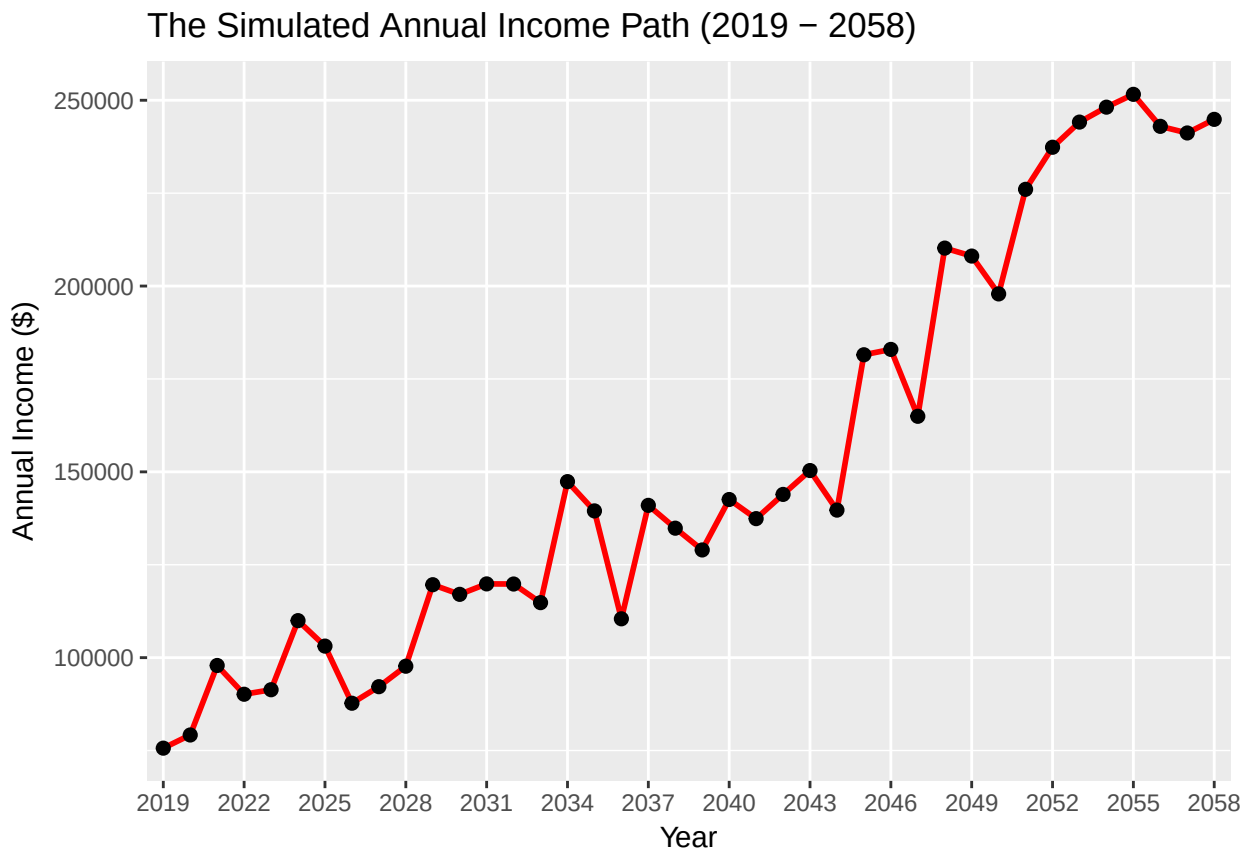
Assignment #4 Simulate Your Income

Xi Chen

November 6, 2017

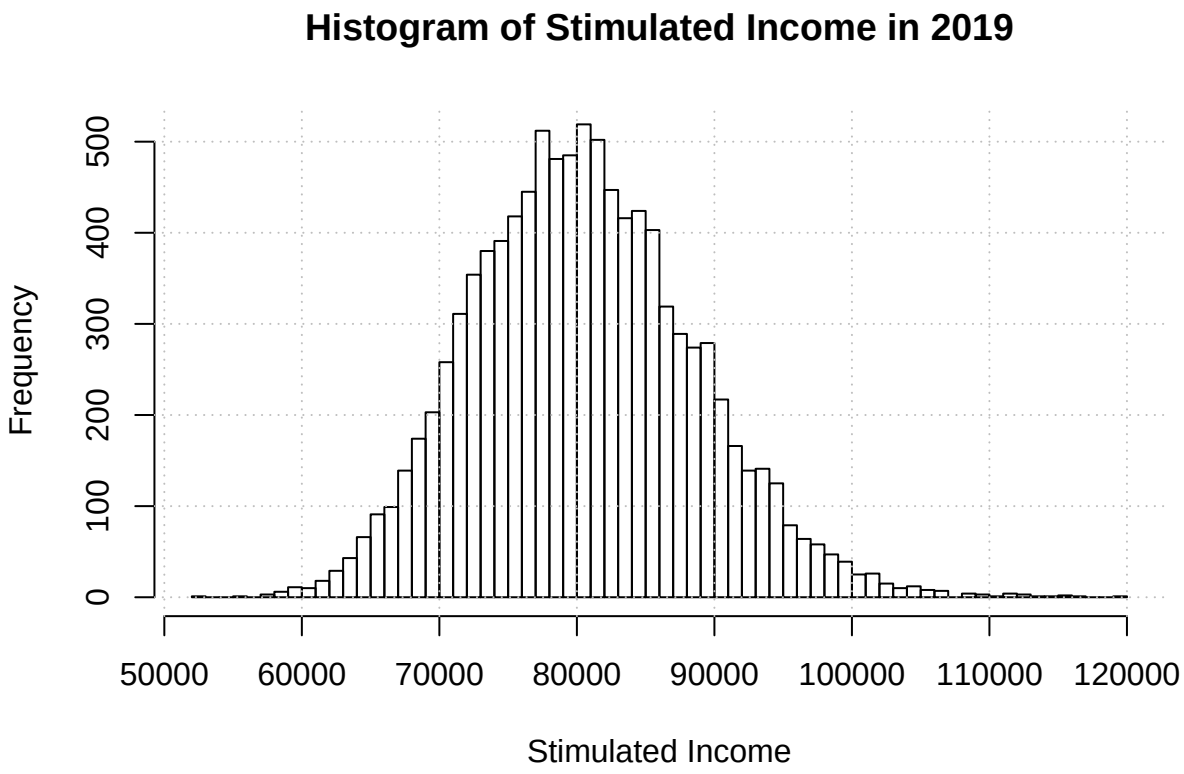
1. Plot one of the lifetime income paths

```
set.seed(123)

# Create a function of stimulating income
income_stimulate = function(error_mean, error_sd, rho, g, inc0, years,n_sample){

  #Create an empty matrix to store the stimulated income
  inc_ln = matrix(nrow=years, ncol=n_sample)

  # Create a matrix for log_error terms
  error_ln = matrix( rnorm(n=years*n_sample, mean=error_mean, sd=error_sd),
                     nrow=years, ncol=n_sample)

  # The log_income for the year of 2019
  inc_ln[1,] = log(inc0) + error_ln[1,]

  # The log_income for the year of 2020 to 2058
  for (t in 2:years){
    inc_ln[t,] = (1-rho)*( log(inc0)+g*(t-1) ) + rho*inc_ln[t-1,] + error_ln[t,]
  }
  inc = exp(inc_ln)
}

# Plug prameteres into the function
inc = income_stimulate(error_mean=0, error_sd=0.1, rho=0.2, g=0.03,
                       inc0=80000, years=40, n_sample=10000)

# Plot one of the lifetime income paths
options(warn=-1)
library(ggplot2)

inc = as.data.frame(inc)
rownames(inc) = c(2019:2058)
ggplot(inc, aes(rownames(inc), inc[,1], group=1)) +
  geom_line(color = "red", size=1) +
  geom_point(size = 2) +
  scale_x_discrete(breaks=seq(2019, 2058, 3)) +
  labs(title="The Simulated Annual Income Path (2019 - 2058)",
       x="Year", y="Annual Income ($)")
```



2. Plot a histogram of 2019 income

```
# Plot a historgram with 50 bins of 2019 Income for 10000 simulations
inc = as.matrix(inc)
inc_2019 = inc[1,]
hist(inc_2019, main="Histogram of Stimulated Income in 2019",
     xlab="Stimulated Income",breaks=50)
grid(col="gray")
```



Comments: The distribution is very close to a normally distribution. However, the right tail is a little bit longer than the left tail, which might indicate right skewness.

```
sum(inc_2019>100000)/10000
```

```
## [1] 0.0124
```

Comments: The percentage of the income which will be more than $100,000 is about 1.24%.

```
sum(inc_2019<70000)/10000
```

```
## [1] 0.0894
```

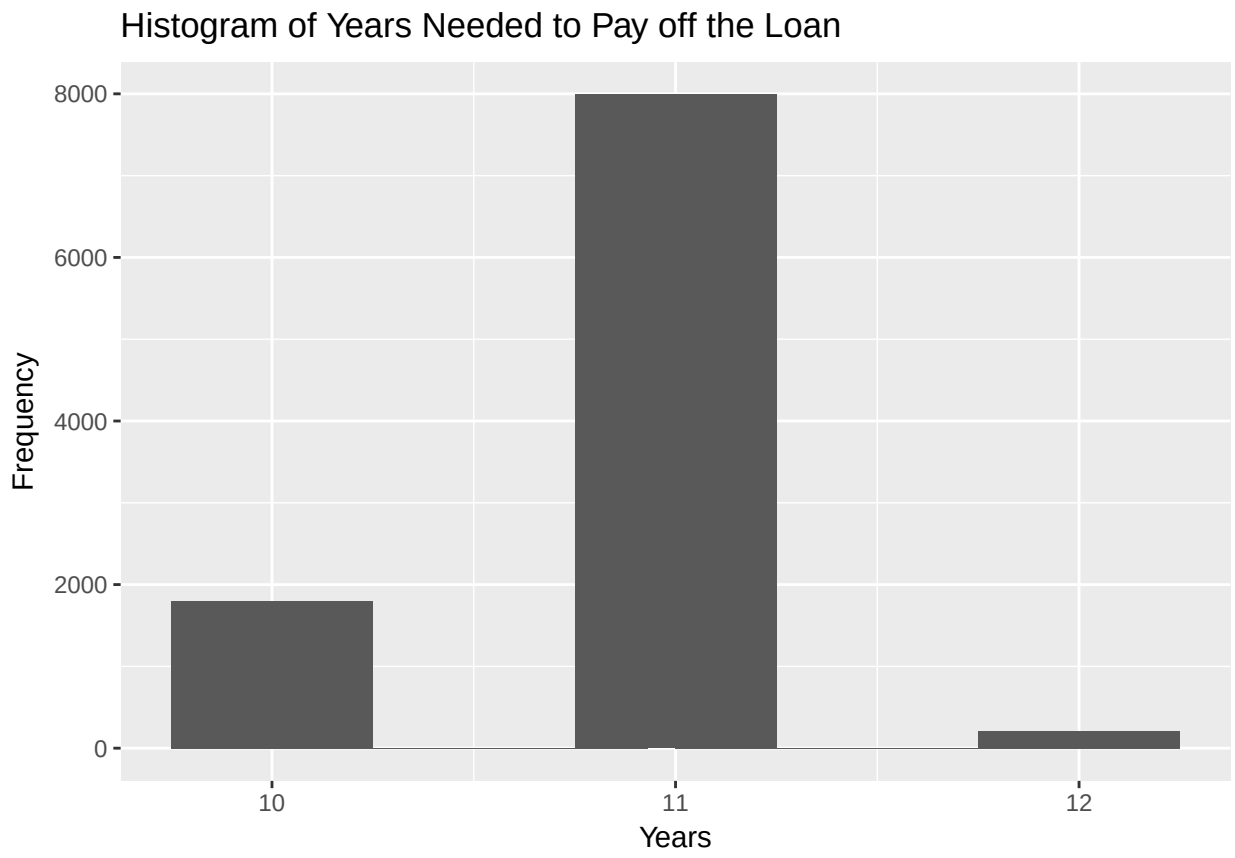
Comments: The percentage of the income which will be less than $70,000 is about 8.94%.

3. Plot the histogram of years needed to pay off the debt

```
total_debt=95000
pay_rate=0.1
n_sample=10000
debt_year = matrix(nrow=n_sample, ncol=1)

# Compute the years needed to pay off the debt for each simulation
for (n in 1:n_sample){

  stimulate_n = inc[,n]
  payment = 0
  pay_year = 0
```

```
    while(payment <= total_debt){
      payment = payment + stimulate_n[pay_year+1]*pay_rate
      pay_year = pay_year + 1
    }

    debt_year[n, 1] = pay_year
  }

  # Histogram of the years to pay off the loan
  debt_year=as.data.frame(debt_year)
  ggplot(debt_year, aes(debt_year)) +
    geom_histogram() +
    scale_x_continuous(breaks=seq(9, 13, 1)) +
    labs(title="Histogram of Years Needed to Pay off the Loan",
         x="Years", y="Frequency") + stat_bin(bins=5)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
  sum(debt_year<=10)/10000
```

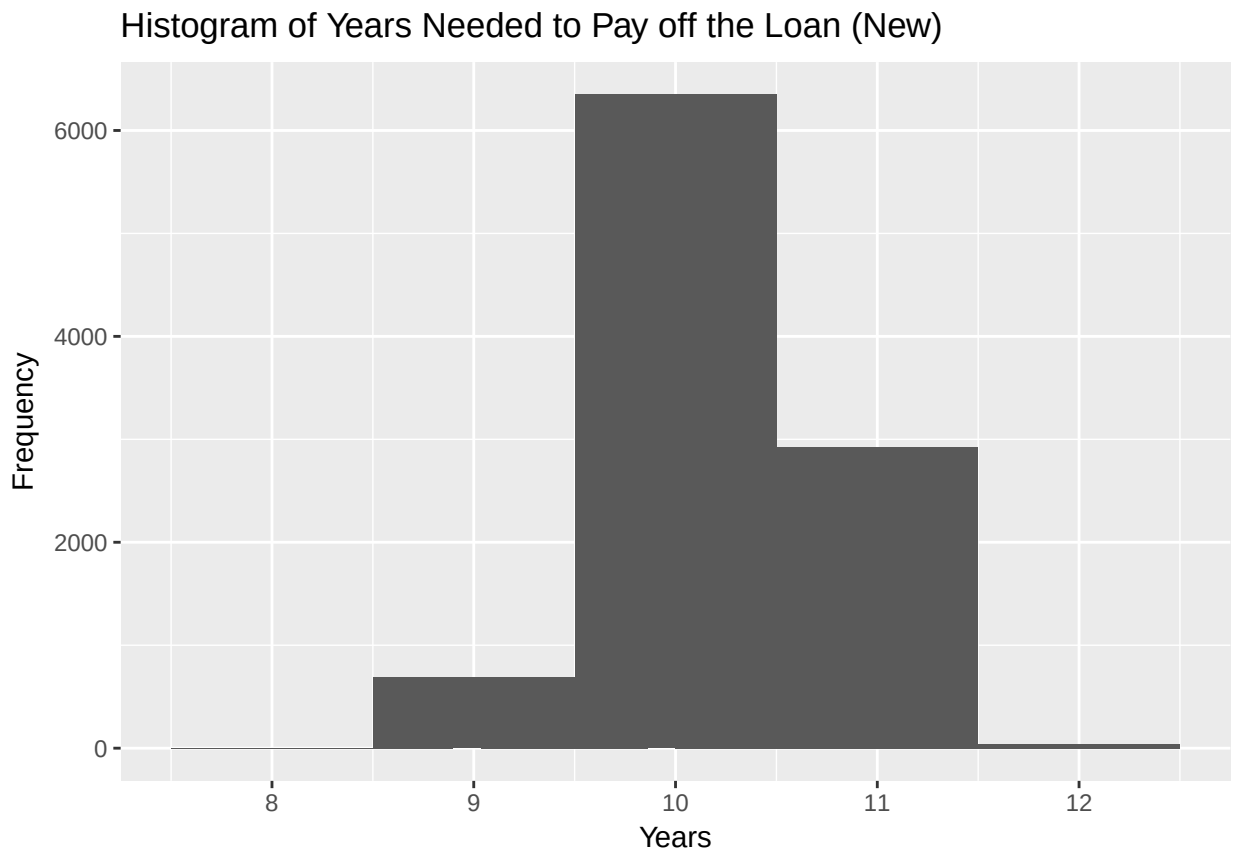
```
## [1] 0.18
```

Comments: The percentage that paying off the loan in 10 years in all stimulations is about 18%.

4. New stimulations with new parameters

```
# Plug new parameters into the function of income stimulation
set.seed(123)
new_inc = income_stimulate(error_mean=0, error_sd=0.15, rho=0.2, g=0.03,
                           inc0=85000, years=40, n_sample=10000)

# The years it takes to pay off the debt
new_debt_year = matrix(nrow=n_sample, ncol=1)
for (n in 1:n_sample){

  stimulate_n = new_inc[,n]
  payment = 0
  pay_year = 0

  while(payment <= total_debt){
    payment = payment + stimulate_n[pay_year+1]*pay_rate
    pay_year = pay_year + 1
  }

  new_debt_year[n, 1] = pay_year
}

# Histogram of the years to pay off the loan
options(warn=-1)
new_debt_year=as.data.frame(new_debt_year)
ggplot(new_debt_year, aes(new_debt_year)) +
  geom_histogram() +
  scale_x_continuous(breaks=seq(8, 12, 1)) +
  labs(title="Histogram of Years Needed to Pay off the Loan (New)",
       x="Years", y="Frequency") + stat_bin(bins=5)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
sum(new_debt_year<=10)/10000
```

```
## [1] 0.7035
```

Comments: With the new parameters, the percentage that paying off the loan in 10 years in all simulations is about 70.35%.