# More Bayesian Inference and STAN

## Key Concepts

- Visual Overview of Bayesian Inference
- Writing Stan Code

### Visual Overview of Bayesian Inference

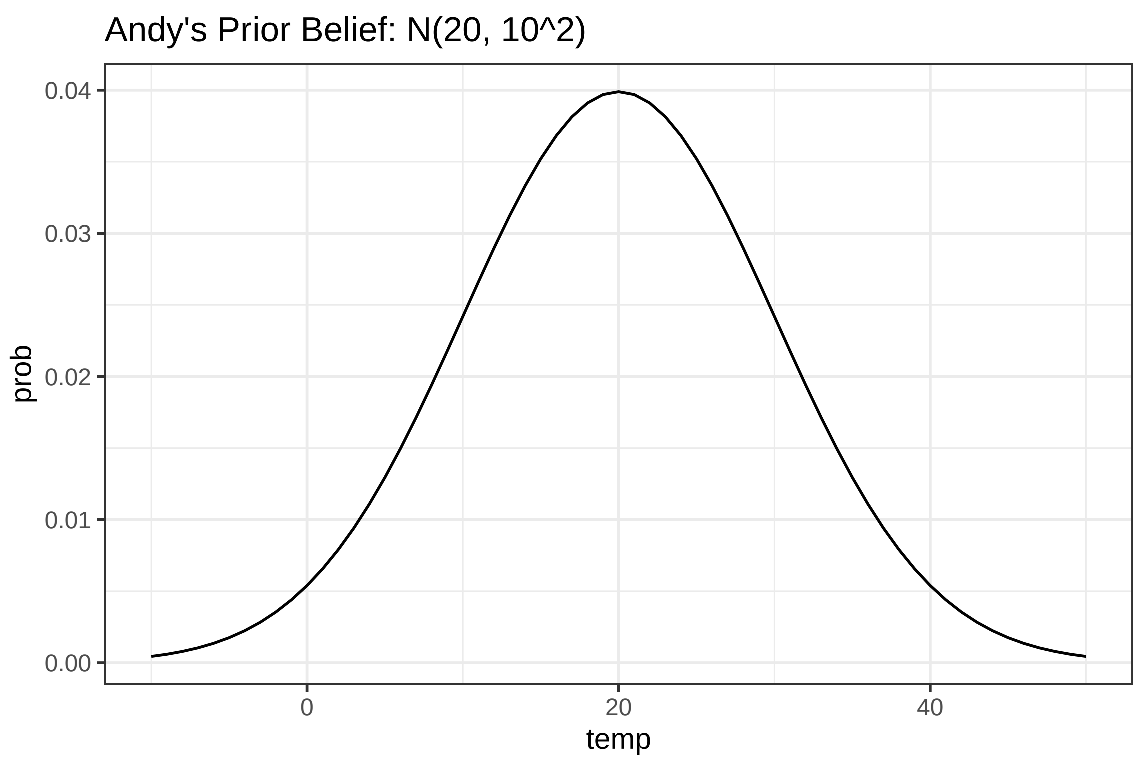
Using some Bridger Bowl weather data we will provide a visual overview of Bayesian Inference. *The goal will be to model the average winter high temperature at the base of Bridger Bowl.*

1. Prior Specification

- First sketch a prior distribution that encapsulates your belief about what you believe the average high temperature would be. *Note this should obey law of total probability*

- *Next we generally need to parameterize (perhaps approximately) this belief with some sort of probability distribution.*

```
temp_seq <- -10:50
prob_seq <- dnorm(temp_seq, mean = 20, sd = 10)
tibble(temp = temp_seq, prob = prob_seq) %>% ggplot(aes(temp, prob)) + geom_line() + theme_bw() +
  ggtitle("Andy's Prior Belief: N(20, 10^2)")
```



*Formally, my prior is on the mean high temp, which we will denote $\mu$.*

$$\mu \sim N(20, 10^2)$$

2. Specify the sampling distribution for the data or perhaps in more familiar language, state the likelihood for the statistical model

- *We will assume that the temperature readings are continuous (or "nearly continuous")*

- *It seems reasonable to start with a normal distribution, so:*

$$X|\mu, \sigma^2 \sim N(\mu, \sigma^2)$$

- Note that we also need to estimate $\sigma$ in this model and need a prior for that parameter too.

- Grab some weather data from Bridger Bowl (roughly the first half of January 2021)

```
temp <- c(26, 45, 44, 36, 22, 25, 31, 31, 37, 34, 35, 37,32, 31)
```

- Any concerns about using this data to inform our research question?

3. Posterior Inference

- Using classical inference, how would you estimate $\mu$.

- *Using maximum likelihood,* $\hat{\mu}_{MLE} = \bar{X}$ *= 33.*

- *With Bayesian inference, our posterior belief is based on the data* ***and*** *our prior belief. Note this can be a blessing or a curse.*

- Formally, we have a distribution for the maximum temperature (a posterior distribution):

$$p(\mu|x) = \int \frac{p(x|\mu, \sigma) \times p(\mu)p(\sigma)}{p(\mu)} d\sigma$$

  , note solving this is not trivial and isn't something we will handle in this class.

- Luckily, there is an elegant computational procedure that will allow us to approximate $p(\mu|x)$ by taking samples from the distribution. *This is, of course, MCMC.*

STAN code for this situation can be written as below. Note that the prior values are hard coded, these could also be passed in as arguments to the model.

```
data {
  int<lower=0> N;
  vector[N] y;
}

parameters {
  real mu;
  real<lower=0> sigma;
}

model {
  y ~ normal(mu, sigma);
  mu ~ normal(20, 10);
}
```

```r
temp_data <- stan("normal.stan", data=list(N = length(temp), y=temp))
```

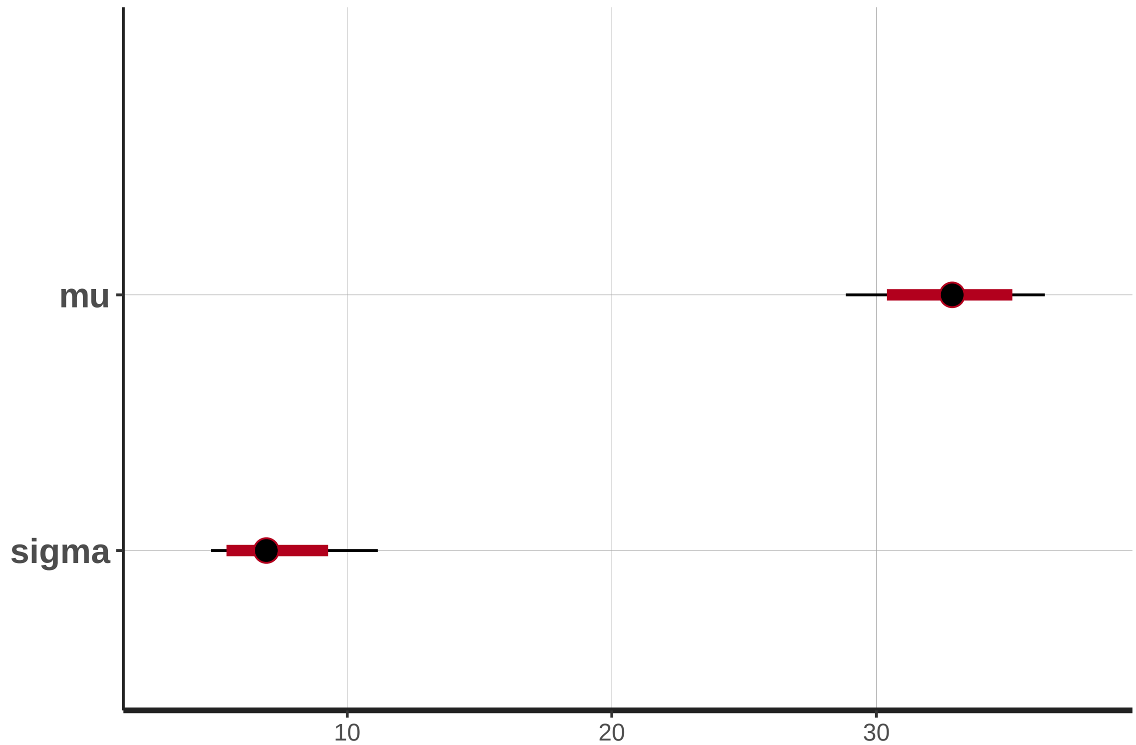
```r
print(temp_data)
```

```
## Inference for Stan model: normal.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##          mean se_mean   sd   2.5%    25%    50%    75%  97.5% n_eff Rhat
## mu      32.81    0.04 1.90  28.85  31.63  32.86  34.06  36.37  2567    1
## sigma    7.22    0.04 1.64   4.85   6.08   6.94   8.07  11.15  2083    1
## lp__   -32.82    0.03 1.04 -35.61 -33.21 -32.50 -32.08 -31.81  1534    1
##
## Samples were drawn using NUTS(diag_e) at Fri Jan 22 11:39:24 2021.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
plot(temp_data)
```

```
## ci_level: 0.8 (80% intervals)
```

```
## outer_level: 0.95 (95% intervals)
```



We can also view the posterior and prior beliefs together on a single figure.

```
tibble(sims = c(extract(temp_data, pars = 'mu')$mu,rnorm(4000, 20, 10)),
       Distribution = rep(c('posterior','prior'), each = 4000)) %>%
  ggplot(aes(x = sims, color = Distribution)) +
  geom_density() + theme_bw() +
  xlab('Temperature (F)') + ylab('') +
  ggtitle("Prior and posterior belief for winter temperature in Bozeman")
```