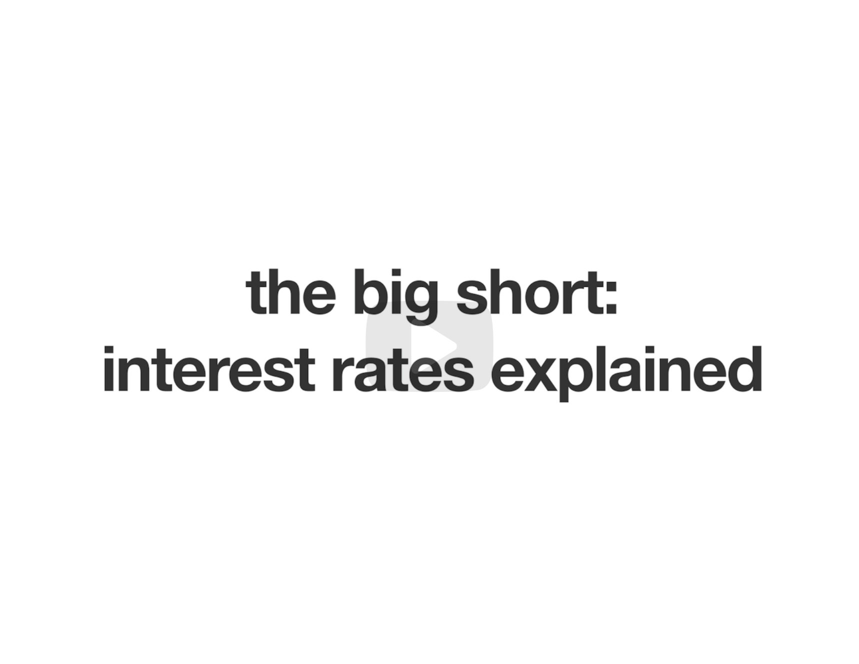
edX

Course > Section... > 4.1 The... > The Big...

The Big Short: Interest Rates Explained
The Big Short: Interest Rates Explained



0:00 / 0:00 1.0x

Start of transcript. Skip to the end.

RAFAEL IRIZARRY: In a way, the sampling models we've been talking about

are also used by banks to decide interest rates.

Let's see how this could be.

Suppose you run a small bank that has a history of identifying

potential homeowners that can be trusted to make payments.

Video
Download video file

Transcripts
Download SubRip (.srt) file
Download Text (.txt) file

Textbook link

This video corresponds to the textbook section on interest rates.

Correction

At 2:35, the displayed results of the code are incorrect. Here are the correct values:

```
n*(p*loss_per_foreclosure + (1-p)*0)
[1] -4e+06
sqrt(n)*abs(loss_per_foreclosure)*sqrt(p*(1-p))
[1] 885438
```

Key points

- Interest rates for loans are set using the probability of loan defaults to calculate a rate that minimizes the probability of losing money.
- We can define the outcome of loans as a random variable. We can also define the sum of outcomes of many loans as a random variable.
- The Central Limit Theorem can be applied to fit a normal distribution to the sum of profits over many loans. We can use properties of the normal distribution to calculate the interest rate needed to ensure a certain probability of losing money for a given probability of default.

Code: Interest rate sampling model

```
n <- 1000
loss_per_foreclosure <- -200000
p <- 0.02
defaults <- sample( c(0,1), n, prob=c(1-p, p), replace = TRUE)
sum(defaults * loss_per_foreclosure)
```

Code: Interest rate Monte Carlo simulation

```
B <- 10000
losses <- replicate(B, {
    defaults <- sample( c(0,1), n, prob=c(1-p, p), replace = TRUE)
  sum(defaults * loss_per_foreclosure)
})
```

Code: Plotting expected losses

```
library(tidyverse)
data.frame(losses_in_millions = losses/10^6) %>%
    ggplot(aes(losses_in_millions)) +
    geom_histogram(binwidth = 0.6, col = "black")
```

Code: Expected value and standard error of the sum of 1,000 loans

```
n*(p*loss_per_foreclosure + (1-p)*0)    # expected value
sqrt(n)*abs(loss_per_foreclosure)*sqrt(p*(1-p))    # standard error
```

Code: Calculating interest rates for expected value of 0

We can calculate the amount x to add to each loan so that the expected value is 0 using the equation $lp + x(1-p) = 0$. Note that this equation is the definition of expected value given a loss per foreclosure l with foreclosure probability p and profit x if there is no foreclosure (probability $1-p$).

We solve for $x = -\frac{lp}{1-p}$ and calculate x:

```
x = - loss_per_foreclosure*p/(1-p)
x
```

On a $180,000 loan, this equals an interest rate of:

```
x/180000
```

Equations: Calculating interest rate for 1% probability of losing money

We want to calculate the value of x for which $\Pr(S < 0) = 0.01$. The expected value $\mathrm{E}[S]$ of the sum of $n = 1000$ loans given our definitions of x, l and p is:

$$\mu_S = (lp + x(1-p)) * n$$

And the standard error of the sum of n loans, $\mathrm{SE}[S]$, is:

$$\sigma_S = \mid x - l \mid \sqrt{np(1-p)}$$

Because we know the definition of a Z-score is $Z = \frac{x-\mu}{\sigma}$, we know that $\Pr(S < 0) = \Pr(Z < -\frac{\mu}{\sigma})$. Thus, $\Pr(S < 0) = 0.01$ equals:

$$\Pr\left(Z < \frac{-\{lp + x(1-p)\}n}{(x-l)\sqrt{np(1-p)}}\right) = 0.01$$

`z<-qnorm(0.01)` gives us the value of z for which $\Pr(Z \leq z) = 0.01$, meaning:

$$z = \frac{-\{lp + x(1-p)\}n}{(x-l)\sqrt{np(1-p)}}$$

Solving for x gives:

$$x = -l\frac{np - z\sqrt{np(1-p)}}{n(1-p) + z\sqrt{np(1-p)}}$$

Code: Calculating interest rate for 1% probability of losing money

```
l <- loss_per_foreclosure
z <- qnorm(0.01)
x <- -l*( n*p - z*sqrt(n*p*(1-p)))/ ( n*(1-p) + z*sqrt(n*p*(1-p)))\x
x/180000    # interest rate
loss_per_foreclosure*p + x*(1-p)    # expected value of the profit per
n*(loss_per_foreclosure*p + x*(1-p)) # expected value of the profit ove
```

Code: Monte Carlo simulation for 1% probability of losing money

Note that your results will vary from the video because the seed is not set.

```
B <- 100000
profit <- replicate(B, {
    draws <- sample( c(x, loss_per_foreclosure), n,
                        prob=c(1-p, p), replace = TRUE)
    sum(draws)
})
mean(profit)    # expected value of the profit over n loans
mean(profit<0)    # probability of losing money
```