Mortgages and Rentals

Nathan Cantafio

May 14, 2024

Contents

List of Figures

List of Tables

Motivations

I was playing around with calculating the total cost of a mortgage, and I realized that for $700000 home, you could easy pay $1.1 million on a mortgage. Growing up my dad has always thought that renting is a waste of money, but this got me wondering if that's actually true. Could renting sometimes be a better financial choice than buying?

```r
# returns the monthly payment required to pay off the mortgage in the specified time (given in years)
# interest rate is annualized and downpayment is given as a percentage
# assumes monthly interest accrual and payments
payment <- function(principal=500000, rate=0.05, downpayment=0.05, time = 25) {
  return((rate/12)*((1+rate/12)^(12*time))*(principal-downpayment*principal)/((1+rate/12)^(12*time)-1))
}

300*payment(principal = 700000) # total cost of mortgage is over 1.1 million!!!
```

```
## [1] 1166257
```

Math Background

Any math that I did or found for this project and thought was worth mentioning is in this section.

Monthly Payment Formula

To find a formula for what the monthly payment must be in order to pay off a fixed rate mortgage in a specified number of years I used linear algebra to solve the recurrence. This is unique compared to the other approaches I've seen online which used sums.

Assume that interest is accrued over the same period as payments are made. Let a_n be the amount owed after n payment periods. Then

$$a_n = (1+r)a_{n-1} - c,$$

where r is the interest rate and c is the monthly payment.

This relation can be rewritten as:

$$\begin{bmatrix} a_n \\ c \end{bmatrix} = \begin{pmatrix} 1+r & -1 \\ 0 & 1 \end{pmatrix} \begin{bmatrix} a_{n-1} \\ c \end{bmatrix},$$

and then further rewritten as:

$$\begin{bmatrix} a_n \\ c \end{bmatrix} = \begin{pmatrix} 1+r & -1 \\ 0 & 1 \end{pmatrix}^n \begin{bmatrix} a_0 \\ c \end{bmatrix} = A^n \begin{bmatrix} a_0 \\ c \end{bmatrix}.$$

Call the matrix A. Then $A = MDM^{-1}$ where D is a diagonal matrix. In fact,

$$A^n = \begin{pmatrix} 1/r & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1+r \end{pmatrix}^n \begin{pmatrix} 0 & 1 \\ 1 & -1/r \end{pmatrix} = \begin{pmatrix} (1+r)^n & \frac{1}{r}(1-(1+r)^n) \\ 0 & 1 \end{pmatrix}.$$

And we have:

$$a_n = a_0(1+r)^n + c\frac{1-(1+r)^n}{r}.$$

If the goal is to pay off the mortgage in N periods, then setting $a_N = 0$ yields:

$$c = a_0 \frac{r(1+r)^N}{(1+r)^N - 1}, \qquad \text{where } a_0 \text{ is the principal.}$$

Vasicek and CIR Models

The Vasicek and CIR models use Stochastic Differential Equations to model interest rates.

The basic idea of both is that there is a value (which we call θ) to which the interest rate should gravitate towards over long enough time scales.

Let r be the interest rate we are modeling. Then the change in the rate, dr should be in the direction $\theta - r$ so that

$$r + dr = r + \theta - r = \theta,$$

in other words r is being pulled towards θ.

We don't want it to be pulled all the way though, otherwise the model wouldn't be very interesting. So we add another parameter α which acts as the speed. Thus

$$dr = \alpha(\theta - r).$$

The above on its own is called the drift term. There is also some volatility or random noise introduced with a parameter sigma and voila:

$$dr = \alpha(\theta - r)dt + \sigma dW.$$

Above is the Vasicek model, we add a dt into the equation for reasons that I do not understand.

The CIR model is similar but volatility is also proportional to the square root of the rate:

$$dr = \alpha(\theta - r)dt + \sigma\sqrt{r}dW.$$

The CIR model is nice because at least analytically, as long as $2\alpha\theta > \sigma^2$ (Feller condition) then the rate will never be negative. This is because when r becomes close to zero, the $\sqrt{r}$ term becomes very small and allows the drift term to dominate over the volatility term; pulling r up. However, this does not work out so nicely in simulation due to errors introduced by discretization which can yield negative rates.

We can discretize the CIR model as follows (I believe this is called the Euler discretization scheme):

$$r[t] = r[t-1] + \alpha(\theta - r[t-1])dt + \sigma\sqrt{r[t-1]}dW.$$

We can then rearrange this as:

$$\frac{r[t]}{\sqrt{r[t-1]}} = a\frac{r[t-1]}{\sqrt{r[t-1]}} + b\frac{1}{\sqrt{r[t-1]}} + \varepsilon,$$

where $a = 1 - \alpha dt$, $b = \alpha\theta dt$ and $\varepsilon = \sigma dW$ so that we can run a linear regression.

Simulating Interest Rates

I chose to use a CIR model to avoid negative interest rates. However, due to errors introduced by the discretization, I added a step of taking the absolute value. The function for simulating a CIR model is below.

```r
cir <- function(alpha, theta, sigma, steps) {
  dt <- 1 # time step

  r <- vector(length=steps)
  r[1] <- theta # initial interest rate


  for (i in 2:steps) {
    dW <- rnorm(1, mean=0, sd=1) # generate random noise

    r[i] <- abs(r[i-1] + alpha*(theta - r[i-1])*dt + sigma*sqrt(r[i-1])*dt*dW) # modified euler
  }

  return(r)
}
```

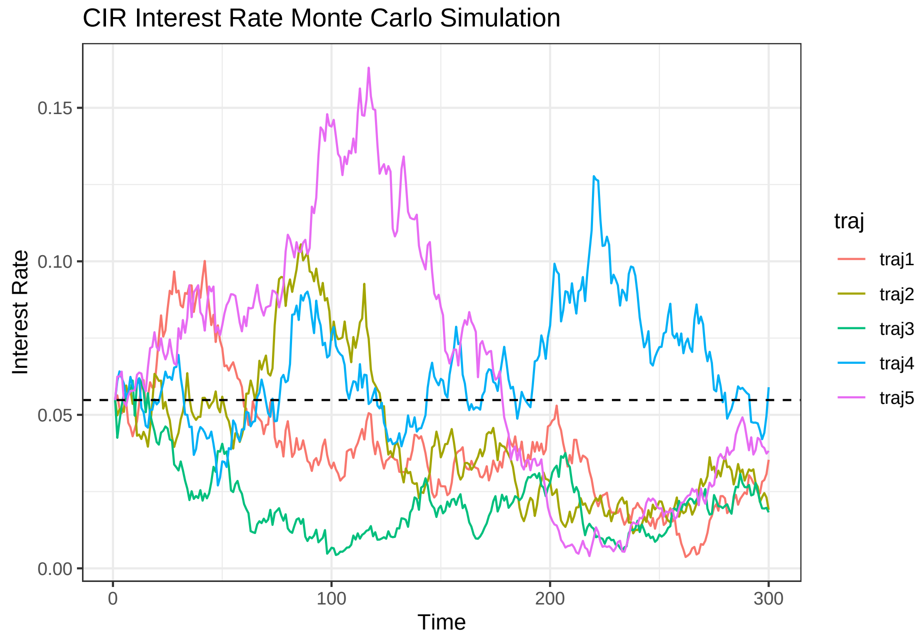
To choose parameters that yield realistic interest rates, I used data of Canada's overnight rate from 1960 to March of 2024. I then ran a linear regression, predicting the last $n-1$ rates from the first $n-1$ rates. Using the formula from before to get the parameters from the coefficients. These parameters act as an initial guess for the MLE estimation which follows. I could have just stopped at the regression, but I wanted to use MLE to refine and hopefully improve the choice of parameters.

```r
# load data
rates <- read.csv(file = "canadarates.csv")
colnames(rates) <- c("date", "prime", "overnight")
rates <- tail(rates, -26)

r = rates$overnight/100
n = length(r)

# predicting last n-1 rates based on the first n-1 to fit CIR Model

x <- r[-n]
y = r[-1]/sqrt(x)

X <- matrix(c(x/sqrt(x), rep(1, n-1)/sqrt(x)), ncol=2)
B <- solve(t(X)%*%X)%*%t(X)%*%y

e <- y - X%*%B # for calculating standard error

# r[t]/sqrt(r[t-1]) ~ a*r[t-1]/sqrt(r[t-1]) + b*(1/sqrt(r[t-1])) + ep
# a = (1 - alpha*dt), b = alpha*theta*dt, ep = sigma*dW

a <- B[1]
b <- B[2]
MSE <- (t(e)%*%e)[1,1]/(n-3)

dt <- 1 # time step

# initial parameters from least squares regression
alpha = (1 - a)/dt # Long-term mean or equilibrium interest rate
theta = b/(1 - a) # Speed of mean reversion
sigma <- sqrt(MSE)/sqrt(dt) # Volatility
```

There are two ways to calculate the Log-Likelihood for the CIR model both of which are implemented below. The first involves the Bessel function, and the second involves Chi-Squared distribution.

```r
# Now refine the value of parameters with ML estimation
# function to calculate the ln Likelihood for CIR Model
lnLhelper <- function(param, data, dt) {
  n = length(data)
  dataF <- data[-1] # data from 2:n
  dataL <- data[-n] # data from 1:n-1

  # parameter values
  alpha <- param[1]
  theta <- param[2]
  sigma <- param[3]

  # ## calculate likelihood (Bessel)
  c = 2*alpha/(sigma^2*(1 - exp(-alpha*dt)))
  q = 2*alpha*theta/sigma^2 - 1
  u = c*exp(-alpha*dt)*dataL
  v = c*dataF
  z = 2*sqrt(u*v)
  bf = besselI(z, q, T) # scaled modified bessel function of the first kind

  lnL = -(n-1)*log(c) + sum(u + v - 0.5*q*log(v/u) - log(bf) - z)

  # calculate likelihood (Chi-Sq)
  # c = 2*alpha/(sigma^2*(1 - exp(-alpha*dt)))
  # q = 2*alpha*theta/sigma^2 - 1
  # u = c*exp(-alpha*dt)*dataL
  # v = c*dataF
  # s = 2*c*dataF
  # nc = 2*u # non-centrality
  # df = 2*q + 2 # degrees of freedom
  #
  # gpdf = dchisq(s, df = df, ncp = nc)
  # ppdf = 2*c*gpdf
  # lnL = sum(-log(ppdf))

  return(lnL)
}

# returns the ln Likelihood function for the CIR Model
# enter the data and time step inside the function definition
logLikelihood <- function(param) {
  result <- lnLhelper(param=param, data=r, dt=1)
  return(result)
}

# optimize using optim()
# for example call optim(c(alpha, theta, sigma), logLikelihood)
# results for Canada overnight rate from January 1960 - March 2024 have been saved in cirParameters.csv

parameter_data <- read.csv(file="cirParameters.csv", row.names = 1)
```

```r
# Simulate the rates
# Set number of simulation steps
steps <- 300

# choose which parameters to use
param <- parameter_data$mleBessel

# Create time vector
time <- 1:steps

# Create data frame to store trajectories
trajectories <- as.data.frame(c(time))

# Choose how many simulated trajectories to generate
N <- 5

# Run loop to generate N different simulated trajectories
for (i in 1:N) {
  trajectories <- cbind(trajectories, cir(param[1], param[2], param[3], steps))
}
colnames(trajectories) <- c('time', paste("traj", 1:N, sep = "")) # rename columns

# for plotting purposes in ggplot
trajectories <- melt(trajectories, id.vars = 'time', variable.name = 'traj')

ggplot(data = trajectories, aes(time, value)) +
  geom_line(aes(colour = traj)) +
  geom_hline(yintercept = param[2], linetype = "dashed") +
  xlab("Time") +
  ylab("Interest Rate") +
  ggtitle("CIR Interest Rate Monte Carlo Simulation") +
  theme_bw()
```



Mortgaging vs Renting

Rather than simulating the rent prices over time which turned out to be difficult and requires lots more data than I had access to, I will simply check if renting is financially wiser than buying for many different rental rates. For each interest rate trajectory that I simulate, I will see how the cost of renting compares to mortgaging for a range of rental rates. I still needed to determine a base rental rate from which to calculate my range, this was a trickier problem than I thought it would be.

The method I finally landed on was calculating what a fixed rate 25 year mortgage's monthly payment would be if you wanted to buy the house at the end of every lease term (usually the lease term is 1 year). What this monthly payment would be changes every year because interest rates vary, but don't forget that property values also increase. Which means the principal would be higher in each year. This needs to be accounted for - because rental rates don't vary semi-randomly like interest rates, instead they just increase over time. Two percent seemed like a reasonable annual appreciation value for property, so that's what I went with.

Lets go through a quick example. Let's say a house is bought in 1990 for $500,000. To calculate what the base rent should be for this property, we use the interest rate in 1990 and the principal of $500,000 to calculate what the monthly payment is on a fixed rate 25 year mortgage. If rent is any higher than this, then it's obvious that we might as well just buy. Then to find the range of potential rent values, we use ratios between say half and the full base rent For example we could increment by 0.1, then we would be checking rents of 0.5 the base rent, 0.6 the base rent, 0.7 the base rent, . . . , the whole base rent. But don't forget we need to update the base rent every lease term (usually 1 year), and then re-calculate the range of rents. So in 1991, the property is expected to increase to $510000 which we use as our new principal to calculate the new base rent.

In the end we will have a range of rents for each of the 25 years. And this is only for 1 interest rate trajectory! But with this information we can compare the cost of mortgaging to the cost of rent for this particular interest rate trajectory. If we do this for several simulated trajectories, we can see how often and by how much is mortgaging or renting better than the other financially.

```r
# a few useful functions

# returns the monthly payment required to pay off the mortgage in the specified time (given in years)
# interest is annualized and downpayment is as a percentage, assumes monthly interest and payments
payment <- function(principal=500000, rate=0.05, downpayment=0.05, time = 25) {
  return((rate/12)*((1+rate/12)^(12*time))*(principal-downpayment*principal)/((1+rate/12)^(12*time)-1))
}

# returns a vector with the monthly payments required to pay off the mortgage in the specified time
# rates is a vector of annualized interest rates for each month in a sequence
# downpayment is given as a percentage
variable_payment <- function(principal=500000, downpayment=0.05, time=25, rates) {
  n = 12*time # time in months
  payments = vector(length=n)
  amount_owed = principal - downpayment*principal # initial amount owed
  for (i in 1:n) {
    # calculate payment for month i so that mortgage is payed off in n-i+1 many months
    payment = (rates[i]/12)*((1+rates[i]/12)^(n-i+1))*(amount_owed)/((1+rates[i]/12)^(n-i+1)-1)
    # records payment on month i
    payments[i] = payment
    # updates new amount owed on month i
    amount_owed = amount_owed - payment + amount_owed*rates[i]/12
  }
  return(payments)
}
```

```r
# principal = initial value of property
# time = number of years for mortgage
# rates = interest rate over time
# investment_appreciation = annual appreciation of investment (i.e. stocks/bonds)
# property_appreciation = annual appreciation of property value
# lease_term = length of the lease term (i.e. how often should price of rent be updated)
# step = step between ratios we test
#
difference <- function(principal = 500000, time = 25, rates, investment_appreciation = 0.06,
                       property_appreciation = 0.02, lease_term = 12, step=0.1, max_ratio=1.2,
                       min_ratio=0.5) {

  n = time*12 # number of months
  r = 1 + investment_appreciation/12 # monthly appreciation of investment
  s = 1 + property_appreciation/12 # monthly appreciation of real estate

  # calculate the mortgage payments over time
  var_payment = variable_payment(principal = principal, rates = rates, time=time)
  # total cost of mortgage = sum of all the payments
  mortgage_cost = sum(var_payment)

  # get adjusted value of home over 25 years based on 2% appreciation per year
  principal = rep(principal, n)
  for (i in 1:n) {
    principal[i] = principal[i]*s^(i-1)
  }

  # calculate the baseline rent for every lease term
  # by finding what the fixed rate mortgage would be
  # for adjusted property value
  baseline_rent = payment(principal = principal, rate = rates, time = time)
  baseline_rent = baseline_rent[c(TRUE,rep(FALSE,lease_term - 1))]
  rep_baseline <- c()
  for(w in 1:length(baseline_rent)) {
    rep_baseline <- rbind(rep_baseline, rep(baseline_rent[w], lease_term))
  }


  m = (max_ratio - min_ratio)/step + 1 # m = number of different rents we test
  result = vector(length = m)

  for (j in 1:m) {
    rental_payment = rep_baseline*(min_ratio+step*(j-1)) # calculate monthly rent for each month

    # calculate how much money is made by investing what we would have payed towards the mortgage
    investment_income = 0
    for (k in 1:n) {
      investment_income = investment_income*r + var_payment[k] - rental_payment[k]
    }

    # calculate the cost of the rental by taking sum of rent payments
    # and subtracting profits from investment
    rental_cost = sum(rental_payment) - investment_income
```

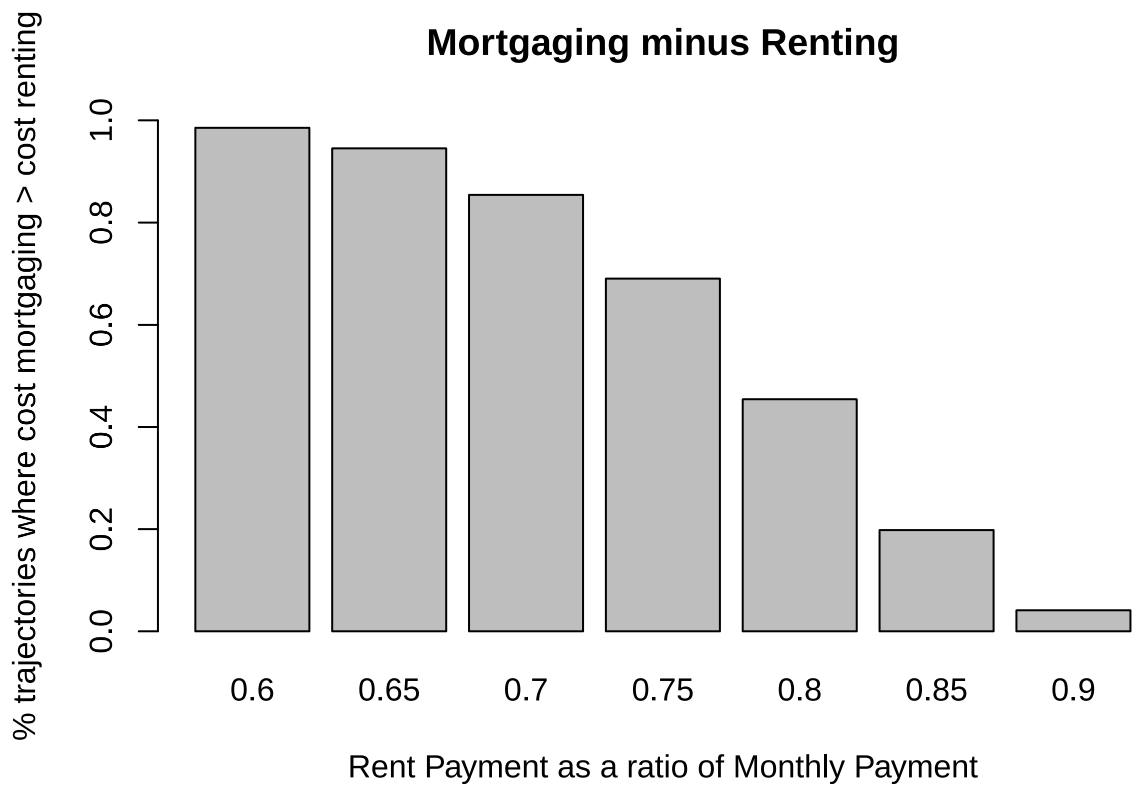
```
    result[j] = mortgage_cost - rental_cost
  }
  return(result)
}
```

Compare rentals to mortgages.

```
# Simulate the rates
# Set number of simulation steps
steps <- 300

# choose which parameters to use
param <- parameter_data$mleBessel

# Create time vector
time <- 1:steps

# Create data frame to store trajectories
trajectories <- as.data.frame(c(time))

# Choose how many simulated trajectories to generate
N <- 100000

# Run loop to generate N different simulated trajectories

for (i in 1:N) {
  trajectories <- cbind(trajectories, cir(param[1], param[2], param[3], steps))
}
colnames(trajectories) <- c('time', paste("traj", 1:N, sep = "")) # rename columns
trajectories$time <- NULL

step = 0.05
max_ratio = 0.9
min_ratio = 0.6

results <- data.frame(as.list(difference(principal=500000, rates = trajectories[,1], step = step,
                                         max_ratio = max_ratio, min_ratio = min_ratio)))
for (i in 2:N) {
  results[nrow(results) + 1, ] = difference(principal=500000, rates = trajectories[,i], step = step,
                                            max_ratio = max_ratio, min_ratio = min_ratio)
}
```

Below is a plot showing the percentage of trajectories that favor rentals, and a plot showing the average difference in cost between renting and mortgaging. From this data, it looks like it makes the most sense to rent when the monthly rent is $0.6 - 0.7$ times what the monthly payment would be on a fixed rate mortgage. In this case you have a greater than 85% chance of being better off renting. If the monthly rent is $0.7 - 0.8$ times the monthly payment, then it's hard to say which will be better financially. And for monthly rent > 0.8 times the monthly payment it is better financially to buy. This assume that property appreciates at about 2% per year. If that isn't the case for you, you can tweak the code and re-run to see how the results change. The more property appreciates, the worse it is for renters.

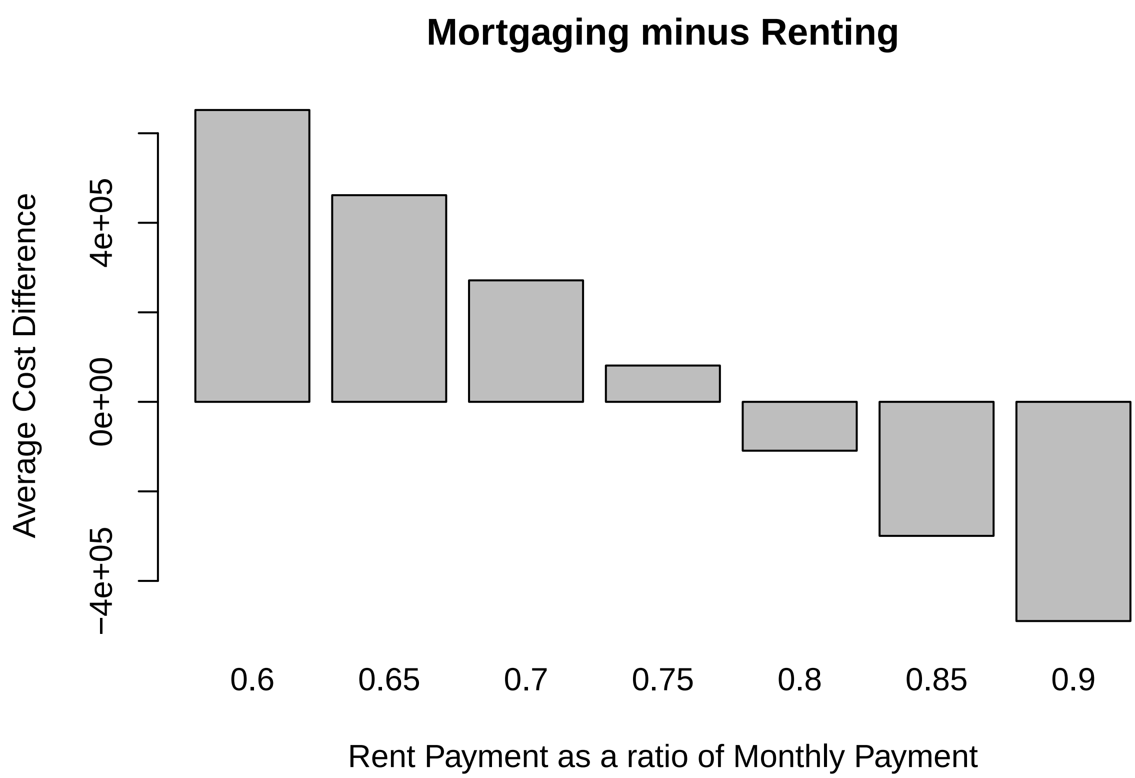
```
rent_ratios <- seq(min_ratio, max_ratio, step)
colnames(results) <- rent_ratios

count_mat <- results > 0
count_mat[,1] <- as.numeric(count_mat[,1])
count <- colSums(count_mat)
barplot(count/N, main='Mortgaging minus Renting',
        ylab = '% trajectories where cost mortgaging > cost renting',
        xlab = 'Rent Payment as a ratio of Monthly Payment', ylim=c(0,1))
```




```
count/N
```

```
##     0.6    0.65     0.7    0.75     0.8    0.85     0.9
## 0.98535 0.94519 0.85406 0.69040 0.45402 0.19818 0.04109
```

```
means <- colMeans(results)
barplot(means, main="Mortgaging minus Renting", ylab="Average Cost Difference",
        xlab="Rent Payment as a ratio of Monthly Payment")
```




```r
means
```

```
##        0.6       0.65        0.7       0.75        0.8       0.85        0.9
##  651931.23  461664.58  271397.94   81131.29 -109135.36 -299402.00 -489668.65
```

From the bottom graph we can see that if rent is 0.65 times the monthly payment, the renter will have on average about \$460000 more than the mortgager at the end of twenty-five years. But this comparison ends at twenty-five years, after the mortgage is payed off the mortgager doesn't have any more monthly payments and could put much more money into an investment every year. But with a such a head start and due to the way interest compounds, the renter will still come out on top as you can see from the approximation of the situation below (this is only an approximation since the monthly injection doesn't change over time).

```r
investment <- function(start = 0, injection, r = 1+0.06/12, n = 300) {
  res = start
  for(i in 1:n) {
    res = res*r + injection
  }
  return(res)
}

investment(start = 460000, injection = 650) # renter
```

```
## [1] 2504332
```

```r
investment(injection = 1000) # mortgager
```

```
## [1] 692994
```

These number are derived for comparing the same property, which isn't realistic since you won't have the option to buy/rent the same property in the real world. However, you can use the rule above to compare two different properties as long as you're happy with either property.

You could use this data/method to answer lots of questions about rental pricing. For example, if you own a condo that you want to rent out you might want to know what you need to price your rental at so that you have a $> 95\%$ chance to cover at least half of your monthly payment.

One thing that I didn't consider in my comparison was the appreciation of property value being added to the mortgager's net gains. I chose to ignore this because that money isn't actually accessible until you sell at which point you need to buy a new property which then ties up all or most of that money again.

Conclusions

This was an interesting project that taught me a lot. For one thing it made me realize that the forces that control the monthly payment on a mortgage are separate from the forces that control rental prices. The interest rates and initial property value (the principal) will inform a mortgage's monthly payment. Whereas the current property value informs the rental payment. So what this comes down to is, are you able to put enough towards an investment that appreciates faster than the property value in order to offset the loss of your rental payment? I also learned a little bit about stochastic differential equations to simulate interest rates.

I would also like to mention that this comparison is purely financial and assumes only one property. My guess is that if you are able to get your hands on multiple properties, that it then becomes better financially (but perhaps worse morally) to mortgage. This also doesn't consider the psychological security you could get from knowing that you own your home and not having to deal with the potential headache of a bad landlord/needing to move.

Another assumption of this comparison is that the renter invests all of the excess money they would have put towards their monthly payment towards investments. For most renters this isn't the case. I have heard people say that renting can be just as good as buying, and from this analysis it actually seems like that is sometimes the case. But that's only true where the potential buyer has the financial capabilities to choose between buying and renting. Simply put, mortgaging is more expensive month-to-month than renting. If you have the baseline income to even consider buying a home, then you are already better off than most people who "choose" to rent. A renter would have to put between 40% to 70% (0.3/0.7 to 0.4/0.6) of their monthly rent towards investments in order to out-earn the appreciation of the property they are living in (and they still need to actually pay the rent, so for renting to be better than mortgaging you can multiply the rent by a factor of about 1.5 to get monthly cost of living). For most people, who don't have the excess money to invest, this analysis does not apply. What I am trying to get at is that just because renting can be financially better than mortgaging **- if you cannot afford to mortgage in the first place then you are still screwed**.

A massive advantage of mortgaging if you have kids is that it gives you something to pass down. A mortgage essentially turns let's say approximately \$1 million of your dollars into $\$5 - \$10$ million of your kid's dollars. And you don't even need all \$1 million at once, you can start with as little as \$25000 as a downpayment. Essentially, mortgaging gives you access to money (and thus wealth appreciation) that you otherwise wouldn't have had access to. This is an amazing way to improve your child's life over yours, in fact it's probably the best way since they essentially won't need to buy or rent for their home.

The other thing to keep in mind is the location of most rentals which are dense cities. If you don't need to own a car for your day-to-day activities that is also a place where financial savings are made. Whereas if you live in the suburbs, a car is necessity for every day which adds to the cost of living. This isn't necessarily a win for renting as much as it is for dense cities, but it is something to consider.

My final thoughts are that if you don't pay off your mortgage and then live in that property for a while, you definitely should have rented instead. If you don't want kids, consider renting instead. If you need to move cities often for work, consider renting instead (this is probably obvious). If you want kids, then consider mortgaging for their financial future (potentially at the expense of yours). Essentially, renting can be better for the generational short term if your goal is to maximize your wealth over your life time. Mortgaging is better over the generational long term and sets your kids up for success. This goes out the window if you own multiple properties and charge rent to tenants, which I would bet makes mortgaging better.

Overall, this project does not aim to say "renting is better than mortgaging" or that "mortgaging is better than renting". Initially my goal was just to get a better understanding of both. Now I have a rule of thumb to make an informed decision about whether a given rental is fairly priced or whether I should just consider buying. But let's be honest, I will be renting for a while.