How To Predict The Future

Joseph T. Ornstein

2025-06-23

Table of contents

Welcome 3

I Uncertainty 4

1 Fundamentals of Probability 5

II Wisdom of Crowds 6

2 Condorcet Jury Theorem 7

3 Median Voter Theorem 13

III Nonlinearity & Chaos 14

4 Exponential Growth & Decay 15

4.0.1 Application 15

References 22

Welcome

This is a Quarto book. Here is a reference Black (1948)

To learn more about Quarto books visit https://quarto.org/docs/books.

```
1 + 1
```

```
[1] 2
```

Part I

Uncertainty

1 Fundamentals of Probability

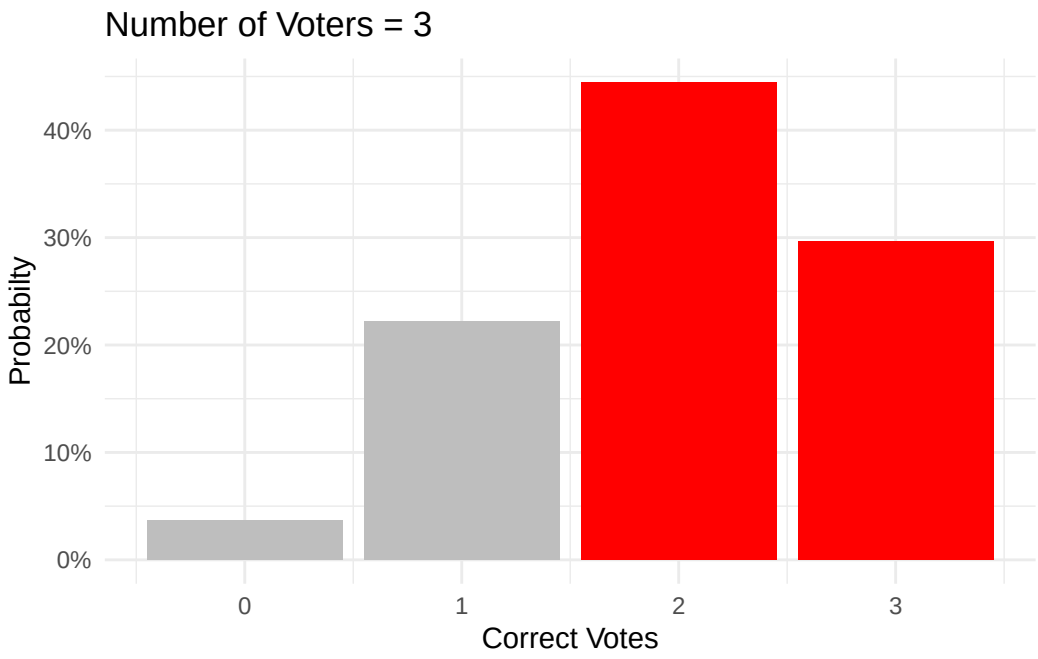
Trees

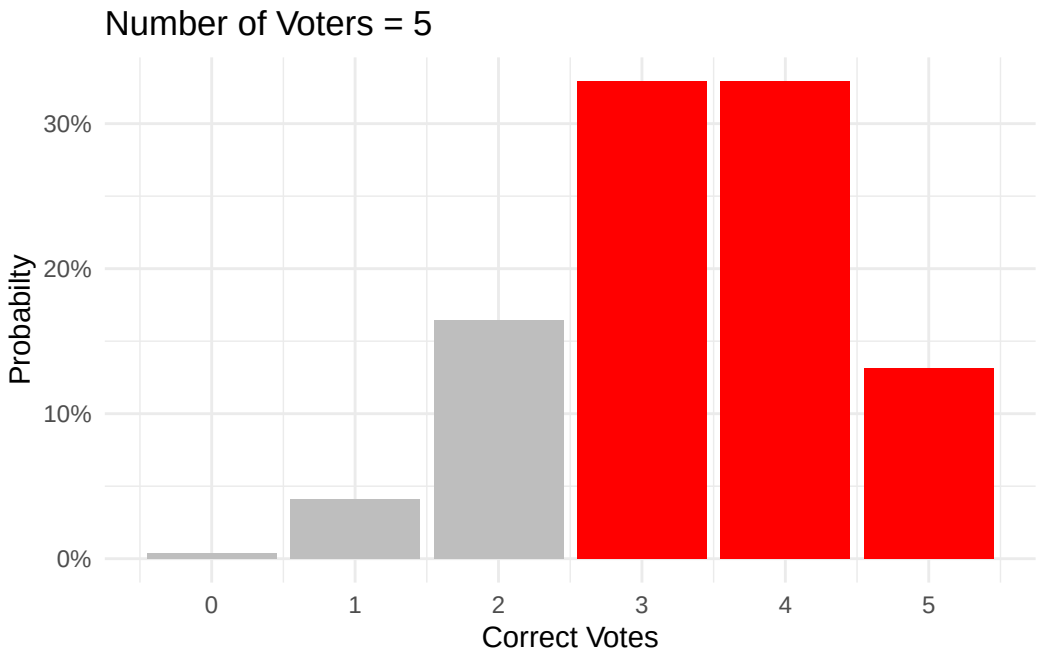
Joint Probability

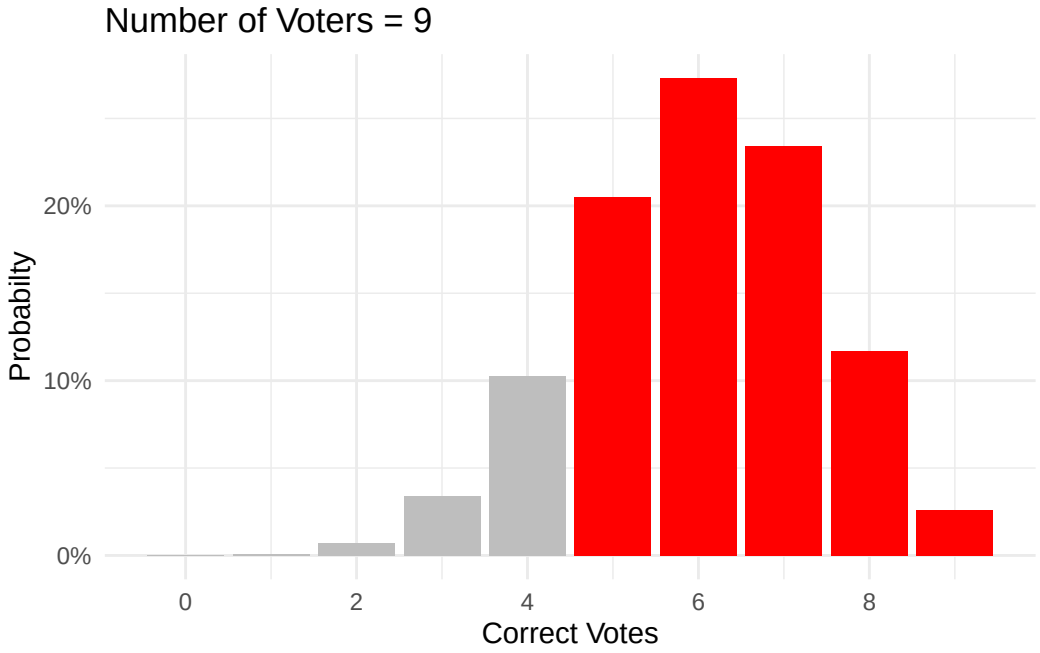
Part II

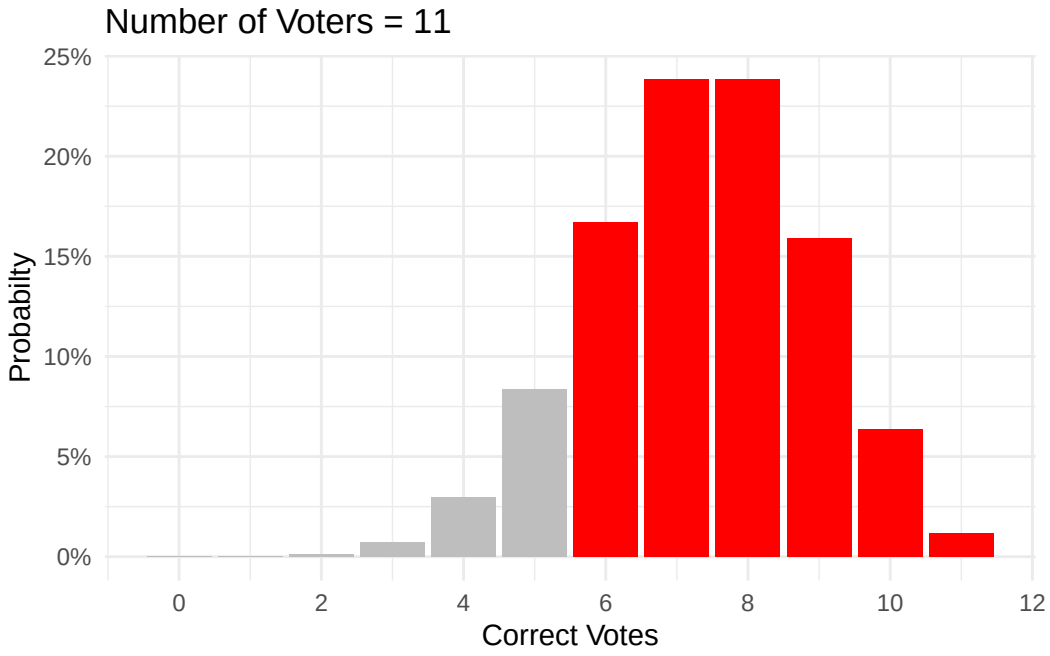
Wisdom of Crowds

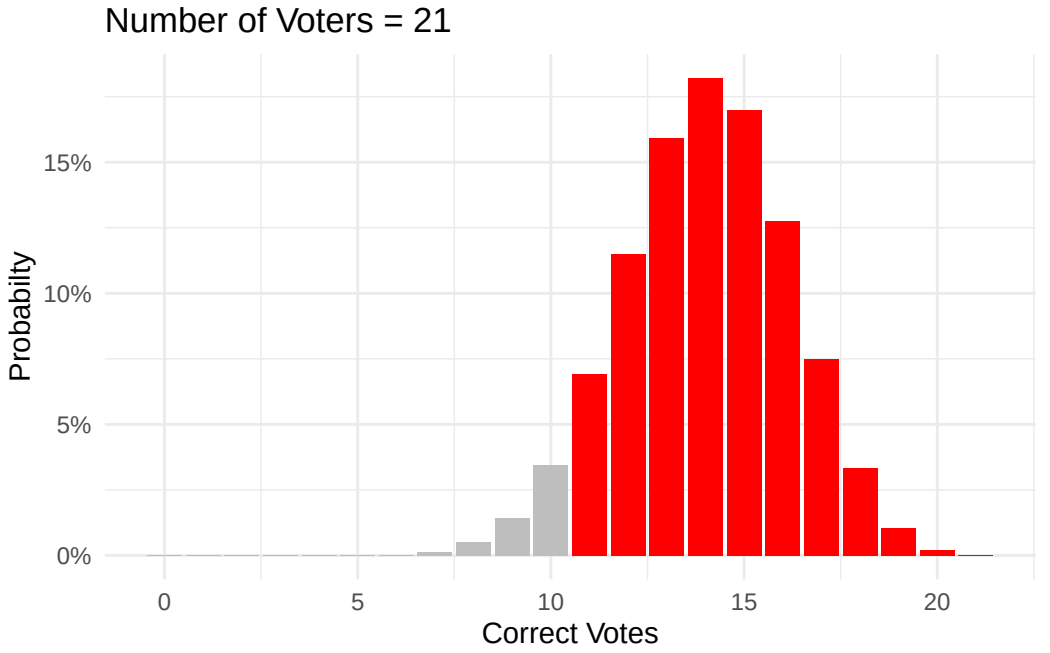
2 Condorcet Jury Theorem

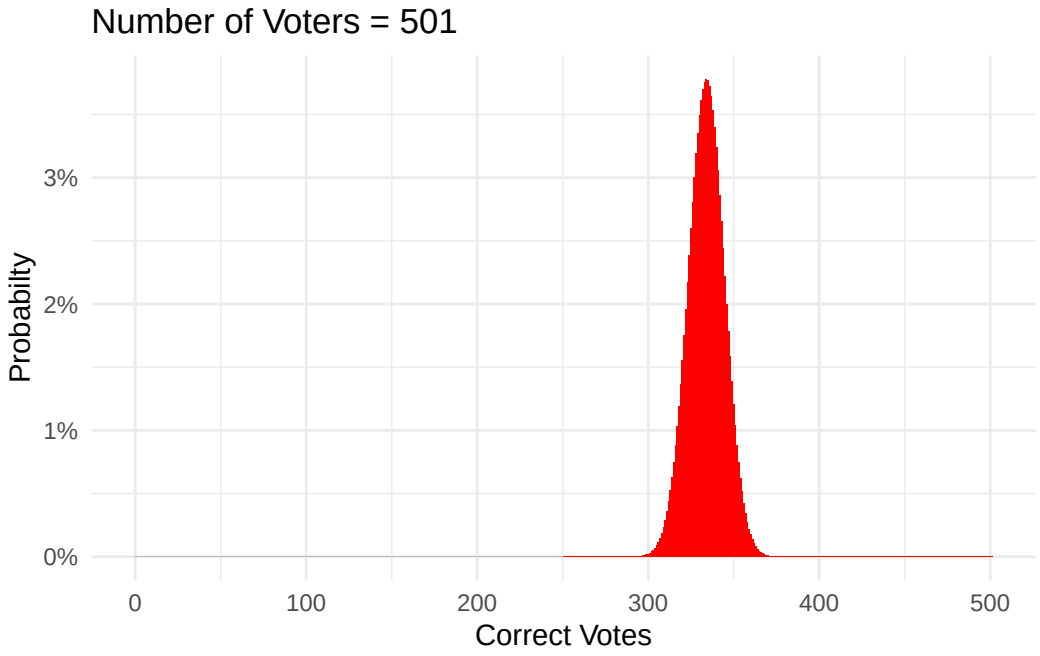



Number of Voters = 5
30%
20%
10%
0%
Probabilty
0
1
2
3
4
5
Correct Votes


Number of Voters = 9
20%
10%
0%
Probabilty
0
2
4
6
8
Correct Votes


Number of Voters = 11
25%
20%
15%
10%
5%
0%
Probabilty
0
2
4
6
8
10
12
Correct Votes


Number of Voters = 21
15%
10%
5%
0%
Probabilty
0
5
10
15
20
Correct Votes



Let's define a **wise crowd** as a group where you're 95% certain that the majority will make the correct decision.How many people do you need in your crowd before you can be certain that the crowd decisions will be wise? That's equivalent to asking what is the minimum number of people n such that the 95% confidence interval of the binomial distribution lies entirely above $\frac{1}{2}$.

The 95% confidence interval is approximately p plus or minus $2\sqrt{\frac{p(1-p)}{n}}$, so we want the value of n such that the lower bound of the confidence interval is greater than $\frac{1}{2}$:[1]

$$p - 2\sqrt{\frac{p(1-p)}{n}} > \frac{1}{2}$$

Rearrange some terms:

$$p - \frac{1}{2} > 2\sqrt{\frac{p(1-p)}{n}}$$

Divide by 2 and square each side:

[1] Readers with a statistics background may note that the 2 rounds up from a more precise 1.96; that makes the following computations somewhat conservative because the confidence intervals will contain slightly more than 95% of observations.

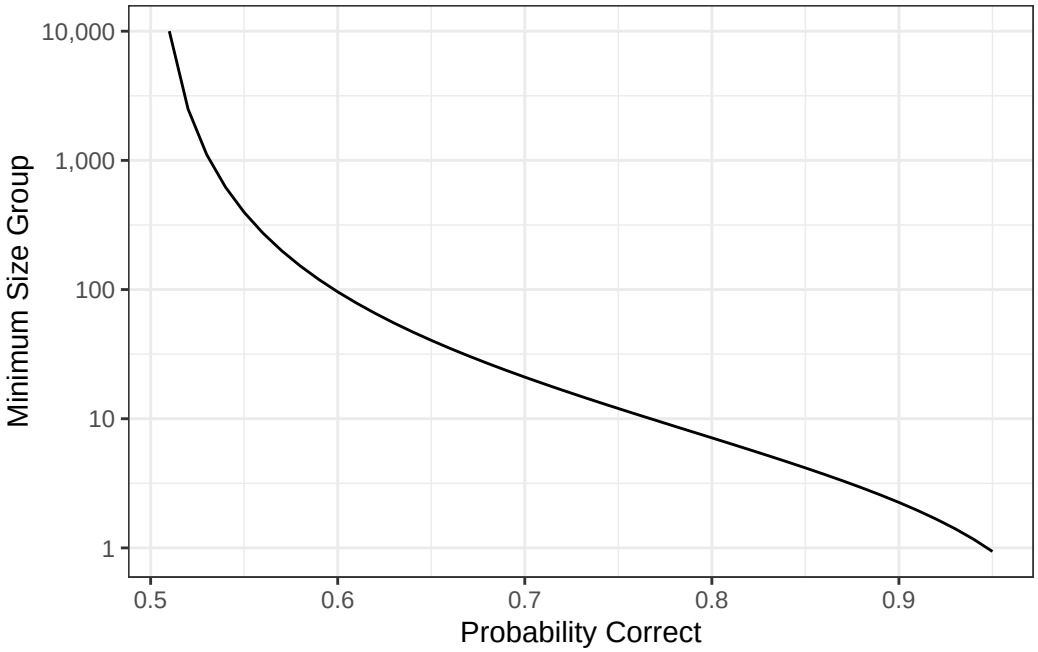
$$\left(\frac{p-\frac{1}{2}}{2}\right)^2 > \frac{p(1-p)}{n}$$

Multiply each side by $4n$ and divide by $(p-\frac{1}{2})^2$:

$$n > \frac{4p(1-p)}{(p-\frac{1}{2})^2}$$

This is a pretty clunky equation, but it helps confirm a few of our earlier intuitions. Notice the denominator on the right hand side. Whenever $p < \frac{1}{2}$, that denominator is *negative*, meaning that when you think your individuals are less likely than chance to select the best answer, you're not going to get better answers by adding more people!

Because of the square term that denominator can get *really small* when p is close to 50%, suggesting that we'll need a *much* larger crowd if the individuals are only barely better than chance at getting the right answer. Let's plot that equation to get a sense of what it looks like:



Notice that the y-axis is on a **logarithmic scale** (each increment corresponds to a tenfold increase). When $p = 0.51$, you need a group of 10,000 people before you can be confident that the majority will get the right answer. But you only a group of 1,000 people if $p = 0.53$! If $p = 0.6$, a group of 100 will do fine. 10 people is enough if you think $p > 0.77$. And, of course, if $p > 0.95$, then you don't have to bother with a crowd at all; one person will suffice.

All this suggests that there's a "sweet spot"; where the problem isn't so hard that it just makes sense to find the expert where you're certain they've got the right answer, or so easy that you can just pluck a person at random and they'll probably give you the right answer. Makes the most sense to harness the wisdom of the crowd for a problem that's hard but not too hard; and the math here suggests that you don't need that big a crowd for this class of problem.

TODO: Cross-reference your math here against `rbinom()`. The symmetry of the confidence interval may be problematic here; see Wald corrections.

Footnote: By the way, phrased differently this looks like a problem we'll return to when we discuss political polling (link): how many people do you need in your poll before you're 95% certain that you've correctly identified which candidate is in the lead?

3 Median Voter Theorem

Extend Condorcet Jury Theorem to continuous choices. Black (1948)

Part III

Nonlinearity & Chaos

4 Exponential Growth & Decay

Story about the chessboard. For me, one of the most important features of the story is that, after a certain amount of time, the king just executes the guy. Suggests that it's unwise to forecast based on naively extrapolating exponential growth!

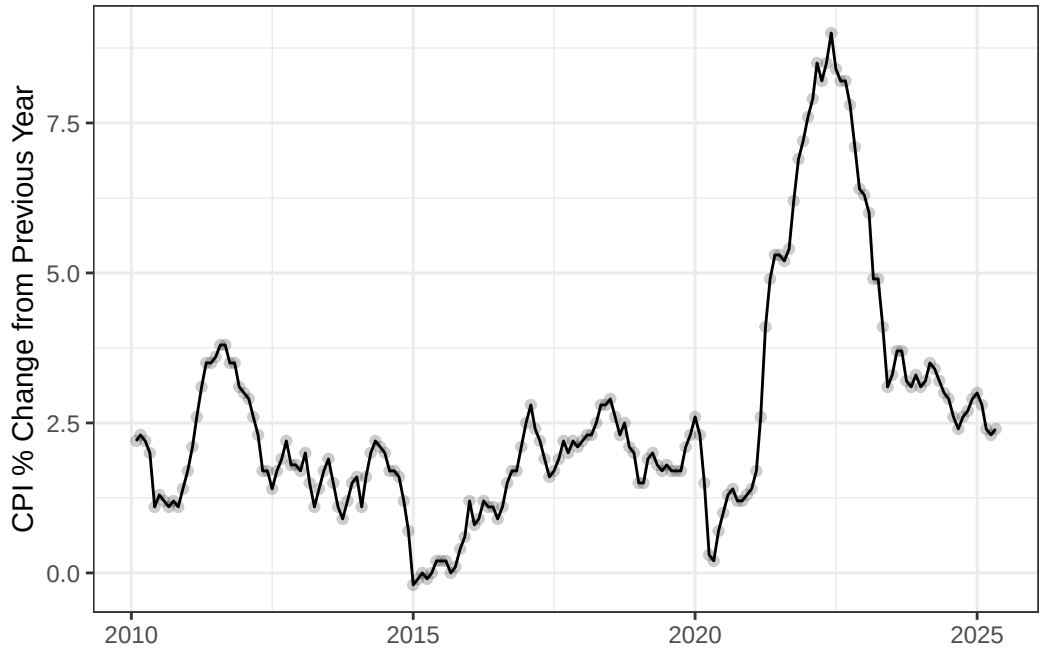
4.0.1 Application

```
library(readxl)
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.2     v tibble    3.2.1
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.0.4
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

```
d <- read_xlsx('data/CPIAUCSL.xlsx')

# compute annual from monthly
d <- d |>
  arrange(observation_date) |>
  mutate(index = 100 * cumprod(1 + pct_change_month/100)) |>
  mutate(annual_computed = (index / lag(index, 12) - 1) * 100)

d |>
  filter(observation_date > '2010-01-01') |>
  ggplot(mapping = aes(x = observation_date,
                       y = pct_change_year)) +
```

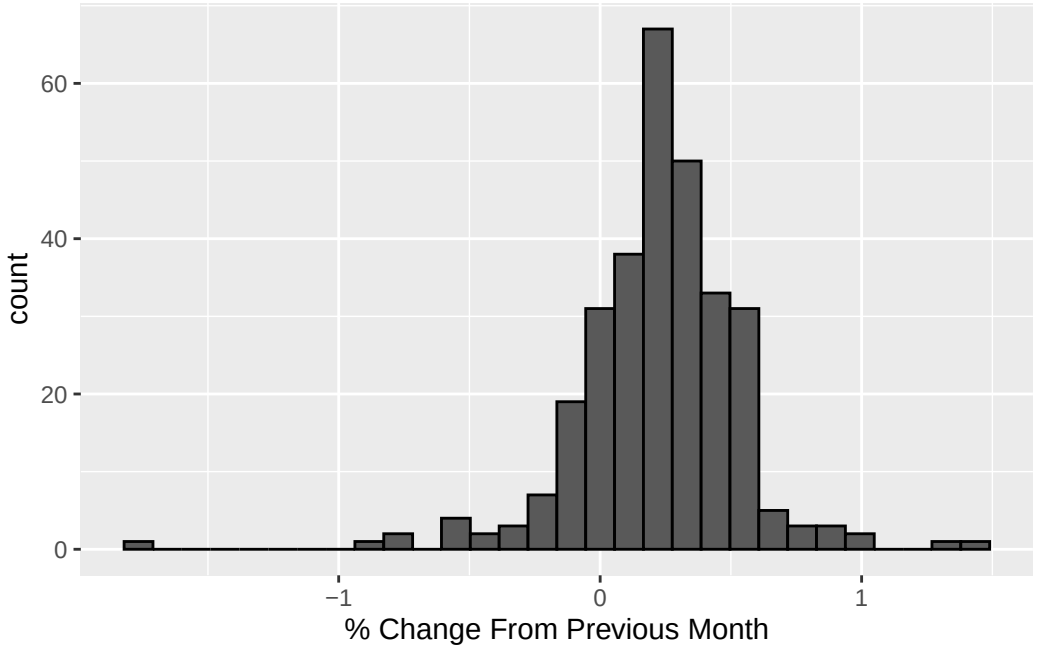
```
  geom_point(alpha = 0.2) +
  geom_line() +
  theme_bw() +
  labs(x = NULL, y = 'CPI % Change from Previous Year')
```




```
d |>
  filter(observation_date > '2010-01-01') |>
  ggplot(mapping = aes(x = observation_date)) +
  geom_line(aes(y = pct_change_year)) +
  geom_line(aes(y = annual_computed), color = 'red') +
  theme_bw() +
  labs(x = NULL, y = 'CPI % Change from Previous Year')
```

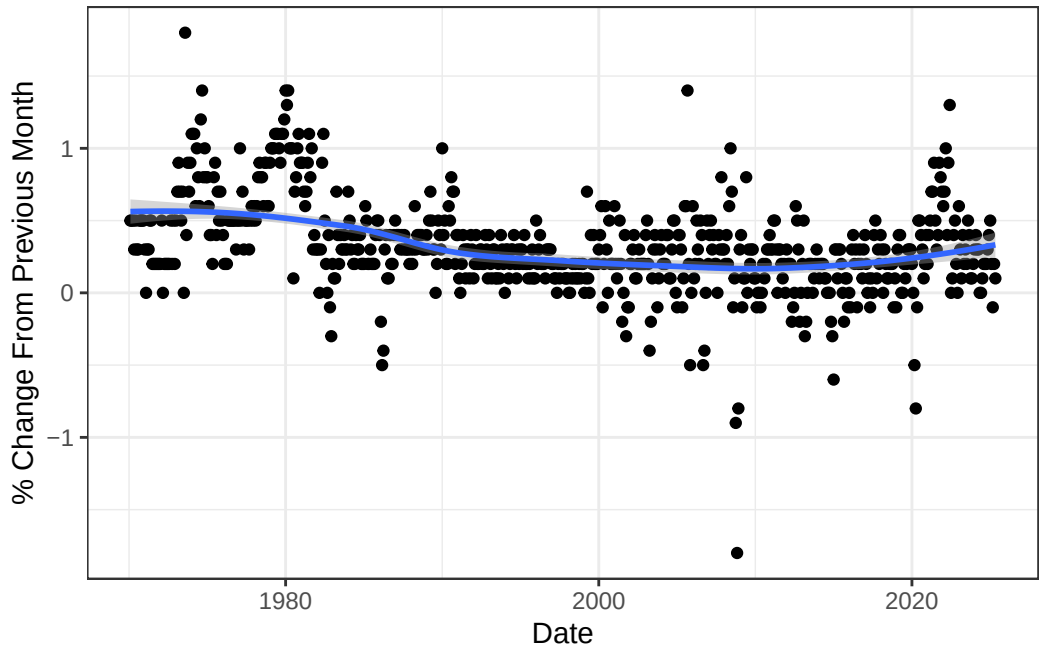

```
# just some rounding errors but otherwise looks perfect

d |> filter(observation_date > '2000-01-01') |> ggplot(mapping = aes(x=pct_change_month)) + g
```

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



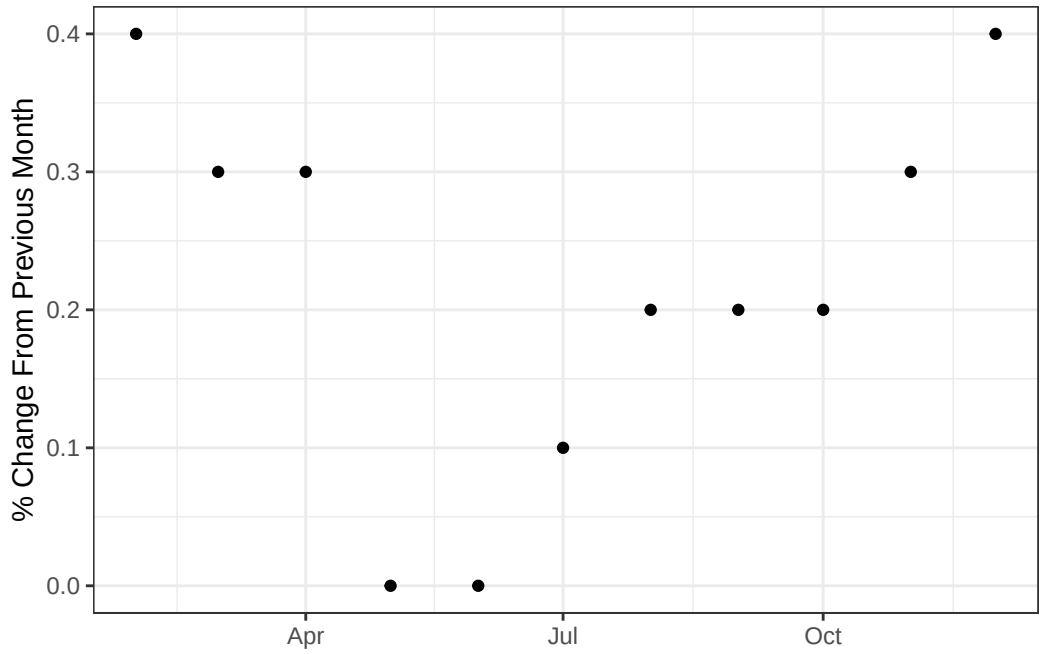
```
d |> filter(observation_date > '1970-01-01') |> ggplot(mapping = aes(x=observation_date, y =
```

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

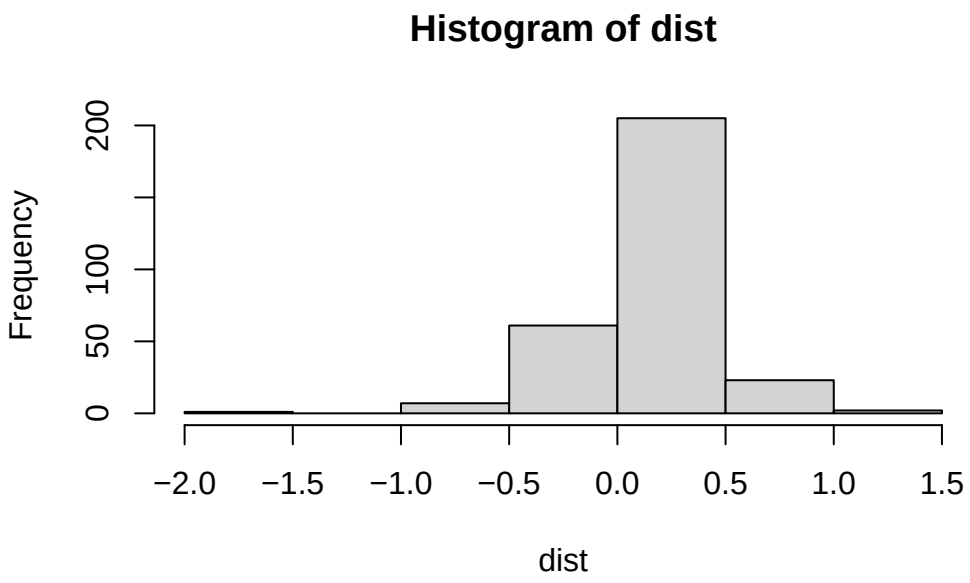


When trying to guess the inflation rate from January 2024 to January 2025, it would be silly to ignore the fact that we *already know what happened in 11 of those months*.

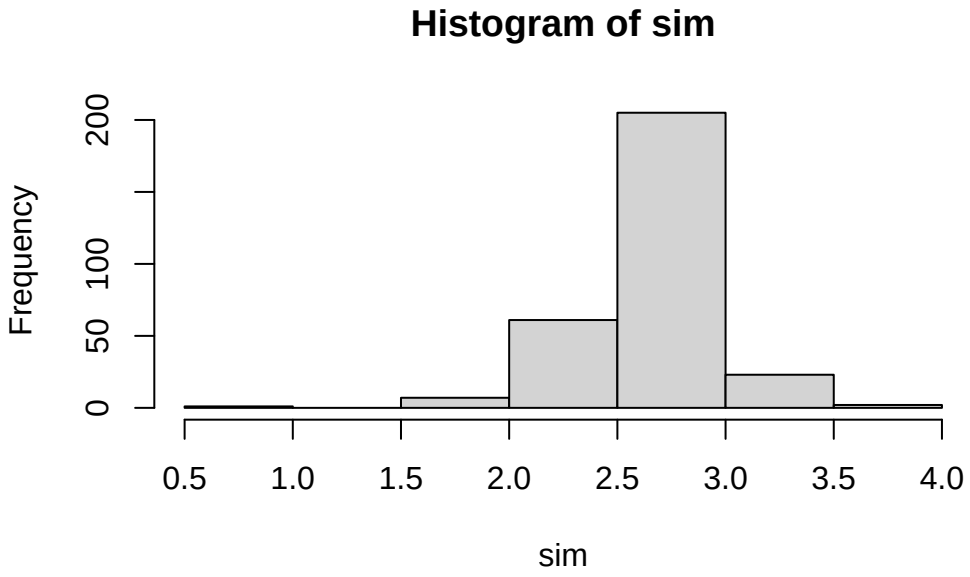
```
d |>
  filter(observation_date > '2024-01-31',
         observation_date < '2024-12-31') |>
  ggplot(mapping = aes(x=observation_date,
                       y=pct_change_month)) +
  geom_point() +
  labs(x = NULL,
       y = '% Change From Previous Month') +
  theme_bw()
```




```
projected_inflation <- function(r){
  prev11 <- d |>
    filter(observation_date > '2024-01-31',
           observation_date < '2024-12-31') |>
    pull(pct_change_month) / 100 + 1
  (prod(prev11) * (1 + r/100) - 1) * 100
}

# pipe observed distribution of month-on-month rates into that function
dist <- d |> filter(observation_date >= '2000-01-01',
            observation_date < '2024-12-31') |>
  pull(pct_change_month)
hist(dist)
```



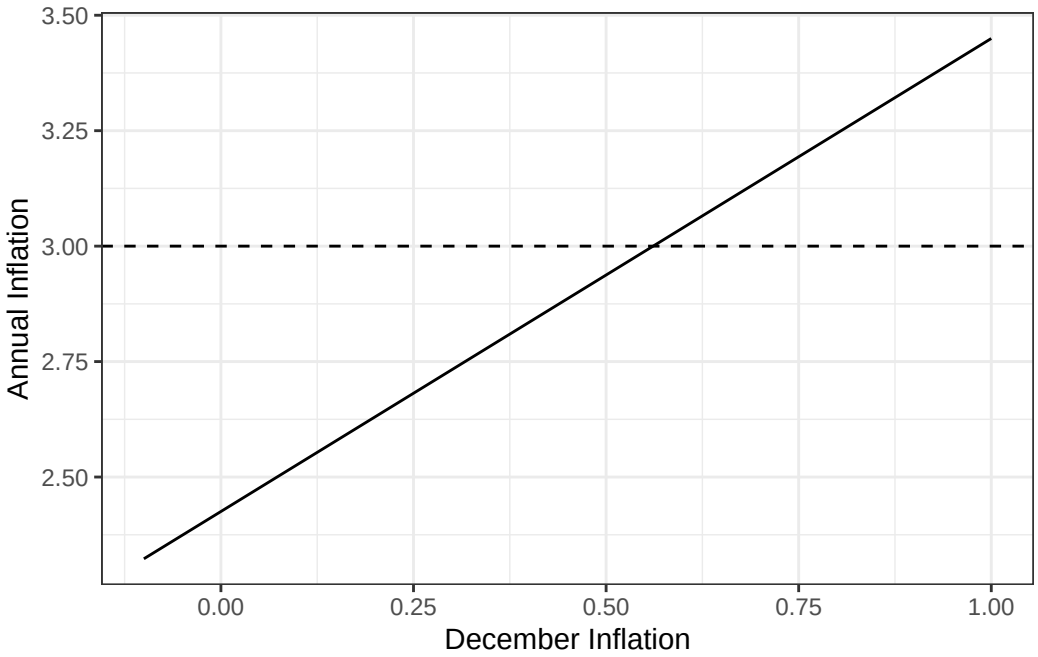

```
sim <- sapply(dist, projected_inflation)

hist(sim)
```




```
sum(sim > 3.05) / length(sim) * 100 # 5%!!
```

```
[1] 5.016722
```

```r
# what's the monthly rate that yields > 3%?
r <- seq(-0.1,1, by = 0.1)
projected <- sapply(r, projected_inflation)

ggplot(mapping = aes(x=r, y=projected)) +
  geom_line() +
  theme_bw() +
  labs(x = 'December Inflation',
       y = 'Annual Inflation') +
  geom_hline(yintercept = 3, linetype = 'dashed')
```



Then you can do the 10-year breakeven bit.

References

Black, Duncan. 1948. "On the Rationale of Group Decision-Making." *Journal of Political Economy* 56 (1): 23–34. https://doi.org/10.1086/256633.