The Power to See: A New Graphical Test of Normality

Steven Lu, Mingyang Xue, Richard Zhu

6/9/2020

Abstract

The Kolmogorov-Smirnov test is by far the only common normality test that has a graphical representation, but it has a few problems such as not being sensitive at tails and does not work well with unknown parameters. In this paper, we discuss a new testing procedure that works better at tails while providing graphical procedures with confidence bands for a normal quantile-quantile plot. We also find out that the MLE is not a good way to estimate the unknown parameters when the data has extreme values in the tails. Instead, a robust estimator such as the one introduced by Croux and Rousseuw (1993) would give more consistent results and precise power.

Introduction

When testing for distributions, graphics help statisticians to see how the observed data fits the distribution. Gaussian distribution, also known as the normal distribution, is a common type of continuous distribution we use in real life. There are many different tests for normality, but only the Kolmogorov-Smirnov(KS) test has a graphical representation. However, there are some down sides with the KS test. The KS test is not sensitive to deviations at the tails of the distribution than at the center. In *The Power to See: A New Graphical Test of Normality*, the authors simulated confidence bands for a normal quantile-quantile plot, and introduced a new procedure called Tail-Sensitive confidence band that detects deviation from normality in the tails better than the KS test. Also, similar to the KS test, they made graphical representations for the TS test to have more insights on how the observed data fits normal distribution than other kinds of tests such as Shapiro-Wilk test. In this paper, we will first show how to construct the TS and KS confidence bands and explain how they test for normality. Then, we will show some examples where the TS test detects deviations from normality that the KS test missed identifying. In the end, we will examine the power of the TS procedure for both known and unknown parameters. We want to show that the TS test could be useful in more applications because of its ability to test fat-tailed distributions and its visibility on graphics.

KS Test for normality

The Kolmogorov-Smirnov test is used to compare one sample with a known probability distribution or to compare two samples. In this paper, we will focus on one sample KS-Test, and use it to test the normality of the observed data.

The KS statistic is defined as $D_n = \sqrt{n} \sup\limits_{-\infty<t<\infty} |F_n(t) - F_0(t)|$. $F_n(t)$ is the empirical distribution for n $i.i.d$ ordered observation X_i where $F_n(x) = \frac{1}{n}\sum_{n=1}^{n} I(X_i \leq x)$. The idea behind this test is to calculate the difference between the distribution of the underlying population with the cdf of a normal distribution. In the KS normality Test, we usually do the hypothesis testing where H_0 : the data follow the normal distribution, and H_1 : the data do not follow the normal distribution. Furthermore, Smirnov created tables for the asymptotic null distribution of D_n under the null hypothesis. The tables provide c_α where $lim_n P(sup_t|F_n(t) - f_0(t) \leq c_\alpha/\sqrt{n}) \approx 1 - \alpha$.

However, there are a few limitations with the KS Test. The KS test is more sensitive at the center than at the tails, which makes it easier to accept the null hypothesis when there are deviations at the tails. Also, the KS Test lacks power when we use estimated parameters. Taking $\mu = \bar{X}, \sigma^2 = s^2$ as estimators, the Monte Carlo critical values are about two-thirds compared to the values from the standard table for the KS Test. This would results in an error that the actual significance level would be a lot lower than the ones given by the KS table. (Lilliefors, 1967)

To make the visualization for the KS-Test, we construct confidence bands based on D_n of the form $\{x, F_0(x) \pm c_\alpha/\sqrt{n}\}$ around the hypothesized cumulative distribution. We reject H_0 if F_n has any point that lies outside of the limits. This confidence band could also be placed in a $Q-Q$ plot. Since the limits for D_n are distribution-free, we can imply samples from a uniform distribution where $F_0(x) = x$ to the limits. The bands will then be $\{x, x \pm c_\alpha/\sqrt{n}\}$, which are parallel diagonal lines that are truncated at 0 and 1, so when we inverse the limits to get the quantiles of the observed sample, the limits do not work in the tails of the data.

Tail-Sensitive Confidence Bands

Next, we will introduce the methodology of Tail-Sensitive(TS) confidence bands and the algorithm to compute the TS bands proposed by Aldor-Noiman et.al (2013). First, we will discuss the method for which the parameters are known and later show a slight modification to account for the case in which the parameters are unknown.

Suppose we want to test whether $X_1, ..., X_n$ are normally distributed with mean μ_0 and standard deviation

σ_0, which both μ_0 and σ_0 are known. We first standardize the sample to be $Z_i = \frac{X_i - \mu_0}{\sigma_0}$. Then the basic idea is to construct the TS bands for $Z_1, ..., Z_n$, and plot the normal Q-Q plot for the standardized sample along with the TS bands. We reject the null hypothesis that $X_1, ..., X_n \overset{i.i.d}{\sim} N(\mu_0, \sigma_0)$ if the standardized sample does not fall entirely inside the TS bands.

To form the TS bands for $Z_1, ..., Z_n$, we first construct the TS bands for a uniform sample $Y_1, ..., Yn \overset{i.i.d}{\sim}$ Uniform(0,1) and then transform back to the normal scale by inverting the standard normal CDF. The reason behind this design is that we know the quantiles of uniform distribution follow a Beta distribution, so we can easily form $1 - \gamma$ confidence intervals for each uniform quantile $Y_{(i)}$ where $Y_{(i)}$ denotes the i^{th} order statistic. After that, we can form the simultaneous confidence bands with a coverage rate of $1 - \alpha$ by adjusting the confidence level γ.

So far we have presented the main ideas for constructing the TS bands. Next, we will provide details about this construction.

Confidence Interval for $Y_{(i)}$

Suppose we simulate $Y_1, ..., Y_n \overset{i.i.d}{\sim}$ Uniform(0,1). To construct confidence interval for the ordered statistic $Y_{(i)}$, we rely on the fact that $Y_{(i)}$ follows a Beta$(i, n-i+1)$ distribution:

$$\begin{aligned}
f_{Y_{(i)}(y)} &= \frac{n!}{(i-1)!(n-i)!}[F_Y(y)]^{i-1}[1-F_Y(y)]^{n-i}f_Y(y) \\
&= \frac{n!}{(i-1)!(n-i)!}y^{i-1}(1-y)^{n-i} \cdot 1 \\
&= \frac{\Gamma(i+n-i+1)}{\Gamma(i)\Gamma(n-i+1)}y^{i-1}(1-y)^{n-i} \\
&= \frac{1}{B(i, n-i+1)}y^{i-1}(1-y)^{n-i} \\
&\sim \text{Beta}(i, n-i+1)
\end{aligned}$$

The $1 - \gamma$ confidence interval for $Y_{(i)}$ is $[L_i(\gamma), U_i(\gamma)]$ such that $P(L_i(\gamma) \leq Y_{(i)} \leq U_i(\gamma)) = 1 - \gamma$. By using the fact that $Y_{(i)} \sim$ Beta$(i, n-i+1)$ and assuming the equal-tail confidence interval, we obtain $L_i(\gamma) = B^{-1}_{(i,n-i+1)}(\frac{\gamma}{2})$ and $U_i(\gamma) = B^{-1}_{(i,n-i+1)}(1 - \frac{\gamma}{2})$.

Simultaneous Confidence Bands

To form simultaneous confidence bands for $Y_{(1)}, ..., Y_{(n)}$, we need to adjust confidence level γ so that

$$P(L_i(\gamma) \leq Y_{(i)} \leq U_i(\gamma), \forall i) = 1 - \alpha.$$

One way is to use iterative simulation to find the simultaneous confidence bands with a coverage rate of $1-\alpha$. That is, we simulate M samples from Uniform(0,1) each with sample size n. We denote $Y_{(i)}^m$ to be the i^{th} order statistic from m^{th} sample where $m = 1, ..., M$ and we compute the proportion of samples in which the confidence bands generated from $1-\gamma$ level contains all the data points. If the proportion is less than $1-\alpha$, decrease γ by a small amount ϵ and generate the confidence bands again from the new γ confidence level. Repeat the process until the proportion reaches $1-\alpha$. However, this optimization process is computationally intensive which is not ideally efficient for a computer to run. Therefore, a statistical method is suggested to avoid this computational difficulty. For each of the M^{th} sample, we want to find the smallest γ such that the collection of confidence intervals $\{[L_i(\gamma), U_i(\gamma)] : \forall i\}$ contains all the quantiles $Y_{(i)}^m$. Here we define the smallest two-sided p-value for the m^{th} sample:

$$C^m = \min\{c : L_i(c) \leq Y_{(i)}^m \leq U_i(c), \forall i\}.$$

Again, by the property that $Y_{(i)}^m$ follows a Beta distribution,

$$\begin{aligned} C^m &= \min\{c : B_{i,n-i+1}^{-1}(c/2) \leq Y_{(i)}^m \leq B_{i,n-i+1}^{-1}(1-c/2), \forall i\} \\ &= \min\{c : c/2 \leq B_{i,n-i+1}(Y_{(i)}^m) \leq 1 - c/2, \forall i\} \\ &= 2 \ \min_i\{\min(B_{i,n-i+1}(Y_{(i)}^m), 1 - B_{i,n-i+1}(Y_{(i)}^m))\} \end{aligned}$$

Then to estimate simultaneous coverage $1-\alpha$, we set γ to be 100α percentile of $C^1, C^2 ..., C^M$. This ensures that the confidence bands for M samples cover all of the data points for a fraction of α times.

Unknown Parameters

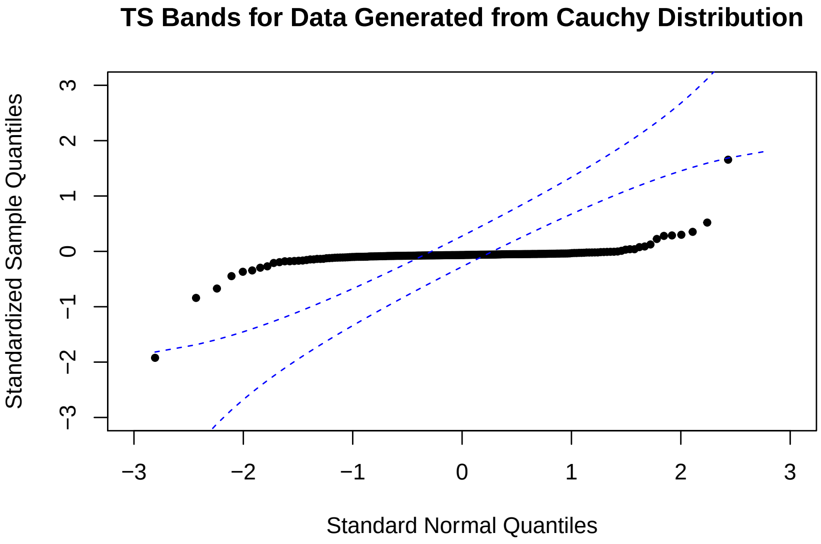
In the previous section, we have specified how we test the normality when the parameters of the normal distribution are known. However, this is not always the case. When the parameters are not known, we need to estimate these parameters and the simultaneous confidence bands should capture the effect of estimating the parameters.

Suppose we want to test whether $X_1, ..., X_n \overset{i.i.d}{\sim} N(\mu, \sigma)$ where μ and σ are not known in this case. We will standardize the sample by the estimates $(\hat{\mu}, \hat{\sigma})$ for μ and σ, i.e. $Z_i = \frac{X_i - \hat{\mu}}{\hat{\sigma}}$. One common estimators for μ and σ are the MLEs $\hat{\mu} = \bar{X}^m$ and $\hat{\sigma} = \sqrt{\frac{\sum_{i=1}^{n}(X_i^m - \bar{X}^m)^2}{n}}$. Then we construct the confidence bands as the similar logic before. We reject the null hypothesis that $X_1, ..., X_n$ are normally distributed if the confidence bands do not contain all the $Z_i's$.

There is one key modification from the steps that generate the confidence bands when the parameters are known. Previously, we generate uniform samples directly and form the TS bands on the uniform scale first. However, in this case, we first simulate $X_1^m, ..., X_n^m \overset{i.i.d}{\sim} N(0, 1)$ where $m = 1, ..., M$. Then we standardize the samples by the estimates $\hat{\mu}^m$ and $\hat{\sigma}^m$ so that $Z_i^m = \frac{X_i^m - \hat{\mu}^m}{\hat{\sigma}^m}$. This step accounts for the error in estimating the parameters. Then we transform the Z_i^m into the uniform scale by applying the standard normal CDF $Y_i^m = \Phi(Z_i^m)$. After that, we follow the same steps as in the case of known parameters.

Estimators Other Than MLEs

In the case of unknown parameters, not only the confidence level γ controls the coverage rate of the bands, but the choice of estimators also plays a huge role in determining power against the alternative distribution. One natural pair of estimators is the MLEs, which are $\hat{\mu} = \bar{X}$ and $\hat{\sigma} = \sqrt{\frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n}}$. However, they lack power when the alternative distribution has fat tails. For example, below shows the TS bands for the 200 data generated from Cauchy distribution with location 0 and scale 1.

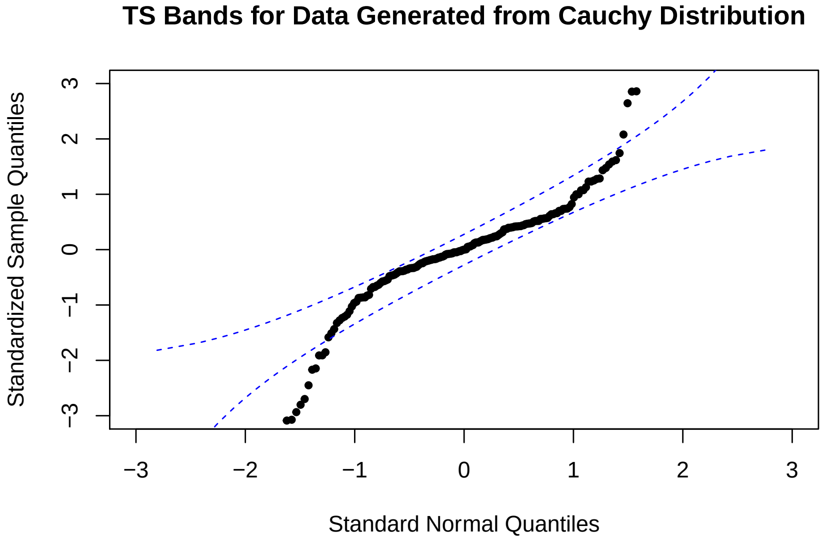


The blue dashed line depicts the TS bands using the MLEs as the estimators. The confidence bands clearly lose track of the scale of the distribution and this is due to extreme values from the tails. There are several robust estimators to solve this problem. One such estimator was proposed by Croux and Rousseeuw (1993).

The estimator is a scale estimate, denoted by Q_n, and has the form

$$Q_n = d\{|x_i - x_j| : i < j\}_{(k)},$$

where d is a constant and $k = \binom{h}{2} \approx \binom{n}{2}/4$, where $h = n/2 + 1$. That means we take the k^{th} order statistic of the $\binom{n}{2}$ interpoint distances. By some analysis and computation, $d = 2.2219$ gives a considerable small-sample bias to Q_n. Then along with median, the robust estimator of location, we form the TS bands for the same Cauchy data:



This time, the TS bands correctly identify the location and scale of the distribution which gives a good reference for testing normality.

Next, we will summarize the algorithm for computing the TS bands with known parameters and unknown parameters.

TS Bands Algorithm

Known Parameters

Our null hypothesis is $H_0 : X_1, ..., X_n \overset{i.i.d}{\sim} N(\mu_0, \sigma_0)$ where μ_0, σ_0 are known. We standardize the data by $Z_i = \frac{X_i - \mu_0}{\sigma_0}$.

1. For m = 1,...,M:

   (a) Simulate a standard uniform sample $Y_1^m, ..., Y_n^m \overset{i.i.d}{\sim} Uniform(0, 1)$.

(b) Sort the sample to have the order statistics $Y_{(1)}^m \leq ... \leq Y_{(n)}^m$.

(c) Find the minimum confidence level $C^m = 2 \min_i\{\min(B_{i,n-i+1}(Y_{(i)}^m), 1 - B_{i,n-i+1}(Y_{(i)}^m))\}$.

2. Set the confidence level γ to be the 100α percentile of $C^1, ..., C^M$.

3. Form the confidence intervals $[\Phi^{-1}(B_{i,n-i+1}^{-1}(\gamma/2)), \Phi^{-1}(B_{i,n-i+1}^{-1}(1-\gamma/2))]$ for each Z_i.

4. The collection $S = \{[\Phi^{-1}(B_{i,n-i+1}^{-1}(\gamma/2)), \Phi^{-1}(B_{i,n-i+1}^{-1}(1-\gamma/2))] : 1 \leq i \leq n\}$ specifies the TS confidence bands for $Z_1, ..., Z_n$

Unknown Parameters

Our null hypothesis is $H_0 : X_1, ..., X_n \overset{i.i.d}{\sim} N(\mu, \sigma)$ where μ, σ are unknown. We standardize the data by $Z_i = \frac{X_i - \hat{\mu}}{\hat{\sigma}}$ where $\hat{\mu}$ and $\hat{\sigma}$ are the estimates based on $X_1, ..., X_n$.

1. For m = 1,...,M:

   (a) Simulate $X_1^m, ..., X_n^m \overset{i.i.d}{\sim} N(0,1)$

   (b) Standardize the sample by $Z_i^m = \frac{X_i^m - \hat{\mu^m}}{\hat{\sigma}^m}$ where $\hat{\mu^m}$ and $\hat{\sigma}^m$ are the estimates based on $X_1^m, ..., X_n^m$.

   (c) Obtain $Y_i = \Phi(Z_i^m)$ and sort the sample to get $Y_{(1)} \leq ... \leq Y_{(m)}$.

   (d) Find the minimum confidence level $C^m = 2 \min_i\{\min(B_{i,n-i+1}(Y_{(i)}^m), 1 - B_{i,n-i+1}(Y_{(i)}^m))\}$.

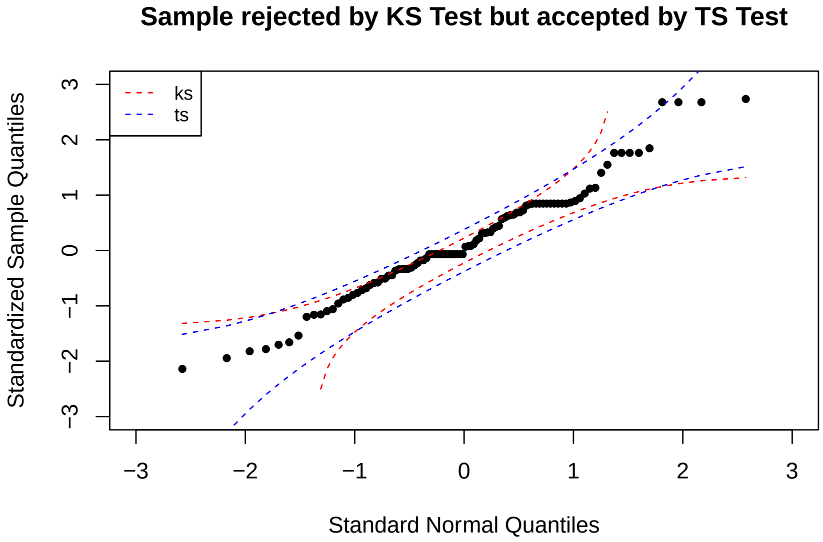
2. Set the confidence level γ to be the 100α percentile of $C^1, ..., C^m$.

3. Form the confidence intervals $[\Phi^{-1}(B_{i,n-i+1}^{-1}(\gamma/2)), \Phi^{-1}(B_{i,n-i+1}^{-1}(1-\gamma/2))]$ for each Z_i.

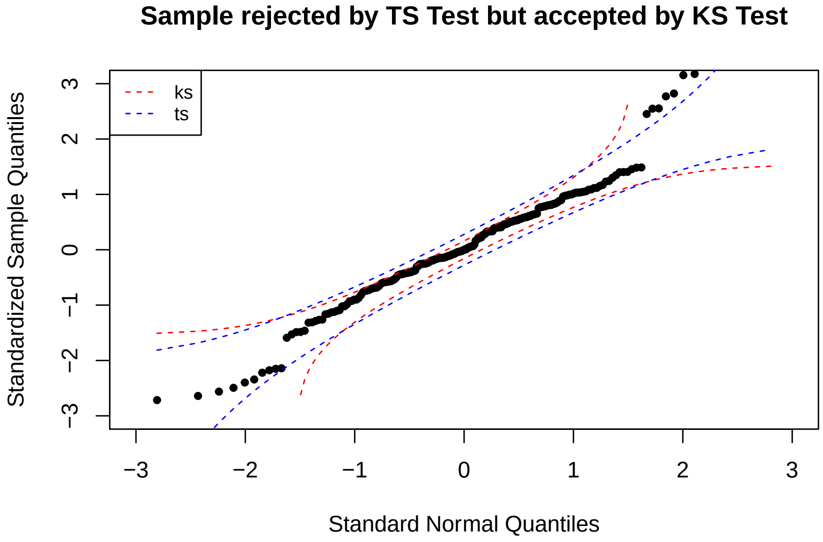
4. The collection $S = \{[\Phi^{-1}(B_{i,n-i+1}^{-1}(\gamma/2)), \Phi^{-1}(B_{i,n-i+1}^{-1}(1-\gamma/2))] : 1 \leq i \leq n\}$ specifies the TS confidence bands for $Z_1, ..., Z_n$.

Examples

According to the description of the KS Test, we know that the KS Test is more sensitive at the center than at the tails. We also know that the TS test is much more sensitive at the tails than the KS Test is. In the following, we generate 100 data where 70 of them come from a standard normal distribution $N(0,1)$ and 30 of them come from a poisson distribution $pois(0.5)$, and then we plot this group of data using the function we conducted in the previous section with robust estimators. In the graph, we can see some data points are outside of KS Confidence Bands while all data points locate inside the TS Confidence Bands. Since this group of data does not come from the normal distribution entirely, we can see that the TS Test makes an incorrect decision in this case, and the KS Test performs better.



On the other hand, TS Test performs better in some cases. In the following, we generate a group of data from a standard normal distribution $N(0,1)$ with size $n = 200$. Then we sort this group of data, subtract 0.5 from the 10 smallest data and add 1 to the 10 biggest data. Then we plot this group of data using the function we conducted in previous section with the robust estimators. It shows that the data we intentionally stretch at the tails goes outside the TS Test Confidence Bands but locates inside the KS Test Confidence Bands.



Power Analysis

In this section, we are going to show the power of the TS Test comparing with the KS Test. We use four alternative distributions as listed in the table, and get the power by calculating how many times these two tests reject alternative distributions in 100 repetitions of stimulation.

Known parameter

In this part, we use theoretical means and standard deviations as parameters to plug in the function we constructed in previous sections. The power of these two tests is shown in the table below. From the table, we can see that TS Test has lower power than KS Test except for $\chi^2(1)$, but the difference between these two tests is not huge for $\chi^2(5)$ and $T(4)$. For $Poisson(\lambda = 10)$, KS Test has much higher power since most data points we generate locate around the center where TS Test is not sensitive.

Alternative distributions	KS Test	TS Test
$\chi^2(1)$	1	1
$\chi^2(5)$	0.58	0.36
$T(4)$	0.5	0.34
$Poisson(\lambda = 10)$	0.71	0.12

Unknown Parameters

In this section, we use the same four alternative distributions but calculate the power in the case that we don't know parameters. For all distributions here, we use robust estimator since mle works poorly as we discussed before. The power we calculated is shown in the table below. From the table, we can see TS Test has higher power in $\chi^2(5)$ and $T(4)$, the reason is that these distributions have heavy tails comparing to others.

Alternative distributions	KS Test	TS Test
$\chi^2(1)$	1	1
$\chi^2(5)$	0.28	0.58
$T(4)$	0.10	0.32
$Poisson(\lambda = 10)$	0.82	0.52

Discussion

Based on the simulation study, TS bands are tighter at the tails than KS bands and thus provide greater power. It is also shown that around the center TS bands are slightly wider than KS bands. Although the difference in width is not significant, there are considerable numbers of datasets that are rejected by KS but not TS due to the points close to the center when the sample size is relatively small. Therefore, this suggests that TS bands tend to make a Type II error more likely than KS bands when the distribution around the mean has more variability or less than expected. In the power analysis section, the power of TS Test and KS

Test we get is very different from the power in the paper. In the case of known parameters, power of KS Test is higher than the power of TS Test in $\chi^2(5)$, $T(4)$ and $Poisson(\lambda = 10)$ in our analysis, but in the paper, they have very different results.

Reference

Lilliefors, H.W. (1967). *On the Kolmogorov-Smirnov test for normality with mean and variance unknown.* J.Am.Statist.Assoc., 62, 399–402.

DasGupta, A. (2011), *Probability for Statistics and Machine Learning: Fundamentals and Advanced Topics*, Berlin: Springer

Smirnov, N. (1939), "On the Estimation of the Discrepancy Between Empirical Curves of Distribution for Two Independent Samples,"

Croux, C., and Rousseeuw, P.J. (1993), "Alternatives to the Median Absolute Deviation," *Journal of the American Statistical Association*, 88, 1273-1283.[257]

Aldor-Noiman, S. et al. (2013). The Power to See: A New Graphical Test of Normality. *The American Statistician.* 67:4.

Appendix

R function that implement TS algorithm:

```
## Displays the normal qqplot for the input data vector. Computes the confidence bands
## based on the Kolmogorov-Smirnov statistic or Tail-sensitive simulation and tests
## the normality of data. For the KS bands, the function from extRemes package is adapted.
## https://rdrr.io/cran/extRemes/man/qqnorm.html
##
## x --- data vector
## method --- specify whether to use KS bands or TS bands or both
## mu,sigma --- parameters for the normal distribution.
##    if both mu and sigma are given, compute the ts bands assuming the parameters are
##    known with the given mu and sigma.
##    if either of mu or sigma is not given, compute the ts bands with the estimated
##    parameters for mu and sigma.
## estimate --- estimators to use. Either MLEs or (median, Q_n), the robust estimators for
##    the location and scale suggested by Croux, C. and Rousseeuw, P.J. (1993). The
##    functions from robustbase package are adapted.
## M --- number of simulation. Default is 5000.
## alpha --- coverage rate is specified to be 1-alpha. Default is 0.05.
## title --- title for the plot
## is.plot --- whether to display the plot
##
## return a list of objects:
##  bands --- dataframe consisting of the confidence bands for KS or TS or both depending.
##    on the specified method.
##  ks.reject --- data points that fall outside of the KS bands.
##  ks.index --- the order index of the data points that fall outside of the KS bands.
##  ts.reject --- analogous to ks.reject for TS bands.
##  ts.index --- analogous to ks.index for TS bands.
##  is.reject --- indicates whether rejecting that data comes from normal distributuion.
##    if method = both, returns a vector (ks, ts) which the 1st element indicates rejection
##    of ks and the 2nd element indicates rejection of ts.
qqnorm_bands <- function(x, method = c("ks", "ts", "both"), mu, sigma, M = 5000, alpha = 0.05,
                         estimate = c("mle","robust"), title = "", is.plot = T) {

  method <- match.arg(method)
  estimate <- match.arg(estimate)
  out <- list()
  bands <- c()
  n <- length(x)
  x <- sort(x)

  # Parameters estimation
  --------------------------------------------------------------------------
  if (missing(mu) || missing(sigma)) {
```

```
  # unknown parameters
  if (estimate == "mle") {
    mu <- mean(x)
    sigma <- sd(x)
  }
  if (estimate == "robust") {
    library(robustbase) # for robust estimates for the mean and sd
    Qn.scale<-function(x){
      Qn(x,finite.corr=FALSE)
    }
    Qn.location<-function(x){
      s_Qn(x,mu.too=TRUE)[[1]]
    }
    mu <- Qn.location(x)
    sigma <- Qn.scale(x)
  }
}

# Draw Q-Q plot
--------------------------------------------------------------------------
# standardize sample
x <- (x-mu)/sigma
# quantile based on ruler method
p <- (1:length(x) - 0.5)/length(x)
q <- qnorm(p)
if (is.plot) {
  plot(q, x, xlim = c(-3,3), ylim = c(-3,3), xlab = "Standard Normal Quantiles",
       ylab = "Standardized Sample Quantiles", main = title, pch = 20)
}

# KS bands
--------------------------------------------------------------------------
```

```
if (method == "ks" || method == "both") {
  k <- 0.895/(sqrt(n) * (1 - 0.01/sqrt(n) + 0.85/n))
  l <- suppressWarnings(qnorm(p - k))
  u <- suppressWarnings(qnorm(p + k))
  if (is.plot) {
    lines(q, l, lty = 2, col = "red")
    lines(q, u, lty = 2, col = "red")
  }
  bands <- cbind("KS.lower.band"=l, "KS.upper.band"=u)
}

# TS bands
---------------------------------------------------------------------------
if (method == "ts" || method == "both") {
  C <- rep(0,M)
  for (m in 1:M) {
    if (!missing(mu) && !missing(sigma)) {
      # known parameters
      y <- runif(n = n, min = 0, max = 1)
    } else {
      # unknown parameters
      xm <- rnorm(n = n, mean = 0, sd = 1)
      if (estimate == "mle") {
        mu <- mean(xm)
        sigma <- sd(xm)
      }
      if (estimate == "robust") {
        mu <- Qn.location(xm)
        sigma <- Qn.scale(xm)
      }
      zm <- (xm-mu)/sigma
      y <- pnorm(zm)
    }
    y <- sort(y)
    # Find minimum confidence level C_m for each simulation
    p.value <- rep(0, n)
    for (i in 1:n) {
      p.value[i] <- min(c(pbeta(q = y[i], shape1 = i, shape2 = n-i+1),
                          1 - pbeta(q = y[i], shape1 = i, shape2 = n-i+1)))
    }
    C[m] <- 2 * min(p.value)
  }
  # Set gamma to be the 100*alpha percentile of C_1,...C_M
  gamma <- quantile(C, probs = alpha)
  l <- rep(0, n)
  u <- rep(0, n)
  for (i in 1:n) {
    l[i] <- qnorm(qbeta(p = gamma/2, shape1 = i, shape2 = n-i+1))
    u[i] <- qnorm(qbeta(p = 1-gamma/2, shape1 = i, shape2 = n-i+1))
  }
  if (is.plot) {
    lines(q, l, lty = 2, col = "blue")
    lines(q, u, lty = 2, col = "blue")
  }
  bands <- cbind(bands, "TS.lower.band"=l, "TS.upper.band"=u)
}
if (is.plot && method == "both") {
  legend("topleft", legend = c("ks", "ts"), col = c("red", "blue"), lty = c(2,2), cex = 0.8)
}

bands <- data.frame(bands)

# output
---------------------------------------------------------------------------
```

```
  out[["bands"]] <- bands

  is.reject <- c()
  if (method == "ks" || method == "both") {
    ks.reject <- na.omit(x[(bands$KS.lower.band > x) | (x > bands$KS.upper.band)])
    ks.reject.index <- na.omit(which((bands$KS.lower.band > x) | (x > bands$KS.upper.band)))
    is.reject <- c(is.reject, length(ks.reject) != 0)
    out[["ks.reject"]] <- ks.reject
    out[["ks.index"]] <- ks.reject.index
  }

  if (method == "ts" || method == "both") {
    ts.reject <- x[(bands$TS.lower.band > x) | (x > bands$TS.upper.band)]
    ts.reject.index <- which((bands$TS.lower.band > x) | (x > bands$TS.upper.band))
    is.reject <- c(is.reject, length(ts.reject) != 0)
    out[["ts.reject"]] <- ts.reject
    out[["ts.index"]] <- ts.reject.index
  }

  out[["is.reject"]] <- is.reject
  return(out)
}
```

```
x1 <- rnorm(70)
x2 <- rpois(30,0.5)
x <- c(x1,x2)
qqnorm_bands(x = x, estimate = "robust", method = "both", title = "Sample rejected by KS Test but accep

x <- rnorm(200)
x <- sort(x)
x1 <- x[1:10] - 0.5
x2 <- x[191:200] + 1
x <- c(x1, x[11:190], x2)
```

```
qqnorm_bands(x = x, estimate = "robust", method = "both", title = "Sample rejected by TS Test but accep
```

```
#Power analysis for known parameter
for (i in 1:100){
  x <- rchisq(50, 1)
  out <- qqnorm_bands(x = x, mu = 1, sigma = sqrt(2), method = "both", M = 1000)
  if (out$is.reject[1] == TRUE){
    pw_1.ks <- pw_1.ks + 0.01
  }
  if (out$is.reject[2] == TRUE){
    pw_1.ts <- pw_1.ts + 0.01
  }
  x <- rchisq(50, 5)
  out <- qqnorm_bands(x = x, mu = 5, sigma = sqrt(10), method = "both", M = 1000)
  if (out$is.reject[1] == TRUE){
    pw_2.ks <- pw_2.ks + 0.01
  }
  if (out$is.reject[2] == TRUE){
    pw_2.ts <- pw_2.ts + 0.01
  }
  x <- rt(50, 4)
  out <- qqnorm_bands(x = x, mu = 0, sigma = sqrt(2), method = "both", M = 1000)
  if (out$is.reject[1] == TRUE){
    pw_3.ks <- pw_3.ks + 0.01
  }
  if (out$is.reject[2] == TRUE){
    pw_3.ts <- pw_3.ts + 0.01
  }
  x <- rpois(50, 10)
  out <- qqnorm_bands(x = x, mu = 10, sigma = sqrt(10), method = "both", M = 1000)
  if (out$is.reject[1] == TRUE){
    pw_4.ks <- pw_4.ks + 0.01
  }
  if (out$is.reject[2] == TRUE){
    pw_4.ts <- pw_4.ts + 0.01
  }
}

#Power analysis for unkown parameter
for (i in 1:100){
  x <- rchisq(50, 1)
  out <- qqnorm_bands(x = x, estimate = "robust", method = "both", M = 1000)
  if (out$is.reject[1] == TRUE){
    pw_1.ks <- pw_1.ks + 0.01
  }
  if (out$is.reject[2] == TRUE){
    pw_1.ts <- pw_1.ts + 0.01
  }
  x <- rchisq(50, 5)
  out <- qqnorm_bands(x = x, estimate = "robust", method = "both", M = 1000)
  if (out$is.reject[1] == TRUE){
    pw_2.ks <- pw_2.ks + 0.01
  }
  if (out$is.reject[2] == TRUE){
    pw_2.ts <- pw_2.ts + 0.01
  }
  x <- rt(50, 4)
  out <- qqnorm_bands(x = x, estimate = "robust", method = "both", M = 1000)
  if (out$is.reject[1] == TRUE){
    pw_3.ks <- pw_3.ks + 0.01
  }
  if (out$is.reject[2] == TRUE){
    pw_3.ts <- pw_3.ts + 0.01
  }
  x <- rpois(50, 10)
  out <- qqnorm_bands(x = x, estimate = "robust", method = "both", M = 1000)
  if (out$is.reject[1] == TRUE){
    pw_4.ks <- pw_4.ks + 0.01
  }
  if (out$is.reject[2] == TRUE){
    pw_4.ts <- pw_4.ts + 0.01
  }
}
```