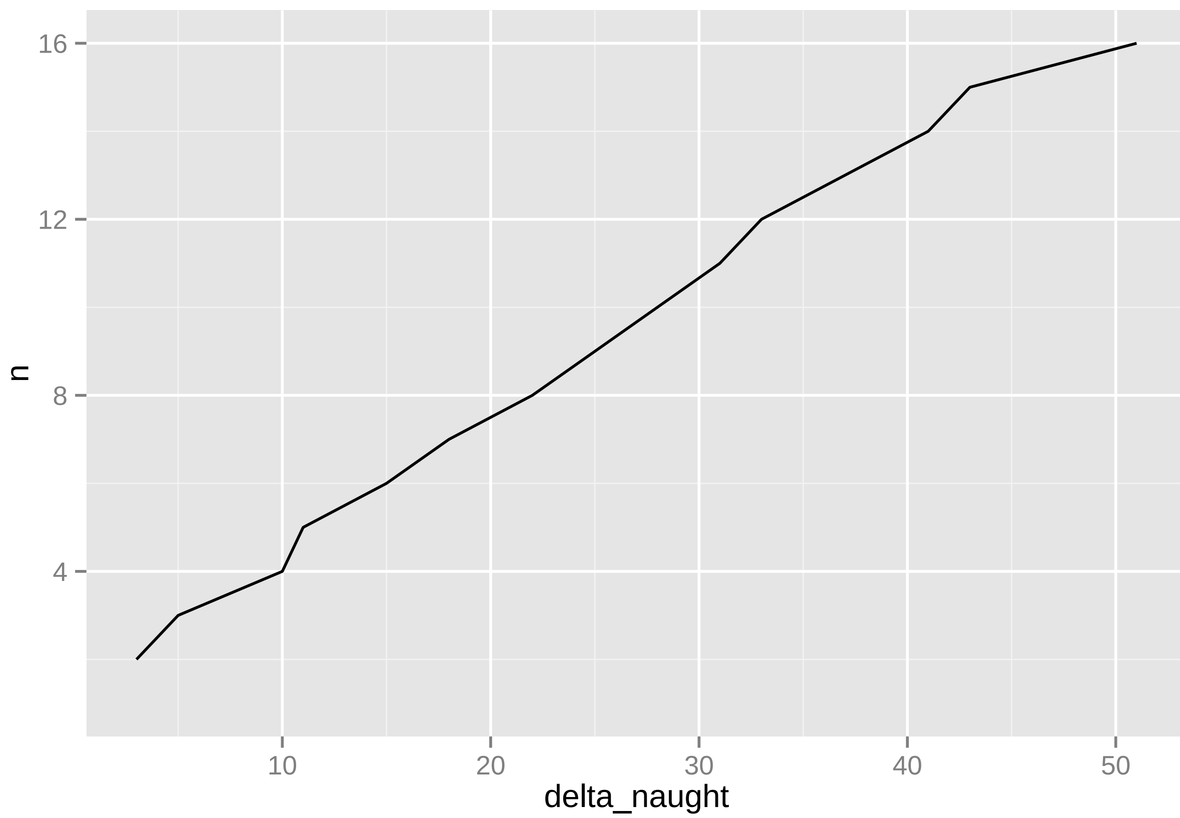
PG 17 #9

The data in the accompanying table show the speed n (in increments of 5 mph) of an automobile and the associated distance an in feet required to stop it once the brakes are applied. For instance, n=6 (representing 6 x 5 = 30 mph) requires a stopping distance of $a_6 = 47$ ft.

a) Calculate and plot the change Δa_n versus n. Does the graph reasonably approximate a linear relationship?

```
n <- 1:16
a_naught <- c(3,6,11,21,32,47,65,87,112,140,171,204,241,282,325,376)
mph <- (n * 5)
```

```r
delta_naught <- c()
d_naught <- NA
for(i in 1:length(a_naught))
{
  delta_naught[i] <- a_naught[i] - a_naught[i - 1]
}
data <- data.frame(n, mph, a_naught, delta_naught)

p1 <- ggplot(data, aes(x=delta_naught, y=n)) + geom_line()
p1
```



Yes there is a reasonable approximate liner relationship.

b) Based on your conclusions in part a, find a difference equation model for the stopping distance data. Test your model by plotting the errors in the predicted values against n. Discuss the appropriateness of the model.

find slope

```r
change_delta <- max(data$delta_naught, na.rm=TRUE)
change_delta_n <- max(data$n)
slope <- change_delta/change_delta_n
slope
```

```
## [1] 3.1875
```

Difference equation model

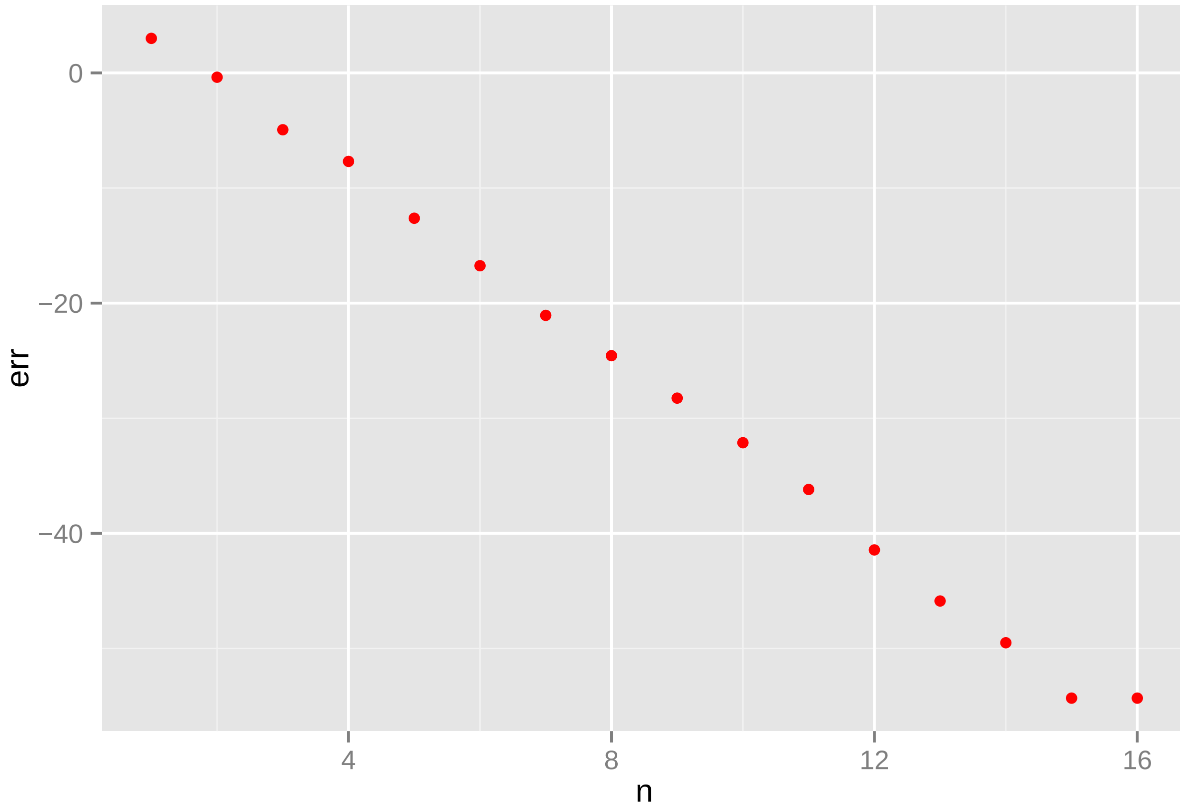
```
a_naught <- c(3,6,11,21,32,47,65,87,112,140,171,204,241,282,325,376)
model <- function(n, a, slope)
{
    an <- slope * n + a
    return(an)
}

m <- c()
m[1] <- 0
for(i in 2:length(a_naught))
{
  m[i] <- model(i, m[i-1], slope)
}
m
```

```
##  [1]   0.0000   6.3750  15.9375  28.6875  44.6250  63.7500  86.0625
##  [8] 111.5625 140.2500 172.1250 207.1875 245.4375 286.8750 331.5000
## [15] 379.3125 430.3125
```

```
data1 <- cbind(data, m)
data1
```

```
##     n mph a_naught delta_naught        m
## 1   1   5        3           NA   0.0000
## 2   2  10        6            3   6.3750
## 3   3  15       11            5  15.9375
## 4   4  20       21           10  28.6875
## 5   5  25       32           11  44.6250
## 6   6  30       47           15  63.7500
## 7   7  35       65           18  86.0625
## 8   8  40       87           22 111.5625
## 9   9  45      112           25 140.2500
## 10 10  50      140           28 172.1250
## 11 11  55      171           31 207.1875
## 12 12  60      204           33 245.4375
## 13 13  65      241           37 286.8750
## 14 14  70      282           41 331.5000
## 15 15  75      325           43 379.3125
## 16 16  80      376           51 430.3125
```

```
data1$err <- data1$a_n - data1$m
data1err <- ggplot(data=data1, aes(x=n)) + geom_point(color="red", aes(y=err))
data1err
```



As values are increase, the error increases.

PG 34 #13

Consider the spreading of a rumor through a company of 1000 employees, allworking in the same building.We assume that the spreading of a rumor is similar to the spreading of a contagious disease (see Example 3, Section 1.2) in that the number of people hearing the rumor each day is proportional to the product of the number who have heard the rumor previously and the number who have not heard the rumor. This is given by

$$r_n + 1 = r_n + kr_n(1000 -_n)$$

where k is a parameter that depends on how fast the rumor spreads and n is the number of days. Assume k = 0.001 and further assume that four people initially have heard the rumor. How soon will all 1000 employees have heard the rumor?

```
n <- 0
r <- 4
data <- data.frame(n,r)
k <- 0.001


for(i in 1:15 )
{
  r <- data$r[i]  +(k *data$r[i]*(1000-data$r[i]))
  n <- i
```

```r
  data <- rbind(data,c(n,r))
}

data
```

```
##     n          r
## 1   0    4.00000
## 2   1    7.98400
## 3   2   15.90426
## 4   3   31.55557
## 5   4   62.11538
## 6   5  120.37244
## 7   6  226.25535
## 8   7  401.31922
## 9   8  641.58132
## 10  9  871.53605
## 11 10  983.49701
## 12 11  999.72765
## 13 12  999.99993
## 14 13 1000.00000
## 15 14 1000.00000
## 16 15 1000.00000
```

All of the 1000 employees would have heard the rumor at the end of the 12^{th} day

PG 55 #6

An economist is interested in the variation of the price of a single product. It is observed that a high price for the product in the market attracts more suppliers. However, increasing the quantity of the product supplied tends to drive the price down. Over time, there is an interaction between price and supply. The economist has proposed the following model, where $\mathrm{P_n}$ represents the price of the product at year n, and $\mathrm{Q_n}$ represents the quantity. Find the equilibrium values for this system.

$$P_{n+1} = P_n - 0.1(Q_n - 500)$$
$$Q_{n+1} = Q_n + 0.2(P_n - 100)$$

a. Does the model make sense intuitively? What is the significance of the constants 100 and 500? Explain the significants of the constance -0.1 and 0.2.

The model does make sense. As $\mathrm{Q_n}$ increases above the equilibrium or ceiling value for quantity the price will be adversely affected. If the price drops below 100 (equilibrium) quantity will be adversely affected .

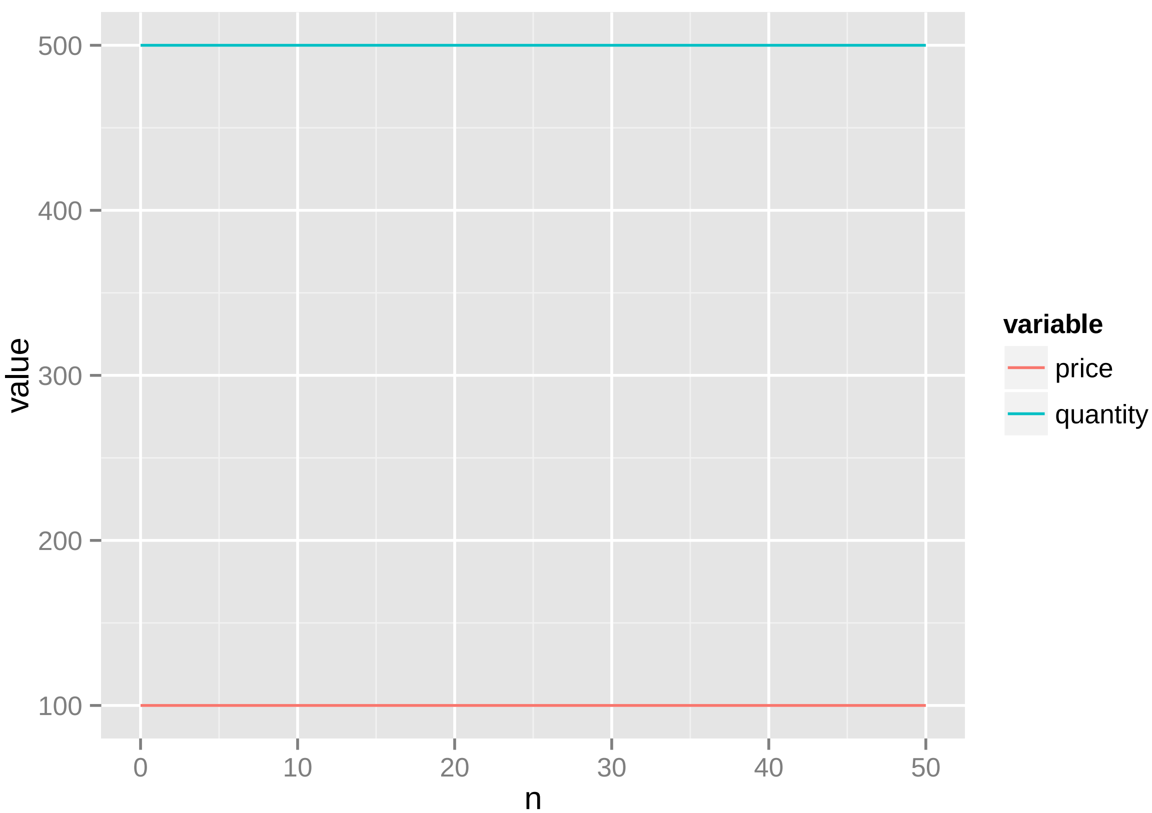
-0.1 modulates the price according to the quantity. If the quantity gets too high it will drive the price down.

+0.2 Since the price determines the quantity. As the price increases so will the quantity.

b. Test the initial conditions in the following table and predict the long-term behavior.

CASE A Price 100 Quantity 500

```
count <- 50
price <-100
quantity <- 500
n <- 0
caseA <- data.frame(n,price,quantity)
for(j in 2:count)
{
  newPrice <- caseA$price[j-1] - (0.1 * (caseA$quantity[j-1]-500))
  newPrice
  newQuantity <- caseA$quantity[j-1] + (0.2 * (caseA$price[j-1]-100))
  newQuantity
  caseA <- rbind(caseA,c(j,newPrice,newQuantity))
}
#caseA

ggplot(melt(caseA, id="n"), aes(x=n, y=value, color=variable)) + geom_line()
```

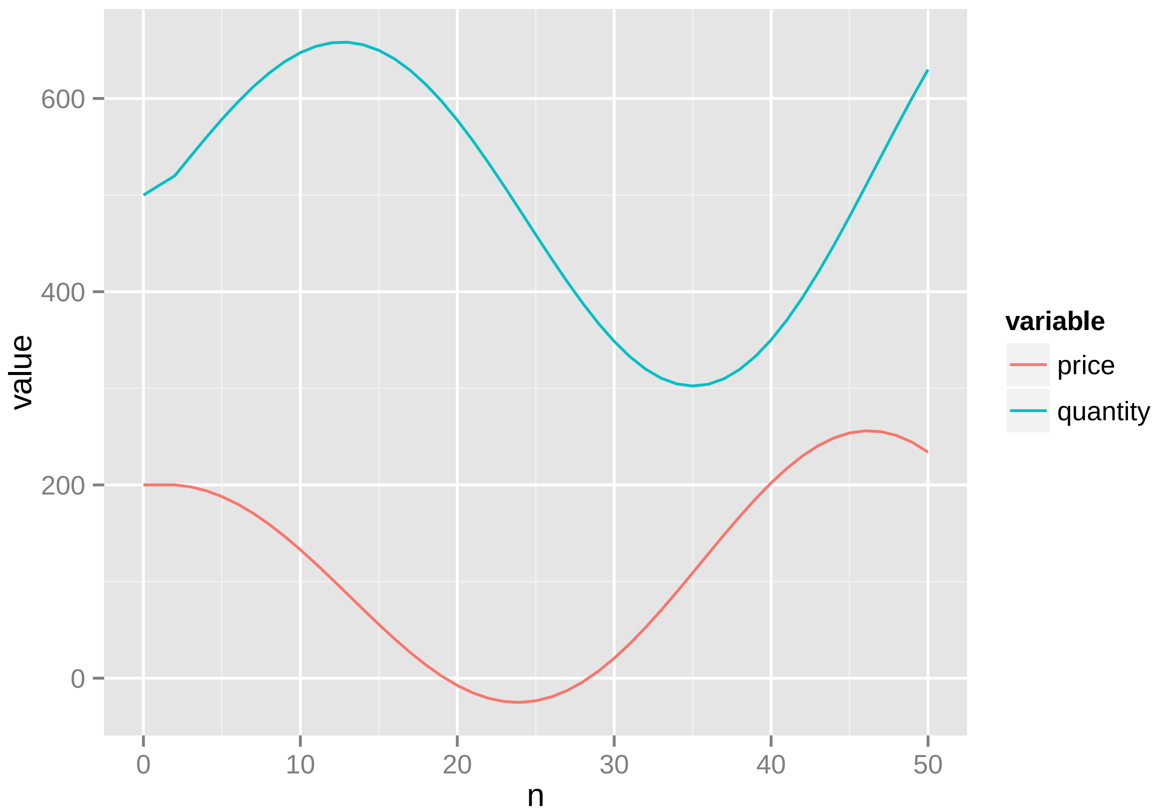


This model will result in an equilibrium price and quantity

Price 200 Quantity 500 Case B

```
price <-200
quantity <- 500
```

```
n <- 0
caseB <- data.frame(n,price,quantity)
for(j in 2:count)
{
  newPrice <- caseB$price[j-1] - (0.1 * (caseB$quantity[j-1]-500))
  newPrice
  newQuantity <- caseB$quantity[j-1] + (0.2 * (caseB$price[j-1]-100))
  newQuantity
  caseB <- rbind(caseB,c(j,newPrice,newQuantity))
}
ggplot(melt(caseB, id="n"), aes(x=n, y=value, color=variable)) + geom_line()
```

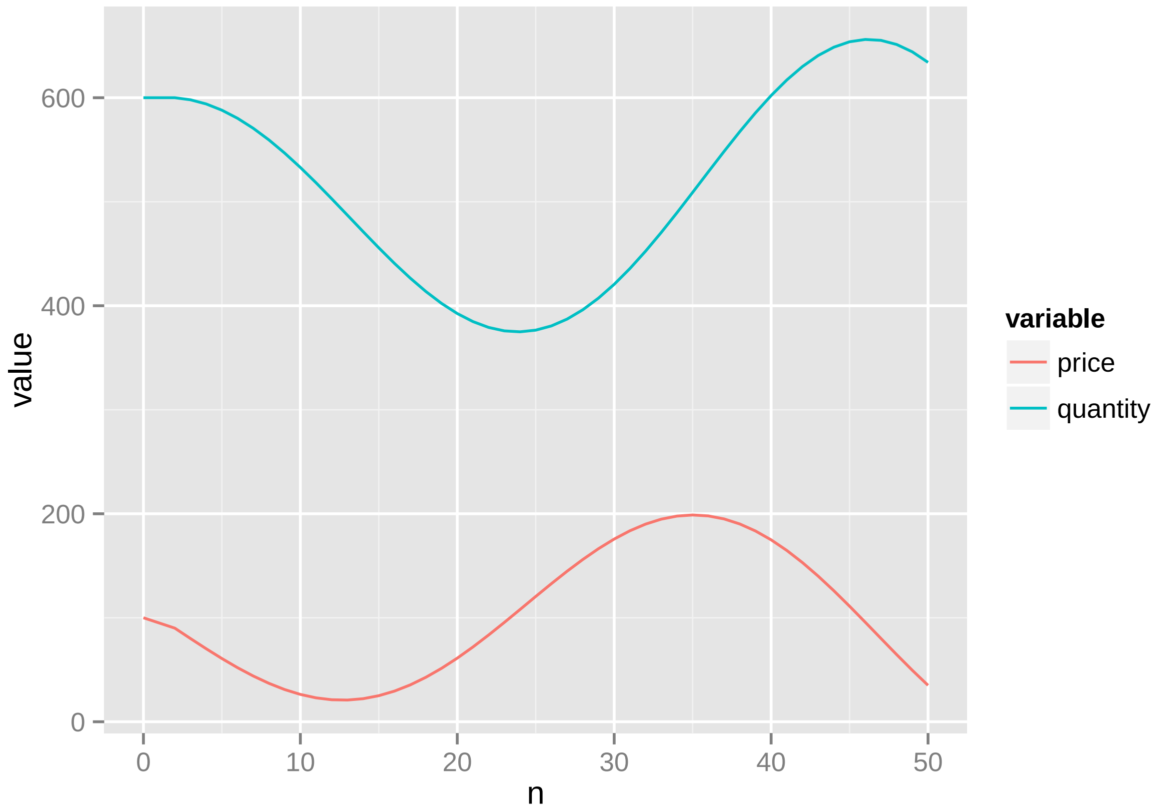


This model will result in a fluctuating price and quantity

Price 100 Quantity 600 Case C

```
price <-100
quantity <- 600
n <- 0
caseC <- data.frame(n,price,quantity)
for(j in 2:count)
{
  newPrice <- caseC$price[j-1] - (0.1 * (caseC$quantity[j-1]-500))
  newQuantity <- caseC$quantity[j-1] + (0.2 * (caseC$price[j-1]-100))
```

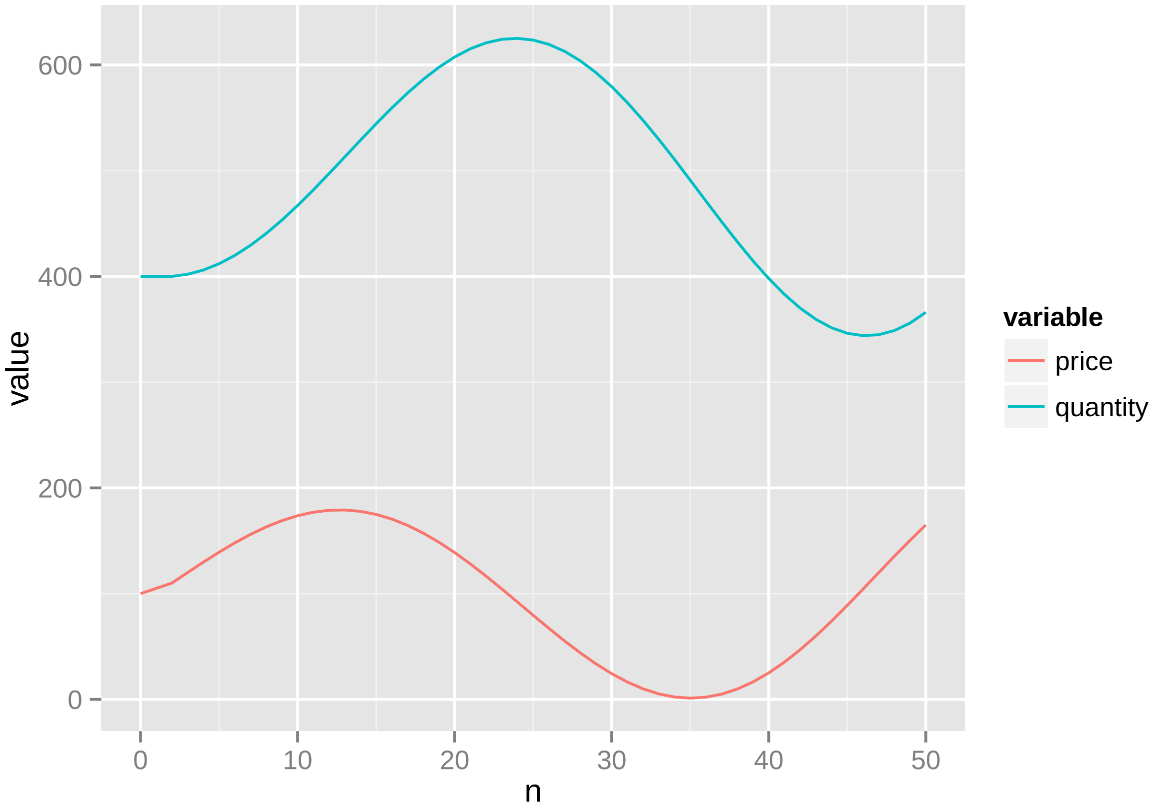
```
  caseC <- rbind(caseC,c(j,newPrice,newQuantity))
}
ggplot(melt(caseC, id="n"), aes(x=n, y=value, color=variable)) + geom_line()
```



This model will result in a fluctuating price and quantity

Price 100 Quantity 400 Case D

```
price <-100
quantity <- 400
n <- 0
caseD <- data.frame(n,price,quantity)
for(j in 2:count)
{
  newPrice <- caseD$price[j-1] - (0.1 * (caseD$quantity[j-1]-500))
  newQuantity <- caseD$quantity[j-1] + (0.2 * (caseD$price[j-1]-100))
  caseD <- rbind(caseD,c(j,newPrice,newQuantity))
}
ggplot(melt(caseD, id="n"), aes(x=n, y=value, color=variable)) + geom_line()
```



This model will result in a fluctuating price and quantity