Scaling Test Scores – An Illustration Using R

 # # The dataset <PIRLS 2011_Morocco_MC reading items_scored.csv> shows responses of 1,512 Moroccan fourth-graders to the 7 multiple-choice reading comprehension items following one reading passage on the 2011 Progress in International Reading Literacy (PIRLS) 2011 test. Item responses are scored as "1" (correct) or "0" (incorrect or missing).

 # Import Excel .csv file

```
> PIRLS_scored <- read.table("C:/Users/username/Desktop/PIRLS 2011_Morocco_MC reading items_scored.csv", header = TRUE, sep = ",")

# The equivalent file pathname for Mac would be "/Users/username/Desktop/PIRLS 2011_Morocco_MC reading items_scored.csv"

# Check that data imported correctly

> dim(PIRLS_scored)

> View(PIRLS_scored)
```

 # Create "sumscore" column

```
> PIRLS_scored$sumscore <- rowSums(PIRLS_scored)

# Use 'attach()' function to recognize columns in the PIRLS_scored dataframe as variables

> attach(PIRLS_scored)

# Compute descriptive statistics for student sum scores at 25th, 50th and 75th percentiles, and mean

> summary(sumscore)

# Produce unformatted frequency table of student sum scores

> table(sumscore)
```

 # Experiment with transforming (scaling) scores.

```
# Compute percentile ranks for reform teaching practices sum score [compile a function, then use the function to compute scores], and verify the format of the new variable. Then append the percentile rank score column (pctilerank) to the "PIRLS_scored" dataframe as a new column called "pctilerank"

> perc.rank <- function(x) floor(100*(rank(x)/length(x)))
> pctilerank <- perc.rank(sumscore)

> PIRLS_scored["pctilerank"] = pctilerank

# Activate the 'CTT' package (or install, if needed, using install.packages("CTT")

> require(CTT)
```

```
# Transform the sumscore variable into z scores (Mean = 0, Standard Deviation = 1) using
the “score.transform” function.  The output object “stdscore0_1” contains the re-scaled
scores (“new.scores”).  Append it to the existing PIRLS_scored dataframe.

> stdscore0_1 <- data.frame(score.transform(sumscore, mu.new = 0, sd.new = 1, normalize =
FALSE))

> PIRLS_scored["stdscore0_1"] = stdscore0_1$new.scores


# Transform sum scores into standardized scores with Mean = 40, Standard Deviation = 15
(or other chosen scale).  Append it to the existing PIRLS_scored dataframe.

> stdscore40_15 <- data.frame(score.transform(sumscore, mu.new = 40, sd.new = 15,
normalize = FALSE))

> PIRLS_scored["stdscore40_15"] = stdscore40_15$new.scores


# Normalize sum scores.  Append the new score variable to the existing PIRLS_scored
dataframe.

> normalscore0_1 <- data.frame(score.transform(sumscore, mu.new = 0, sd.new = 1,
normalize = TRUE))

> PIRLS_scored["normalscore0_1"] = normalscore0_1$new.scores


# Create dataframe

> scores <- data.frame(PIRLS_scored$sumscore, PIRLS_scored$pctilerank,
PIRLS_scored$stdscore0_1, PIRLS_scored$stdscore40_15, PIRLS_scored$normalscore0_1)

> colnames(scores) <- c("Sum_score", "Percentile_rank", "Standardized_score_0_1",
"Standardized_score_40_15", "Normalized_score_0_1")


# Compute quartiles, mean, and Pearson correlations among scores

> summary(scores)

> cor(scores)


# Plot score histograms [first use commands: install.packages("ggplot2") and/or
install.packages("reshape2") if needed]

> require(ggplot2)

> require(reshape2)

> ggplot(data = melt(scores), mapping = aes(x = value)) + geom_histogram(bins = 8) +
facet_wrap(~variable, scales = 'free_x')
```