Group31 ETC5242 Assignment 1

Christy Lai 29252229 MA Sun Yan Joanna 33775877
Jiwen Zhou 33529086 Sarah Liu 34500006

This report conducts statistic analysis of bank customers, based on a database including data of 200 customers' annual gross income and education level.

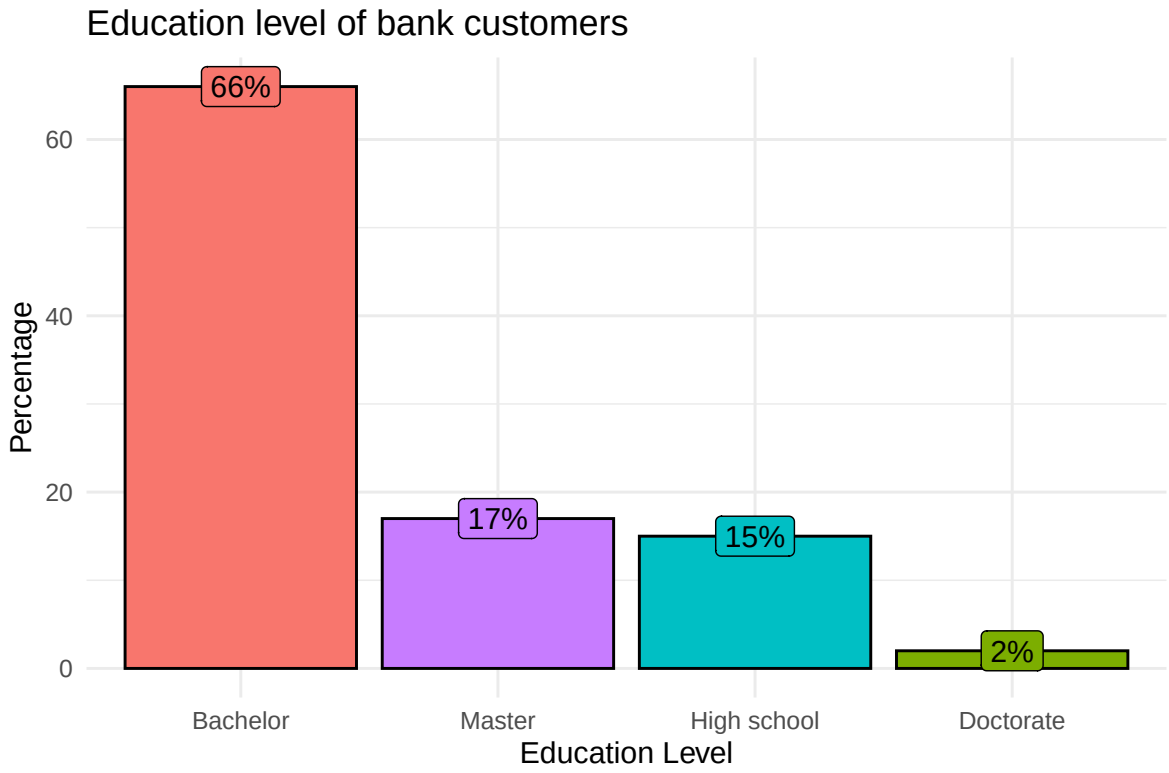
Task 1

First part would be some distribution plots and descriptive statistics of the data, and fitted distribution model.

Load data and libraries we needed:

The first plot is about the education level of bank customers.

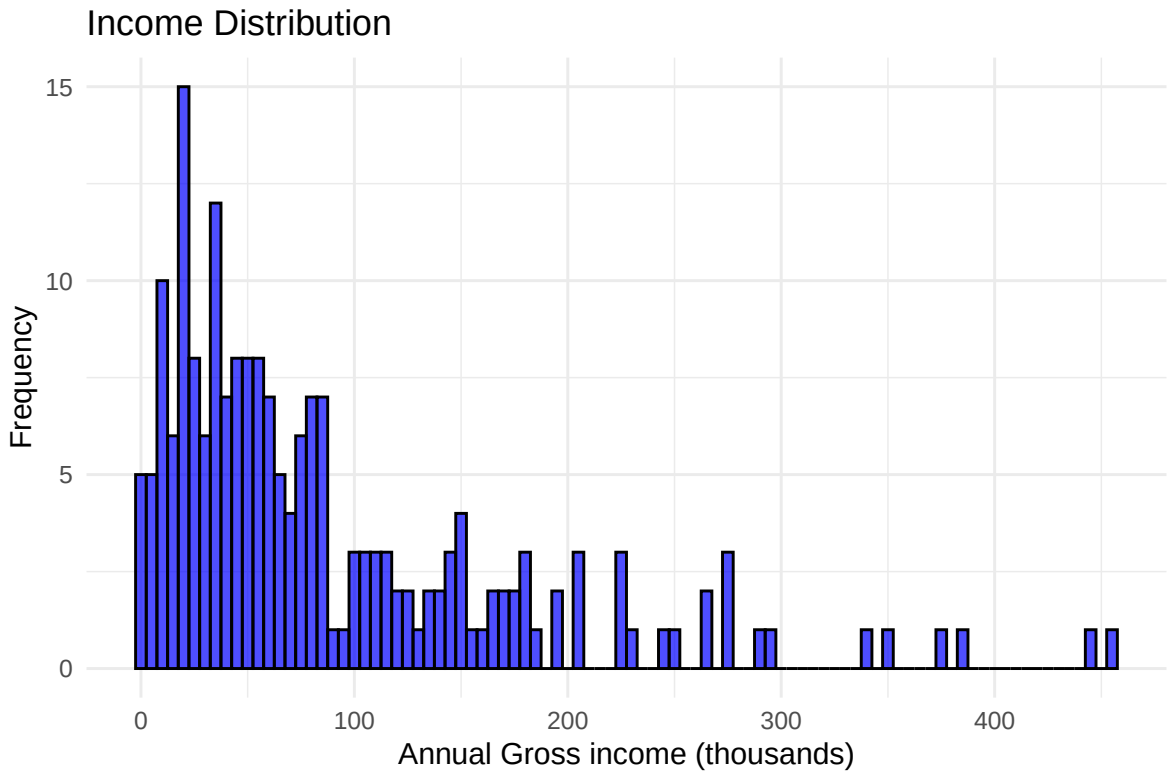
```
# Descriptive bar plot
bank_data %>%
  group_by(education) %>%
  summarise(perc = n()/nrow(bank_data) * 100) %>%
  as.data.frame() %>%
  ggplot(aes(x = fct_reorder(education, -perc),
             y = perc,
             fill = education)) +
  geom_col(color = "black") +
  theme_minimal() +
  labs(x = "Education Level",
       y = "Percentage",
       title = "Education level of bank customers") +
  geom_label(aes(label = paste0(perc, "%"),
                 vjust = 0.5)) +
  guides(fill = "none")
```



The bar plot shows that most of the surveyed bank customers achieved a highest education level of bachelor's degree (66%), followed by master's and high school with similar percentage, and 2% hold a doctorate degree.

The second part is visualizing the distribution of customer incomes with a histogram, which gives us insight into how incomes are spread across our customer sample.

```
# Histogram of incomes
ggplot(bank_data, aes(x = income)) +
  geom_histogram(binwidth = 5, fill = "blue", color = "black", alpha = 0.7) +
  labs(title = "Income Distribution", x = "Annual Gross income (thousands)", y = "Frequency")
  theme_minimal()
```



The histogram reveals a right-skewed distribution, with most customers having incomes clustered in the lower range and a tail extending toward higher income levels.

We calculate the summary statistics to get more detailed insights into the distribution of incomes.

```
# Descriptive statistics
summary_stats <- bank_data %>%
  summarize(
    Mean = round(mean(income), 1),
    Median = median(income),
    SD = round(sd(income), 1),
    Min = min(income),
    Max = max(income)
  )

kable(summary_stats,
      caption = "Statistics of annual gross income of bank customers (thousands)")
```

Table 1: Statistics of annual gross income of bank customers (thousands)

Mean	Median	SD	Min	Max
89.2	58.7	87.6	0.1	453.9

The average income is $89,200, but the median income of $58,700 provides a more accurate representation of the typical customer due to the skewed distribution. The standard deviation of $87,600 reflects a significant variation in incomes.

The next part will be exploring fitting models, including normal, exponential and gamma distribution.

```
# Fit normal distribution
fit_norm <- fitdist(bank_data$income, "norm")
# Fit exponential distribution
fit_exp <- fitdist(bank_data$income, "exp")
# Fit gamma distribution
fit_gamma <- fitdist(bank_data$income, "gamma")

# Normal distribution summary
tibble(Parameter = c("Mean", "Standard Deviation"),
       Estimate = fit_norm$estimate,
       'Standard Error' = fit_norm$sd) %>%
  kable(caption = "Maximum likelihood estimates of fitted normal distribution model")
```

Table 2: Maximum likelihood estimates of fitted normal distribution model

Parameter	Estimate	Standard Error
Mean	89.2250	6.176770
Standard Deviation	87.3526	4.367631

```
# Exponential distribution summary
tibble(Parameter = "Rate",
       Estimate = fit_exp$estimate,
       'Standard Error' = fit_exp$sd) %>%
  kable(caption = "Maximum likelihood estimates of fitted exponential distribution model")
```

Table 3: Maximum likelihood estimates of fitted exponential distribution model

Parameter	Estimate	Standard Error
Rate	0.0112076	0.0007861

```
# Gamma distribution summary
tibble(Parameter = c("Shape", "Rate"),
       Estimate = fit_gamma$estimate,
       'Standard Error' = fit_gamma$sd) %>%
  kable(caption = "Maximum likelihood estimates of fitted gamma distribution model")
```

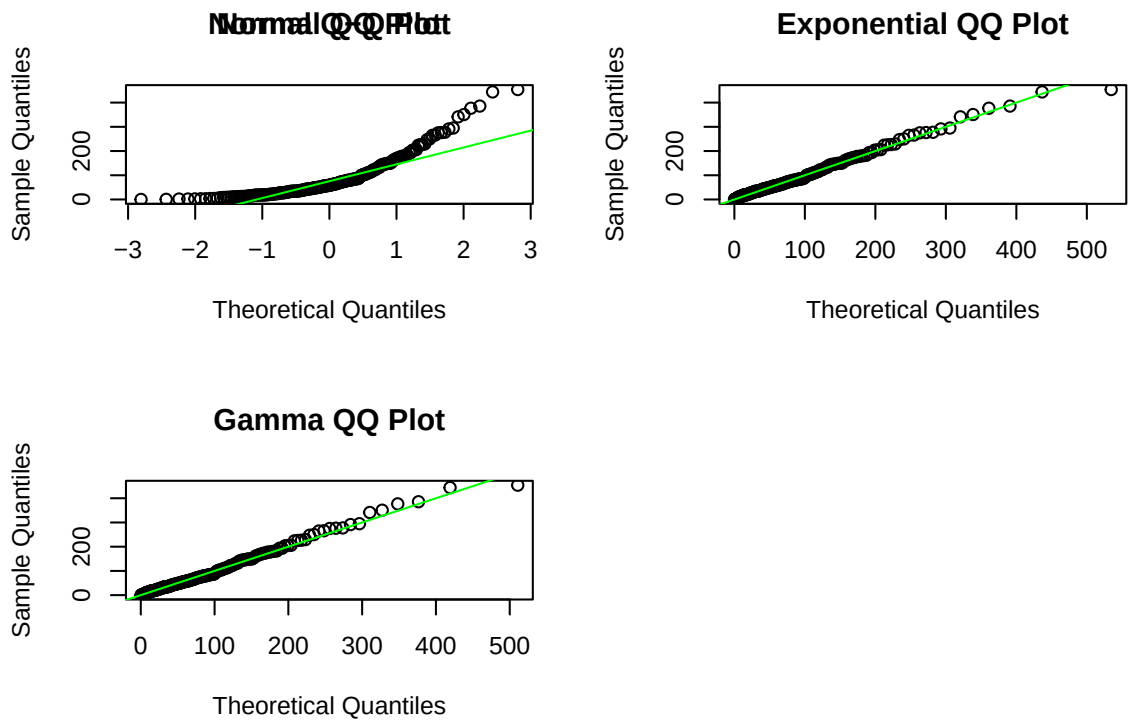
Table 4: Maximum likelihood estimates of fitted gamma distribution model

Parameter	Estimate	Standard Error
Shape	1.083319	0.0953152
Rate	0.012132	0.0013355

Above tables showed the maximum likelihood estimates of the 3 fitted models.

```
# QQ plots for the normal, exponential, and gamma distributions
par(mfrow = c(2, 2)) # Set up plot window for multiple plots

# QQ plot for normal distribution
qqnorm(bank_data$income)
qqline(bank_data$income,col = "green")
title("Normal QQ Plot")

# QQ plot for exponential distribution
qqplot(qexp(ppoints(length(bank_data$income)),
            rate = fit_exp$estimate),
       sort(bank_data$income),
       xlab = "Theoretical Quantiles",
       ylab = "Sample Quantiles")
abline(0, 1,col = "green")
title("Exponential QQ Plot")

# QQ plot for gamma distribution
qqplot(qgamma(ppoints(length(bank_data$income)),
              shape = fit_gamma$estimate[1],
              rate = fit_gamma$estimate[2]),
```

```
       sort(bank_data$income),
       xlab = "Theoretical Quantiles",
       ylab = "Sample Quantiles")
abline(0, 1,col = "green")
title("Gamma QQ Plot")
```



Seems like both exponential and gamma distribution models fit the data better than normal distribution. In particular, exponential distribution is slightly better than gamma.

Task 2

In this section, it will focus on estimating the 80th percentile of annual income of customers. In addition, 95% confidence intervals based on each of the estimators will be calculated. In task1, we have already fitted three distributions, which are normal, exponential and gamma distribution. Therefore, in this analysis, we use these three distributions to estimate the 80th percentile.

```r
# Estimate the 80th percentile using estimators based on fitted model
q80_normal <- qnorm(0.80, mean = fit_norm$estimate[1], sd = fit_norm$estimate[2])
q80_exponential <- qexp(0.80, rate=fit_exp$estimate)
q80_gamma <- qgamma(0.80, shape=fit_gamma$estimate[1], rate=fit_gamma$estimate[2])

# Create a tibble based on these results
results <- tibble(
  Distribution = c("Normal", "Exponential", "Gamma"),
  `80th Percentile Estimate` = c(q80_normal, q80_exponential, q80_gamma))


kable(results, caption = "80th Percentile Estimates based on Different Models")
```

Table 5: 80th Percentile Estimates based on Different Models

Distribution	80th Percentile Estimate
Normal	162.7428
Exponential	143.6021
Gamma	142.7391

The parameter of interest is the 80th percentile of the annual income distribution. It indicates the income level at which 80% of the customers earn less than this amount. Through Table5, we can see that the normal model has the highest 80th percentile estimate, while the exponential and gamma estimates are lower. However, annual income shows a right-skewed distribution, which indicates that most people are at a lower income level, and few of them has a high salary. The normal distribution assumes the data is normally distributed, therefore, it will make the 80th percentile larger, which is not true. So, exponential and gamma estimates seem better.

Although we can estimate the 80th percentile of annual income based on a particular model, it is also possible to use a statistic called 'sample quantile', which is an estimator that does not assume any specific model.

```r
# Using sample quantile to estimate
q80_sample <- quantile(bank_data$income, probs = 0.80)

new_results <- tibble(
  Distribution = c("Normal", "Exponential", "Gamma","Sample"),
  `80th Percentile Estimate` = c(q80_normal, q80_exponential, q80_gamma,q80_sample))


kable(new_results, caption = "80th Percentile Estimates based on Different Models")
```

Table 6: 80th Percentile Estimates based on Different Models

Distribution	80th Percentile Estimate
Normal	162.7428
Exponential	143.6021
Gamma	142.7391
Sample	146.5600

After that, we can calculate the 95% confidence intervals for the estimates. We can first calculate the stand error for each distribution, and then use the formula $CI = \hat{\theta} \pm Z_{\alpha/2} \times SE(\hat{\theta})$, here, $\hat{\theta}$ refers to the 80th percentile of each distribution, $Z_{\alpha/2}$ equals to '1.96', and the $SE(\hat{\theta})$ is the stand error which is calculated.

```
# Calculate the 95% CI for the estimates

# Normal distribution 95% CI
se_normal <- fit_norm$estimate[2] / sqrt(length(bank_data$income))
ci_normal <- q80_normal + c(-1.96, 1.96) * se_normal

# Exponential distribution 95% CI
se_exp <- sqrt(1 / (length(bank_data$income) * fit_exp$estimate^2))
ci_exp <- q80_exponential + c(-1.96, 1.96) * se_exp

# Gamma distribution 95% CI
se_gamma <- sqrt(fit_gamma$estimate[1] / (fit_gamma$estimate[2]^2 * length(bank_data$income))
ci_gamma <- q80_gamma + c(-1.96, 1.96) * se_gamma

ci_results <- tibble(
  Distribution = c("Normal", "Exponential", "Gamma"),
  `80th Percentile Estimate` = c(q80_normal, q80_exponential, q80_gamma),
  `Lower Bound (95% CI)` = c(ci_normal[1], ci_exp[1], ci_gamma[1]),
  `Upper Bound (95% CI)` = c(ci_normal[2], ci_exp[2], ci_gamma[2])
)

kable(ci_results,
      caption = "80th Percentile Estimates and 95% Confidence Intervals for Different Distril
```

Table 7: 80th Percentile Estimates and 95% Confidence Intervals for Different Distributions

Distribution	80th Percentile Estimate	Lower Bound (95% CI)	Upper Bound (95% CI)
Normal	162.7428	150.6364	174.8493
Exponential	143.6021	131.2361	155.9681
Gamma	142.7391	130.8490	154.6293

In conclusion, the key statistical idea is that we use three different models, normal, exponential and gamma to estimate the 80th percentile, along with the sample based estimate for comparison. Based on the results, the 80th percentile of the normal model(162.7428) is significantly higher than the sample estimate(146.5600). However, the exponential and gamma estimates are similar and close to the sample estimate. Therefore, they are more useful and reliable than the normal estimate.

Task 3

Simulation Process

The simulation is generated in the below steps:

1. To begin with, a few numbers should be clarified. As the mean = 1 / lambda, given mean = $100,000, lambda would equal to 1 / 100,000. We chose to set up the sample size to be 100, and bootstrapping for 1000 times. The assumption is that the annual income follows an exponential distribution, hence the original data are simulated on the basis of exponential function.

```
set.seed(2420)

true_mean <- 100
lambda <- 1 / true_mean
n_sample <- 100
n_bootstrap <- 1000

# Generate simulated data following an exponential distribution
original_data <- rexp(n_sample, rate = lambda)
```

2. We aim to simulate each of the estimators aforementioned, that are Normal, Gamma and Exponential to estimate the 80th quantile of the annual income.

```r
# Define a function to calculate the 80th percentile
estimate_quantiles <- function(data, indices) {
  sample_data <- data[indices]  # Bootstrap

   # Fit normal, exponential, and gamma distributions
  normal_fit <- fitdist(sample_data, "norm")
  exp_fit <- fitdist(sample_data, "exp")
  gamma_fit <- fitdist(sample_data, "gamma")

  # Return to 80th percentile
  return(
  c(
    qnorm(0.80, mean = normal_fit$estimate[1], sd = normal_fit$estimate[2]),
    qexp(0.80, rate = exp_fit$estimate),
    qgamma(0.80, shape = gamma_fit$estimate[1], rate = gamma_fit$estimate[2])
  ))
}
```

3. We draw 1000 samples from the original data based on the estimates calculated about the 80th quantile of income.

Using bootstrap re-sampling, we draw 1000 samples from the original data to calculate estimates of the 80th percentile based on normal, exponential, and gamma distributions. The true

The main goal is to compare the 80th percentile (P80) estimates from each model using bootstrap resampling, a method that allows us to evaluate the variability of our estimates by repeatedly sampling from the data.

```r
# Bootstrap sampling
bootstrap_results <- boot(data = original_data, statistic = estimate_quantiles,
                          R = n_bootstrap)

# Extract results
normal_estimates <- bootstrap_results$t[, 1]
exp_estimates <- bootstrap_results$t[, 2]
gamma_estimates <- bootstrap_results$t[, 3]
```

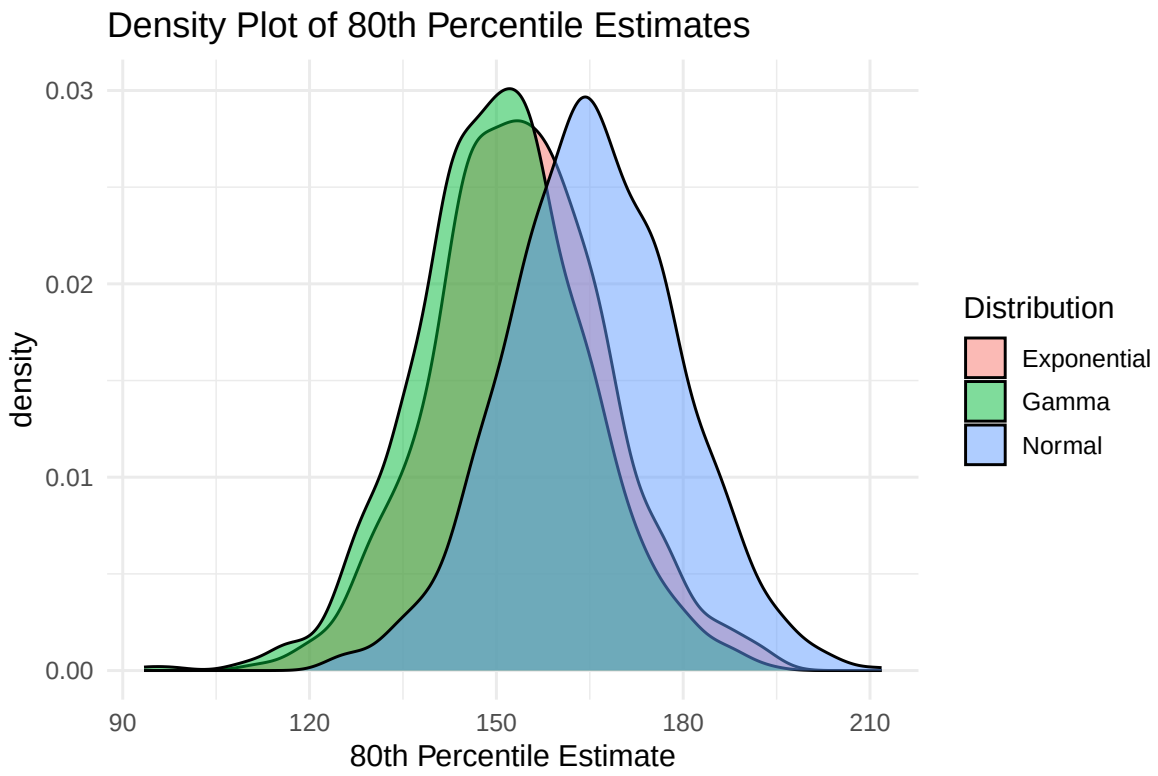
4. It is crucial to ensure that the estimator is both accurate and consistent, hence bias and standard deviation(SD) are utilised to select an appropriate estimator. Bias measures the how far, on average, the estimated P80 is from the true P80, and SD measures the variability of the estimates.

```r
# Calculate the bias and standard deviation
true_80th_percentile <- qexp(0.80, rate = lambda)

bias_normal <- mean(normal_estimates) - true_80th_percentile
bias_exp <- mean(exp_estimates) - true_80th_percentile
bias_gamma <- mean(gamma_estimates) - true_80th_percentile

sd_normal <- sd(normal_estimates)
sd_exp <- sd(exp_estimates)
sd_gamma <- sd(gamma_estimates)
```

Simulation Results

Density plots of 80th Percentile Estimates

```r
combined_estimates <- data.frame(
  estimate = c(normal_estimates, exp_estimates, gamma_estimates),
  distribution = factor(rep(c("Normal", "Exponential", "Gamma"),
                            times = c(length(normal_estimates),
                                      length(exp_estimates),
                                      length(gamma_estimates))))
)

# Create the combined density plot
ggplot(combined_estimates, aes(x = estimate, fill = distribution)) +
  geom_density(alpha = 0.5) + # Use density plot with transparency
  labs(title = "Density Plot of 80th Percentile Estimates",
       x = "80th Percentile Estimate",
       fill = "Distribution") +
  theme_minimal()
```



The normal peak is further to the right, elucidating potential overestimation of the annual income. Exponential and Gamma are similar and overlap to a large extent, though gamma exhibit more of a left-skewness, which tend to underestimate the income compared to the other two.

The most Appropriate estimator

```
# Print in table
estimates_bias_sd <- tibble(
  Estimate = c("Normal", "Exponential", "Gamma"),
  Bias = c(bias_normal, bias_exp, bias_gamma),
  SD = c(sd_normal, sd_exp, sd_gamma)
) %>% kable(caption = "The bias and SD of estimates")

estimates_bias_sd
```

Table 8: the bias and standard deviation of estimators

Table 8: The bias and SD of estimates

Estimate	Bias	SD
Normal	4.532751	13.78083
Exponential	-7.061224	13.69364
Gamma	-10.330769	13.59149

The desired estimator would have low bias and low SD. According to Table 8 - Normal: the bias(4.53) of normal estimator indicate an overestimation of consumer income, which could cause potential problems when designing the financial product. The highest SD, meaning the highest variability among three estimators also serve as another factor rejecting it to be an appropriate estimator.

- Exponential and **Gamma**: exponential and gamma distributions are alike in terms of the variability and the left-skewness, both underestimate the income as evident of a negative bias. However, Gamma has the minimum SD, providing more consistent estimates while capturing the left-skewness of the income distribution.

Gamma simulation

If the simulation is based from a gamma distribution with shape parameter not equal to 1, the bias of the estimators would be expected to decrease, since the gamma estimator is assumed to capture the true data generating process better than the exponential. However, the normal estimator may see increase in bias since the normal fit may overestimate the central tendency. Exponential estimator would not perform well as it is expected to show significant underestimation (negative bias) and high variability because of the mismatch in the data's characteristics.

Insights

In our analysis of customer income, we evaluated various statistical methods to estimate the 80th percentile of annual income, crucial for financial decision-making. The gamma distribution emerged as the most effective estimator due to its ability to accurately model income data, particularly for skewed distributions, providing lower variability and bias compared to normal and exponential distributions. This approach allows us to make better-informed decisions regarding product offerings and marketing strategies, ensuring that our financial products align with the actual income distribution of our customers. We recommend adopting the gamma distribution method for future analyses to enhance the relevance and effectiveness of our services.

Task 4

This section will be focusing on the relationship between customer education level and income.

```
# Recode education levels into two groups:
# "University Degree" vs "No University Degree"
bank_data <- bank_data %>%
  mutate(education_group = ifelse(education %in%
                                    c("Bachelor", "Master", "Doctorate"),
                                    "University Degree", "No University Degree"))

# Group summary of income by education group
income_education_summary <- bank_data %>%
  group_by(education_group) %>%
  summarise(mean_income = round(mean(income), 1),
            sd_income = round(sd(income), 1),
            n = n())

kable(income_education_summary,
      caption = "Income comparison of university degree holders")
```

Table 9: Income comparison of university degree holders

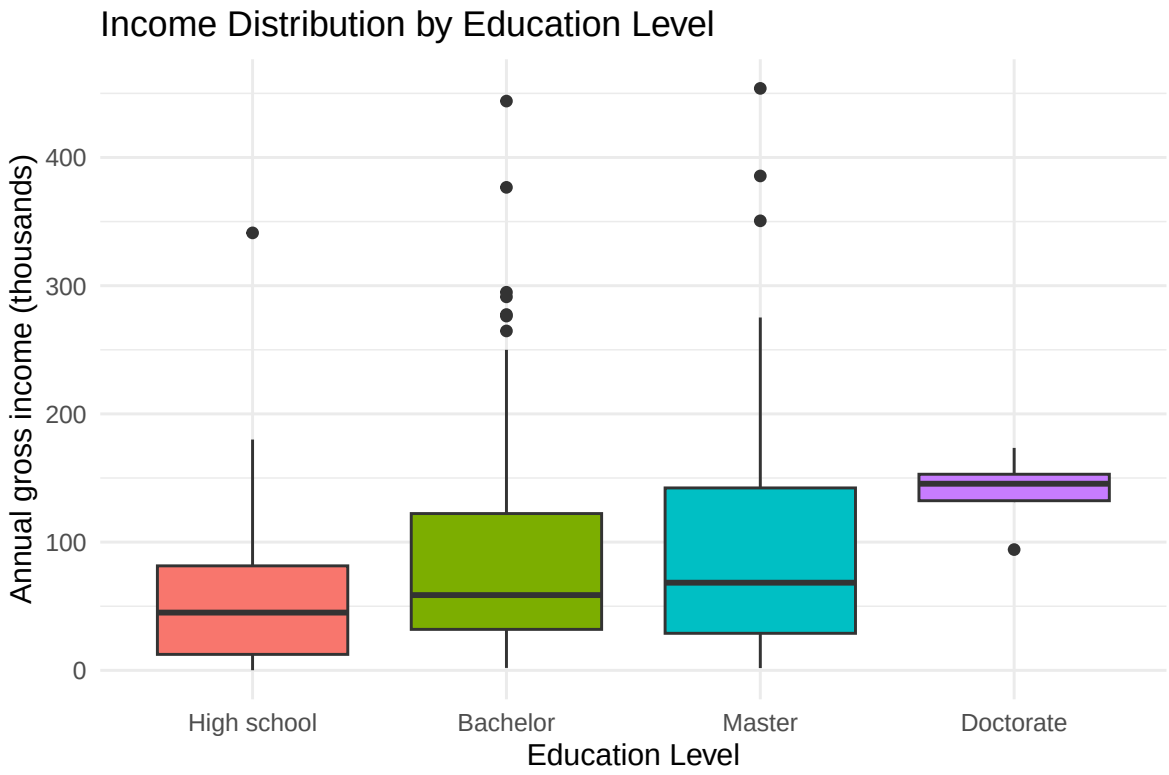
education_group	mean_income	sd_income	n
No University Degree	65.1	72.8	30
University Degree	93.5	89.4	170

University degree holders have a higher mean income and standard deviation, with a difference of $28,400 in mean compared to non-university degree holders.

Visualizing Income Distribution by Education Level

```
# Order education levels
bank_data$education <- factor(bank_data$education,
                          levels = c("High school",
                                     "Bachelor",
                                     "Master",
                                     "Doctorate"))
```

```r
# Boxplot comparing income across all education levels
ggplot(bank_data, aes(x = education, y = income, fill = education)) +
  geom_boxplot() +
  theme_minimal() +
  ggtitle('Income Distribution by Education Level') +
  xlab('Education Level') +
  ylab('Annual gross income (thousands)') +
  guides(fill = "none")
```



The boxplot showed that the median of annual gross income increases as the education level gets higher. While there is a huge gap between median income of master and doctorate degree holders, the range of doctorate's income is much narrower, and the maximum point is lower than the other 3 education groups as well.

Confidence Interval for the Difference in Mean Incomes

To compare the difference in income of customer who hold a university degree or not, t-test will be conducted to calculate the confidence interval for the difference in mean incomes.

```
# Subset data into two groups
university_graduates <- bank_data %>%  filter(education_group == "University Degree")
non_graduates <- bank_data %>%  filter(education_group == "No University Degree")

# Calculate the mean difference and confidence interval
mean_diff <- mean(university_graduates$income) - mean(non_graduates$income)

# Perform a t-test to get the confidence interval
t_test_result <- t.test(university_graduates$income, non_graduates$income,
                        var.equal = FALSE)

# Print the results
kable(round(mean_diff, 1),
      caption = "Difference in mean income (thousands)",
      col.names = "")
```

Table 10: Difference in mean income (thousands)

28.4

```
kable(round(t_test_result$conf.int, 1),
      caption = "95% confidence interval of difference in mean income (thousands)",
      col.names = "")
```

Table 11: 95% confidence interval of difference in mean income (thousands)

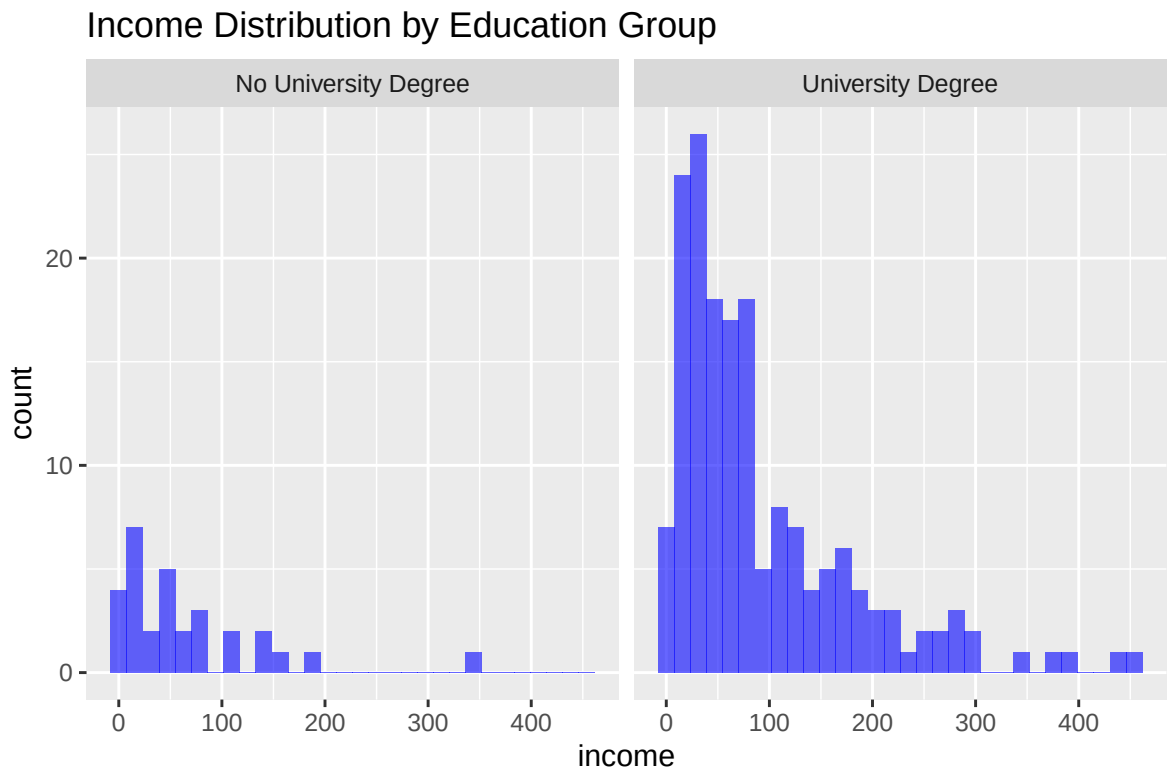
-1.7
58.5

The difference in mean of annual gross income of customers who hold a university degree or not is $28,400, with a 95% confidence interval of -$1,700 and $58,500.

Central Limit Theorem

This part discusses whether it is appropriate to use the Central Limit Theorem when calculating the above confidence interval.

```r
# Histograms to check normality
ggplot(bank_data, aes(x = income)) +
  geom_histogram(bins = 30, fill = 'blue', alpha = 0.6) +
  facet_wrap(~education_group) +
  ggtitle('Income Distribution by Education Group')
```




```r
# Check the sample sizes of each group
table(bank_data$education_group) %>%
  kable(col.names = c("Education level", "Count"),
        caption = "Sample size of education groups")
```

Table 12: Sample size of education groups

Education level	Count
No University Degree	30
University Degree	170

To use t-test to calculate confidence interval, the sample needs to be normally distributed. But according to the above plots, both groups are not normally distributed. In this case, Central Limit Theorem is needed, which states that the sample mean gets closer to normal distribution as the sample size increases, even if the data are not under normal distribution. Since both groups have a sample size of at least 30, Central Limit Theorem is applicable and t-test could still be used to calculate confidence interval.

Causation of Education and Income

This section of analysis about education level and income seem to show that the average income would be higher as education level increases. However, these are not evidences of a causal effect. As shown in the t-test results that the confidence interval showed a large positive difference, this could imply that income has a positive relationship with education level, but it could not be summarized with causation that providing greater education would lead to higher incomes.

There could be other underlying factors that are not revealed in this analysis, such as people with higher income have adequate financial support for education, or people in higher level positions are likely to be required by company to complete certain level of education. A positive relationship does not imply causation.

Task 5

This section uses Bayesian inference approach compares the difference in 80th quantile of bank customers annual income of those who holds an university degree or not. Assume an exponential distribution for the annual income, so conjugate prior $\lambda \sim \text{Gamma}(\alpha, \beta)$, and posterior $\lambda \mid x_1, x_2, \ldots, x_n \sim \text{Gamma}(\tilde{\alpha} = \alpha + n, \tilde{\beta} = \beta + n\bar{x})$. Next is simulate 10,000 values from it and calculate the difference in 80th quantile of different groups, lastly calculate 95% credible interval for it. In addition, we assume that the mean and the variance of the prior all equal to '1', so we can calculate the alpha and the beta.

```
# Prior
n0 <- nrow(non_graduates)
n1 <- nrow(university_graduates)
alpha <- 1
beta <- 1

# Posterior
alpha_tilda0 <- alpha + n0
alpha_tilda1 <- alpha + n1
beta_tilda0 <- beta + sum(non_graduates$income)
```

```r
beta_tilda1 <- beta + sum(university_graduates$income)

tibble(Parameter = c("N_0", "N_1","Alpha","Beta",
                     "Alpha_tilda_0","Alpha_tilda_1",
                     "Beta_tilda_0","Beta_tilda_1"),
       Value = c(n0,n1,alpha,beta,alpha_tilda0,alpha_tilda1,
                 beta_tilda0,beta_tilda1),
       Description = c("Number of those who doesn't have a university degree",
                       "Number of those who has a university degree",
                       "Shape of Gamma distribution",
                       "Rate of Gamma distribution",
                       "Updated shape of posterior(Gamma distribution) for non-uni",
                       "Updated shape of posterior(Gamma distribution) for uni",
                       "Updated rate of posterior(Gamma distribution) for non-uni",
                       "Updated rate of posterior(Gamma distribution) for uni")) %>%
  kable(caption = "Parameter matrix")
```

Table 13: Parameter matrix

Parameter	Value	Description
N_0	30.0	Number of those who doesn't have a university degree
N_1	170.0	Number of those who has a university degree
Alpha	1.0	Shape of Gamma distribution
Beta	1.0	Rate of Gamma distribution
Alpha_tilda_0	31.0	Updated shape of posterior(Gamma distribution) for non-uni
Alpha_tilda_1	171.0	Updated shape of posterior(Gamma distribution) for uni
Beta_tilda_0	1952.8	Updated rate of posterior(Gamma distribution) for non-uni
Beta_tilda_1	15894.2	Updated rate of posterior(Gamma distribution) for uni

So, the posterior for non-uni is X ~ Gamma(31,1952.8), and the posterior for uni X~Gamma(171,15894.2). The below code simulates 10,000 examples from the posterior distribution of the difference between the 80th percentile incomes between the two groups.

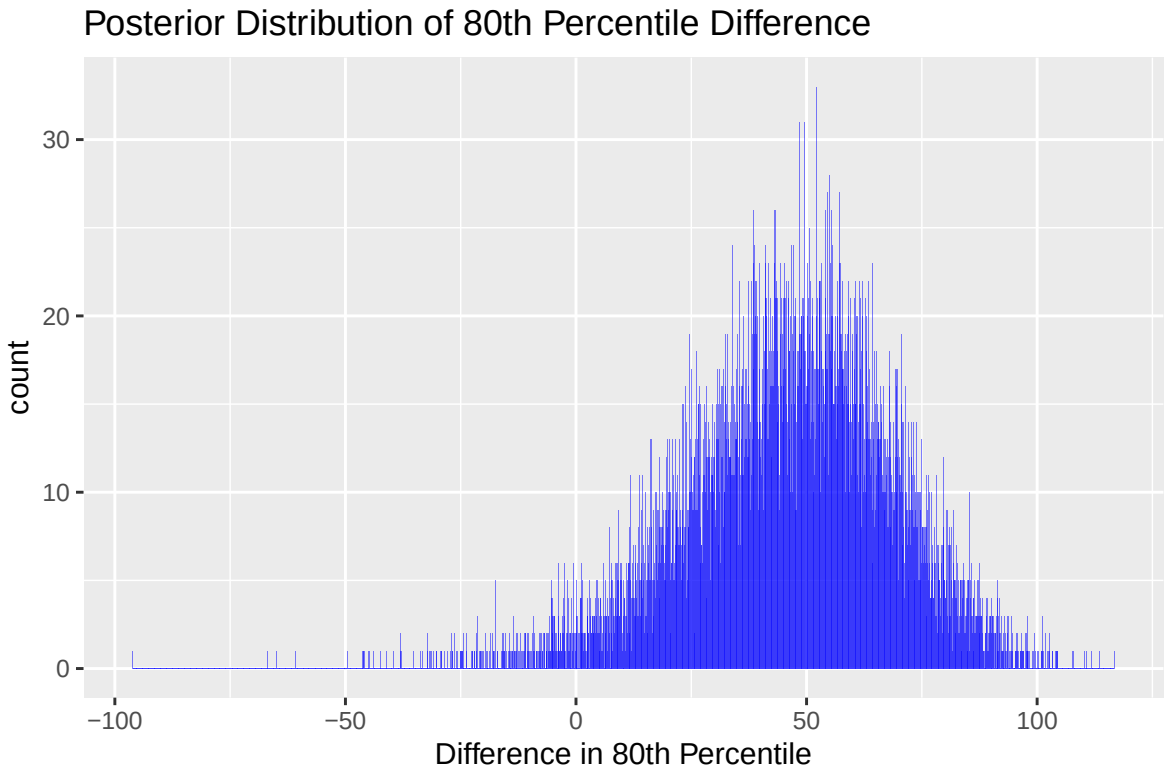
```r
# Simulate 10,000 examples
set.seed(2420)
lambda_0_samples <- rgamma(10000, shape = alpha_tilda0, rate = beta_tilda0)
lambda_1_samples <- rgamma(10000, shape = alpha_tilda1, rate = beta_tilda1)

# Manually calculate the 80th percentile of each lambda
p80_0 <- log(5) / lambda_0_samples
```

```r
p80_1 <- log(5) / lambda_1_samples

# Calculate the the 80th percentile diff
p80_diff <- p80_1 - p80_0

# Calculate the 95% CI
credible_interval <- quantile(p80_diff, c(0.025, 0.975))


tibble('Credible interval' = credible_interval) %>%
kable(caption = "95% credible interval of difference between 80th percentile incomes")
```

Table 14: 95% credible interval of difference between 80th percentile incomes

Credible interval
-2.782999
86.071030

```r
# Plot results
ggplot(data.frame(p80_diff), aes(x = p80_diff)) +
  geom_histogram(binwidth = 0.1, fill = "blue", alpha = 0.5) +
  labs(title = "Posterior Distribution of 80th Percentile Difference", x =
         "Difference in 80th Percentile")
```



The 95% credible interval for the difference between the 80th percentile incomes between the two groups is -2.78 and 86.07. The interval is mostly positive, which indicates that bank customers with university degree is more likely to earn a higher income than those who don't under 80th percentile.