This vignette shows how to reproduce the main figures in “A Joint Confidence Region for an Overall Ranking of Populations” (Klein, Wright, and Wieczorek, 2020, *Journal of the Royal Statistical Society: Series C*).

Note: For this vignette itself, we automatically save the figures below using the *knitr* package with option `dev="tikz"` instead of saving them individually. For an example of how to save individual plots using the *tikz()* function in the *tikzDevice* package, please see the *Primer* vignette: `vignette("primer", package = "RankingProject")`

**Workflow to reproduce figures from the article**

First, we load the package and the *TravelTime2011.1dec* dataset used in the paper. Note that we are using the version of the data where estimates and Margins of Error have been rounded to 1 decimal place, causing some ranks to be tied.

```r
library(RankingProject)
data(TravelTime2011.1dec)
USdata <- TravelTime2011.1dec
head(USdata)
```

```plaintext
# Rank State Estimate.1dec MOE.1dec Abbreviation Region
# 1  2 South Dakota  16.9  0.5   SD MIDWEST
# 2  2 North Dakota  16.9  0.6   ND MIDWEST
# 3  4 Wyoming       18.1  0.8   WY WEST
# 4  4 Nebraska      18.1  0.3   NE MIDWEST
# 5  4 Montana       18.2  0.5   MT WEST
# 6  6 Alaska        18.4  0.5   AK PACIFIC
# FIPS
# 1  46
# 2  38
# 3  56
# 4  31
# 5  30
# 6  2
```

n = nrow(USdata)
alpha = 0.1
Z = qnorm(1-alpha/2)
Z.Indep = qnorm(1-(1-(1-alpha)^(1/n))/2) # around 3.081
USdata$IndepCiLo = with(USdata, round(Estimate.1dec - Z.Indep/Z*MOE.1dec, 1))
USdata$IndepCiHi = with(USdata, round(Estimate.1dec + Z.Indep/Z*MOE.1dec, 1))

attach(USdata)
## We could have used a Bonferroni correction instead.

### Not run:

```r
# Z.Bonf = qnorm(1-alpha/(n*2)) # around 3.096
# USdata$BonfCiLo = with(USdata, round(Estimate.1dec - Z.Bonf/Z*MOE.1dec, 1))
# USdata$BonfCiHi = with(USdata, round(Estimate.1dec + Z.Bonf/Z*MOE.1dec, 1))
```

Reproduce Figure 1, the plot of the 90% joint confidence region for the overall ranking:

```r
par(xpd = TRUE,
    mar = c(6.3, 2.8, 0.3, 0.3) + 0.1)
plot(c(0, n+1), c(0, n), type='n', bty='n', xaxt='n', yaxt='n', xlab='', ylab='',
     xaxs = 'i', yaxs = 'i')
text(-3.5, n, "$r_k$", cex = 1, pos = 4)
wd = 0.5; ht = 0.5
for(ii in seq(1, n-2, by = 6)){
  polygon(c(1-wd, n+wd, n+wd, 1-wd),
          c(ii+ht, ii+ht, ii+ht+2, ii+ht+2),
          border = NA, col = "grey90")
}
for(ii in 1:n){
  SigDiffLo = sum(IndepCiHi <= IndepCiLo[ii])
  SigDiffHi = sum(IndepCiLo >= IndepCiHi[ii])
  NotSigDiff = (SigDiffLo+1):(n-SigDiffHi)
  mycex = 0.5
  ## Add text
  text(ii, (1:n)[NotSigDiff], Abbreviation[ii], cex = mycex, family = "mono", font = 2)
  ## Draw box
  wd = .5
  ht = .5
  polygon(c(ii-wd, ii+wd, ii+wd, ii-wd),
          c(Rank[ii]-ht, Rank[ii]-ht, Rank[ii]+ht, Rank[ii]+ht),
          border = NA, col = "grey20")
  text(ii, Rank[ii], Abbreviation[ii], cex = mycex, family = "mono", font = 2, col = "white")
}
axis(1, at = 1:n, labels = FALSE)
text(1:n + 0.5, par("usr")[3] - 2.0, labels = State, srt = 45, pos = 2, xpd = TRUE, cex = 0.7)
axis(2, at = 1:n, las = 2, cex.axis = 0.7)
```
Reproduce Figure 2, the plot of 90% joint confidence intervals for the travel times:

```r
stopifnot(15 <= min(IndepCiLo) & max(IndepCiHi) <= 35)
theamin = 15.5
thetamax = 33
mycex = 0.5
tickWidth = 2/n
par(xpd = TRUE, mar = c(6.3, 2.8, 0.3, 0.3) + 0.1)
plot(c(0, n+1), c(thetamin, thetamax),
     type='n', bty='n', xaxt='n', yaxt='n', xlab='', ylab='',
     xaxs = 'i', yaxs = 'i')
wd = 0.5; ht = 0.5
for(ii in seq(1, n-2, by = 6)){
  polygon(c(ii-ht, ii-ht, ii+ht+2, ii+ht+2),
          c(thetamin, thetamax, thetamax, thetamin),
          border = NA, col = "grey90")
}
text(-3.5, thetamax - 0.3, "$\theta_k$", cex = 1, pos = 4)
for(ii in 1:n){
  points(ii, Estimate.1dec[ii], pch=16, cex=mycex)
  arrows(y0 = Estimate.1dec[ii], x0 = ii, y1 = IndepCiLo[ii],
         angle = 90, length = tickWidth)
  arrows(y0 = Estimate.1dec[ii], x0 = ii, y1 = IndepCiHi[ii],
         angle = 90, length = tickWidth)
}
axis(1, at = 1:n, labels = FALSE)
text(1:n + 0.5, par("usr")[3] - 0.8, labels = State, srt = 45, pos = 2, xpd = TRUE, cex = 0.7)
axis(2, at = seq(16, 32, by = 2), las = 2, cex.axis = 0.7)
```