

7)c)

```
require(ggplot2)

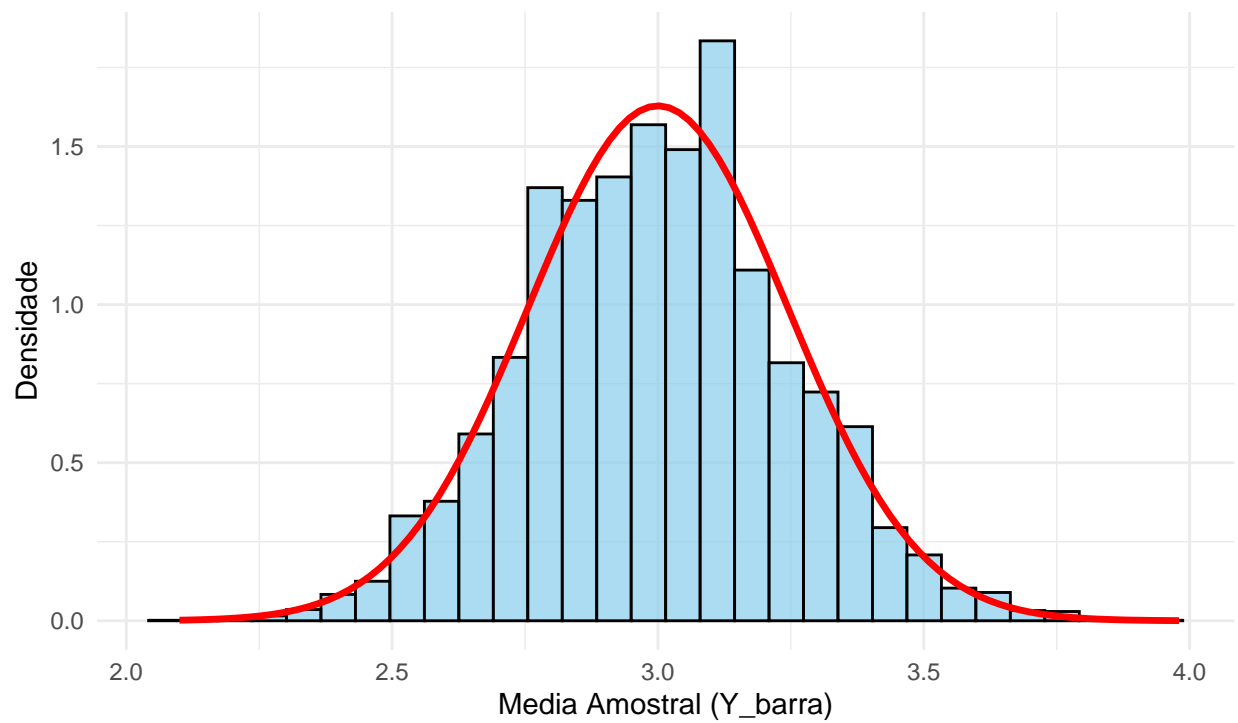
lambda <- 3
n <- 50
num_simulacoes <- 10000

medias_amostrais <- numeric(num_simulacoes)
for (i in 1:num_simulacoes) {
  amostra <- rpois(n, lambda)
  medias_amostrais[i] <- mean(amostra)}

df_medias <- data.frame(media = medias_amostrais)
media_teorica <- lambda
variancia_teorica <- lambda / n
desvio_padrao_teorico <- sqrt(variancia_teorica)

ggplot(df_medias, aes(x = media)) +
  geom_histogram(aes(y = ..density..), bins = 30, fill = "skyblue",
    color = "black", alpha = 0.7) +
  stat_function(fun = dnorm, args = list(mean = media_teorica, sd = desvio_padrao_teorico),
    color = "red", linewidth = 1.2) +
  labs(title = "Distribuicao Empirica das Medias Amostrais vs. Aproximacao Normal (TCL)",
    subtitle = paste("Lambda =", lambda, "| n =", n, "| Simulacoes =", num_simulacoes),
    x = "Media Amostral (Y_barra)", y = "Densidade",
    caption = "A curva vermelha representa a distribuicao Normal teorica.") +
  theme_minimal()
```

Distribuicao Empirica das Medias Amostrais vs. Aproximacao Normal (TCL)
Lambda = 3 | n = 50 | Simulacoes = 10000



A curva vermelha representa a distribuicao Normal teorica.

8)c)

```
require(dplyr)
require(purrr)

mu <- 10
sigma2 <- 4
num_simulacoes <- 10000
n_values <- c(50, 250, 1000)

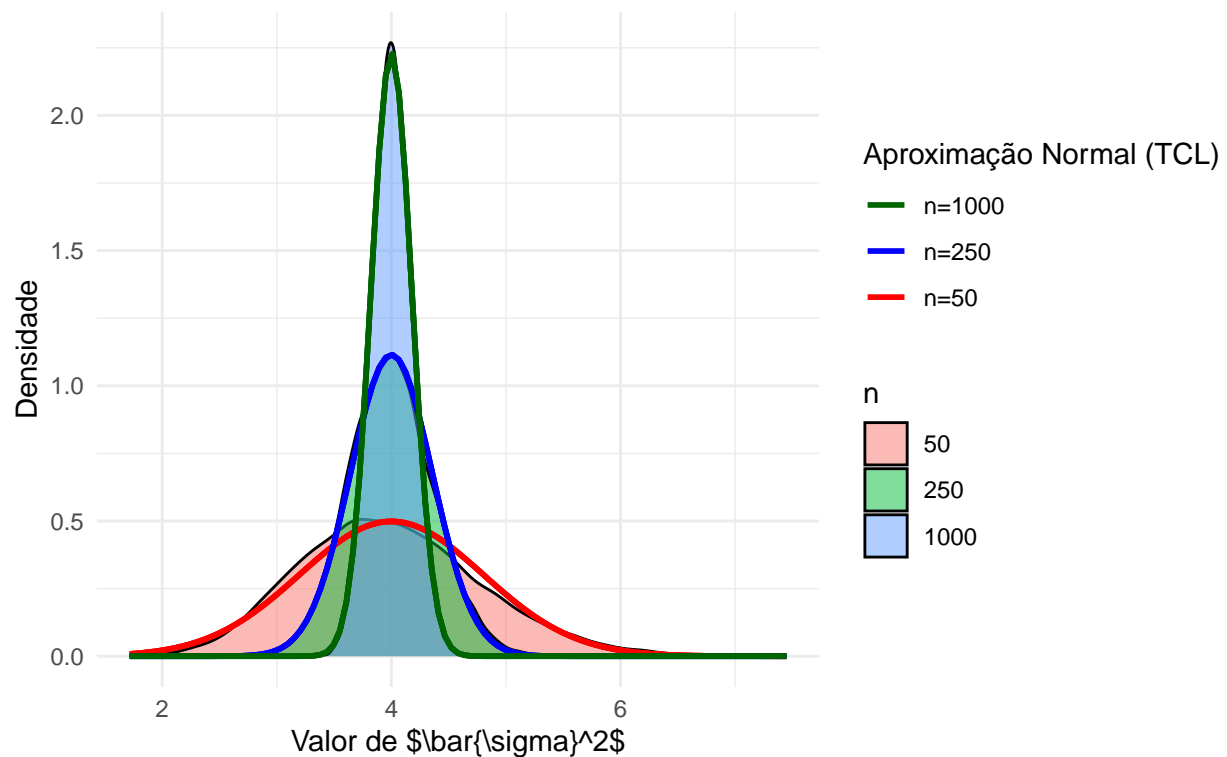
simular_sigma_barra2 <- function(n, mu, sigma2, sims) {
  amostras <- matrix(rnorm(n * sims, mean = mu, sd = sqrt(sigma2)), nrow = sims)
  desvios_sq <- (amostras - mu)^2
  sigma_barra2_valores <- rowMeans(desvios_sq)
  return(data.frame(n = n, sigma_barra_sq = sigma_barra2_valores))}

resultados <- map_df(n_values, ~simular_sigma_barra2(n = .x, mu = mu, sigma2 = sigma2,
                                                    sims = num_simulacoes))

resultados$n <- as.factor(resultados$n)

ggplot(resultados, aes(x = sigma_barra_sq, fill = n)) +
  geom_density(alpha = 0.5) +
  stat_function(fun = dnorm, args = list(mean = sigma2, sd = sqrt(2*sigma2^2 / 50)),
               aes(color = "n=50"), linewidth = 1) +
  stat_function(fun = dnorm, args = list(mean = sigma2, sd = sqrt(2*sigma2^2 / 250)),
               aes(color = "n=250"), linewidth = 1) +
  stat_function(fun = dnorm, args = list(mean = sigma2, sd = sqrt(2*sigma2^2 / 1000)),
               aes(color = "n=1000"), linewidth = 1) +
  scale_color_manual(name = "Aproximação Normal (TCL)",
                    values = c("n=50" = "red", "n=250" = "blue", "n=1000" = "darkgreen")) +
  labs(title = "Distribuição de  $\bar{\sigma}^2$  conforme n cresce",
       subtitle = "As curvas representam a aproximação teórica pelo TCL",
       x = "Valor de  $\bar{\sigma}^2$ ", y = "Densidade") + theme_minimal()
```

Distribuição de $\bar{\sigma}^2$ conforme n cresce
 As curvas representam a aproximação teórica pelo TCL



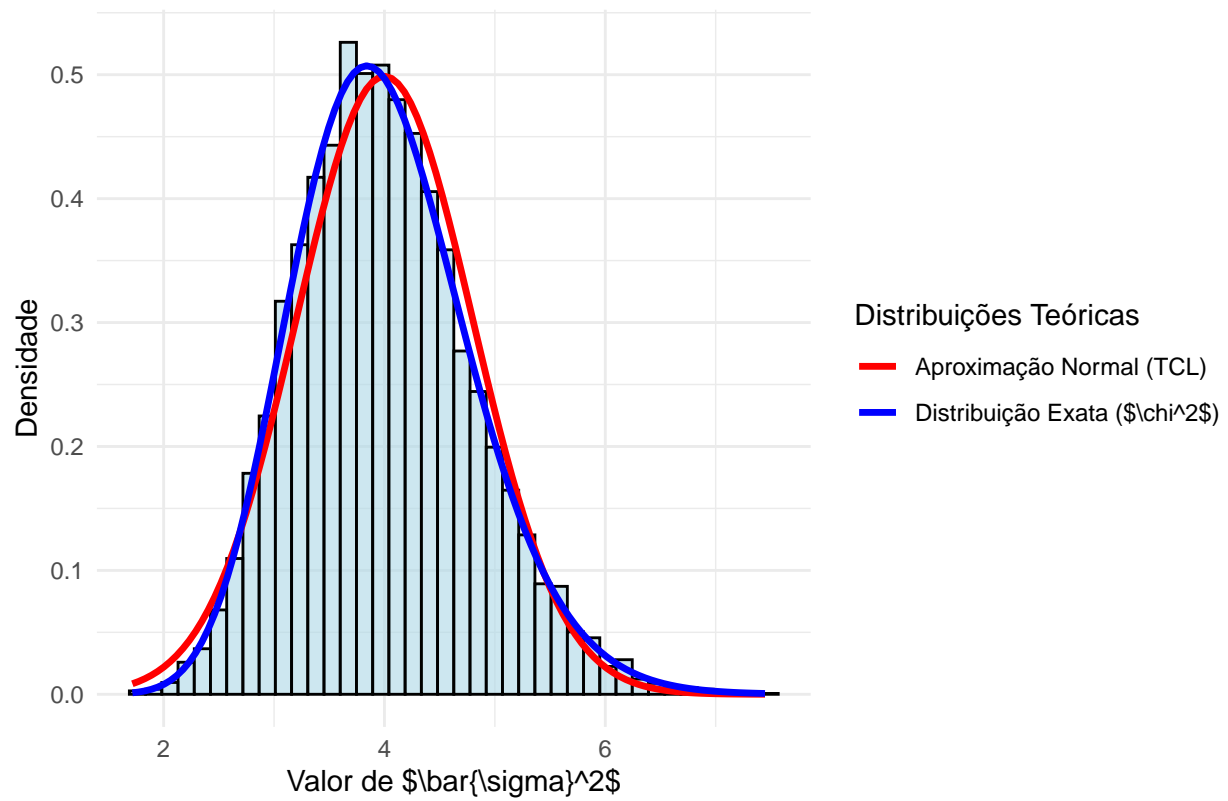
8)d)

```
resultados_n50 <- resultados %>% filter(n == 50)

dscaled_chisq <- function(x, n, sigma2) {
  dchisq(x * n / sigma2, df = n) * (n / sigma2)}

ggplot(resultados_n50, aes(x = sigma_barra_sq)) +
  geom_histogram(aes(y = ..density..), bins = 40, fill = "lightblue", color = "black", alpha = 0.6) +
  stat_function(fun = dnorm, args = list(mean = sigma2, sd = sqrt(2*sigma2^2 / 50)),
    aes(color = "Aproximação Normal (TCL)", linewidth = 1.2) +
  stat_function(fun = dscaled_chisq, args = list(n = 50, sigma2 = sigma2),
    aes(color = "Distribuição Exata ( $\chi^2$ )", linewidth = 1.2) +
  scale_color_manual(name = "Distribuições Teóricas",
    values = c("Aproximação Normal (TCL)" = "red",
      "Distribuição Exata ( $\chi^2$ )" = "blue")) +
  labs(title = "Comparação das Distribuições Empírica, Aproximada e Exata (n=50)",
    x = "Valor de  $\bar{\sigma}^2$ ", y = "Densidade") + theme_minimal()
```

Comparação das Distribuições Empírica, Aproximada e Exata (n=50)



9)b)

```
rate <- 0.5
mu_verdadeiro <- 1 / rate
num_simulacoes <- 10000
n_values <- c(50, 250, 1000)

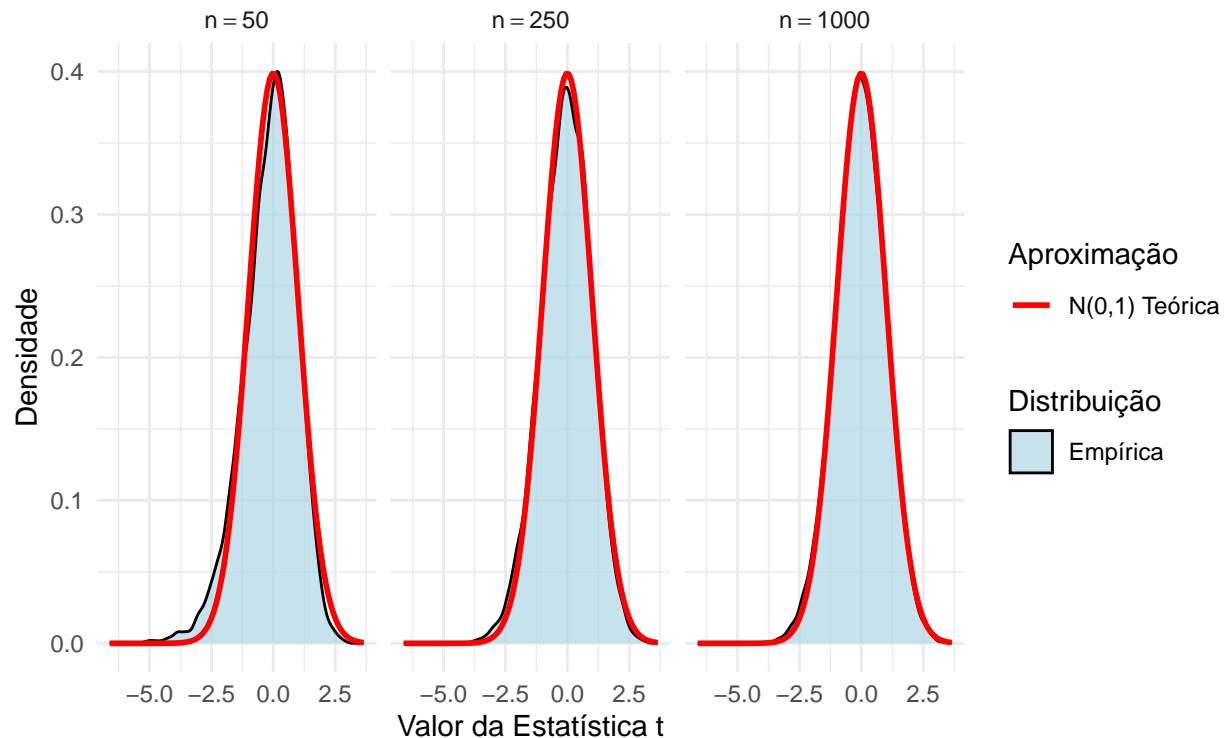
simular_t_stat <- function(n, rate, mu, sims) {
  amostras <- matrix(rexp(n * sims, rate = rate), nrow = sims)
  medias_amostrais <- rowMeans(amostras)
  sds_amostrais <- apply(amostras, 1, sd)
  t_stats <- (medias_amostrais - mu) / (sds_amostrais / sqrt(n))
  return(data.frame(n = n, t_statistic = t_stats))}

resultados_b <- map_df(n_values, ~simular_t_stat(n = .x, rate = rate,
                                                mu = mu_verdadeiro, sims = num_simulacoes))
resultados_b$n <- factor(resultados_b$n, levels = c(50, 250, 1000))

ggplot(resultados_b, aes(x = t_statistic)) +
  geom_density(aes(fill = "Empírica"), alpha = 0.7) +
  stat_function(fun = dnorm, args = list(mean = 0, sd = 1),
               aes(color = "N(0,1) Teórica"), linewidth = 1) +
  facet_wrap(~n, labeller = label_bquote(n == .(as.character(n)))) +
  scale_fill_manual(name = "Distribuição", values = c("Empírica" = "lightblue")) +
  scale_color_manual(name = "Aproximação", values = c("N(0,1) Teórica" = "red")) +
  labs(title = "Convergência da Estatística t para a Normal Padrão",
       subtitle = "A distribuição empírica se aproxima da curva N(0,1) conforme n aumenta",
       x = "Valor da Estatística t", y = "Densidade") + theme_minimal()
```

Convergência da Estatística t para a Normal Padrão

A distribuição empírica se aproxima da curva $N(0,1)$ conforme n aumenta



9)c)

```
n_exato <- 50
mu_normal <- 10
sigma_normal <- 2

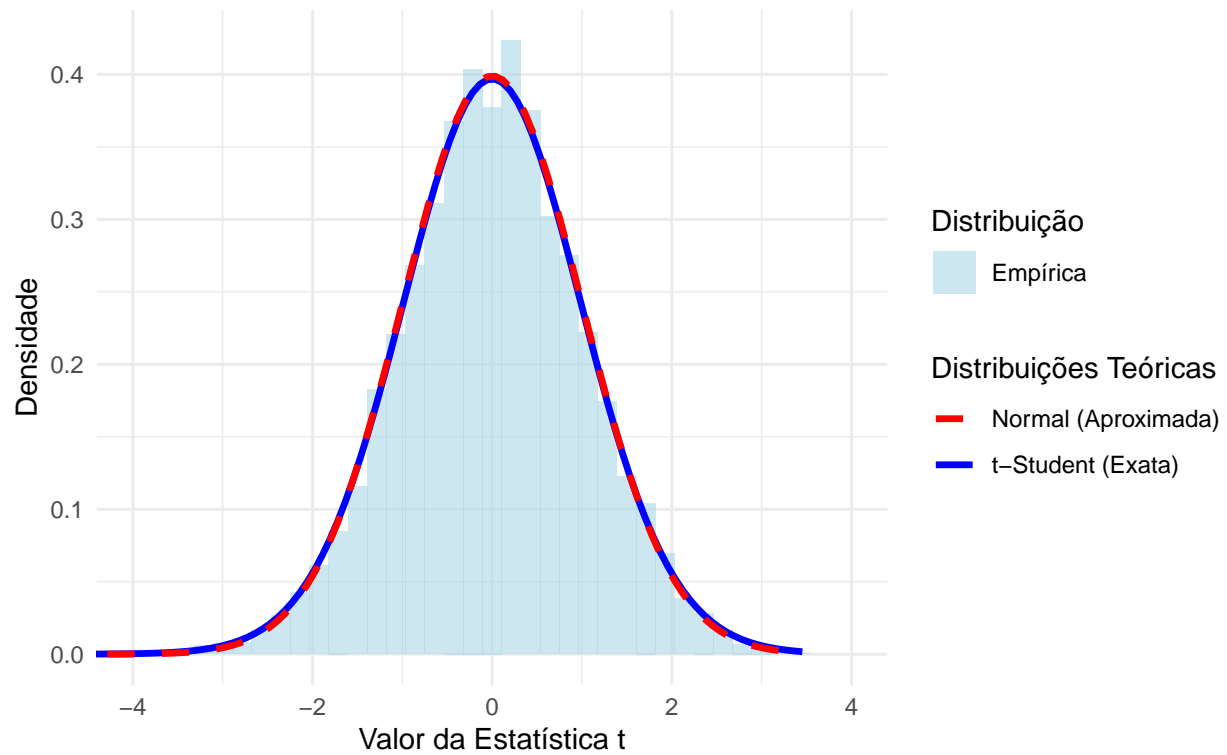
amostras_normais <- matrix(rnorm(n_exato * num_simulacoes, mean = mu_normal,
                                sd = sigma_normal), nrow = num_simulacoes)
medias_normais <- rowMeans(amostras_normais)
sds_normais <- apply(amostras_normais, 1, sd)
t_stats_normais <- (medias_normais - mu_normal) / (sds_normais / sqrt(n_exato))
resultados_c <- data.frame(t_statistic = t_stats_normais)

ggplot(resultados_c, aes(x = t_statistic)) +
  geom_histogram(aes(y = ..density.., fill = "Empírica"), bins = 40, alpha = 0.6) +
  stat_function(fun = dt, args = list(df = n_exato - 1),
               aes(color = "t-Student (Exata)", linewidth = 1.2)) +
  stat_function(fun = dnorm, args = list(mean = 0, sd = 1),
               aes(color = "Normal (Aproximada)", linewidth = 1.2, linetype = "dashed")) +
  scale_fill_manual(name = "Distribuição",
                   values = c("Empírica" = "lightblue")) +
  scale_color_manual(name = "Distribuições Teóricas",
                    values = c("t-Student (Exata)" = "blue", "Normal (Aproximada)" = "red")) +
  labs(title = paste("Comparação das Distribuições da Estatística t (n=", n_exato, ")", sep=""),
       subtitle = "População Original: Normal",
```

```
x = "Valor da Estatística t", y = "Densidade") +  
coord_cartesian(xlim = c(-4, 4)) + theme_minimal()
```

Comparação das Distribuições da Estatística t (n=50)

População Original: Normal



10)b)

```
n <- 100
m <- 120
num_simulacoes <- 20000

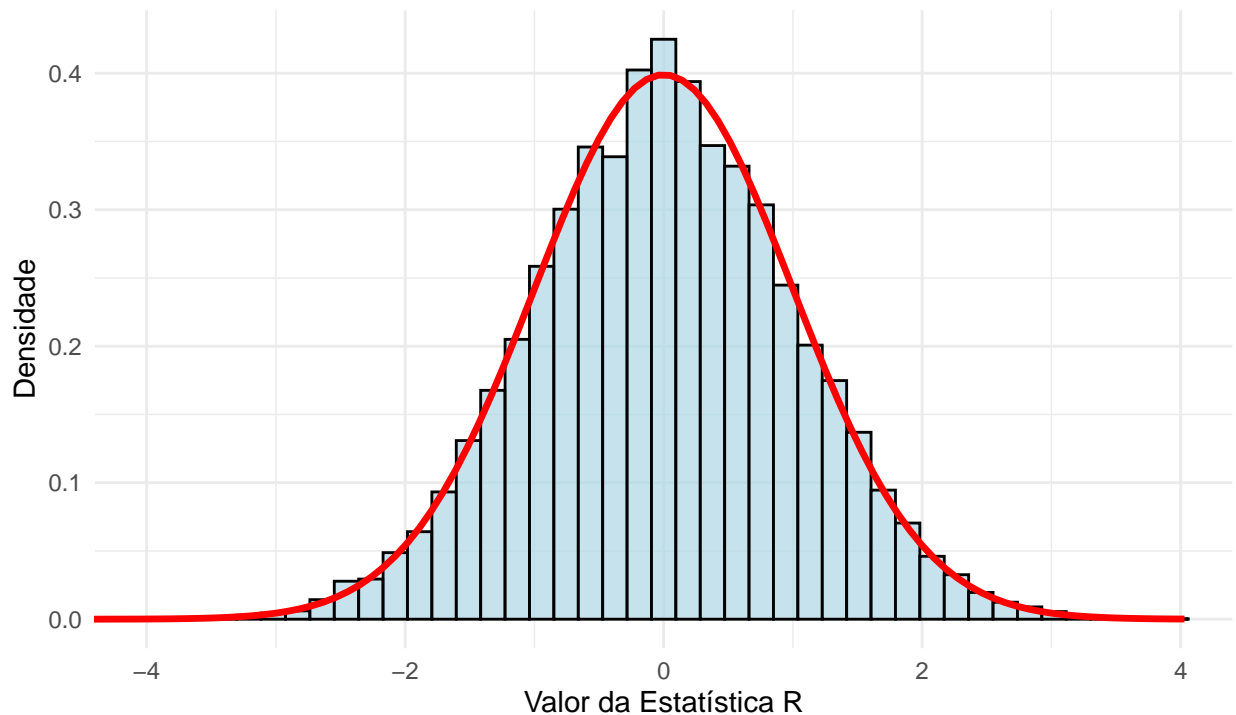
amostras_Xn <- rpois(num_simulacoes, lambda = n)
amostras_Ym <- rpois(num_simulacoes, lambda = m)

R_valores <- ((amostras_Xn - n) - (amostras_Ym - m)) / sqrt(amostras_Xn + amostras_Ym)
df_R <- data.frame(R = R_valores)

ggplot(df_R, aes(x = R)) +
  geom_histogram(aes(y = ..density..), bins = 50, fill = "lightblue",
    color = "black", alpha = 0.7) +
  stat_function(fun = dnorm, args = list(mean = 0, sd = 1),
    color = "red", linewidth = 1.2) +
  labs(title = "Distribuição Empírica da Estatística R vs. Normal Padrão",
    subtitle = paste("n =", n, "| m =", m, "| Simulações =", num_simulacoes),
    x = "Valor da Estatística R", y = "Densidade",
    caption = "A curva vermelha representa a distribuição N(0,1) teórica.") +
  coord_cartesian(xlim = c(-4, 4)) + theme_minimal()
```

Distribuição Empírica da Estatística R vs. Normal Padrão

n = 100 | m = 120 | Simulações = 20000



A curva vermelha representa a distribuição N(0,1) teórica.