Analyzing shape, accuracy, and precison of shooting results with `shotGroups`

Daniel Wollschläger*

September 7, 2015

Contents

*Email: dwoll@kuci.org

1 Introduction

The `shotGroups` package adds functionality to the open source statistical environment R (R Development Core Team, 2014a).[1] It provides functions to read in, plot, statistically describe, analyze, and compare shooting data with respect to group shape, precision, and accuracy. This includes graphical methods, descriptive statistics, and inference tests using standard, but also non-parametric and robust statistical techniques. The data can be imported from files produced by OnTarget PC and OnTarget TDS (Block, 2014), Taran (Trofimov, 2015), or from custom data files in text format with a similar structure.

The package includes limited support for the analysis of three-dimensional data (see sections 3.2.1, 3.2.2),

Use `help(package="shotGroups")` for a list of all functions and links to the detailed help pages with information on options, usage and output.

For a `shiny`-based (RStudio Inc., 2014) web application that implements most of the functionality of `shotGroups`, see `http://dwoll.shinyapps.io/shotGroupsApp/`.

2 Analyzing bullet hole data

Analyzing shot groups usually takes the following steps:

- Read in data (section 2.1)
- Perform either a comprehensive numerical as well as graphical analysis of a group's shape, location (accuracy), and spread (precision) with `analyzeGroup()` (section 2.2) ...
- ...or analyze these aspects of a group separately with `groupShape()` (section 2.3), `groupSpread()` (section 2.4), `groupLocation()` (section 2.5)
- Numerically and visually compare different groups in terms of their shape, location (accuracy), and spread (precision) with `compareGroups()` (section 2.6)
- Use additional utility functions (section 3) to individually explore different aspects of a given group

Grubbs (1964b) and `http://ballistipedia.com/` are good sources for statistical methods for analyzing shot groups.

2.1 Reading in data

To import data into R, it should be saved as a text file with the following format:

- The file should have one row for each shot, and one column for each coordinate as well as for any other variable such as distance to target, point-of-aim coordinates.
- Columns should be separated by commas, tabs or other whitespace. This type of text file can be exported from OnTarget PC/TDS, or from a spreadsheet application like Excel or Calc.

[1] For an introduction to R, see Dalgaard (2008), TryR (`http://tryr.codeschool.com/`) or Quick-R (`http://www.statmethods.net/`).

- The file needs a header in the first line giving the variable names, and should contain at least the coordinates of points of impact, either with variable names `Point.X`, `Point.Y` or just `X`, `Y`.
- For several analysis functions, the following additional variables are useful: `Group` (group number), `Distance` (distance to target), and `Aim.X`, `Aim.Y` (point of aim). If these variables are missing, default values are assumed with a warning.
- If you have output files from OnTarget PC/TDS, you can read multiple files with `readDataOT1()` (for OnTarget PC v1.*), or with `readDataOT2()` (for OnTarget PC v2.* and OnTarget TDS v3.*).
- If you have other whitespace or comma-separated text files with the structure outlined above, you can read multiple files with `readDataMisc()`. For three-dimensional data, this function also recognizes variables `Point.Z` or `Z` and `Aim.Z`.
- If your data is saved in some other text file format, consult the help for `read.table()` or the R import/export manual (R Development Core Team, 2014b).

```
library(shotGroups, verbose=FALSE)       # load shotGroups package

## read text files and save to data frame
## not run, we later use data frame provided in package instead
DFgroups <- readDataMisc(fPath="c:/path/to/files",
                         fNames=c("series1.dat", "series2.dat"))
```

By default, OnTarget's "Export Point Data" places the origin of the coordinate system in the top-left corner. This can be taken into account by correctly setting the option `xyTopLeft` in functions `analyzeGroup()` (section 2.2), `compareGroups` (section 2.6), and `drawGroup()` (section 3.4). In OnTarget TDS, the orientation of the y-axis can be changed by checking the box "Tools → Options → Options tab → Data Export → Invert Y-Axis on Export". If groups appear to be upside-down, `xyTopLeft` is the setting to change.

When analyzing different aspects of a group separately using `groupShape()` (section 2.3), `groupSpread()` (section 2.4), and `groupLocation()` (section 2.5), the scatterplots will be upside-down if the default option of OnTarget was used.

2.2 Performing a combined analysis

`analyzeGroup()`: This function is a convencience wrapper for the functions presented in sections 2.3, 2.4, and 2.5. It analyzes a group's shape, precision, and accuracy in one go, and collects the results.

```
library(shotGroups, verbose=FALSE)       # load shotGroups package
analyzeGroup(DFtalon, conversion="m2mm")

## output not shown, see following sections for results
```

2.3 Analyzing group shape

`groupShape()`: Assess (multivariate) normality, identify outliers and get a sense for the shape of the bivariate distribution.

Reported statistical parameters and tests:

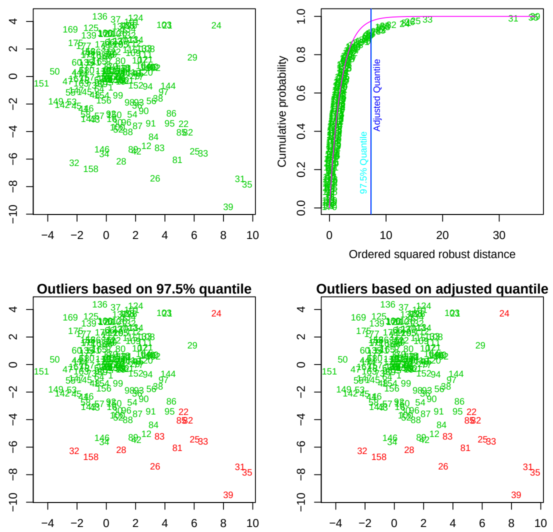
- Correlation matrix including a robust estimate using the MCD method (from package `robustbase`; Rousseeuw et al., 2014)
- Outlier identification: Either using squared robust Mahalanobis distances and adjusted quantiles from the χ^2-distribution, or using robust principal components analysis (PCA) with options to tune the sensitivity (from package `mvoutlier`; Filzmoser & Gschwandtner, 2014)
- Shapiro-Wilk normality tests for the distribution of x- and y-coordinates. For more than 5000 observations, the drop-in Kolmogorov-Smirnov-test is reported instead.
- Energy test for bivariate normality of (x, y)-coordinates (from package `energy`; Rizzo & Szekely, 2014)

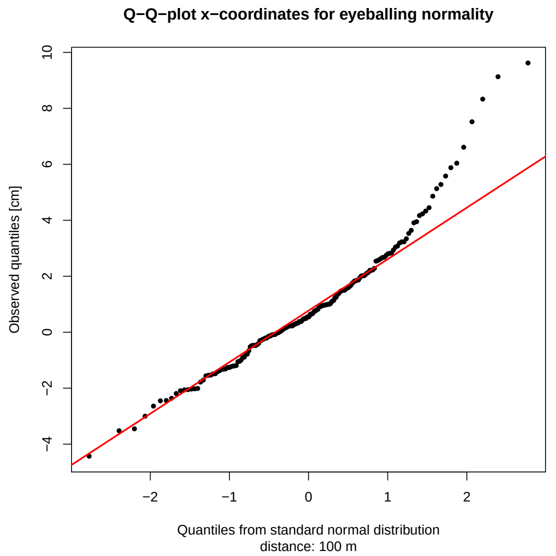
Plots:

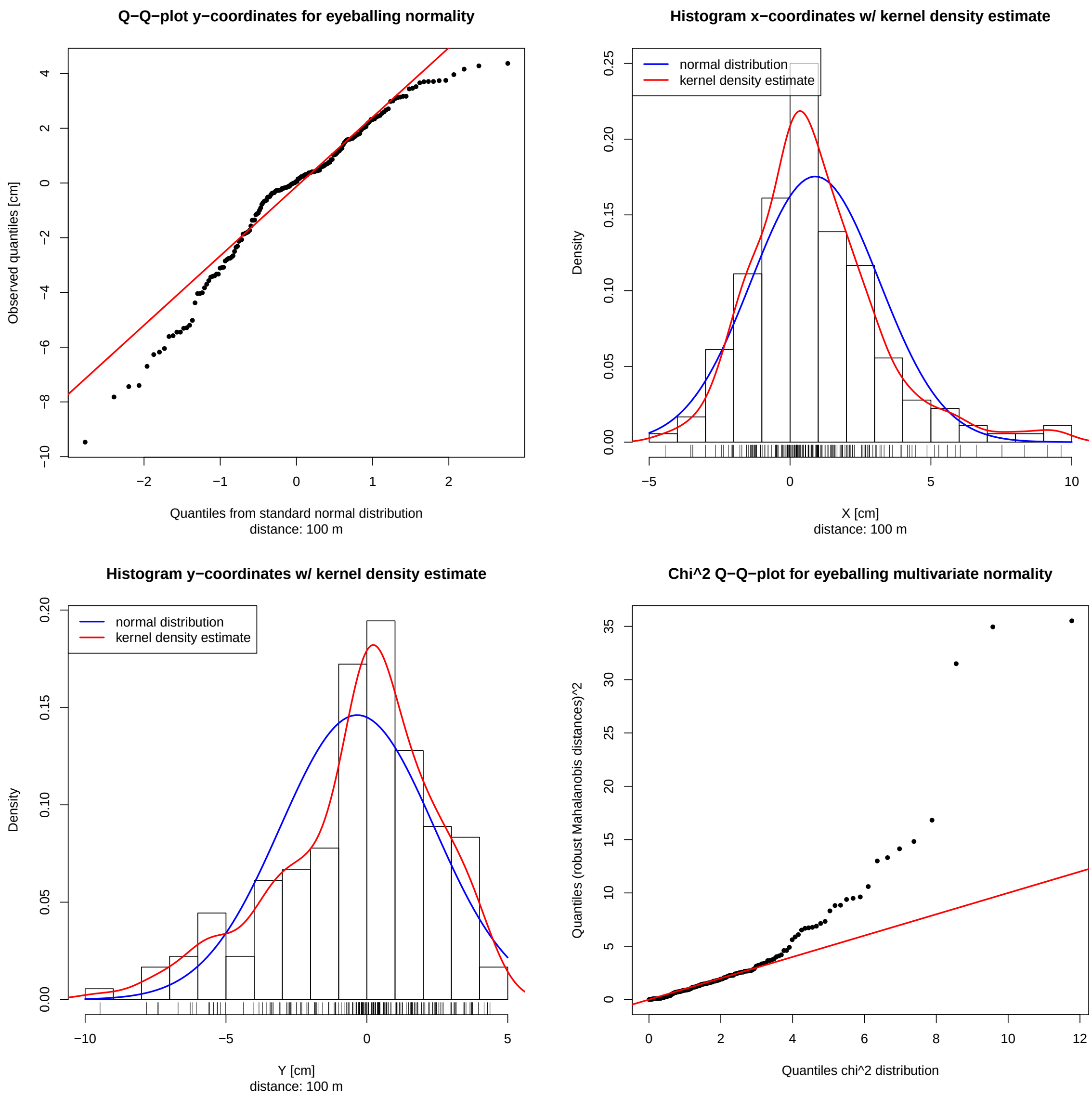
- Combined plot for multivariate outlier identification using squared robust Mahalanobis distances and adjusted quantiles from the χ^2-distribution (from package `mvoutlier`)
- χ^2 QQ-plot of squared robust Mahalanobis distances to group center for eyeballing multivariate normality of (x, y)-coordinates
- Heatmap of a non-parametric 2D-kernel density estimate for the (x, y)-coordinates (from package `KernSmooth`; Wand, 2013) together with robust group center and robust error ellipse
- QQ-plots of x- and y-coordinates for eyeballing normality
- Histogram of x- and y-coordinates including a fitted normal distribution as well as a non-parametric kernel density estimate

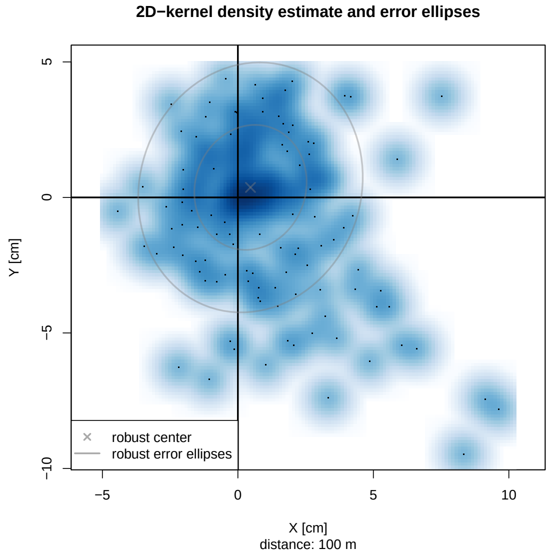
```
library(shotGroups, verbose=FALSE)       # load shotGroups package
groupShape(DFtalon, bandW=0.4, outlier="mcd")

$corXY
        x       y
x  1.0000 -0.2931
y -0.2931  1.0000

$corXYrob
        x       y
x 1.00000 0.08223
y 0.08223 1.00000

$Outliers
 [1]  22  24  25  26  28  31  32  33  35  39  81  82  83  85 158
```

```
$ShapiroX

Shapiro-Wilk normality test

data:  X
W = 0.95, p-value = 3e-06


$ShapiroY

Shapiro-Wilk normality test

data:  Y
W = 0.96, p-value = 2e-05


$multNorm

Energy test of multivariate normality: estimated parameters

data:  x, sample size 180, dimension 2, replicates 1499
E-statistic = 3.7, p-value <2e-16
```






Q−Q−plot y−coordinates for eyeballing normality
Observed quantiles [cm]
Quantiles from standard normal distribution
distance: 100 m
Histogram x−coordinates w/ kernel density estimate
normal distribution
kernel density estimate
Density
X [cm]
distance: 100 m
Histogram y−coordinates w/ kernel density estimate
normal distribution
kernel density estimate
Density
Y [cm]
distance: 100 m
Chi^2 Q−Q−plot for eyeballing multivariate normality
Quantiles (robust Mahalanobis distances)^2
Quantiles chi^2 distribution



2.4 Analyzing group spread – precision

`groupSpread()`: Assess precision using empirical and parametric spread measures with confidence intervals. Where possible, also use the MCD method for a robust estimate of the covariance matrix (from package `robustbase`). Bootstrap confidence intervals are from package `boot` (Canty & Ripley, 2013) with 1499 replications.

Reported statistical parameters and tests:

- (Robust) Standard deviations of x- and y-coordinates together with parametric and bootstrap confidence intervals (in original measurement units, MOA, SMOA, mrad)
- (Robust) Covariance matrix of (x, y)-coordinates
- Empirical mean and median radius as well as estimated Rayleigh precision parameter σ, estimated Rayleigh radial standard deviation RSD $= \sigma\sqrt{\frac{4-\pi}{2}}$, and estimated Rayleigh mean radius MR $= \sigma\sqrt{\frac{\pi}{2}}$ together with parametric and bootstrap confidence intervals for σ, RSD, and MR (in original measurement units, MOA, SMOA, mrad)
- Maximum pairwise distance (center-to-center, = maximum spread, in original measurement units, MOA, SMOA, mrad)
- Width and height of bounding box with length of diagonal and figure of merit as well as of the (oriented) minimum-area bounding box (in original measurement units, MOA, SMOA, mrad)
- Radius for the minimum enclosing circle (in original measurement units, MOA, SMOA, mrad)
- Length of semi-major and semi-minor axis of the (robust) confidence ellipse (in original measurement units, MOA, SMOA, mrad)
- Aspect ratio $\sqrt{\kappa}$ (with condition index κ) and flattening $1 - \frac{1}{\sqrt{\kappa}}$ of the (robust) confidence ellipse as well as the trace and determinant of the covariance matrix
- Estimate for the circular error probable CEP (see section 3.2.1; in original measurement

units, MOA, SMOA, mrad)

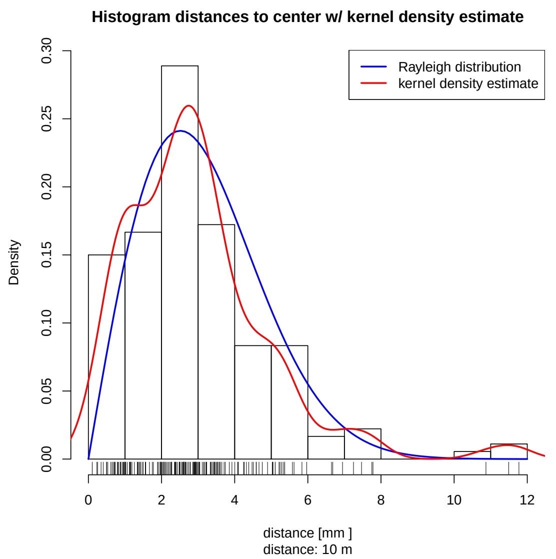
Plots:

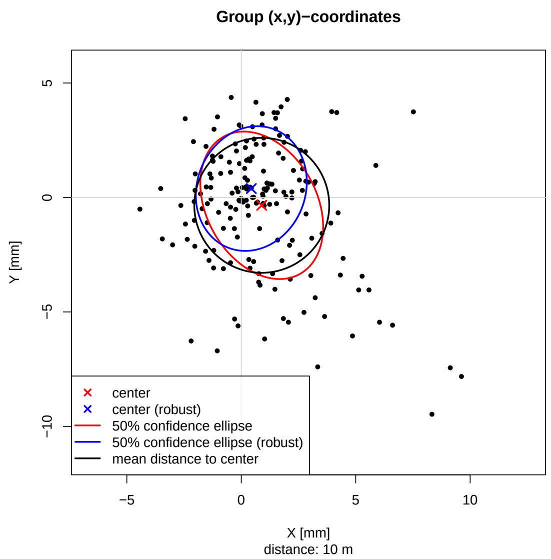
- Scatterplot of the (x, y)-coordinates together with group center, circle with average distance to center, and (robust) confidence ellipse
- Scatterplot of the (x, y)-coordinates together with the bounding box, minimum-area bounding box, minimum enclosing circle, and maximum group spread
- Histogram of distances to group center including a Rayleigh fit and a non-parametric kernel density estimate

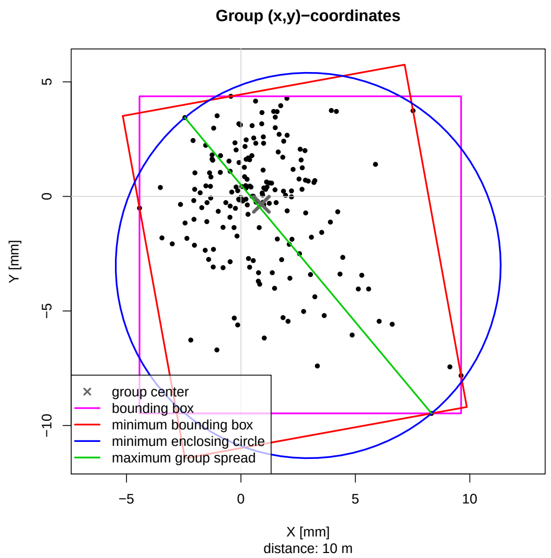
```
library(shotGroups, verbose=FALSE)       # load shotGroups package
groupSpread(DFtalon, CEPtype=c("CorrNormal", "GrubbsPatnaik", "Rayleigh"),
            CEPlevel=0.5, CIlevel=0.95, bootCI="basic", dstTarget=10,
            conversion="m2mm")

$sdXY
          x      y
unit 2.2746 2.7308
MOA  0.7819 0.9388
SMOA 0.8188 0.9831
mrad 0.2275 0.2731

$sdXci
      sdX (   sdX ) sdX basic ( sdX basic )
unit 2.0614 2.5374      1.9738      2.6261
MOA  0.7087 0.8723      0.6785      0.9028
SMOA 0.7421 0.9134      0.7106      0.9454
mrad 0.2061 0.2537      0.1974      0.2626

$sdYci
      sdY (   sdY ) sdY basic ( sdY basic )
unit 2.4749 3.0463      2.4375      3.0583
MOA  0.8508 1.0472      0.8380      1.0514
SMOA 0.8909 1.0967      0.8775      1.1010
mrad 0.2475 0.3046      0.2438      0.3058

$sdXYrob
          x      y
unit 2.0556 2.3094
MOA  0.7066 0.7939
SMOA 0.7400 0.8314
mrad 0.2056 0.2309

$covXY
       x      y
x  5.174 -1.820
```

```
y -1.820  7.457

$covXYrob
       x      y
x 4.2253 0.5141
y 0.5141 5.3332

$distToCtr
       mean median    max  sigma    RSD     MR
unit 2.9486 2.6696 11.772 2.5148 1.6476 3.1519
MOA  1.0137 0.9178  4.047 0.8645 0.5664 1.0835
SMOA 1.0615 0.9611  4.238 0.9053 0.5931 1.1347
mrad 0.2949 0.2670  1.177 0.2515 0.1648 0.3152

$sigmaCI
     sigma ( sigma ) sigma basic ( sigma basic )
unit  2.3433  2.7136        2.2504        2.8030
MOA   0.8056  0.9329        0.7736        0.9636
SMOA  0.8436  0.9769        0.8101        1.0091
mrad  0.2343  0.2714        0.2250        0.2803

$RSDci
      RSD (  RSD ) RSD basic ( RSD basic )
unit 1.5352 1.7778      1.4743      1.8363
MOA  0.5278 0.6112      0.5068      0.6313
SMOA 0.5527 0.6400      0.5308      0.6611
mrad 0.1535 0.1778      0.1474      0.1836

$MRci
       MR (   MR ) MR basic ( MR basic )
unit 2.9369 3.4010     2.8205     3.5130
MOA  1.0096 1.1692     0.9696     1.2077
SMOA 1.0573 1.2244     1.0154     1.2647
mrad 0.2937 0.3401     0.2820     0.3513

$maxPairDist
  unit    MOA   SMOA   mrad
16.819  5.782  6.055  1.682

$groupRect
      width height    FoM   diag
unit 14.050 13.840 13.945 19.722
MOA   4.830  4.758  4.794  6.780
SMOA  5.058  4.982  5.020  7.100
mrad  1.405  1.384  1.394  1.972
```

```
$groupRectMin
      width height    FoM   diag
unit 15.185 12.517 13.851 19.679
MOA   5.220  4.303  4.762  6.765
SMOA  5.466  4.506  4.986  7.084
mrad  1.518  1.252  1.385  1.968

$minCircleRad
  unit    MOA   SMOA   mrad
8.4095 2.8910 3.0274 0.8409

$confEll
     semi-major semi-minor
unit     3.4321     2.4081
MOA      1.1799     0.8278
SMOA     1.2356     0.8669
mrad     0.3432     0.2408

$confEllRob
     semi-major semi-minor
unit     2.7545     2.4062
MOA      0.9469     0.8272
SMOA     0.9916     0.8662
mrad     0.2755     0.2406

$confEllShape
aspectRatio  flattening       trace         det
     1.4253      0.2984     12.6311     35.2687

$confEllShapeRob
aspectRatio  flattening       trace         det
     1.1448      0.1265      9.6123     22.6821

$CEP
     CorrNormal GrubbsPatnaik Rayleigh
unit     2.9013        2.8913   2.9610
MOA      0.9974        0.9939   1.0179
SMOA     1.0445        1.0409   1.0660
mrad     0.2901        0.2891   0.2961
```







2.5 Analyzing group location – accuracy

`groupLocation()`: Assess accuracy of a group using empirical and parametric measures. Where possible, also use the MCD method for a robust estimate of the covariance matrix (from package `robustbase`). Bootstrap confidence intervals are from package `boot` with 1499 replications.

Reported statistical parameters and tests:

- (x, y)-offset of (robust) group center relative to point of aim
- Distance from (robust) group center to point of aim (in original measurement units, MOA, SMOA, mrad)
- Hotelling's T^2-test result for equality of the true group center with point of aim
- Parametric and bootstrap confidence intervals for the true center's x- and y-coordinate

Plots:

- Scatterplot of the (x, y)-coordinates together with (robust) group center.

```
library(shotGroups, verbose=FALSE)       # load shotGroups package
groupLocation(DFtalon, dstTarget=10, conversion="m2cm",
              level=0.95, plots=FALSE, bootCI="basic")

$ctr
      x       y
 0.8947 -0.3432

$ctrRob
     x      y
0.4391 0.3890

$distPOA
  unit    MOA   SMOA   mrad
0.9583 3.2943 3.4498 0.9583

$distPOArob
  unit    MOA   SMOA   mrad
0.5867 2.0168 2.1120 0.5867

$Hotelling
Analysis of Variance Table

             Df Hotelling-Lawley approx F num Df den Df  Pr(>F)
(Intercept)   1            0.156     13.9      2    178 2.5e-06
Residuals   179

$ctrXci
         x (   x )
t     0.5602 1.229
basic 0.5608 1.216

$ctrYci
          y (     y )
t     -0.7448 0.05849
basic -0.7523 0.07032
```

2.6 Comparing groups

`compareGroups()`: Compare two or more groups with regard to their precision and accuracy using empirical measures and statistical tests.

`compareGroups()` requires that the data includes a variable `series` that identifies shot groups. OnTarget PC/TDS' variable `Group` identifies groups just within one file, `series` should number groups also across different original files. When you read in data with `readDataOT1()`, `series` is added automatically (same for `readDataOT2()` and `readDataMisc()`). For data from just one file, you can otherwise copy variable `group` to `series` in a data frame called `shots` with

```
shots$series <- shots$group
```

Reported statistical parameters and tests:

- Group center offset from the respective point of aim
- Distances from group centers to their respective point of aim (in original measurement units, MOA, SMOA, mrad)
- MANOVA result from testing equality of group center offset from the respective point of aim
- Group correlation matrices for the (x, y)-coordinates
- Group standard deviations of the x- and y-coordinates including parametric 95%-confidence intervals (in original measurement units, MOA, SMOA, mrad)
- Average distances from points to their respective group center (in original measurement units, MOA, SMOA, mrad)
- Maximum pairwise distance between points for each group (center-to-center, = maximum spread, in original measurement units, MOA, SMOA, mrad)
- Figure of merit FoM and diagonal of the (oriented) minimum-area bounding box for each group (in original measurement units, MOA, SMOA, mrad)
- Radius of the minimum enclosing circle for each group (in original measurement units, MOA, SMOA, mrad)
- Eestimated Rayleigh parameter σ (precision) for each group (in original measurement units, MOA, SMOA, milliradian)
- Estimated Rayleigh mean radius MR for each group (in original measurement units, MOA, SMOA, milliradian)
- Parametric χ^2 confidence intervals for Rayleigh σ and MR (in original measurement units, MOA, SMOA, milliradian)
- Estimate for the 50% circular error probable (CEP) in each group (see section 3.2.1; in original measurement units, MOA, SMOA, mrad)
- Ansari-Bradley-test results from testing equality of group variances for x- and y-coordinates – when two groups are compared. With more than two groups, the Fligner-Killeen-test is used
- Wilcoxon-Rank-Sum-test (= Mann-Whitney-U-test) result from testing equality of average point distances to their respective group center – when two groups are compared. With more than two groups, the Kruskal-Wallis-test is used

The Ansari-Bradley-, Fligner-Killeen-, Wilcoxon-Rank-Sum-, and Kruskal-Wallis-tests are implemented as permutation tests using the `coin` package (Hothorn, Hornik, van de Wiel, & Zeileis, 2008). The tests for two groups (Ansari-Bradley, Wilcoxon) use the exact permutation distribution, the tests for more than two groups (Fligner-Killeen, Kruskal-Wallis) use the approximate permutation distribution with 9999 random permutations.

Plots:

- Scatterplot showing all groups as well as their respective center and 50%-confidence ellipse
- Scatterplot showing all groups as well as their respective (minimum) bounding box and maximum group spread
- Scatterplot showing all groups as well as their respective minimum enclosing circle and circle with average distance to center
- Boxplot for the distances to group center per group
- Stripchart showing the distances to group center per group together with the estimated Rayleigh mean radius and its confidence interval

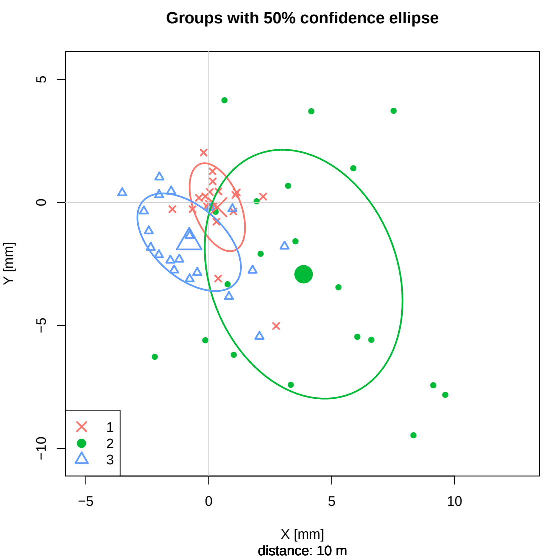
```
library(shotGroups, verbose=FALSE)       # load shotGroups package

## only use first 3 groups of DFtalon
DFsub <- subset(DFtalon, series %in% 1:3)
compareGroups(DFsub, conversion="m2mm")

$ctr
        1      2       3
x  0.3475  3.856 -0.7985
y -0.1910 -2.913 -1.6140

$distPOA
           1      2      3
unit 0.39653 4.8329 1.8007
MOA  0.13632 1.6614 0.6190
SMOA 0.14275 1.7399 0.6483
mrad 0.03965 0.4833 0.1801

$MANOVA
Analysis of Variance Table

            Df Wilks approx F num Df den Df  Pr(>F)
(Intercept)  1 0.676     13.4      2     56 1.7e-05
series       2 0.504     11.4      4    112 7.9e-08
Residuals   57

$corXY
$corXY$`1`
        x       y
x  1.0000 -0.4632
y -0.4632  1.0000

$corXY$`2`
        x       y
x  1.0000 -0.2143
y -0.2143  1.0000
```

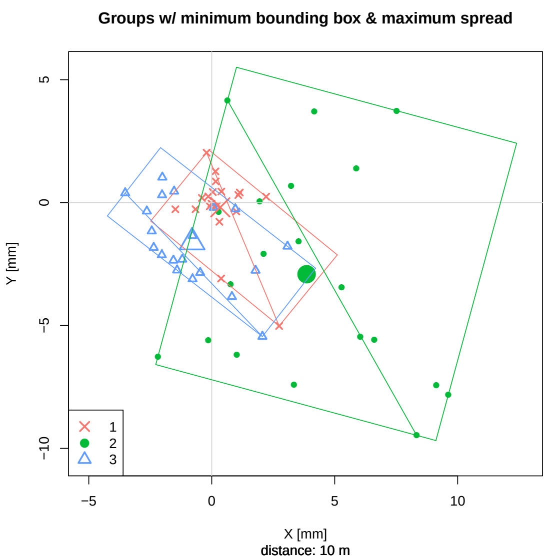
```
$corXY$`3`
        x       y
x  1.0000 -0.5081
y -0.5081  1.0000


$sdXY
$sdXY$`1`
           x      y
unit 0.94037 1.4912
MOA  0.32327 0.5126
SMOA 0.33853 0.5368
mrad 0.09404 0.1491

$sdXY$`2`
          x     y
unit 3.3539 4.220
MOA  1.1530 1.451
SMOA 1.2074 1.519
mrad 0.3354 0.422

$sdXY$`3`
          x      y
unit 1.7549 1.6556
MOA  0.6033 0.5691
SMOA 0.6318 0.5960
mrad 0.1755 0.1656


$sdXYci
$sdXYci$`1`
       sdX (  sdX )  sdY (  sdY )
unit 0.71514 1.3735 1.1340 2.1780
MOA  0.24585 0.4722 0.3898 0.7487
SMOA 0.25745 0.4945 0.4082 0.7841
mrad 0.07151 0.1373 0.1134 0.2178

$sdXYci$`2`
      sdX (  sdX )  sdY (  sdY )
unit 2.5506 4.8986 3.2094 6.1638
MOA  0.8768 1.6840 1.1033 2.1190
SMOA 0.9182 1.7635 1.1554 2.2190
mrad 0.2551 0.4899 0.3209 0.6164

$sdXYci$`3`
      sdX (  sdX )  sdY (  sdY )
```

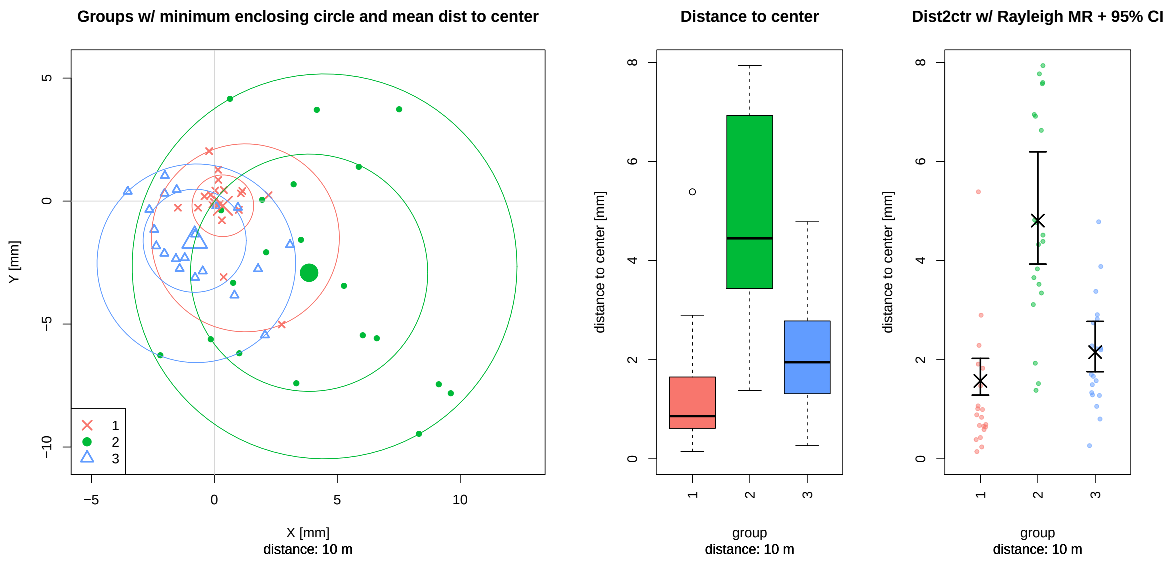
```
unit 1.3346 2.5631 1.2591 2.4181
MOA  0.4588 0.8811 0.4328 0.8313
SMOA 0.4804 0.9227 0.4533 0.8705
mrad 0.1335 0.2563 0.1259 0.2418


$meanDistToCtr
          1      2      3
unit 1.2526 4.8246 2.0961
MOA  0.4306 1.6586 0.7206
SMOA 0.4509 1.7368 0.7546
mrad 0.1253 0.4825 0.2096

$maxPairDist
          1      2      3
unit 7.6423 15.650 8.0773
MOA  2.6272  5.380 2.7768
SMOA 2.7512  5.634 2.9078
mrad 0.7642  1.565 0.8077

$bbFoM
          1      2      3
unit 5.2415 12.170 5.7565
MOA  1.8019  4.184 1.9790
SMOA 1.8869  4.381 2.0724
mrad 0.5241  1.217 0.5757

$bbDiag
          1      2      3
unit 7.7121 17.219 8.7324
MOA  2.6512  5.920 3.0020
SMOA 2.7764  6.199 3.1437
mrad 0.7712  1.722 0.8732

$minCircleRad
          1      2      3
unit 3.8212 7.8248 4.0386
MOA  1.3136 2.6900 1.3884
SMOA 1.3756 2.8169 1.4539
mrad 0.3821 0.7825 0.4039

$sigma
          1      2      3
unit 1.2548 3.8369 1.7172
MOA  0.4314 1.3190 0.5903
SMOA 0.4517 1.3813 0.6182
```

```
mrad 0.1255 0.3837 0.1717

$MR
          1      2      3
unit 1.5727 4.8088 2.1522
MOA  0.5406 1.6531 0.7399
SMOA 0.5662 1.7312 0.7748
mrad 0.1573 0.4809 0.2152

$sigmaMRci
$sigmaMRci$`1`
     sigma ( sigma )   MR (   MR )
unit  1.0255  1.6172 1.2852 2.0268
MOA   0.3525  0.5559 0.4418 0.6968
SMOA  0.3692  0.5822 0.4627 0.7296
mrad  0.1025  0.1617 0.1285 0.2027

$sigmaMRci$`2`
     sigma ( sigma )  MR (   MR )
unit  3.1357  4.9449 3.930 6.1975
MOA   1.0780  1.6999 1.351 2.1305
SMOA  1.1288  1.7801 1.415 2.2311
mrad  0.3136  0.4945 0.393 0.6197

$sigmaMRci$`3`
     sigma ( sigma )   MR (   MR )
unit  1.4034  2.2131 1.7589 2.7737
MOA   0.4824  0.7608 0.6047 0.9535
SMOA  0.5052  0.7967 0.6332 0.9985
mrad  0.1403  0.2213 0.1759 0.2774


$CEP
          1      2      3
unit 1.3724 4.4169 1.9169
MOA  0.4718 1.5184 0.6590
SMOA 0.4941 1.5901 0.6901
mrad 0.1372 0.4417 0.1917

$FlignerX

Approximative Fligner-Killeen Test

data:  x by series (1, 2, 3)
chi-squared = 18, p-value <2e-16
```

```
$FlignerY

Approximative Fligner-Killeen Test

data:  y by series (1, 2, 3)
chi-squared = 21, p-value <2e-16


$Kruskal

Approximative Kruskal-Wallis Test

data:  dstCtr by series (1, 2, 3)
chi-squared = 30, p-value <2e-16
```







3 Additional functionality

The `shotGroups` package also provides a number of utility functions that can be used separately to . . .

- calculate individual descriptive precision measures (section 3.1)
- estimate hit probabilities: either get the region that is expected to contain a certain fraction of shots, or get the estimated fraction of shots expected to be within a given region (section 3.2)
- plot a group to scale on a target background and add precision indicators (section 3.4)
- simulate the ring count for a given group, bullet diameter, and target type (section 3.5)
- convert between absolute and angular size units MOA, SMOA, and mrad (section 3.6)
- try an analysis on collections of empirical data included in the package (section 3.7)

3.1 Descriptive precision measures

The following functions can be used to calculate precision measures that summarize some feature of the group's geometry. Section 3.4 illustrates how to add these precision indicators to a plot of the group.

- `getBoundingBox()`: Calculates the vertices, length of diagonal, and figure of merit (FoM) of the axis-aligned bounding box. This is the smallest rectangle that contains all points (bullet hole centers), and has edges parallel to the x- and y-axes.
- `getMinBBox()`: Calculates the vertices, length of diagonal, and figure of merit (FoM) of the minimum-area bounding box. This is the smallest, possibly oriented rectangle that contains all points (bullet hole centers). Uses the rotating calipers algorithm (Toussaint, 1983).
- `getMinCircle()`: Calculates center and radius of the minimum enclosing circle. This is

the smallest circle that contains all points (bullet hole centers). Uses the Skyum algorithm (Skyum, 1991).

- `getDistToCtr()`: Calculates the distances of a set of points to their center. The mean or median can then be taken as a precision measure.
- `getMaxPairDist()`: Calculates the maximum of all pairwise distances between points.

```
library(shotGroups, verbose=FALSE)        # load shotGroups package
getBoundingBox(DFtalon)                   # axis-aligned bounding box

$pts
  xleft ybottom  xright    ytop
  -4.43   -4.37    9.62    9.47

$width
[1] 14.05

$height
[1] 13.84

$FoM
[1] 13.95

$diag
[1] 19.72

getMinBBox(DFtalon)                       # minimum-area bounding box

$pts
          x      y
[1,] -2.447 11.428
[2,] -5.161 -3.512
[3,]  7.155 -5.750
[4,]  9.869  9.190

$width
[1] 15.18

$height
[1] 12.52

$FoM
[1] 13.85

$diag
[1] 19.68
```

```
$angle
   y
79.7

getMinCircle(DFtalon)                    # minimum enclosing circle

$ctr
[1] 2.940 3.015

$rad
[1] 8.409

getMaxPairDist(DFtalon)                  # maximum pairwise distance

$d
[1] 16.82

$idx
[1] 169  39
```

3.2 Estimating hit probability

Beyond calculating decriptive/geometric precision measures, `shotGroups` also includes functions that provide inferential statistics to estimate hit probabilities.

- Section 3.2.1 shows how to estimate the circular, spherical or elliptical region that is expected to contain a given fraction of shots.
- Section 3.2.2 describes how to estimate the fraction of shots expected to be within a given distance to the true group center.
- Section 3.2.3 covers the extrapolation of hit probabilities to different distances other than the one a group was actually shot at.

3.2.1 Region for a given hit probability: CEP, SEP and confidence ellipse

The following functions estimate the region that is expected to contain a given fraction of shots (bullet hole centers) under different assumptions. The given fraction of shots is the same as the probability for one shot to lie within the calculated region. The functions can use the MCD method for a robust estimate of the group center and covariance matrix (from package `robustbase`).

See section 3.2.4 for more references to relevant literature, and section 3.3 for a discussion of the different distributions of radial error mentioned below.

- `getCEP()`: Calculates estimates for the Circular Error Probable CEP. For three-dimensional data, the Spherical Error Probable SEP is returned. The CEP/SEP estimate is the radius of the circle/sphere around the point of aim (POA) that is expected to cover a certain fraction of points. If systematic accuracy bias is ignored, the POA is is assumed to coincide with the true group center. If systematic accuracy bias is taken into account, the POA is in the origin 0, possibly offset from the true group center. The following estimates are available:

- `CorrNormal`: If systematic accuracy bias is ignored, and for two-dimensional data, this estimate is based on the Hoyt distribution for radial error in correlated bivariate normal variables re-written in polar coordinates (radius and angle; Hoyt, 1947; Paris, 2009a, 2009b). If systematic accuracy bias is taken into account, package `CompQuadForm` (Duchesne & Lafaye de Micheaux, 2010) is used to calculate the cdf of radial error using numerical integration of the multivariate normal distribution over an offset disc (DiDonato & Jarnagin, 1961a; Evans, Govindarajulu, & Barthoulot, 1985) or sphere (DiDonato, 1988). The `CorrNormal` estimate is available for all probability levels
- `GrubbsPearson`: The Grubbs-Pearson estimate (Grubbs, 1964a) is based on the Pearson three-moment central χ^2-approximation (Imhof, 1961; Pearson, 1959) of the cumulative distribution function of radial error in bivariate normal variables. Shot coordinates may be correlated and have unequal variances. The eigenvalues of the covariance matrix of coordinates are used as variance estimates since they are the variances of the principal components (the PCA-rotated = decorrelated data). For probabilities ≥ 0.25, the approximation is very close to the true cumulative distribution function (cdf) used in `CorrNormal` – but easier to calculate. For probabilities < 0.25 and some distribution shapes, the approximation can diverge from the true cdf. The Grubbs-Pearson estimate is available for all probability levels, and generalizes to three dimensions.
- `GrubbsPatnaik`: The Grubbs-Patnaik estimate (Grubbs, 1964a) differs from the Grubbs-Pearson estimate insofar as it is based on the Patnaik (1949) two-moment central χ^2-approximation of the true cumulative distribution function of radial error. For probabilities < 0.5 and some distribution shapes, the approximation can diverge from the true cdf.
- `GrubbsLiu`: The Grubbs-Liu estimate was not proposed by Grubbs but follows the same principle as his original estimates. It differs from them insofar as it is based on the Liu, Tang, and Zhang (2009) four-moment non-central χ^2-approximation of the true cumulative distribution function of radial error. For `accuracy=FALSE`, it is identical to `GrubbsPearson`.
- `Rayleigh`: If systematic accuracy bias is ignored, and for two-dimensional data, this estimate uses the Rayleigh distribution (H. P. Singh, 1992). It is valid for uncorrelated bivariate normal coordinates with equal variances. For `accuracy=FALSE` and three-dimensional data, the Maxwell-Boltzmann distribution is used. For `accuracy=TRUE` and two-dimensional data, the estimate uses the Rice distribution. For `accuracy=TRUE` and three-dimensional data, it is based on the offset sphere probabilities for the multivariate normal distribution set to have equal variances. This estimate is available for all probability levels.
- `Ethridge`: The Ethridge estimate (Ethridge, 1983) is not based on the assumption of multivariate normality of shot coordinates but uses a robust unbiased estimator for the median radius (Hogg, 1967). The Ethridge estimate is also documented in

Puhek (1992).[2] This estimate can only be reported for probability 0.5 but generalizes to three dimensions.

  - `RAND`: The modified RAND R-234 estimate (RAND Corporation, 1952) is a weighted sum of the square root of the eigenvalues of the covariance matrix, that is of the standard deviations of the two principal components. The bias correction with `accuracy=TRUE` is based on a cubic regression fit to tabulated data (Pesapane & Irvine, 1977; Puhek, 1992). This estimate can only be reported for probability 0.5 and does not generalize to three dimensions.

- `getConfEll()`: Calculates the confidence ellipse for the true mean of the distribution under the assumption of multivariate normality of shot coordinates. The coordinates may be correlated and have unequal variances. The confidence ellipse gives the iso-probability contour, the points on its rim all have the same Mahalanobis distance to the center. The result also includes the ellipse based on a robust estimate for the covariance matrix of the coordinates using the MCD algorithm (from package `robustbase`). The confidence ellipse generalizes to three-dimensional data.

```
## circular error probable
getCEP(DFscar17, type=c("GrubbsPatnaik", "Rayleigh"), CEPlevel=0.5,
       dstTarget=100, conversion="yd2in")

$CEP
     GrubbsPatnaik Rayleigh
unit        0.8415   0.8751
MOA         0.8036   0.8357
SMOA        0.8415   0.8751
mrad        0.2337   0.2431

$ellShape
aspectRatio  flattening
     1.4503      0.3105

$ctr
    x     y
2.599 2.299

## confidence ellipse
getConfEll(DFscar17, level=0.95,
           dstTarget=100, conversion="yd2in")

$ctr
    x     y
2.599 2.299
```

[2]Note that the formula for the Hogg weighted location estimate is wrong in Puhek (1992); Tongue (1993); Wang, Yang, Jia, and Wang (2013); Wang, Yang, Yan, Wang, and Song (2014); Williams (1997).

```
$ctrRob
    x     y
2.804 2.283

$cov
        x       y
x  0.4492 -0.1695
y -0.1695  0.6253

$covRob
        x       y
x 0.03677 0.00779
y 0.00779 1.97410

$size
     semi-major semi-minor
unit     2.4900     1.7168
MOA      2.3778     1.6395
SMOA     2.4900     1.7168
mrad     0.6917     0.4769

$sizeRob
     semi-major semi-minor
unit      4.099     0.5592
MOA       3.915     0.5340
SMOA      4.099     0.5592
mrad      1.139     0.1553

$shape
aspectRatio  flattening       trace         det
     1.4503      0.3105      1.0745      0.2522

$shapeRob
aspectRatio  flattening       trace         det
    7.33035     0.86358     2.01087     0.07253

$magFac
[1] 2.918
```

Function `getRayParam()` estimates the Rayleigh distribution's radial precision parameter σ together with its radial standard deviation RSD $= \sigma\sqrt{\frac{4-\pi}{2}}$, and its mean radius MR $= \sigma\sqrt{\frac{\pi}{2}}$, including parametric confidence intervals. Function `getMaxParam()` likewise estimates σ, MR $= \sigma\sqrt{\frac{8}{\pi}}$ and RSD $= \sigma\sqrt{\frac{3\pi-8}{\pi}}$ of the Maxwell-Boltzmann distribution from three-dimensional data.

```
## Rayleigh parameter estimates with 95% confidence interval
getRayParam(DFscar17, level=0.95)

$sigma
  sigma sigCIlo sigCIup
 0.7432  0.5616  1.0991

$RSD
    RSD RSDciLo RSDciUp
 0.4869  0.3679  0.7201

$MR
    MR MRciLo MRciUp
0.9315 0.7039 1.3775

## Maxwell-Boltzmann parameter estimates with 95% confidence interval
xyz <- matrix(rnorm(60), ncol=3)
getMaxParam(xyz, level=0.95)

$sigma
  sigma sigCIlo sigCIup
 0.8697  0.7353  1.0648

$mean
    MR MRciLo MRciUp
 1.388  1.173  1.699

$sd
    RSD RSDciLo RSDciUp
 0.5857  0.4952  0.7171
```

3.2.2 Hit probability for a given region

Given a circle or sphere with radius r around the true mean of the bullet hole distribution, `getHitProb()` estimates the expected fraction of shots that has at most distance r to the group center. The estimated fraction of shots is the same as the estimated probability for one shot to lie in the circle with radius r. The probability can be calculated using the correlated bivariate normal, Grubbs-Pearson χ^2, Grubbs-Patnaik χ^2, Grubbs-Liu χ^2, and Rayleigh distribution as explained in section 3.2.1.

In the example given below, we plug in the results for the 50%-CEP as calculated by `getCEP()` in section 3.2.1 for r, and therefore expect a hit probability of 50%.

```
## for the Grubbs-Patnaik estimate
getHitProb(DFscar17, r=0.8414825, unit="in", doRob=FALSE,
           dstTarget=100, conversion="yd2in", type="GrubbsPatnaik")
```

```
GrubbsPatnaik
          0.5

## for the Rayleigh estimate
getHitProb(DFscar17, r=0.8290354, unit="in", doRob=FALSE,
           dstTarget=100, conversion="yd2in", type="Rayleigh")

Rayleigh
  0.4632
```

Another calculation gives the estimated fraction of shots within a circle with radius 1 MOA.

```
getHitProb(DFscar17, r=1, unit="MOA", doRob=FALSE,
           dstTarget=100, conversion="yd2in", type="CorrNormal")

CorrNormal
    0.6508
```

3.2.3 Extrapolating CEP and confidence ellipse to different distances

Function `getCEP()` returns the radius of the circular error probable (CEP) in absolute and angular size units, as does `getConfEll()` for the size of the confidence ellipse (section 3.2.1). Since angular size measures can be converted back to absolute size for arbitrary distances (3.6.1), it is possible to estimate the absolute size of the CEP and confidence ellipse for distances different than the one a group was actually shot at.

Given an observed group shot at 100 yd, one might, for example, calculate the radius of the circle at 300 m that is expected to contain 50% of the shots. This calculation is highly idealized as it makes the assumption that all influences on precision scale linearly with distance. Under most circumstances, this assumption is invalid. Generally, extrapolating beyond observed data can often be misleading. However, projecting CEP to slightly different distances might still give a sufficient approximation.

```
## 50% circular error probable for group shot at 100yd
CEP100yd <- getCEP(DFscar17, type=c("GrubbsPatnaik", "Rayleigh"),
                   CEPlevel=0.5, dstTarget=100, conversion="yd2in")

## CEP in absolute and angular size units
CEP100yd$CEP

     GrubbsPatnaik Rayleigh
unit        0.8415   0.8751
MOA         0.8036   0.8357
SMOA        0.8415   0.8751
mrad        0.2337   0.2431
```

```
## extract CEP in MOA
CEPmoa <- CEP100yd$CEP["MOA", c("GrubbsPatnaik", "Rayleigh")]

## 50% CEP in inch for the same group extrapolated to 100m
fromMOA(CEPmoa, dst=100, conversion="m2in")

GrubbsPatnaik      Rayleigh
       0.9203        0.9570
```

Given a group shot at 100 yd, one may be interested in the expected fraction of shots within a circular region with radius $r = 1$ inch around the group center at the distance of 100 m (section 3.2.2). To this end, we first convert 1 inch at 100 m to MOA, and then supply the MOA value to `getHitProb()`.

```
## 1 inch at 100 m in MOA
MOA <- getMOA(1, dst=100, conversion="m2in")

getHitProb(DFscar17, r=MOA, unit="MOA", doRob=FALSE,
           dstTarget=100, conversion="yd2in", type="GrubbsPatnaik")

GrubbsPatnaik
       0.5545
```

3.2.4 Literature related to CEP

The literature on the circular error probable (CEP) is extensive and diverse: Applications for CEP are found in areas such as target shooting, missile ballistics, or positional accuracy of navigation and guidance systems like GPS. The statistical foundations in quadratic forms of normal variables are important for analyzing the power of inference tests. The Hoyt and Rayleigh distribution have applications in (wireless) signal processing, the Rice distribution is important for medical imaging techniques like MRI, the Maxwell-Boltzmann distribution comes from the physics of ideal gases.

The following list is by no means intended to be complete. Beware that the quality of the cited articles is not uniformly high. The relevant publications may be roughly categorized into different groups:

- Articles that develop a CEP estimator or the modification of one – e. g., RAND-234 (RAND Corporation, 1952), modified RAND-234 (Pesapane & Irvine, 1977), Grubbs (1964a), Rayleigh (Culpepper, 1978; Saxena & Singh, 2005; H. P. Singh, 1992), Ethridge (1983), Spall and Maryak (1992), correlated bivariate normal (DiDonato & Jarnagin, 1961a; Evans et al., 1985). Some articles focus on the confidence intervals for CEP (DiDonato, 2007; Sathe, Joshi, & Nabar, 1991; Taub & Thomas, 1983b; Thomas, Crigler, Gemmill, & Taub, 1973; Zhang & An, 2012).
- Articles or Master's theses comparing the characteristics of CEP estimators in different scenarios (Blischke & Halpin, 1966; Elder, 1986; Kamat, 1962; McMillan & McMillan,

2008; Moranda, 1959, 1960; Nelson, 1988; Puhek, 1992; Tongue, 1993; Taub & Thomas, 1983a; Wang, Jia, Yang, & Wang, 2013; Wang, Yang, et al., 2013; Wang et al., 2014; Williams, 1997).

- Publications studying the correlated bivariate normal distribution re-written in polar coordinates radius and angle (Chew & Boyce, 1962; Greenwalt & Shultz, 1962; Harter, 1960; Hoover, 1984; Hoyt, 1947). The distribution of the radius is known as the Hoyt (1947) distribution. The closed form expression for its cumulative distribution function has only recently been identified as the symmetric difference between two first-order Marcum Q-functions (Marcum, 1950; Paris, 2009a, 2009b). The latter are special cases of the non-central χ^2-distribution (Nuttall, 1975). The statistical literature on coverage problems in the multivariate normal distribution is reviewed in Guenther and Terragno (1964).
- DiDonato and Jarnagin (1961a, 1961b, 1962a, 1962b); Evans et al. (1985) develop methods to use the correlated bivariate normal distribution for CEP estimation when systematic accuracy bias must be taken into account. This requires integrating the distribution over a disc that is not centered on the true mean of the shot group but on the point of aim. This so-called *offset circle probability* is the probability of a quadratic form of a normal variable (Duchesne & Lafaye de Micheaux, 2010). The exact distribution of quadratic forms is a weighted average of non-central χ^2-distributions and difficult to calculate without numerical tools. Therefore, the Patnaik (1949) two-moment central χ^2-approximation or the Pearson (Imhof, 1961; Pearson, 1959) three-moment central χ^2-approximation are often used. Recently, Liu et al. (2009) proposed a four-moment non-central χ^2-approximation.
- A number of articles present algorithms for the efficient numerical calculation of the Hoyt cumulative distribution function (cdf), as well as for its inverse, the quantile function (DiDonato, 2004, 2007; Pyati, 1993; Shnidman, 1995). Numerical algorithms to efficiently and precisely calculate the distribution of quadratic forms of normal random variables were proposed by Davies (1980); Farebrother (1984, 1990); Imhof (1961); Sheil and O'Muircheartaigh (1977). A comparison and implementation can be found in Duchesne and Lafaye de Micheaux (2010).
- The Spherical Error Probable is developed in DiDonato (1988); N. Singh (1962, 1970).

3.3 Distributions for radial error

`shotGroups` implements several distributions for radial error in multivariate normal variables which apply to different situations. In general, the following functions are available:

- `d⟨Name⟩()`: The probability density function (pdf).
- `p⟨Name⟩()`: The cumulative distribution function (cdf, the integral over the pdf).
- `q⟨Name⟩()`: The quantile function (cdf^{-1}, the inverse of the cdf).
- `r⟨Name⟩()`: The function that generates random numbers from the given distribution.

The following distributions are available, illustrated in figure 1:

- Rayleigh: The radius around the true mean in a bivariate uncorrelated normal random variable with equal variances, re-written in polar coordinates (radius and angle), follows a Rayleigh distribution. Fully defined in closed form.

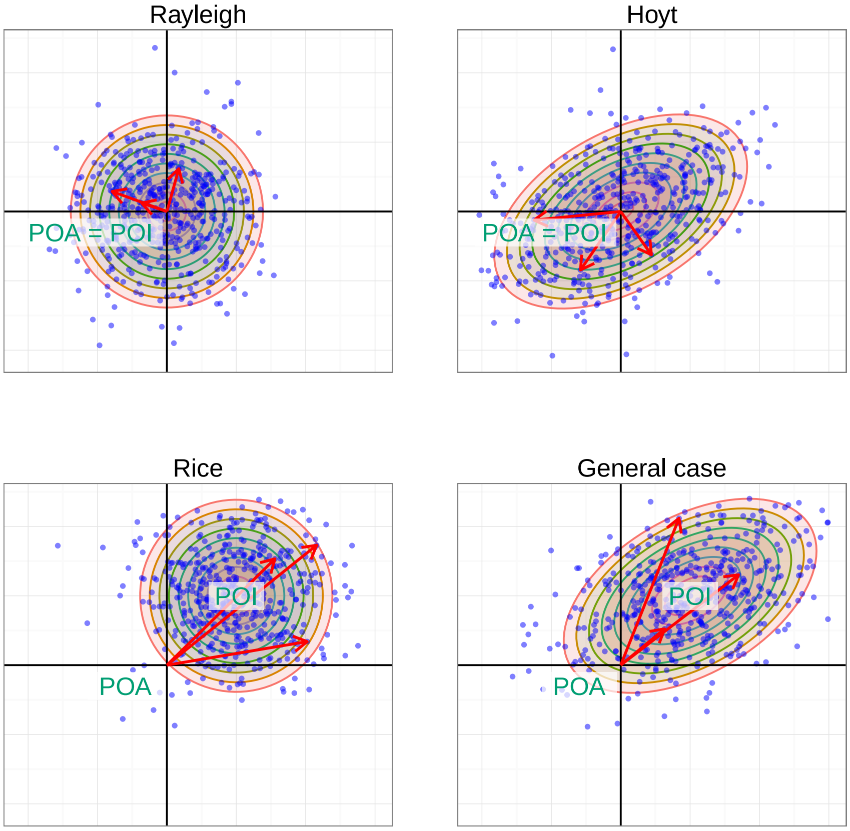


Figure 1: Distribution of radial error (red arrows) for bivariate normal shot distributions with point of aim (POA) and true mean (point of impact, POI).

- Maxwell-Boltzmann: The radius around the true mean in a trivariate uncorrelated normal random variable with equal variances, re-written in polar coordinates (radius, azimuth, elevation), follows a Maxwell-Boltzmann distribution. Fully defined in closed form (using the cdf of the normal distribution).
- Rice: The radius around the origin in a bivariate uncorrelated normal random variable with equal variances and an offset mean, re-written in polar coordinates (radius and angle), follows a Rice distribution. The pdf, cdf, and inverse cdf are defined in closed form with the Marcum Q-function. Reduces to the Rayleigh distribution if the mean has no offset.
- Hoyt: The radius around the true mean in a bivariate correlated normal random variable with unequal variances, re-written in polar coordinates (radius and angle), follows a Hoyt distribution. The pdf and cdf are defined in closed form, numerical root finding is used to find cdf^{-1}. Reduces to the Rayleigh distribution if the correlation is 0 and the variances are equal.
- General case: The distribution of the radius around the origin in a bivariate correlated normal variable with unequal variances and an offset mean. The cdf of radial error is equal to the integral of the bivariate normal distribution over an offset disc. Not defined in

closed form. Numerical methods for quadratic forms provided by package `CompQuadForm` are used to find the pdf and cdf. Numerical root finding is used to find cdf^{-1}. Reduces to the Rice distribution if the correlation is 0 and the variances are equal. Reduces to the Hoyt distribution if the mean has no offset.

```
## probability of staying within 10cm of the point of aim
## Rayleigh distribution
pRayleigh(10, scale=5)

[1] 0.8647

## Rice distribution with offset x=1, y=1
pRice(10, nu=sqrt(2), sigma=5)

[1] 0.8538

## Hoyt distribution - unequal variances
sdX <- 8                              # standard deviation along x
sdY <- 4                              # standard deviation along y
hp  <- getHoytParam(c(sdX^2, sdY^2))  # convert to Hoyt parameters
pHoyt(10, qpar=hp$q, omega=hp$omega)

[1] 0.7332

## general case: unequal variances and offset x=1, y=1
sigma <- cbind(c(52, 20), c(20, 28))  # covariance matrix
pmvnEll(r=10, sigma=sigma, mu=c(1, 1), e=diag(2), x0=c(0, 0))

[1] 0.7251
```

The following examples show how the general correlated normal case encompasses the specialized distributions for radial error. The radial error in each case is 1.5. First, the case of equal variances and no offset mean.

```
## 1D - normal distribution with mean 0 for interval [-1.5, 1.5]
pnorm(1.5, mean=0, sd=2) - pnorm(-1.5, mean=0, sd=2)

[1] 0.5467

pmvnEll(1.5, sigma=4, mu=0, e=1, x0=0)

[1] 0.5467

## 2D - Rayleigh distribution
pRayleigh(1.5, scale=2)
```

```
[1] 0.2452

pmvnEll(1.5, sigma=diag(rep(4, 2)), mu=rep(0, 2), e=diag(2), x0=rep(0, 2))

[1] 0.2452

## 3D - Maxwell-Boltzmann distribution
pMaxwell(1.5, sigma=2)

[1] 0.09504

pmvnEll(1.5, sigma=diag(rep(4, 3)), mu=rep(0, 3), e=diag(3), x0=rep(0, 3))

[1] 0.09504
```

Next, the case of equal variances and an offset mean.

```
## 1D - normal distribution with mean 1 for interval [-1.5, 1.5]
pnorm(1.5, mean=1, sd=2) - pnorm(-1.5, mean=1, sd=2)

[1] 0.4931

pmvnEll(1.5, sigma=4, mu=1, e=1, x0=0)

[1] 0.4931

## 2D - Rice distribution
pRice(1.5, nu=1, sigma=2)

[1] 0.22

pmvnEll(1.5, sigma=diag(c(4, 4)), mu=c(1, 0), e=diag(2), x0=c(0, 0))

[1] 0.22
```

Next, the case of unequal variances and no offset mean.

```
## 2D - Hoyt distribution
sdX <- 4                                # standard deviation along x
sdY <- 2                                # standard deviation along y
hp  <- getHoytParam(c(sdX^2, sdY^2))  # convert to Hoyt parameters
pHoyt(1.5, qpar=hp$q, omega=hp$omega)

[1] 0.1291
```

```r
pmvnEll(1.5, sigma=diag(c(sdX^2, sdY^2)), mu=c(0, 0), e=diag(2), x0=c(0, 0))

[1] 0.1291
```

3.4 Plotting scaled bullet holes on a target background

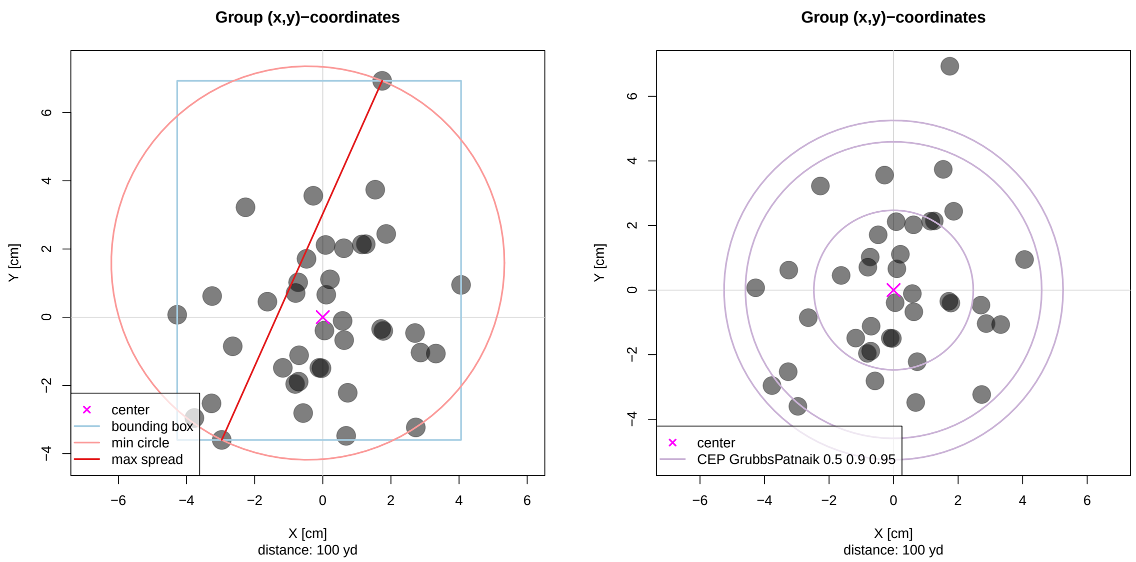
Function `drawGroup()` serves to illustrate a group of bullet holes by drawing the holes to scale on a target background, possibly adding the following features:

- The diagram can be drawn in original measurement units, in absolute size units m, cm, mm, yd, ft, in, or in angular measures MOA, SMOA, mrad.
- A target background can be selected from a number of pre-defined circular target types from different shooting federations (ISSF, DSB, BDS, BDMP, DSU, see `help(targets)`). Targets can also be plotted just by themselves using `drawTarget()`.
- Precision indicators can be added to the plot individually:
  - (Minimum-area) bounding box with diagonal
  - Minimum enclosing circle
  - Maximum group spread
  - Circle with mean distance to group center
  - (Robust) confidence ellipse
  - Circular error probable CEP
- If a known target is supplied, the simulated ring value for each shot can be displayed (see section 3.5)

`drawGroup()` invisibly returns all the information that is shown in the diagram converted to the requested measurement unit. In the following example, the original measurement unit for (x, y)-coordinates was inch, the group is here drawn converted to cm. The second example shows how to plot a CEP estimator for multiple levels.

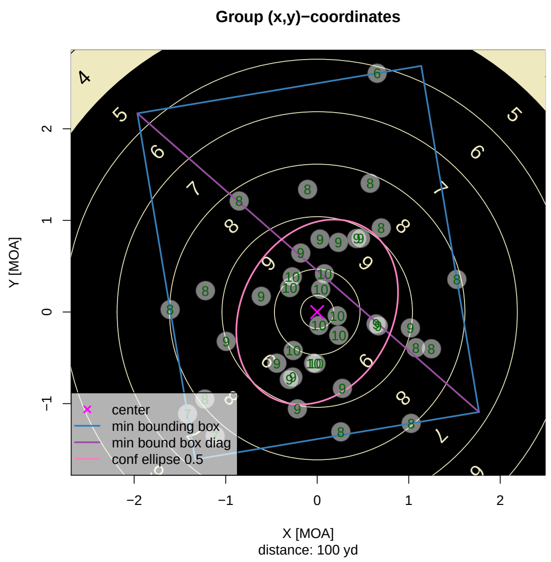
```r
library(shotGroups, verbose=FALSE)       # load shotGroups package
dg1 <- drawGroup(DFcciHV, xyTopLeft=TRUE, bb=TRUE, minCirc=TRUE,
                 maxSpread=TRUE, scaled=TRUE, dstTarget=100,
                 conversion="yd2in", caliber=5.56, unit="cm", alpha=0.5,
                 target=NA)

## minimum enclosing circle parameters in cm
dg1$minCirc

$ctr
[1] -0.4395  1.5890

$rad
[1] 5.771
```

```r
## show Grubbs CEP estimate for 50%, 90% and 95%
dg2 <- drawGroup(DFcciHV, xyTopLeft=TRUE, CEP="GrubbsPatnaik", scaled=TRUE,
                 level=c(0.5, 0.9, 0.95), dstTarget=100, conversion="yd2in",
                 caliber=5.56, unit="cm", alpha=0.5, target=NA)

## Grubbs CEP estimate for 50%, 90% and 95%
dg2$CEP

[1] 2.471 4.592 5.254
```



Now draw the group with coordinates converted to MOA, add the minimum-area bounding box, 50%-confidence ellipse, use the ISSF 100 yd target, and show the ring value for each shot (see section 3.5).

```r
library(shotGroups, verbose=FALSE)        # load shotGroups package
dg3 <- drawGroup(DFcciHV, xyTopLeft=TRUE, bbMin=TRUE, bbDiag=TRUE,
                 confEll=TRUE, ringID=TRUE, level=0.5, scaled=TRUE,
                 dstTarget=100, conversion="yd2in", caliber=5.56, unit="MOA",
                 alpha=0.5, target="ISSF_100yd")

## simulated total ring count with maximum possible
dg3$ringCount

count   max
  351   400
```



3.5 Simulate ring count

Given the (x, y)-coordinates of a group, bullet diameter, and target type with definition of ring diameters, `simRingCount()` calculates a simulated ring count. This is an idealized calculation as it assumes that bullet holes exactly have the bullet diameter, and that rings exactly have the diameter given in the target definition. The count thus ignores the possibility of ragged bullet holes as well as the physical width of the ring markings. The simulated ring count therefore need not be equal to the calculated ring count from the corresponding physical target.

As an example, we simulate the ring count for the `DFscar17` data from shooting a .308 rifle (bullet diameter 7.62 mm) at 100 yd, using the ISSF target made for rifle shooting at 100 m.

```
library(shotGroups, verbose=FALSE)       # load shotGroups package
## simulated ring count and maximum possible with given number of shots
simRingCount(DFscar17, target="ISSF_100m", caliber=7.62, unit="in")

$count
[1] 71

$max
[1] 100

$rings
 [1] 7 6 7 7 7 7 7 7 8 8
Levels: 10 9 8 7 6 5 4 3 2 1 0
```

3.6 Conversion between absolute and angular size units

In addition to absolute length units, group size is often reported in terms of its angular diameter. Angles can be measured equivalently either in degree or in radian. If x is the angular measurement in radian, and φ the angular measurement in degree for the same angle, then $\frac{x}{2\pi} = \frac{\varphi}{360}$ such that conversion between degree and radian is given by $x = \frac{2\pi}{360} \cdot \varphi$ and $\varphi = \frac{360}{2\pi} \cdot x$ (figure 2).

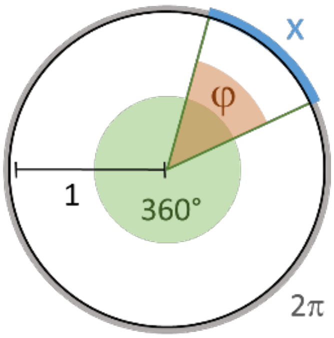


Figure 2: Angle φ (degree) with corresponding arc length x (radian) in the unit circle.

The angular size of an object with absolute size s is its angular diameter at a given distance d. This is the angle α subtended by the object with the line of sight centered on it (figure 3).

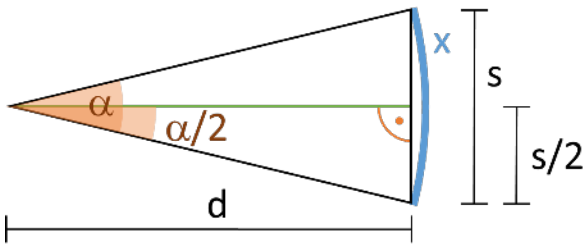


Figure 3: Angular diameter of object with absolute size s at distance to target d. Right triangle formed by d and object of size $s/2$. s corresponds to angle α (degree) and arc length x (radian).

The `shotGroups` package includes functions `getMOA()` and `fromMOA()` to convert from absolute object size to the angular measures MOA, SMOA, mrad and vice versa. The functions need the distance to target d, object sizes s and measurement units for d and s. The option `type` controls which angular measure is returned:

- `type="MOA"`: Convert to/from MOA = minute of angle = arcmin. The circle is divided into 360 degrees, 1 MOA = 1/60 degree such that the circle has $360 \cdot 60 = 21600$ MOA.
- `type="SMOA"`: Convert to/from SMOA = Shooter's MOA = Inches Per Hundred Yards IPHY. 1 inch at 100 yards = 1 SMOA.
- `type="mrad"`: Convert to/from mrad = milliradian = 1/1000 radian. 1 radian is 1 unit of arc length on the unit circle which has a circumference of 2π. The circle is thus divided into $2\pi \cdot 1000 \approx 6283.19$ mrad.
- `type="mil"`: Convert to/from NATO mil. 1 mil = $\frac{2\pi}{6400}$ radian – the circle circumference is divided into 6400 mils.

Function `getDistance()` returns the distance to an object given its absolute size and angular size in MOA, SMOA, mrad, or mil.

3.6.1 Calculating the angular diameter of an object

Figure 3 shows how the angle α subtended by an object of size s at distance d can be calculated from the right triangle with hypotenuse d and cathetus $s/2$: $\tan\left(\frac{\alpha}{2}\right) = \frac{s}{2} \cdot \frac{1}{d}$, therefore $\alpha = 2 \cdot \arctan\left(\frac{s}{2d}\right)$.

Assuming that the argument for $\tan(\cdot)$ and the result from $\arctan(\cdot)$ are in radian, and that distance to target d and object size s are measured in the same unit, this leads to the following formulas for calculating α in MOA, SMOA as well as x in mrad and NATO mil based on d and s:

- Angle α in MOA: $\alpha = 60 \cdot \frac{360}{2\pi} \cdot 2 \cdot \arctan\left(\frac{s}{2d}\right) = \frac{21600}{\pi} \cdot \arctan\left(\frac{s}{2d}\right)$
- Angle α in SMOA: By definition, size $s = 1$ inch at $d = 100$ yards ($= 3600$ inch) is 1 SMOA.

  Conversion factors to/from MOA are $\frac{21600}{\pi} \cdot \arctan\left(\frac{1}{2 \cdot 3600}\right) \approx 0.95493$ (fairly close to $3/\pi$), and $\frac{\pi}{21600} \cdot \frac{1}{\arctan(1/7200)} \approx 1.04720$ (fairly close to $\pi/3$).

  $\alpha = \frac{\pi}{21600} \cdot \frac{1}{\arctan(1/7200)} \cdot \frac{21600}{\pi} \cdot \arctan\left(\frac{s}{2d}\right) = \frac{1}{\arctan(1/7200)} \cdot \arctan\left(\frac{s}{2d}\right)$
- Arc length x in mrad: $x = 1000 \cdot 2 \cdot \arctan\left(\frac{s}{2d}\right) = 2000 \cdot \arctan\left(\frac{s}{2d}\right)$.

  Conversion factors to/from MOA are $\frac{21600}{2000\pi} \approx 3.43775$ and $\frac{2000\pi}{21600} \approx 0.29089$.
- Arc length x in NATO mil: $x = \frac{6400}{\pi} \cdot \arctan\left(\frac{s}{2d}\right)$.

  Conversion factors to/from MOA are $\frac{21600}{6400} = 3.375$ and $\frac{6400}{21600} \approx 0.2962963$.

```
## convert object sizes in cm to MOA, distance in m
getMOA(c(1, 2, 10), dst=100, conversion="m2cm", type="MOA")

[1] 0.3438 0.6875 3.4377
```

Likewise, absolute object size s can be calculated from angular size and distance to target d:

- From angle α in MOA: $s = 2 \cdot d \cdot \tan\left(\frac{(2\pi/360)(\alpha/60)}{2}\right) = 2 \cdot d \cdot \tan\left(\alpha \cdot \frac{\pi}{21600}\right)$
- From angle α in SMOA: $s = \frac{21600}{\pi} \cdot \arctan\left(\frac{1}{7200}\right) \cdot 2 \cdot d \cdot \tan\left(\alpha \cdot \frac{\pi}{21600}\right)$
- From arc length x in mrad: $s = 2 \cdot d \cdot \tan\left(x \cdot \frac{1}{2000}\right)$
- From arc length x in NATO mil: $s = 2 \cdot d \cdot \tan\left(x \cdot \frac{\pi}{6400}\right)$

```
## convert from SMOA to object sizes in inch, distance in yard
fromMOA(c(0.5, 1, 2), dst=100, conversion="yd2in", type="SMOA")

[1] 0.5 1.0 2.0

## convert from object sizes in mm to mrad, distance in m
fromMOA(c(1, 10, 20), dst=100, conversion="m2mm", type="mrad")

[1]  100 1000 2000
```

Conversely, distance to target d can be calculated from absolute object size s and angular size:

- From angle α in MOA: $d = \frac{s}{2} \cdot \frac{1}{\tan(\alpha \cdot \pi/21600)}$
- From angle α in SMOA: $d = \frac{s}{2} \cdot \frac{1}{\tan(\alpha \cdot \arctan(1/7200))}$
- From arc length x in mrad: $d = \frac{s}{2} \cdot \frac{1}{\tan(x/2000)}$
- From arc length x in NATO mil: $d = \frac{s}{2} \cdot \frac{1}{\tan(x \cdot \pi/6400)}$

```
## get distance in yard from object size in inch and angular size in MOA
getDistance(2, angular=5, conversion="yd2in", type="MOA")

[1] 38.2

## get distance in m from object size in mm and angular size in mrad
getDistance(2, angular=0.5, conversion="m2mm", type="mrad")

[1] 4
```

3.6.2 Less accurate calculation of angular size

Sometimes, a slightly different angular size is reported as corresponding to absolute size s at distance d: This is the angle α' subtended by the object if it “sits” on the line of sight (figure 4). α' can be calculated from the right triangle with hypotenuse d and cathetus s: $\tan(\alpha') = \frac{s}{d}$, therefore $\alpha' = \arctan(\frac{s}{d})$.

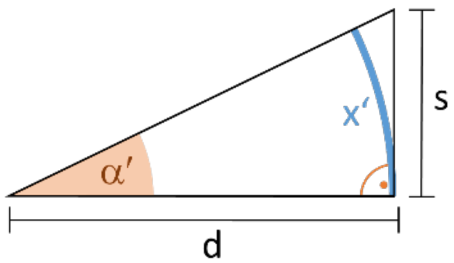


Figure 4: Object “sits” on line of sight: right triangle formed by distance to target d and object of size s. s corresponds to angle α' (degree) and arc length x' (radian).

If size s is small compared to distance d, the difference between the actual angular diameter α and approximate angular size α' is negligible, but it becomes noticeable once s gets bigger in relation to d (figure 5).

3.7 Included data sets

The `shotGroups` package includes a number of empirical data sets with shooting results:

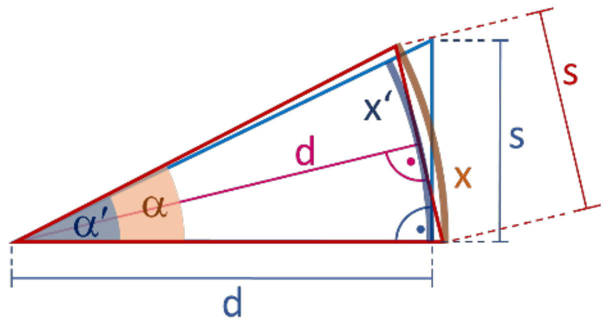


Figure 5: Comparison between actual angular diameter α (red) and the approximate angular size α' (blue) as well as between arc lengths x (red) and x' (blue) corresponding to s at distance d.

- `DF300BLK`: One group of shooting a Noveske AR-15 rifle in 300BLK at 100 yd with factory ammunition (20 observations)[3]
- `DF300BLKhl`: Three groups of shooting a Noveske AR-15 rifle in 300BLK at 100 yd with handloaded ammunition (60 observations, see footnote 3)
- `DFcciHV`: Two groups of shooting a PWS T3 rifle in .22LR at 100 yd (40 observations, see footnote 3)
- `DFcm`: Several groups of shooting a 9x19mm pistol at 25 m (487 observations)
- `DFtalon`: Several groups of shooting a Talon SS air rifle at 10 m (180 observations)[4]
- `DFsavage`: Several groups of shooting a Savage 12 FT/R rifle in .308 Win at distances from 100 to 300 m (180 observations, see footnote 4)
- `DFscar17`: One group of shooting an FN SCAR 17 rifle in .308 Win at 100 yd (10 observations, see footnote 3)

References

Blischke, W. R., & Halpin, A. H. (1966). Asymptotic properties of some estimators of quantiles of circular error. *Journal of the American Statistical Association*, *61* (315), 618–632. URL http://www.jstor.org/stable/2282775

Block, J. (2014). OnTarget TDS [Computer software]. URL http://www.ontargetshooting.com/ (Version 3.81)

Canty, A., & Ripley, B. D. (2013). boot: Bootstrap R (S-Plus) Functions [Computer software]. URL http://CRAN.R-project.org/package=boot (R package version 1.3-9)

Chew, V., & Boyce, R. (1962). Distribution of radial error in bivariate elliptical normal distributions. *Technometrics*, *4* (1), 138–140. URL http://www.jstor.org/stable/1266181

Culpepper, G. A. (1978). *Statistical analysis of radial error in two dimensions* (Tech. Rep.). White Sands Missile Range, NM: U.S. Army Material Test and Evaluation Directorate. URL http://handle.dtic.mil/100.2/ADA059117

[3]Thanks: David Bookstaber http://ballistipedia.com/

[4]Thanks: Charles & Paul McMillan http://statshooting.com/

Dalgaard, P. (2008). *Introductory Statistics with R* (2. ed.). London, UK: Springer. URL http://www.biostat.ku.dk/~pd/ISwR.html

Davies, R. B. (1980). Algorithm AS 155: The distribution of a linear combination of χ^2 random variables. *Journal of the Royal Statistical Society, C*, *29*, 323–333. URL http://www.jstor.org/stable/2346911

DiDonato, A. R. (1988). *Integration of the trivariate normal distribution over an offset spehere and an inverse problem* (Tech. Rep. No. NSWC TR 87-27). Dahlgren, VA: U.S. Naval Surface Weapons Center Dahlgren Division. URL http://www.dtic.mil/dtic/tr/fulltext/u2/a198129.pdf

DiDonato, A. R. (2004). *An inverse of the generalized circular error function* (Tech. Rep. No. NSWCDD/TR-04/43). Dahlgren, VA: U.S. Naval Surface Weapons Center Dahlgren Division. URL http://handle.dtic.mil/100.2/ADA476368

DiDonato, A. R. (2007). *Computation of the Circular Error Probable (CEP) and Confidence Intervals in Bombing Tests* (Tech. Rep. No. NSWCDD/TR-07/13). Dahlgren, VA: U.S. Naval Surface Weapons Center Dahlgren Division. URL http://handle.dtic.mil/100.2/ADA476368

DiDonato, A. R., & Jarnagin, M. P. (1961a). Integration of the general bivariate Gaussian distribution over an offset circle. *Mathematics of Computation*, *15*(76), 375–382. URL http://www.jstor.org/stable/2003026

DiDonato, A. R., & Jarnagin, M. P. (1961b). *Integration of the general bivariate Gaussian distribution over an offset ellipse* (Tech. Rep. No. NWL TR 1710). Dahlgren, VA: U.S. Naval Weapons Laboratory.

DiDonato, A. R., & Jarnagin, M. P. (1962a). A method for computing the circular coverage function. *Mathematics of Computation*, *16*(79), 347–355. URL http://www.jstor.org/stable/2004054

DiDonato, A. R., & Jarnagin, M. P. (1962b). *A method for computing the generalized circular error function and the circular coverage function* (Tech. Rep. No. NWL TR 1786). Dahlgren, VA: U.S. Naval Weapons Laboratory.

Duchesne, P., & Lafaye de Micheaux, P. (2010). Computing the distribution of quadratic forms: Further comparisons between the Liu-Tang-Zhang approximation and exact methods. *Computational Statistics and Data Analysis*, *54*(4), 858–862.

Elder, R. L. (1986). *An examination of circular error probable approximation techniques* (Tech. Rep. No. AFIT/GST/ENS/86M-6). Wright-Patterson AFB, OH: U.S. Air Force Institute of Technology. URL http://handle.dtic.mil/100.2/ADA172498

Ethridge, R. A. (1983). *Robust estimation of circular error probable for small samples* (Tech. Rep. No. ACSC 83-0690). Maxwell AFB, AL: U.S. Air Command and Staff College.

Evans, M. J., Govindarajulu, Z., & Barthoulot, J. (1985). *Estimates of circular error probabilities* (Tech. Rep. No. TR 367). Arlington, VA: U.S. Office of Naval Research. URL http://handle.dtic.mil/100.2/ADA163257

Farebrother, R. W. (1984). Algorithm AS 204: The distribution of a positive linear combination of χ^2 random variables. *Journal of the Royal Statistical Society, C*, *33*, 332–339. URL http://www.jstor.org/stable/2347721

Farebrother, R. W. (1990). Algorithm AS 256: The distribution of a quadratic form in normal variables. *Journal of the Royal Statistical Society, C*, *39*, 294–309. URL http://www.jstor.org/stable/2347778

Filzmoser, P., & Gschwandtner, M. (2014). mvoutlier: Multivariate outlier detection based on robust methods [Computer software]. URL http://CRAN.R-project.org/package=

mvoutlier (R package version 2.0.5)

Greenwalt, C. R., & Shultz, M. E. (1962). *Principles of Error Theory and Cartographic Applications* (Tech. Rep. No. ACIC TR-96). St. Louis, MO: U.S. Aeronautical Chart & Information Center. URL http://earth-info.nga.mil/GandG/publications/tr96.pdf

Grubbs, F. E. (1964a). Approximate circular and noncircular offset probabilities of hitting. *Operations Research*, *12*(1), 51–62. URL http://www.jstor.org/stable/167752

Grubbs, F. E. (1964b). *Statistical measures of accuracy for riflemen and missile engineers.* Ann Arbor, ML: Edwards Brothers. URL http://ballistipedia.com/images/3/33/Statistical_Measures_for_Riflemen_and_Missile_Engineers_-_Grubbs_1964.pdf

Guenther, W. C., & Terragno, P. J. (1964). A review of the literature on a class of coverage problems. *Annals of Mathematical Statistics*, *35*(1), 232–260. URL http://projecteuclid.org/euclid.aoms/1177703747

Harter, H. L. (1960). Circular error probabilities. *Journal of the American Statistical Association*, *55*(292), 723–731. URL http://www.jstor.org/stable/2281595

Hogg, R. V. (1967). Some observations on robust estimation. *Journal of the American Statistical Association*, *62*(320), 1179–1186. URL http://www.jstor.org/stable/2283768

Hoover, W. E. (1984). *Algorithms for confidence circles, and ellipses* (Tech. Rep. No. NOAA TR NOS 107 C&GS 3). Rockville, MD: U.S. National Oceanic and Atmospheric Administration. URL http://www.ngs.noaa.gov/PUBS_LIB/Brunswick/NOAATRNOS107CGS3.pdf

Hothorn, T., Hornik, K., van de Wiel, M. A., & Zeileis, A. (2008). Implementing a Class of Permutation Tests: The coin Package. *Journal of Statistical Software*, *28*(8), 1–23. URL http://www.jstatsoft.org/v28/i08/

Hoyt, R. S. (1947). Probability functions for the modulus and angle of the normal complex variate. *Bell System Technical Journal*, *26*(2), 318–359. URL http://www3.alcatel-lucent.com/bstj/vol26-1947/articles/bstj26-2-318.pdf

Imhof, J. P. (1961). Computing the distribution of quadratic forms in normal variables. *Biometrika*, *48*(3–4), 419–426. URL http://www.jstor.org/stable/2332763

Kamat, A. R. (1962). Some more estimates of circular probable error. *Journal of the American Statistical Association*, *57*(297), 191–195. URL http://www.jstor.org/stable/2282450

Liu, H., Tang, Y., & Zhang, H. H. (2009). A new chi-square approximation to the distribution of non-negative definite quadratic forms in non-central normal variables. *Computational Statistics & Data Analysis*, *53*(4), 853–856.

Marcum, J. I. (1950). *Table of Q functions* (Tech. Rep. No. RAND M-339). Santa Monica, CA: RAND Corporation.

McMillan, C., & McMillan, P. (2008). *Characterizing rifle performance using circular error probable measured via a flatbed scanner.* URL http://statshooting.com/

Moranda, P. B. (1959). Comparison of estimates of circular probable error. *Journal of the American Statistical Association*, *54*(288), 794–780. URL http://www.jstor.org/stable/2282503

Moranda, P. B. (1960). Effect of bias on estimates of the circular probable error. *Journal of the American Statistical Association*, *55*(292), 732–735. URL http://www.jstor.org/stable/2281596

Nelson, W. (1988). *Use of circular error probability in target detection* (Tech. Rep. Nos. ESD-TR-88-109, MTR-10293). Bedford, MA: MITRE Corporation. URL http://

handle.dtic.mil/100.2/ADA199190
Nuttall, A. H. (1975). Some integrals involving the Q-M function. *IEEE Transactions on Information Theory*, *21*(1), 95–96.
Paris, J. F. (2009a). Erratum for "Nakagami-q (Hoyt) distribution function with applications". *Electronics Letters*, *45*(8), 432. URL http://dx.doi.org/10.1049/el.2009.0828
Paris, J. F. (2009b). Nakagami-q (Hoyt) distribution function with applications. *Electronics Letters*, *45*(4), 210–211.
Patnaik, P. B. (1949). The non-central χ^2- and F-distributions and their applications. *Biometrika*, *36*(1–2), 202–232. URL http://www.jstor.org/stable/2332542
Pearson, E. S. (1959). Note on an approximation to the distribution of non-central χ^2. *Biometrika*, *46*(3–4), 364. URL http://www.jstor.org/stable/2333533
Pesapane, J., & Irvine, R. B. (1977). *Derivation of CEP formula to approximate RAND-234 tables* (Tech. Rep.). Offut AFB, NE: U.S. Ballistic Missile Evaluation, HQ SAC.
Puhek, P. (1992). *Sensitivity analysis of circular error probable approximation techniques* (Tech. Rep. No. AFIT/GOR/ENS/92M-23). Wright-Patterson AFB, OH: U.S. Air Force Institute of Technology. URL http://handle.dtic.mil/100.2/ADA248105
Pyati, V. P. (1993). Computation of the circular error probability (CEP) integral. *IEEE Transactions on Aerospace and Electronic Systems*, *29*(3), 1023–1024.
R Development Core Team. (2014a). R: A Language and Environment for Statistical Computing [Computer software manual]. Vienna, Austria. URL http://www.r-project.org/
R Development Core Team. (2014b). R: Data Import/Export [Computer software manual]. Vienna, Austria. URL http://CRAN.R-project.org/doc/manuals/R-data.html
RAND Corporation. (1952). *Offset circle probabilities* (Tech. Rep. No. RAND-234). Santa Monica, CA: RAND Corporation. URL http://www.rand.org/pubs/reports/2008/R234.pdf
Rizzo, M. L., & Szekely, G. J. (2014). energy: E-statistics (energy statistics) [Computer software]. URL http://CRAN.R-project.org/package=energy (R package version 1.6.1)
Rousseeuw, P. J., Croux, C., Todorov, V., Ruckstuhl, A., Salibian-Barrera, M., Verbeke, T., & Maechler, M. (2014). robustbase: Basic Robust Statistics [Computer software]. URL http://CRAN.R-project.org/package=robustbase (R package version 0.90-2)
RStudio Inc. (2014). shiny: Web application framework for r [Computer software]. URL http://CRAN.R-project.org/package=shiny (R package version 0.10.0)
Sathe, Y. S., Joshi, S. M., & Nabar, S. P. (1991). Bounds for circular error probabilities. *Naval Research Logistics (NRL)*, *38*(1), 33–40.
Saxena, S., & Singh, H. P. (2005). Some estimators of the dispersion parameter of a chi-distributed radial error with applications to target analysis. *Austrial Journal of Statistics*, *34*(1), 51–63. URL http://www.stat.tugraz.at/AJS/ausg051/051Saxena&Singh.pdf
Sheil, J., & O'Muircheartaigh, I. (1977). Algorithm AS 106. The distribution of non-negative quadratic forms in normal variables. *Applied Statistics*, *26*(1), 92–98. URL http://www.jstor.org/stable/2346884
Shnidman, D. A. (1995). Efficient computation of the circular error probability (CEP) integral. *IEEE Transactions on Automatic Control*, *40*(8), 1472–1474.
Singh, H. P. (1992). Estimation of Circular Probable Error. *The Indian Journal of Statistics, Series B*, *54*(3), 289–305. URL http://www.jstor.org/stable/25052751

Singh, N. (1962). Spherical probable error. *Nature*, *193*(4815), 605. URL http://www.nature.com/nature/journal/v193/n4815/abs/193605a0.html

Singh, N. (1970). Spherical probable error (SPE) and its estimation. *Metrika*, *15*(1), 149–163.

Skyum, S. (1991). A simple algorithm for computing the smallest enclosing circle. *Information Processing Letters*, *37*(3), 121–125. URL http://ojs.statsbiblioteket.dk/index.php/daimipb/article/viewFile/6704/5821

Spall, J. C., & Maryak, J. L. (1992). A feasible Bayesian estimator of quantiles for projectile accuracy from non-iid data. *Journal of the American Statistical Association*, *87*(419), 676–681. URL http://www.jstor.org/stable/2290205

Taub, A. E., & Thomas, M. A. (1983a). *Comparison of CEP estimators for elliptical normal errors* (Tech. Rep. No. ADP001580). Dahlgren, VA: U.S. Naval Surface Weapons Center Dahlgren Division. URL http://handle.dtic.mil/100.2/ADA153828

Taub, A. E., & Thomas, M. A. (1983b). *Confidence Intervals for CEP When the Errors are Elliptical Normal* (Tech. Rep. No. NSWC/TR-83-205). Dahlgren, VA: U.S. Naval Surface Weapons Center Dahlgren Division. URL http://handle.dtic.mil/100.2/ADA153828

Thomas, M. A., Crigler, J. R., Gemmill, G. W., & Taub, A. E. (1973). *Tolerance limits for the Rayleigh (radial normal) distribution with emphasis on the CEP* (Tech. Rep. No. NWL TR 2946). Dahlgren, VA: U.S. Naval Weapons Laboratory. URL http://handle.dtic.mil/100.2/AD0759989

Tongue, W. L. (1993). *An empirical evaluation of five circular error probable estimation techniques and a method for improving them* (Tech. Rep. No. AFIT/GST/ENS/93M-13). Wright-Patterson AFB, OH: U.S. Air Force Institute of Technology. URL http://handle.dtic.mil/100.2/ADA266528

Toussaint, G. T. (1983). Solving geometric problems with the rotating calipers. In *Proceedings of the 1983 IEEE Mediterranean Electrotechnical Conference.* Athens, Greece: IEEE Computer Society.

Trofimov, A. (2015). Taran [Computer software]. URL http://taran.ptosis.ch/taran.html (Version 1.0)

Wand, M. (2013). KernSmooth: Functions for kernel smoothing for Wand & Jones (1995) [Computer software]. URL http://CRAN.R-project.org/package=KernSmooth (R package version 2.23-10)

Wang, Y., Jia, X. R., Yang, G., & Wang, Y. M. (2013). Comprehensive CEP evaluation method for calculating positioning precision of navigation systems. *Applied Mechanics and Materials*, *341–342*, 955–960.

Wang, Y., Yang, G., Jia, X. R., & Wang, Y. M. (2013). Comprehensive TCEP assessment of methods for calculating MUAV navigation position accuracy based on visual measurement. *Advanced Materials Research*, *765–767*, 2224–2228.

Wang, Y., Yang, G., Yan, D., Wang, Y. M., & Song, X. (2014). Comprehensive assessment algorithm for calculating CEP of positioning accuracy. *Measurement*, *47*(January), 255–263.

Williams, C. E. (1997). *A comparison of circular error probable estimators for small samples* (Tech. Rep. No. AFIT/GOA/ENS/97M-14). Wright-Patterson AFB, OH: U.S. Air Force Institute of Technology. URL http://handle.dtic.mil/100.2/ADA324337

Zhang, J., & An, W. (2012). Assessing circular error probable when the errors are elliptical normal. *Journal of Statistical Computation and Simulation*, *82*(4), 565–586.