2 Examine the Distribution

Immanuel Klein

```
library("tidyverse")
library("ggplot2")
```

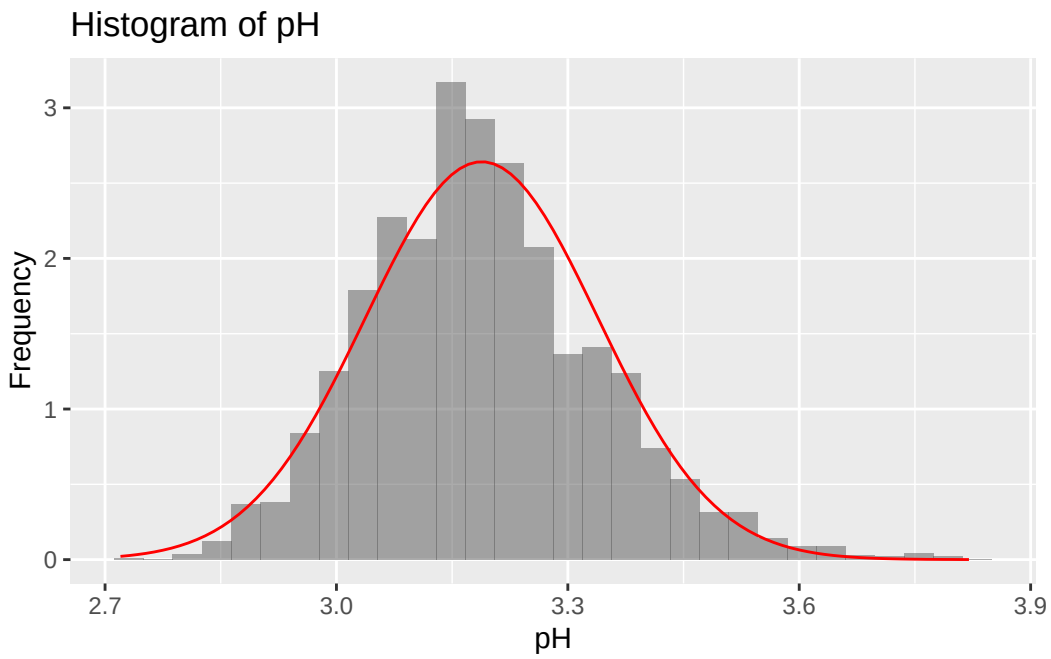
This task is about analyzing the `pH` variable from the white wine quality dataset from task 1, comparing its distribution between good and bad wines, and all wines collectively. The problem requires plotting histograms with normal density curves, generating QQ-plots and PP-plots to check for normality, and comparing the ECDFs of `pH` for the different groups. Furthermore, the analysis includes adding point-wise and uniform confidence bands to the ECDFs to further assess differences.

```
# Reading the whole data set
winequality.white <- read.csv("wine+quality/winequality-white.csv", sep = ";")

# Using only volatile.acidity, residual.sugar,
# and pH and adding binary variable good (1 if quality > 5 and 0 otherwise).
working.df <- winequality.white %>%
  mutate(good = ifelse(quality > 5, 1, 0)) %>%
  select(volatile.acidity, residual.sugar, pH, good)

head(working.df)
```

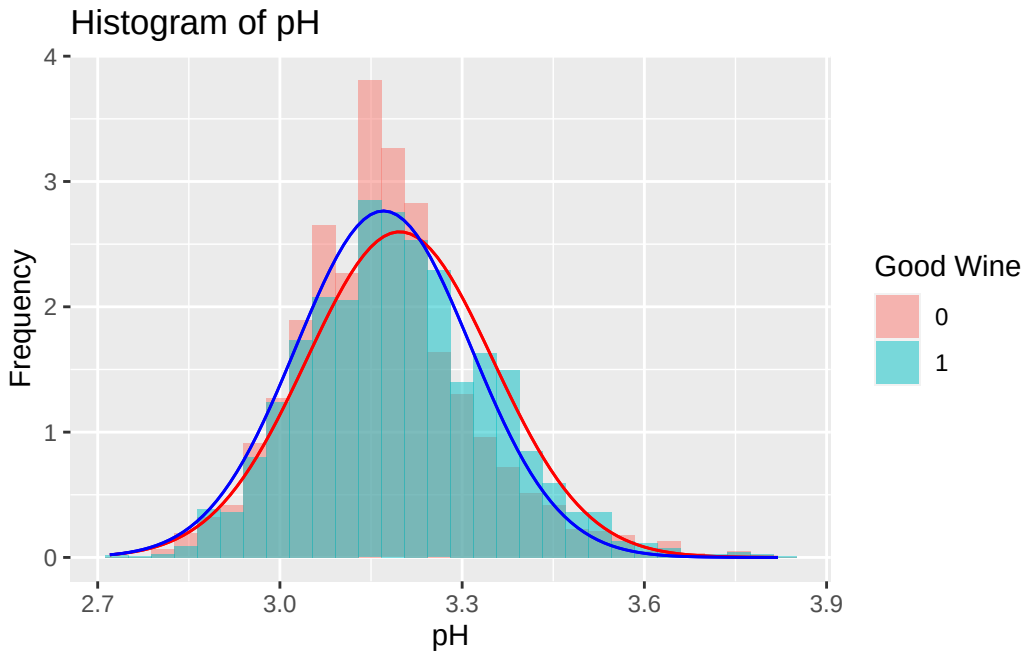
```
  volatile.acidity residual.sugar   pH good
1             0.27           20.7 3.00    1
2             0.30            1.6 3.30    1
3             0.28            6.9 3.26    1
4             0.23            8.5 3.19    1
5             0.23            8.5 3.19    1
6             0.28            6.9 3.26    1
```

Exercise (a)

```
pH.mean <- working.df %>% pull(pH) %>% mean()
pH.sd <- working.df %>% pull(pH) %>% sd()

pH.good.mean <- working.df %>% filter(good == 1) %>% pull(pH) %>% mean()
pH.good.sd <- working.df %>% filter(good == 1) %>% pull(pH) %>% sd()

pH.bad.mean <- working.df %>% filter(good == 0) %>% pull(pH) %>% mean()
pH.bad.sd <- working.df %>% filter(good == 0) %>% pull(pH) %>% sd()

# Plot histogram of pH for all wines
ggplot(working.df, aes(x = pH)) +
  geom_histogram(aes(y = after_stat(density)), bins = 30, alpha = 0.5) +
  labs(title = "Histogram of pH",
       x = "pH",
       y = "Frequency") +
  stat_function(fun = dnorm,
                args = c(mean = pH.mean, sd = pH.sd),
                color = "red")
```



```r
ggplot(working.df) +
  aes(x = pH, fill = factor(good)) +
  geom_histogram(aes(y = after_stat(density)),
                 position = "identity",
                 alpha = 0.5,
                 bins = 30) +
  labs(title = "Histogram of pH",
       x = "pH",
       y = "Frequency",
       fill = "Good Wine") +
  stat_function(fun = dnorm,
                args = c(mean = pH.good.mean, sd = pH.good.sd),
                color = "red") +
  stat_function(fun = dnorm,
                args = c(mean = pH.bad.mean, sd = pH.bad.sd),
                color = "blue")
```



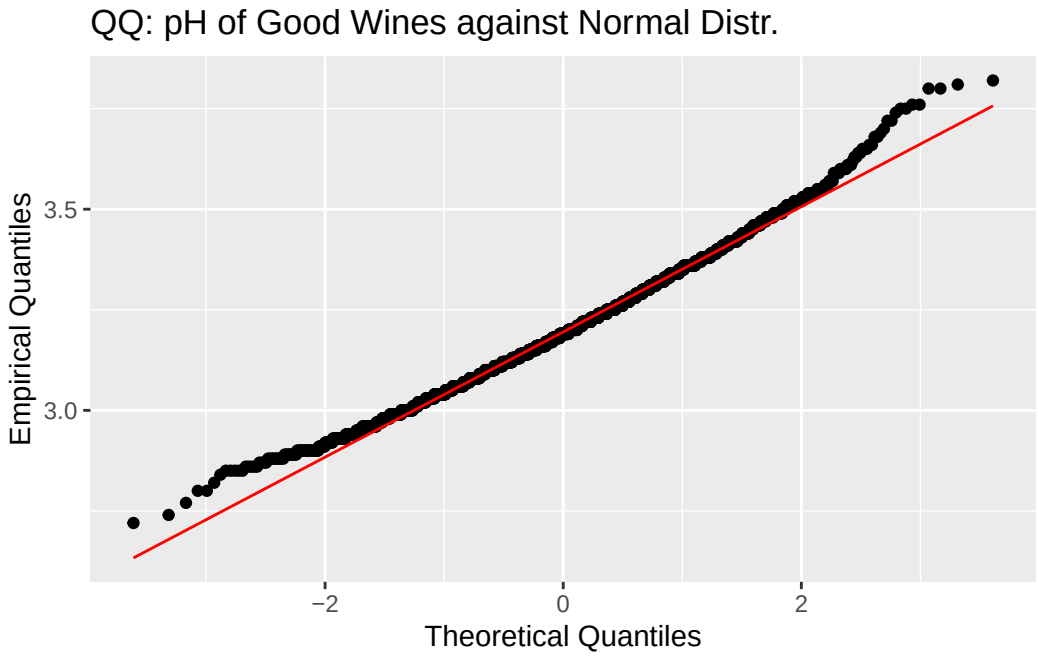
The distributions of good and bad wines deviate from their normal distributions in the following ways:

- Good Wines: The histogram is quite similar to the normal distribution, suggesting that the pH values of good wines are approximately normally distributed.

- Bad Wines: The histogram has a higher peak than the normal distribution and is slightly right-skewed compared to the normal density.

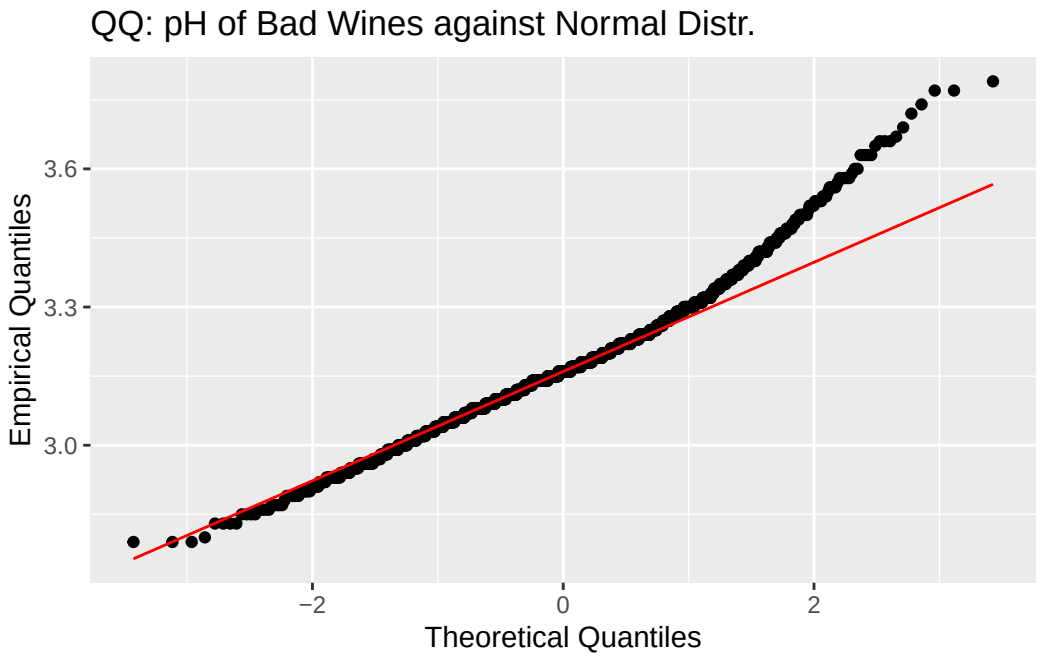
Exercise (b)

```
pH.good <- working.df %>% filter(good == 1) %>% pull(pH)
pH.bad <- working.df %>% filter(good == 0) %>% pull(pH)
pH.all <- working.df %>% pull(pH)

# QQ-Plots:
ggplot(data.frame(pH = pH.good), aes(sample = pH)) +
  geom_qq() +
  geom_qq_line(col = "red") +
  labs(x = "Theoretical Quantiles",
       y = "Empirical Quantiles",
       title = "QQ: pH of Good Wines against Normal Distr.")
```

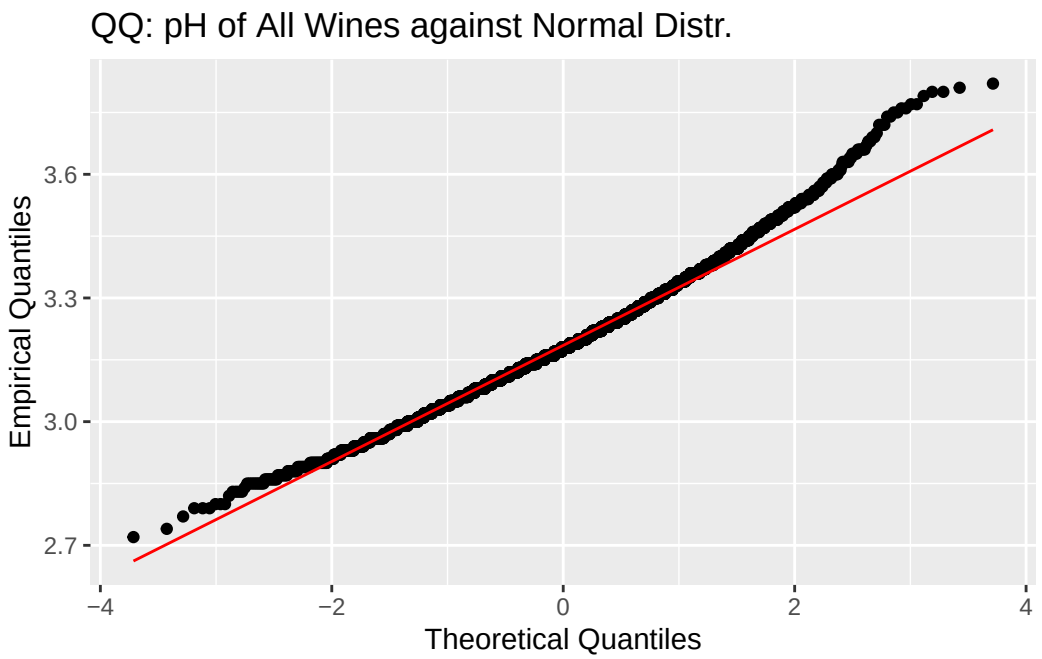



```
ggplot(data.frame(pH = pH.bad), aes(sample = pH)) +
  geom_qq() +
  geom_qq_line(col = "red") +
  labs(x = "Theoretical Quantiles",
```

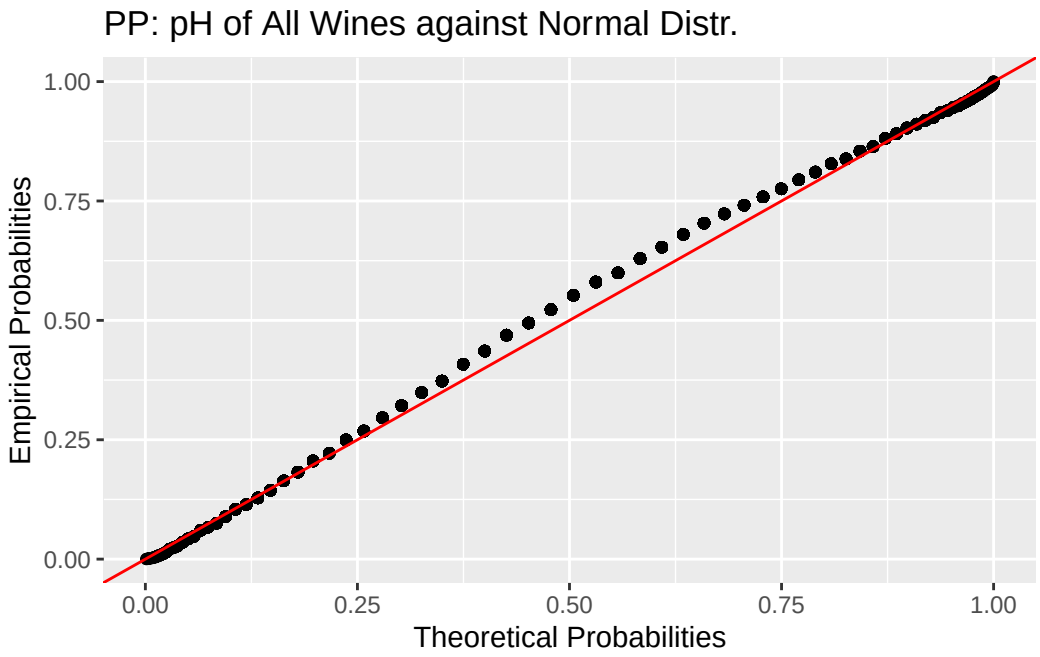
```
       y = "Empirical Quantiles",
       title = "QQ: pH of Bad Wines against Normal Distr.")
```



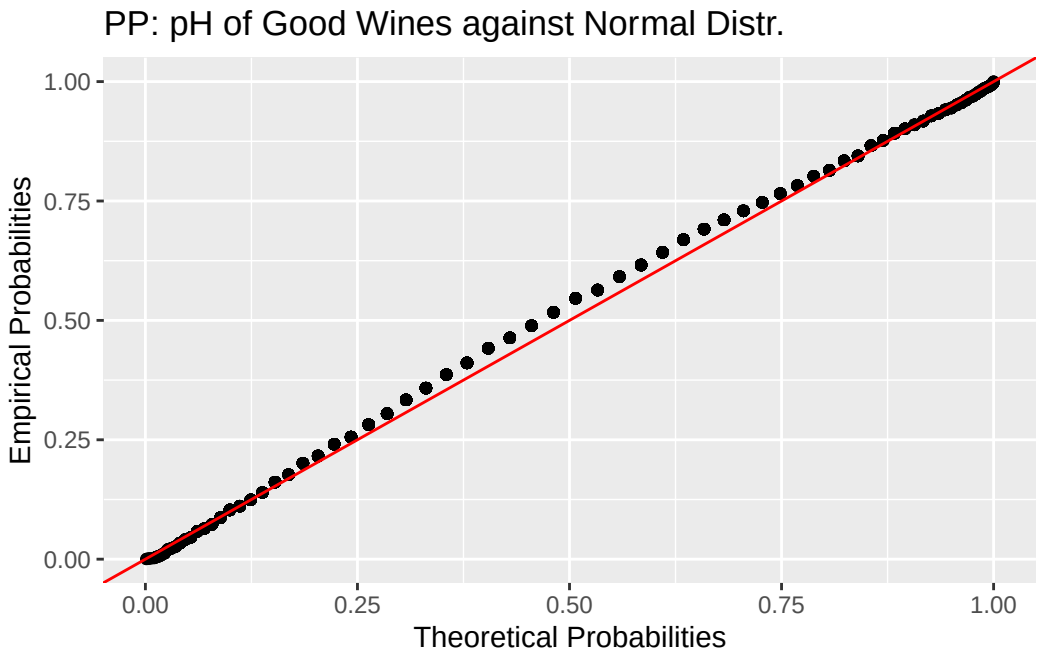

```
ggplot(data.frame(pH = pH.all), aes(sample = pH)) +
  geom_qq() +
  geom_qq_line(col = "red") +
  labs(x = "Theoretical Quantiles",
       y = "Empirical Quantiles",
       title = "QQ: pH of All Wines against Normal Distr.")
```



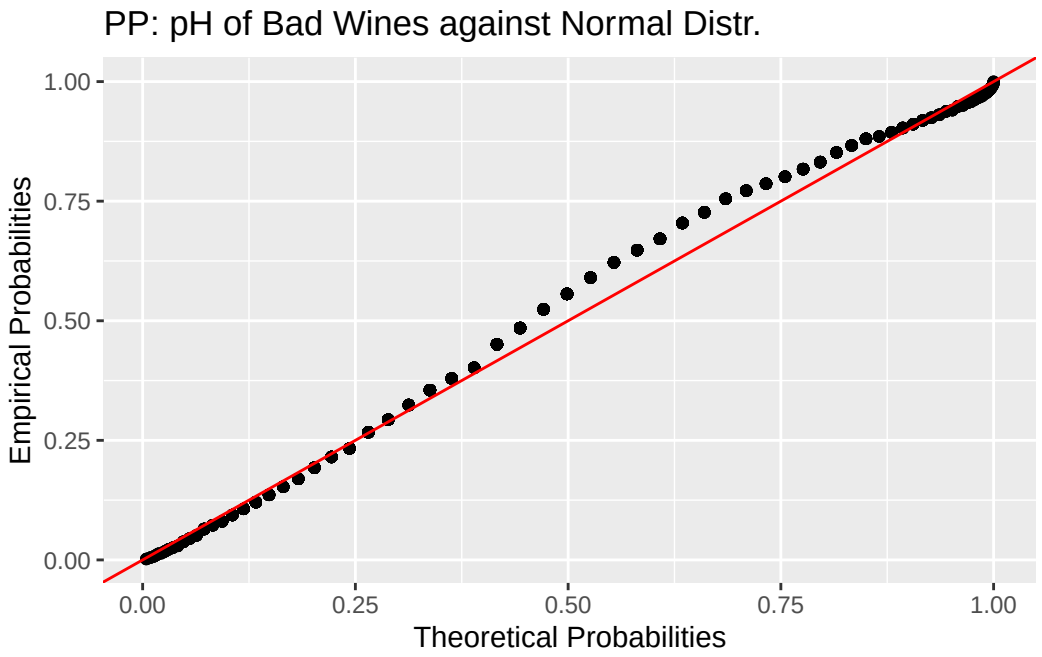

```
# Data-Prep for PP-Plots
ecdf.pH.all <- ecdf(sort(pH.all))
norm.all <- pnorm(sort(pH.all), mean(pH.all), sd = sd(pH.all))

ecdf.pH.good <- ecdf(sort(pH.good))
norm.good <- pnorm(sort(pH.good), mean(pH.good), sd = sd(pH.good))

ecdf.pH.bad <- ecdf(sort(pH.bad))
norm.bad <- pnorm(sort(pH.bad), mean(pH.bad), sd = sd(pH.bad))

# PP-Plots
ggplot(data.frame(norm.all, ecdf.pH.all(sort(pH.all))),
       aes(x = norm.all, y = ecdf.pH.all(sort(pH.all)))) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1, color = "red") +
  labs(x = "Theoretical Probabilities",
       y = "Empirical Probabilities",
       title = "PP: pH of All Wines against Normal Distr.")
```




```
ggplot(data.frame(norm.good, ecdf.pH.good(sort(pH.good))),
       aes(x = norm.good, y = ecdf.pH.good(sort(pH.good)))) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1, color = "red") +
  labs(x = "Theoretical Probabilities",
       y = "Empirical Probabilities",
       title = "PP: pH of Good Wines against Normal Distr.")
```




```
ggplot(data.frame(norm.bad, ecdf.pH.bad(sort(pH.bad))),
       aes(x = norm.bad, y = ecdf.pH.bad(sort(pH.bad)))) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1, color = "red") +
  labs(x = "Theoretical Probabilities",
       y = "Empirical Probabilities",
       title = "PP: pH of Bad Wines against Normal Distr.")
```



QQ-plots compare the quantiles of the sample data to the quantiles of a theoretical distribution, in this case the normal distribution, whereas PP-plots compare the ECDFs of the sample data and the theoretical distribution. For both holds: If the data follows the theoretical distribution, the points should lie approximately along a straight line.

Here are some key observations that are also in line with what has already been mentioned in Exercise (a):

Good Wines:

- PP-Plot: The PP-plot looks good, very close to the normal distribution, indicating a high goodness-of-fit overall.
- QQ-Plot: There is an upwards curve in the tails, indicating heavier tails with more extreme values (outliers) than the theoretical distribution.

Bad Wines:

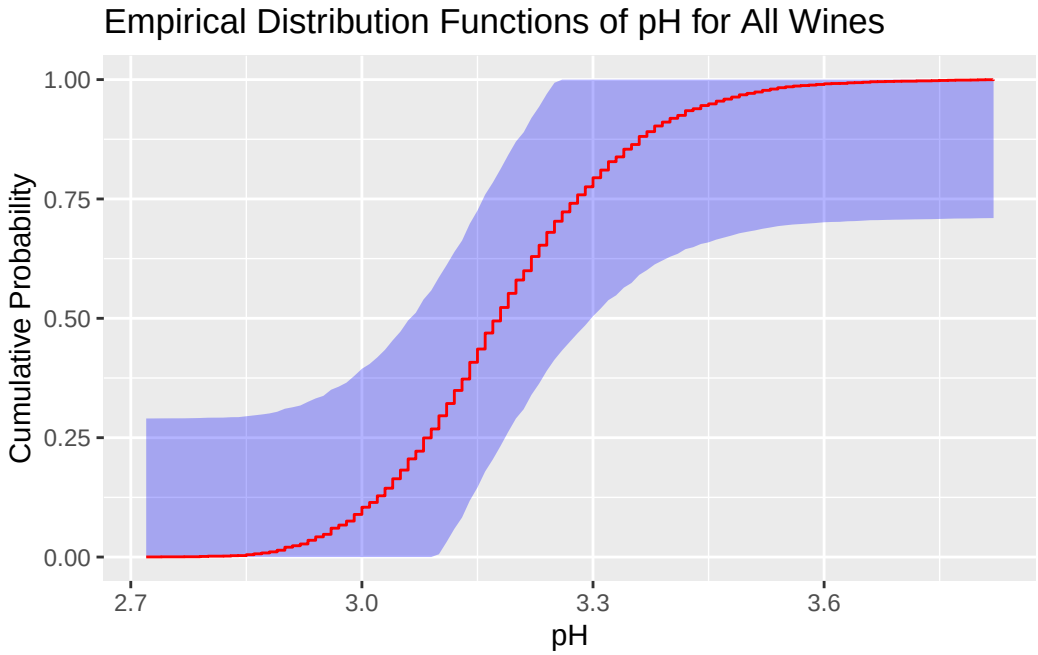
- PP-Plot: The PP-plot shows some deviation from the straight line, indicating a less perfect fit.
- QQ-Plot: Points deviate above the line in the upper tail but follow the line in the lower tail, suggesting that the sample data is right-skewed.

All Wines:

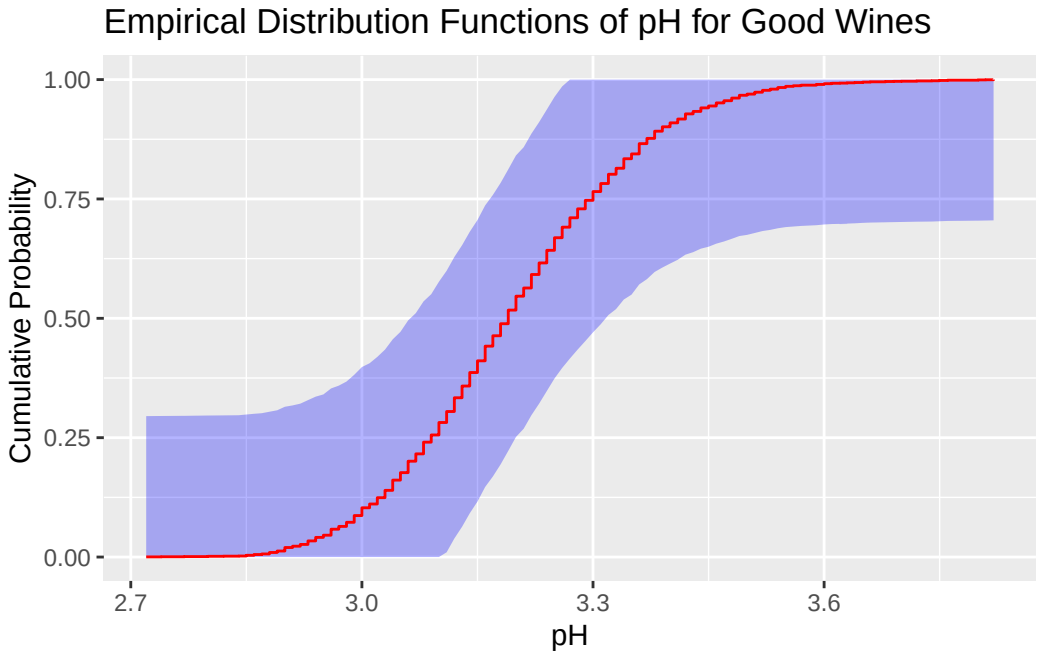
- PP-Plot: Again, the PP-plot shows some deviation from the straight line, indicating a less perfect fit.
- QQ-Plot: There are slightly heavy tails, more pronounced in the upper tail, likely due to the skewness of the bad wines.

Exercise (c)

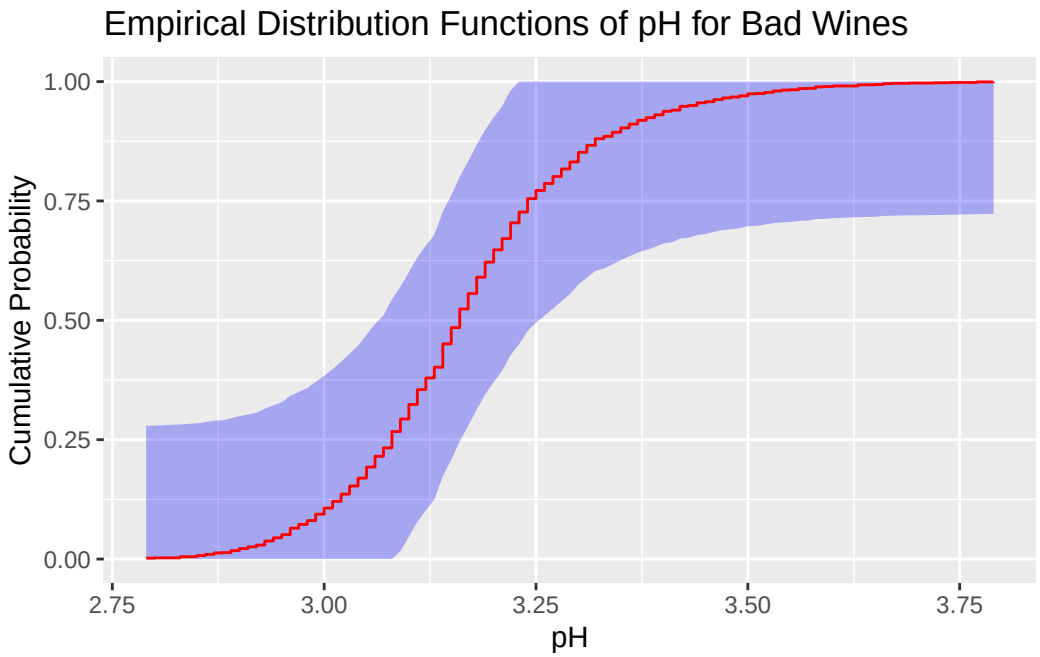
```
# Function to calculate ecdf and confidence intervals
ecdf.with.ci <- function(data, alpha = 0.05) {
  n <- length(data)
  ecdf.data <- ecdf(data)
  x <- sort(unique(data))
  y <- ecdf.data(x)

  # Slutzky's lemma and general limit theorem:
  # Using var of the sample to approx. real var
  epsilon <- sqrt(var(data) * log(2 / alpha))
  lower <- pmax(y - epsilon, 0)
  upper <- pmin(y + epsilon, 1)

  data.frame(x = x, y = y, lower = lower, upper = upper)
}

pH.good <- working.df %>% filter(good == 1) %>% pull(pH)
pH.bad <- working.df %>% filter(good == 0) %>% pull(pH)
pH.all <- working.df %>% pull(pH)

ecdf.all <- ecdf.with.ci(pH.all)
ecdf.good <- ecdf.with.ci(pH.good)
ecdf.bad <- ecdf.with.ci(pH.bad)

# Plot all wines
ggplot(ecdf.all, aes(x = x, y = y)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), alpha = 0.3, fill = "blue") +
  geom_step(color = "red") +
  labs(title = "Empirical Distribution Functions of pH for All Wines",
       x = "pH",
       y = "Cumulative Probability")
```




```
# Plot good wines
ggplot(ecdf.good, aes(x = x, y = y)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), alpha = 0.3, fill = "blue") +
  geom_step(color = "red") +
  labs(title = "Empirical Distribution Functions of pH for Good Wines",
       x = "pH",
       y = "Cumulative Probability")
```



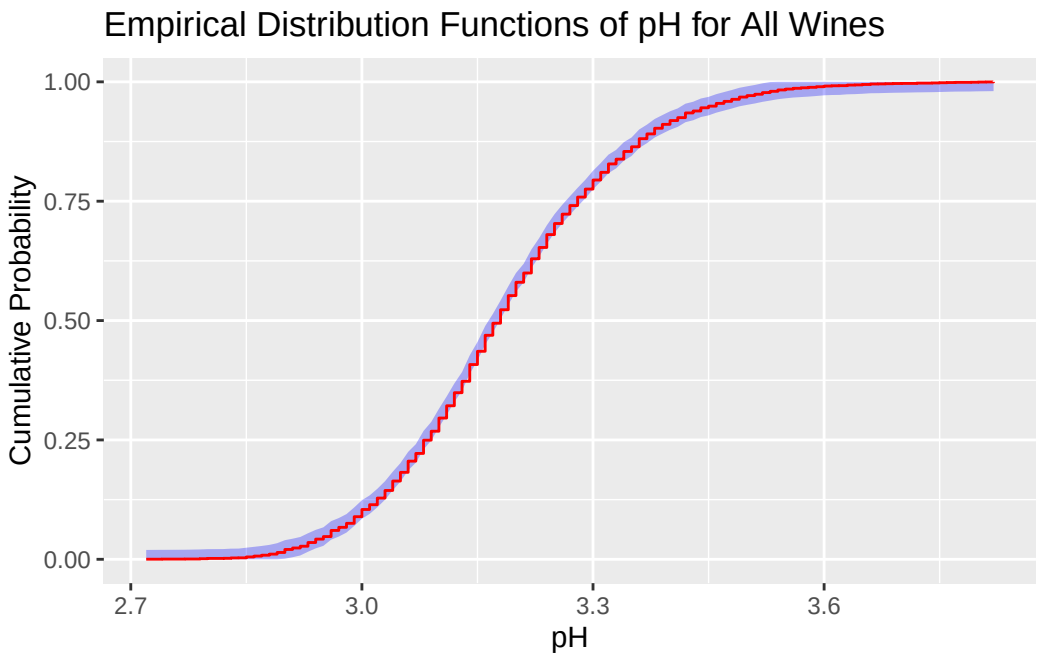

```
# Plot bad wines
ggplot(ecdf.bad, aes(x = x, y = y)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), alpha = 0.3, fill = "blue") +
  geom_step(color = "red") +
  labs(title = "Empirical Distribution Functions of pH for Bad Wines",
       x = "pH",
       y = "Cumulative Probability")
```



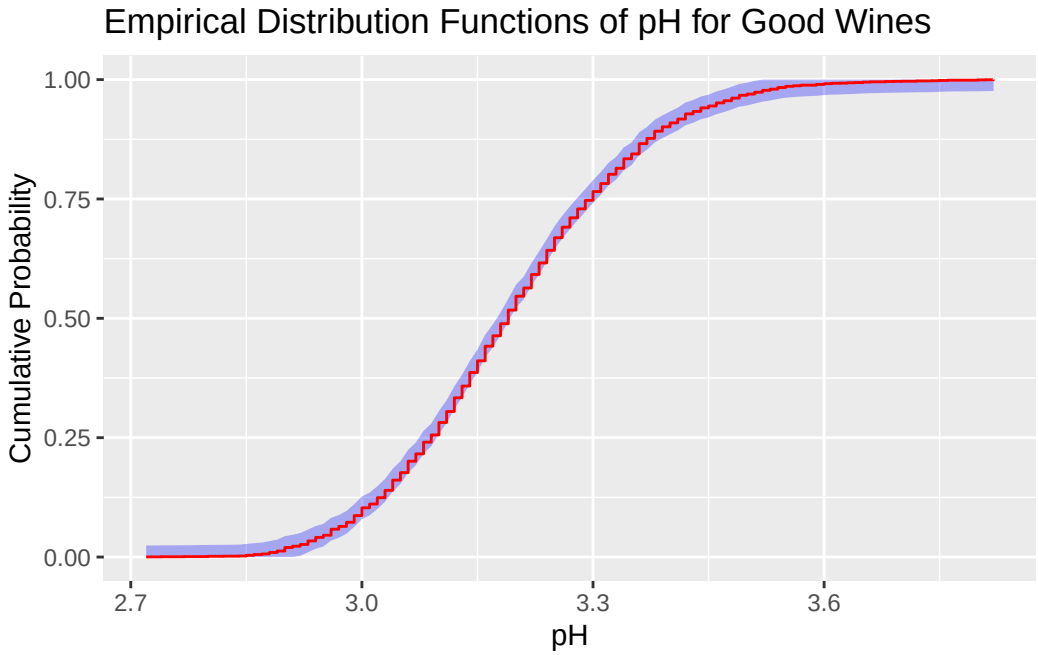
Exercise (d)

```r
# Function to calculate ecdf and uniform confidence intervals using KS method
ecdf.with.ks.ci <- function(data, alpha = 0.05) {
  n <- length(data)
  ecdf.data <- ecdf(data)
  x <- sort(unique(data))
  y <- ecdf.data(x)

  # Kolmogorov-Smirnov criterion for confidence bands
  epsilon <- sqrt(-0.5 * log(alpha / 2)) / sqrt(n)
  lower <- pmax(y - epsilon, 0)
  upper <- pmin(y + epsilon, 1)

  data.frame(x = x, y = y, lower = lower, upper = upper)
}

# From this line on: same code as in Exercise (c)

pH.good <- working.df %>% filter(good == 1) %>% pull(pH)
pH.bad <- working.df %>% filter(good == 0) %>% pull(pH)
```

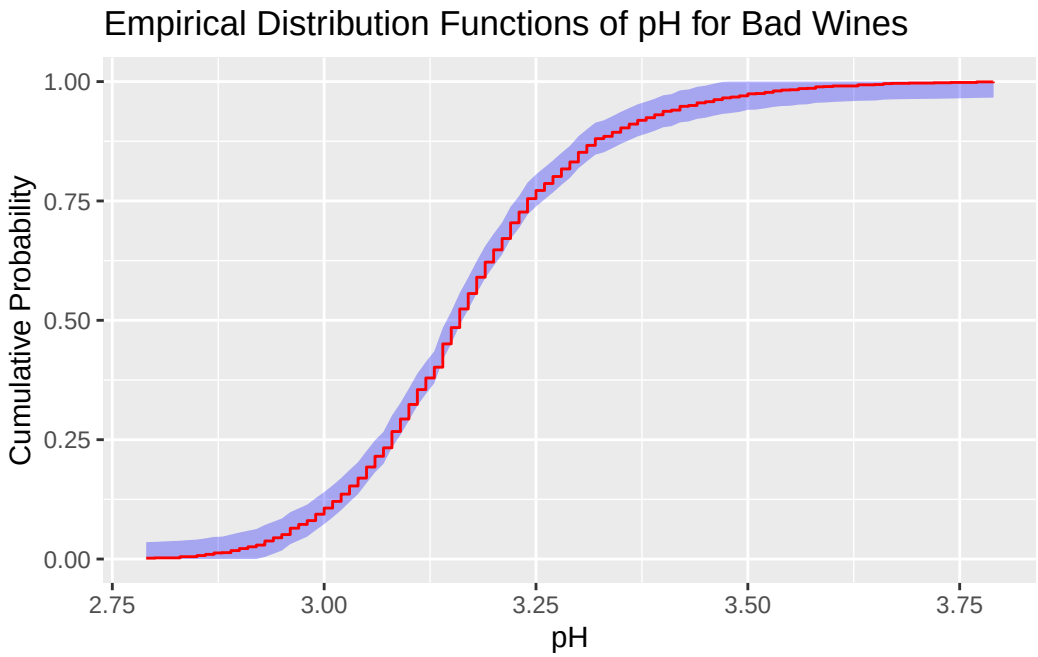
```r
pH.all <- working.df %>% pull(pH)

ecdf.all <- ecdf.with.ks.ci(pH.all)
ecdf.good <- ecdf.with.ks.ci(pH.good)
ecdf.bad <- ecdf.with.ks.ci(pH.bad)

# Plot all wines
ggplot(ecdf.all, aes(x = x, y = y)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), alpha = 0.3, fill = "blue") +
  geom_step(color = "red") +
  labs(title = "Empirical Distribution Functions of pH for All Wines",
       x = "pH",
       y = "Cumulative Probability")
```




```r
# Plot good wines
ggplot(ecdf.good, aes(x = x, y = y)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), alpha = 0.3, fill = "blue") +
  geom_step(color = "red") +
  labs(title = "Empirical Distribution Functions of pH for Good Wines",
       x = "pH",
       y = "Cumulative Probability")
```



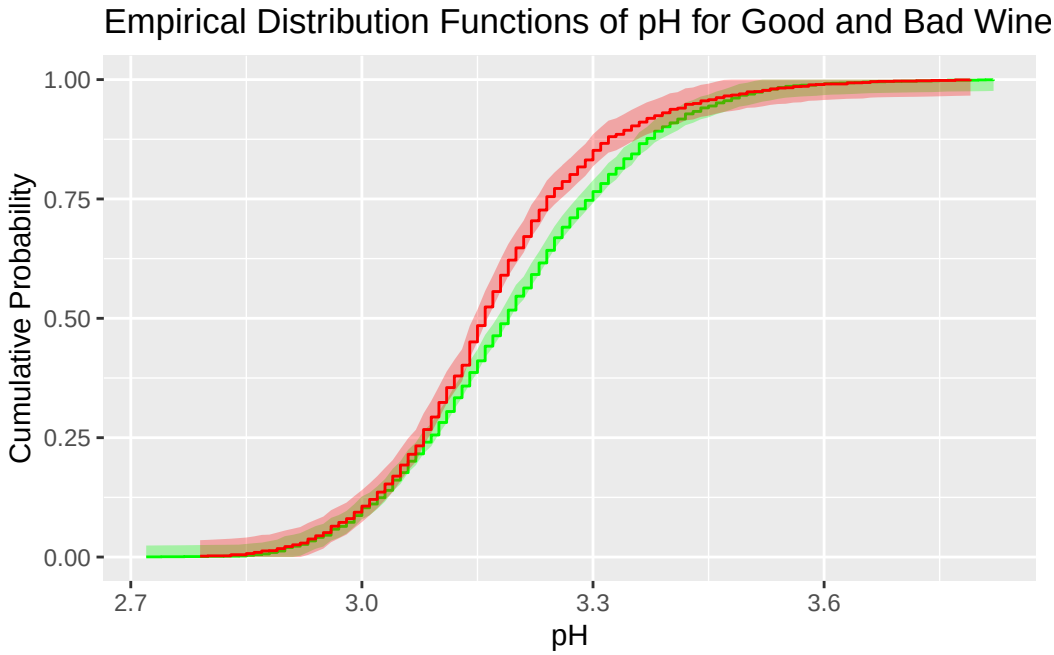

```r
# Plot bad wines
ggplot(ecdf.bad, aes(x = x, y = y)) +
  geom_ribbon(aes(ymin = lower, ymax = upper), alpha = 0.3, fill = "blue") +
  geom_step(color = "red") +
  labs(title = "Empirical Distribution Functions of pH for Bad Wines",
       x = "pH",
       y = "Cumulative Probability")
```



Using the uniform confidence bands and calculating with the Kolmogorov-Smirnov criterion yields much smaller confidence bands for the same alpha than using point-wise confidence bands does.

Exercise (e)

```
# Combine good and bad wine ECDFs with uniform confidence bandsin one plot
ggplot() +
  # Good wines
  geom_ribbon(data = ecdf.good, aes(x = x, ymin = lower, ymax = upper),
              alpha = 0.3, fill = "green") +
  geom_step(data = ecdf.good, aes(x = x, y = y), color = "green") +
  # Bad wines
  geom_ribbon(data = ecdf.bad, aes(x = x, ymin = lower, ymax = upper),
              alpha = 0.3, fill = "red") +
  geom_step(data = ecdf.bad, aes(x = x, y = y), color = "red") +
  labs(title = "Empirical Distribution Functions of pH for Good and Bad Wines",
       x = "pH",
       y = "Cumulative Probability")
```



While there may be some tendency for good wines to have slightly higher `pH` values, the `pH` distributions of good and bad wines mostly overlap. That suggests that `pH` alone might not be a definitive factor for distinguishing between good and bad wines. The overlap between the confidence intervals for good and bad wines indicates that the differences in the distributions are not significant for most of the `pH` range and that they are likely due to chance.