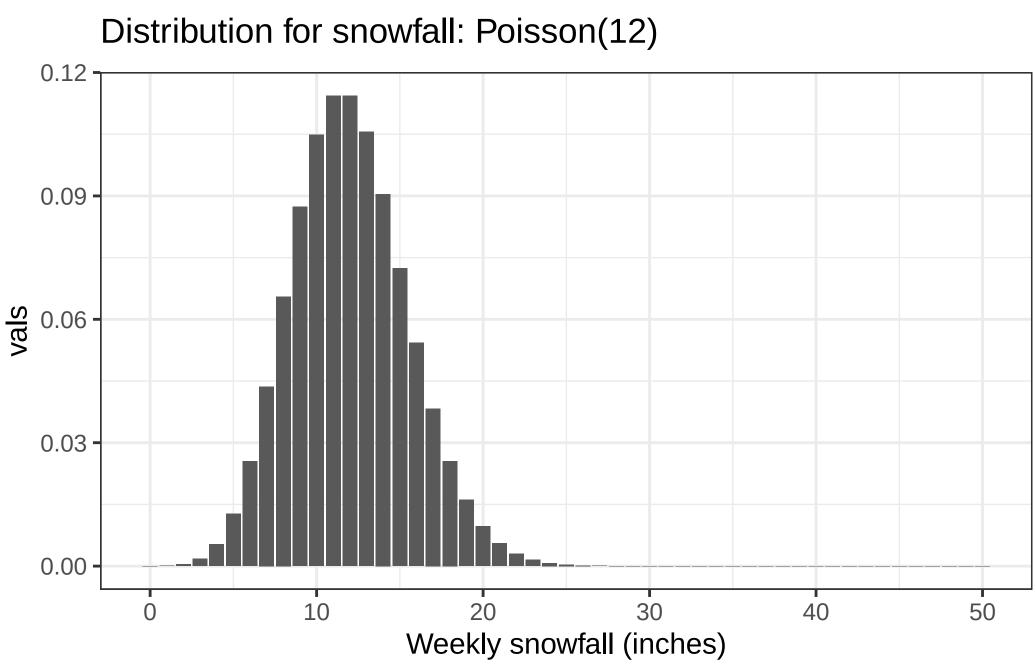
# Lab 2: Key

**Q1.**

Recall the probability mass function for weekly winter snowfall at Bridger Bowl that we saw on Tuesday.

```r
library(tidyverse)
snow_seq <- 0:50
tibble(vals = c(dpois(snow_seq, 12)),
       `Weekly snowfall (inches)` = snow_seq) %>%
  ggplot(aes(x = `Weekly snowfall (inches)`, y = vals)) +
  geom_col() + theme_bw() +
  ggtitle(expression(paste('Distribution for snowfall: Poisson(12)')))
```

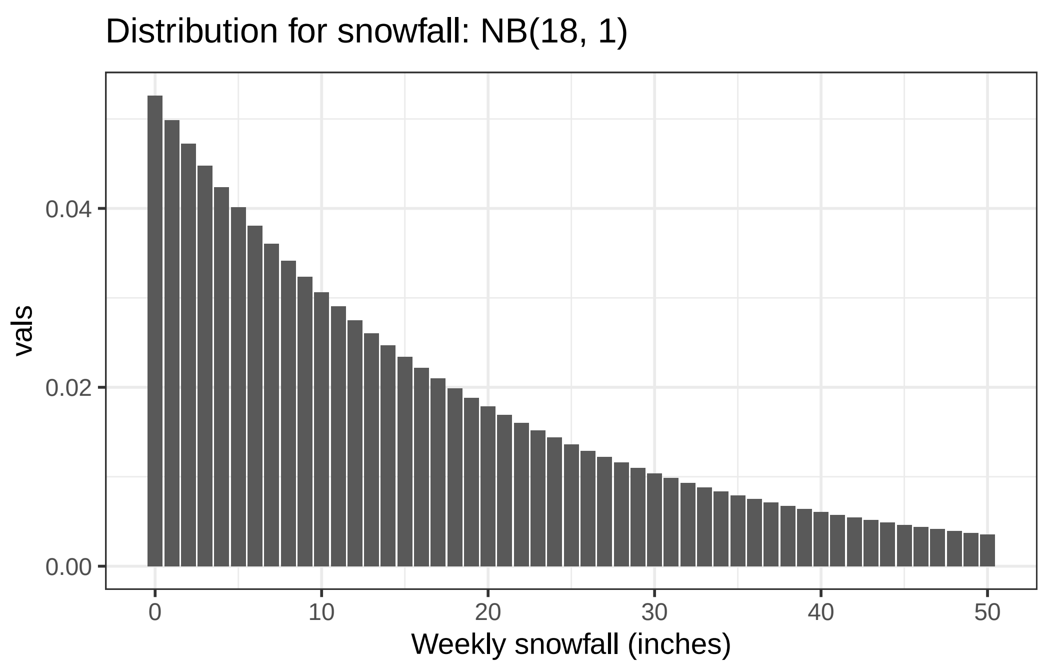


We decided this distribution did a poor job handling larger snowfall events and events near zero.

### a. (4 points)

Now use a negative binomial distribution (`dnbinom()`) to create a better model for weekly snowfall. (Hint: using the parameterization of `dnbinom()` with `mu` and `size` might be more intuitive than default values.) Create a figure to show this distribution.

```
library(tidyverse)
snow_seq <- 0:50
tibble(vals = c(dnbinom(snow_seq, mu = 18, size = 1)),
       `Weekly snowfall (inches)` = snow_seq) %>%
  ggplot(aes(x = `Weekly snowfall (inches)`, y = vals)) +
  geom_col() + theme_bw() +
  ggtitle(expression(paste('Distribution for snowfall: NB(18, 1)')))
```




### b. (4 points)

Describe the parameters in the negative binomial model and defend the choices you made for those parameters.

*With the alternative parameterization, the negative binomial distribution has a parameter related to the mean and a parameter (size) related to the dispersion. I've chosen 18 as the weekly mean. The variance is a function of the size parameter $\frac{\mu+\mu^2}{size}$. For the size parameter, I've chosen 1.*

#### c. (2 points)

According to this distribution, what is the probability of a week having 23 inches of snow?

*The probability of this occurrence is fairly small, specifically it is 0.0151768, but we'd still expect to see these types of values every year or two.*

#### d. (2 points)

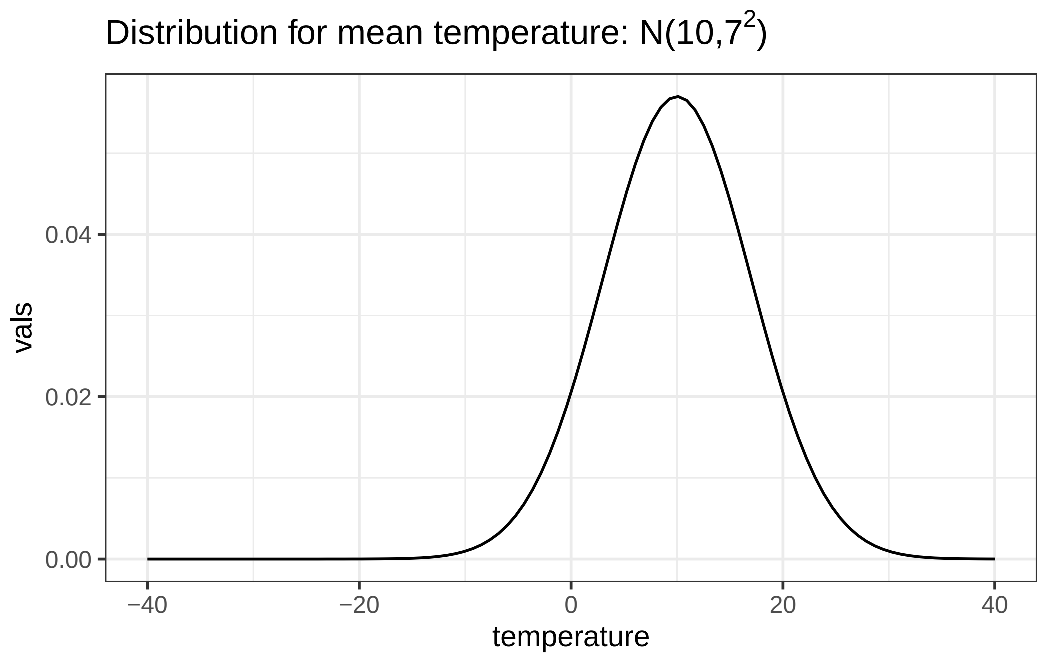
According to this distribution, what is the probability of a week having 0 inches of snow

*The probability of this occurrence is very small, specifically it is 0.0526316.*

### Q2.

Recall, the probability distribution for average temperature in Hyalite that is Normal with mean = 10 and standard deviation = 7.

```
temp_seq <- seq(-40,40, length.out = 100)
tibble(vals = c(dnorm(temp_seq, 10, 7)),
       temperature = temp_seq) %>%
  ggplot(aes(x = temperature, y = vals)) +
  geom_line() + theme_bw() +
  ggtitle(expression(paste('Distribution for mean temperature: N(10,',7^2,')')))
```




Answer the following questions with an numeric answer and a description of why that answer is true.

### a. (2 points)

What would be the lower 95% interval?

*The lower 95% interval would be [$-\infty$, 21.51]*

### b. (2 points)

What would be the upper 95% interval?

*The upper 95% interval would be [-1.51, $\infty$ ]*

### c. (4 points)

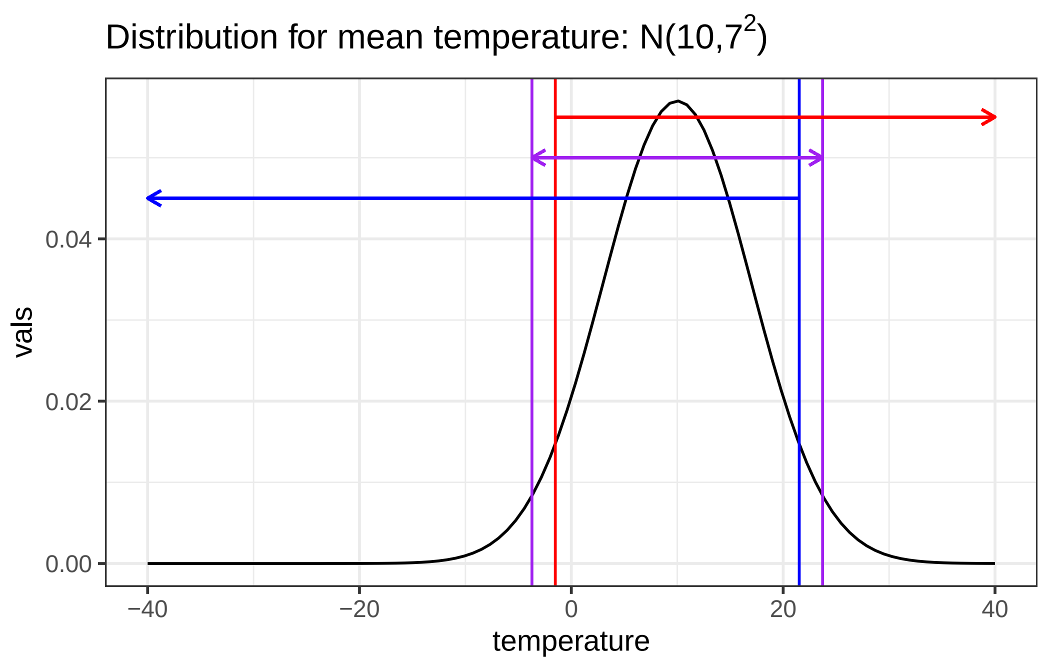
What is the shortest 95% interval, why?

*The shortest 95% interval would be [-3.72, 23.72 ]. Given that this is a symmetric, unimodal distribution the interval is going to be centered at the mean. We then exclude the tails that account for 2.5% of the distributions area.*

**d. (2 points)**

Recreate the plot from above and denote these three intervals on the figure.

```
temp_seq <- seq(-40,40, length.out = 100)
tibble(vals = c(dnorm(temp_seq, 10, 7)),
       temperature = temp_seq) %>%
  ggplot(aes(x = temperature, y = vals)) +
  geom_line() + theme_bw() +
  ggtitle(expression(paste('Distribution for mean temperature: N(10,',7^2,')'))) +
  geom_vline(xintercept = qnorm(.05,10,7), color = 'red' ) +
  geom_vline(xintercept = qnorm(.95,10,7), color = 'blue' ) +
  geom_vline(xintercept = qnorm(.025,10,7), color = 'purple' ) +
  geom_vline(xintercept = qnorm(.975,10,7), color = 'purple' ) +
  geom_segment(x = qnorm(.05,10,7), y = .055, xend = 40, yend = .055,
               arrow = arrow(length = unit(0.03, "npc"), ends = "last"), color = 'red') +
  geom_segment(x = qnorm(.025,10,7), y = .05, xend = qnorm(.975,10,7), yend = .05,
               arrow = arrow(length = unit(0.03, "npc"), ends = "both"), color = 'purple') +
  geom_segment(x = qnorm(.95,10,7), y = .045, xend = -40, yend = .045,
               arrow = arrow(length = unit(0.03, "npc"), ends = "last"), color = 'blue')
```



### Q3.

For the following Beta distributions find the mean of the distribution and the shortest 95 percent interval.

#### a. (2 points)

Beta(1,10)

*The mean of this distribution is* $\frac{1}{11}$*. This distribution is skewed to right, so more an interval omitting most of the mass in the right tail, [0, .259], would be close to the shortest interval.*

#### b. (2 points)

Beta(1,100)

*The mean of this distribution is* $\frac{1}{101}$*. This distribution is skewed to right, so more an interval omitting most of the mass in the right tail, [0, .03], would be close to the shortest interval.*

#### c. (2 points)

Beta(1, 1)

*This is a uniform distribution with mean* $= \frac{1}{2}$*. Any 95% interval will be the shortest interval. So [.025,.975] would be reasonable.*

### Q4. (10 points)

What is the probability that dealer "busts" in black jack. To simplify the calculation (albeit not being entirely correct), you can treat all Aces as being worth 11 points.

Note that the dealer must continue to take cards until the total is 17 or more, at which point the dealer must stand.

```r
num_hands <- 100000
bust <- rep(FALSE, num_hands)
values <- rep(c(2:10,10,10,10,11), each = 4)

for (i in 1:num_hands){
  shuffle <- sample(values)
  num_cards <- 2
  total <- sum(shuffle[1:num_cards])
  while (total < 17){
```

```
    num_cards <- num_cards + 1
    total <- sum(shuffle[1:num_cards])
  }
  if (total > 21) {
    bust[i] = TRUE
  }
}
```

With our modified blackjack game, the dealer busts with probability 0.32

```
cards <- c(rep(c(2:5,7:10,10,10,10,11), each = 4), 6, 6, 6)

bust <- rep(0, 1000)
for (i in 1:1000){
  more_cards <- sum(sample(cards, 2))
  if (more_cards > 15){
    bust[i] <- 1
  }
}
mean(bust)
```

```
[1] 0.459
```