Modeling measurement data

Stat 340: Bayesian Statistics

1. Inference for μ with known σ

(Problem topics 1-5)

Change from 2016 to 2020

Image source: https://www.washingtonpost.com/graphics/2020/elections/electorate-changes-2016-election-vs-2020/

Change in the democratic vote

- How did democratic share of the two party vote change from 2016 to 2020?
- MIT Election Data and Science Lab has county-level election results
- We'll look at the percent change in the two-party vote

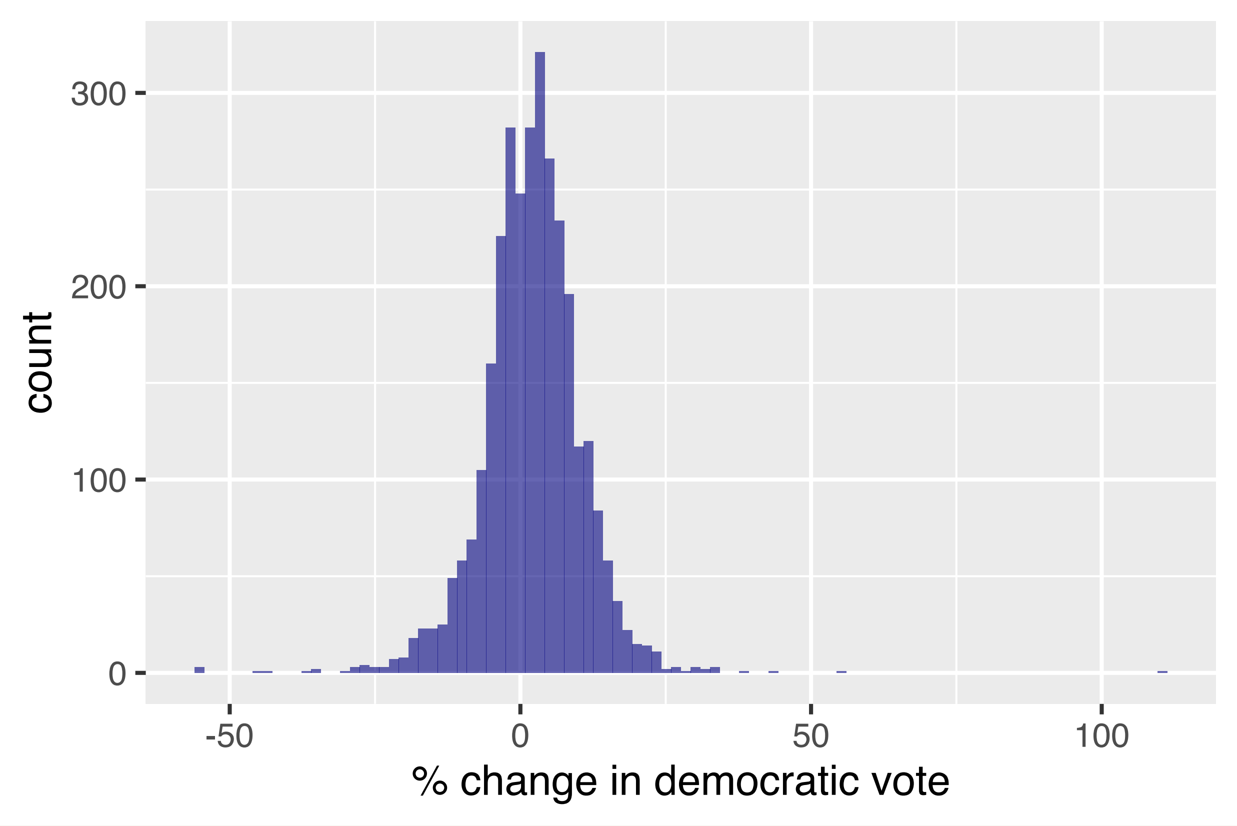
$$Y_i = 100\,(A_i/B_i - 1)$$

$A_i =$ % of two-party vote cast for democrats in 2020

$B_i =$ % of two-party vote cast for democrats in 2016

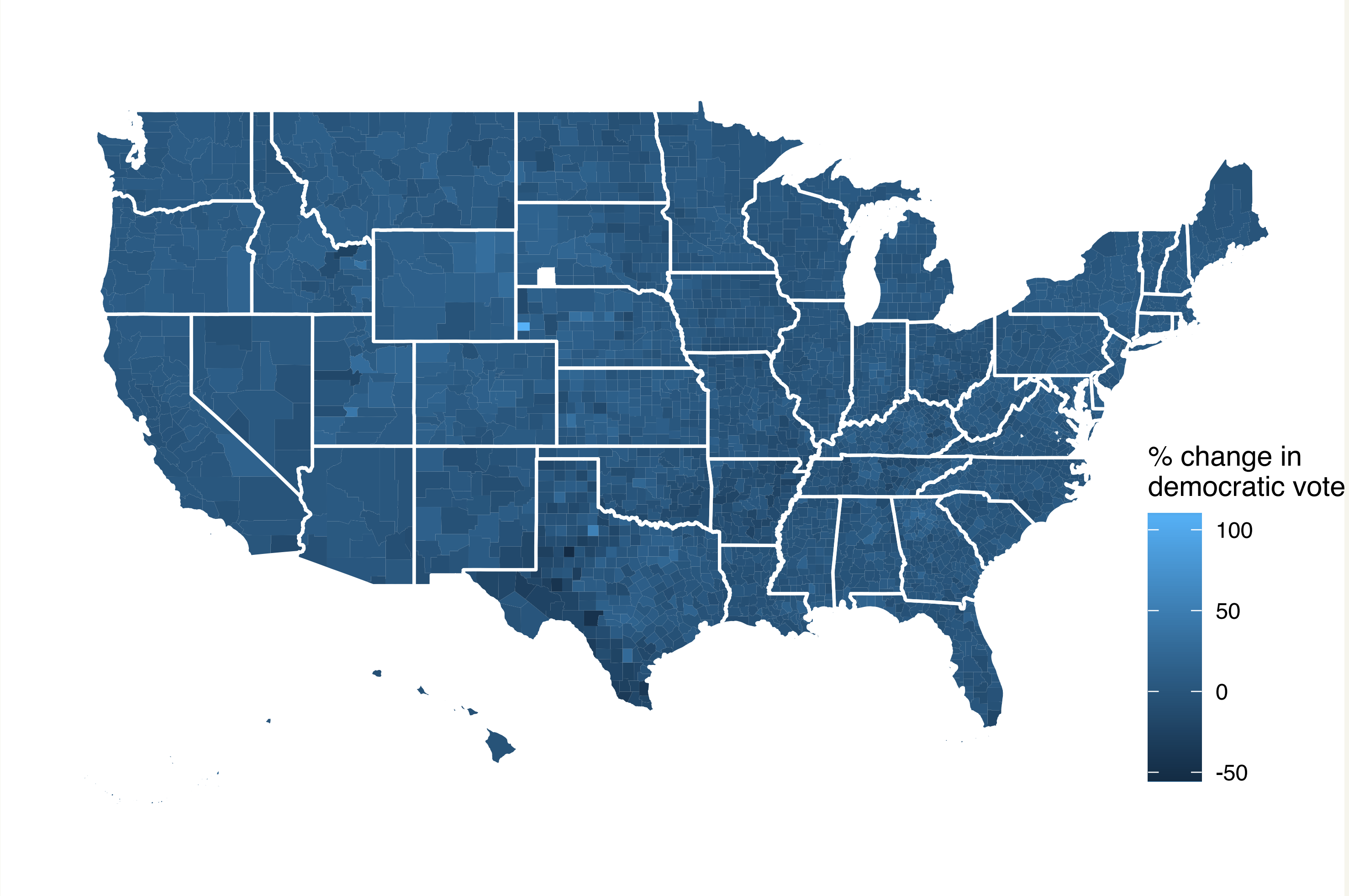
MIT Election Data and Science Lab, "County Presidential Election Returns 2000-2020", https://doi.org/10.7910/DVN/VOQCHQ, Harvard Dataverse, V9

EDA

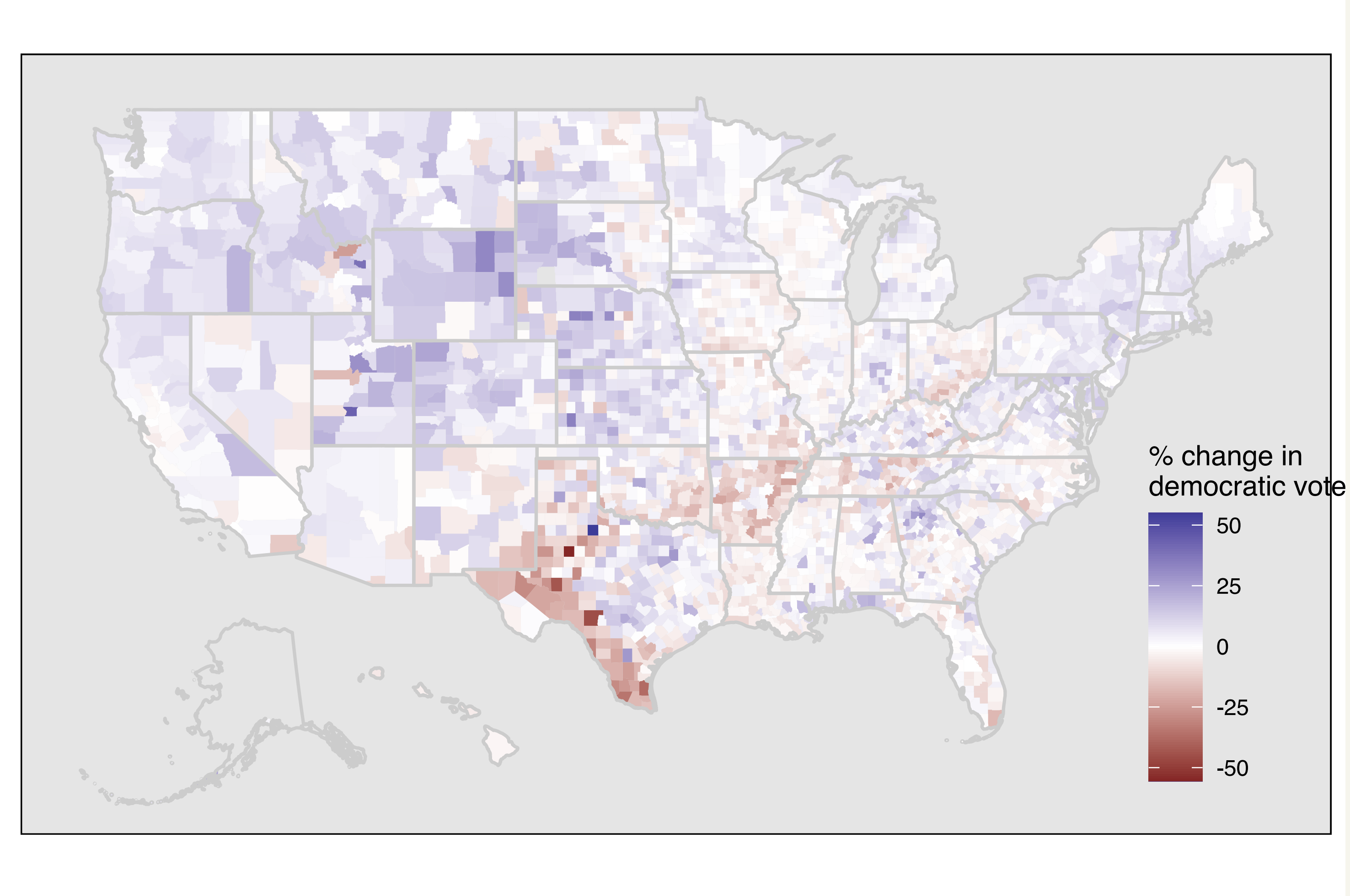


- Normal model seems reasonable
- Potential outliers

EDA

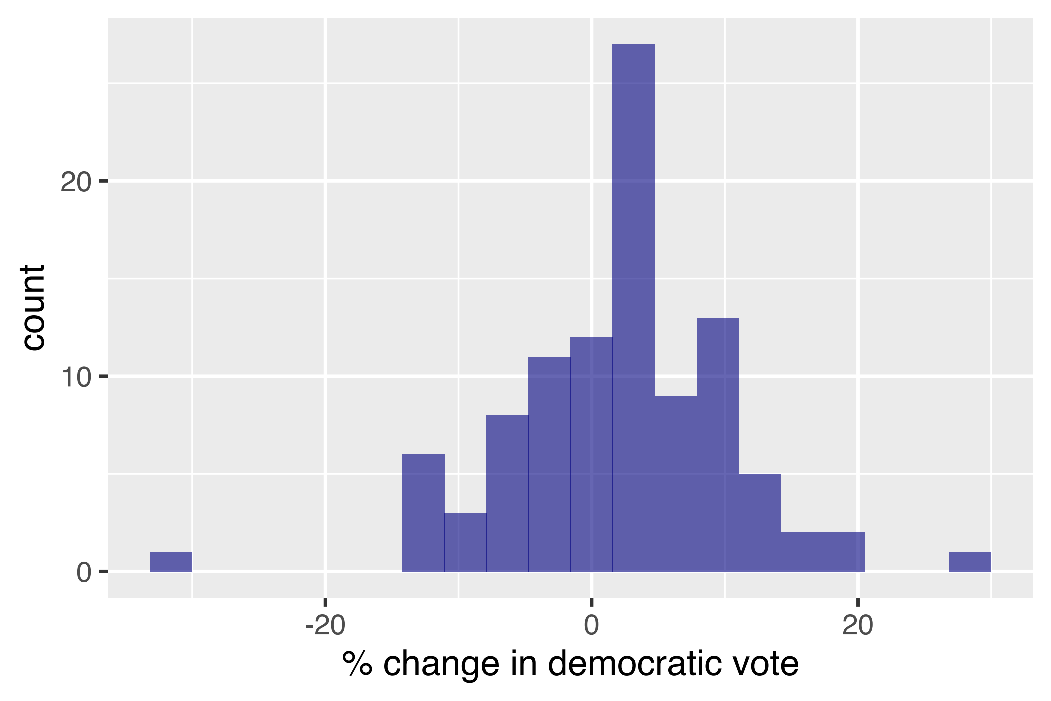


Without outliers



Hypothetical sample

We'll work with a hypothetical sample of 100 counties



Your turn 1

Derive the likelihood if we assume $Y_i \overset{\text{iid}}{\sim} \mathcal{N}(\mu, \sigma)$

Normal PDF:

$$f(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right), \quad -\infty < y < \infty, \quad -\infty < \mu < \infty, \quad \sigma > 0$$

02:30

Your turn 2

For simplicity, assume that $\sigma = 8$.

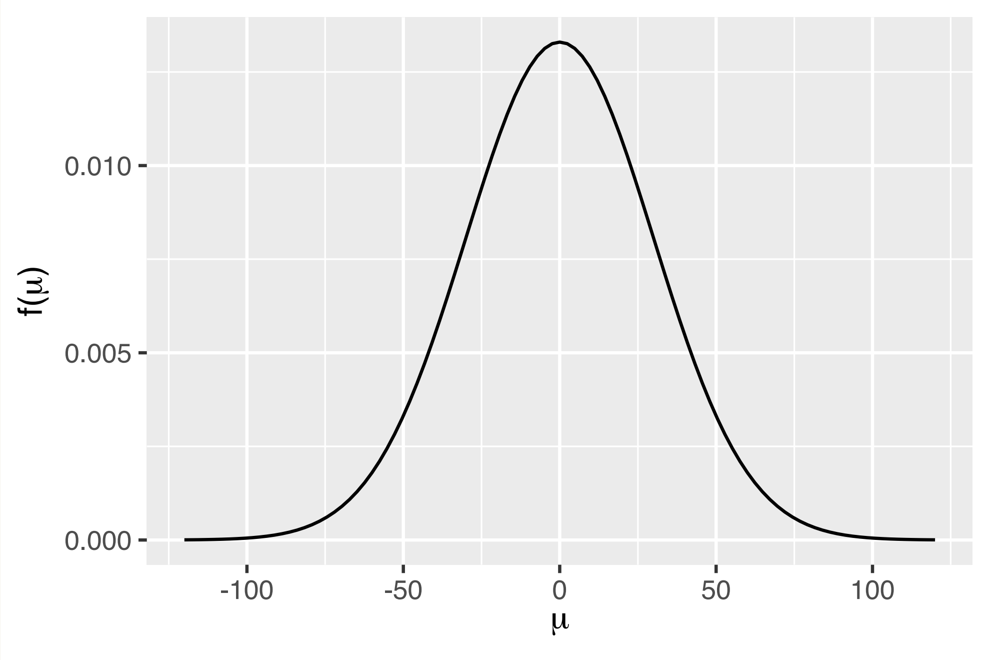
Prior for μ: $\mu \sim \mathcal{N}(\mu_0, \sigma_0)$

Derive the posterior for μ.

03:00

Using a weak prior

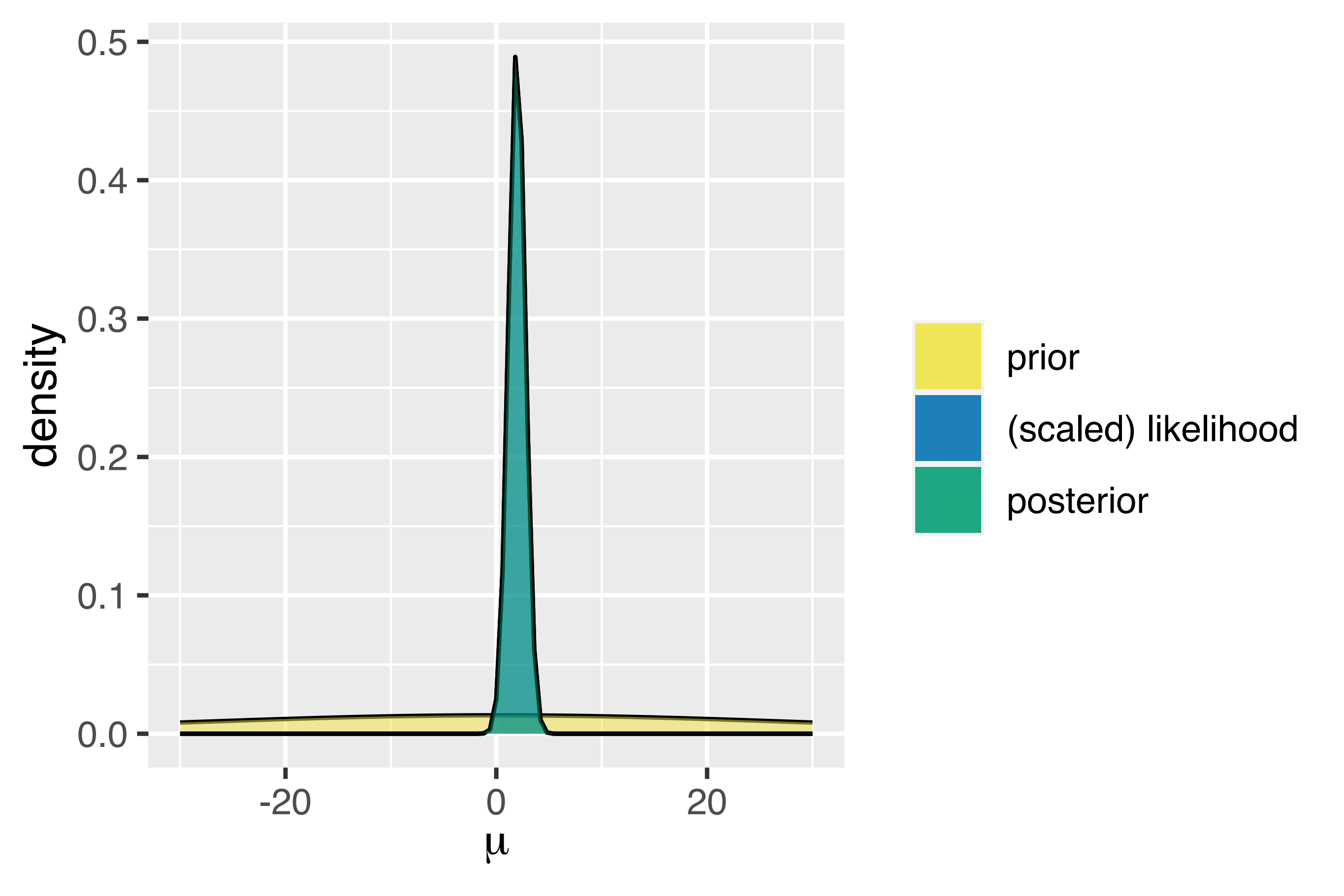
Suppose we thought the election was a toss up, then a weakly informative prior would be reasonable



$$\mu \sim \mathcal{N}(0, 30)$$

- swings to either the left or right possible
- more likely to be closer to 0 (no change) than an extreme swing

Posterior



0.5
0.4
0.3
0.2
0.1
0.0
density
-20
0
20
μ
prior
(scaled) likelihood
posterior

Bayesian hypothesis tests

How do we test $H_0 : \mu \leq 0$ vs. $H_1 : \mu > 0$?

How do we test $H_0 : \mu = 0$ vs. $H_1 : \mu > 0$?

Posterior predictive checks

```r
sigma <- 8   # known sigma
n     <- 100 # sample size
post_mean <- 1.96 # posterior mean
post_sd   <- 0.64 # posterior sd
S <- 1000  #number of sims

# First draw S mus from the posterior
mu_draws <- rnorm(S, post_mean, post_sd)
y_draws  <- matrix(nrow = S, ncol = 100)

# Then, simulate 20 observations for each mu
for(i in 1:S) {
  y_draws[i,] <- rnorm(100, mu_draws[i], sigma)
}
```

Posterior predictive checks

We need to calculate some testing function, $T(\tilde{y})$, on each replicated data set

```
y_bar_sims <- rowMeans(y_draws)
```