Find the Closest Point Along a Line to Another Point

Fan Wang

2021-06-25

Contents

1 Find Nearest

Go to the **RMD**, **R**, **PDF**, or **HTML** version of this file. Go back to fan's REconTools Package, R Code Examples Repository (bookdown site), or Intro Stats with R Repository (bookdown site).

1.1 Neart Point Along a Line Through Orgin to Another Point

We first have X_1 an Y_1, given these, we are able to generate $R = \frac{Y_2}{X_2}$, a ratio. We want to iteratively update X_1 and Y_1, where 1 subscript indicates the first iteration, but we only know the ratio. Think of R as a line through the origin with R as the slope.

We generate X_2 and Y_2 by finding the point along the R as slope origin line that is the closest to the X_1 and Y_1. At the resulting point, R will be respected, and it will differ least in distance to the earlier iteration's X_1 and Y_1 points.

1. The slope of the diagonal line is $-\frac{X_2}{Y_2} = -\frac{1}{R}$
2. The diagonal line must cross X_1 and Y_1, solve for this line's y-intercept
3. Solve for the intersection of the diagonal line and the origin line with R as slope

Implementing step (2):

$$\begin{aligned}
Y_1 &= I - \frac{1}{R} \cdot X_1 \\
I &= Y_1 + \frac{X_1}{R} \\
I &= \frac{Y_1 \cdot R + X_1}{R}
\end{aligned}$$

Implementing step (3):

$$
\begin{aligned}
Y &= \frac{Y_1 \cdot R + X_1}{R} - \frac{1}{R} \cdot X \\
Y &= R \cdot X \\
R \cdot X &= \frac{Y_1 \cdot R + X_1}{R} - \frac{1}{R} \cdot X \\
\left(R + \frac{1}{R}\right) \cdot X &= \frac{Y_1 \cdot R + X_1}{R} \\
\frac{R^2 + 1}{R} \cdot X &= \frac{Y_1 \cdot R + X_1}{R} \\
X &= \frac{Y_1 \cdot R + X_1}{R} \cdot \frac{R}{R^2 + 1}
\end{aligned}
$$

And we have:

$$
\begin{aligned}
X &= \frac{Y_1 \cdot R + X_1}{R^2 + 1} \\
Y &= \frac{Y_1 \cdot R^2 + X_1 \cdot R}{R^2 + 1}
\end{aligned}
$$

Visualize the results:

```r
# Set random parameter Values for X1, Y1, and X2/Y2 ratio
set.seed(3)
fl_x1 <- runif(1) * 10
fl_y1 <- runif(1) * 10
fl_r <- runif(1) * 5

# Diaganol
fl_diag_slope <- -1 / fl_r
fl_diag_yintercept <- (fl_y1 * fl_r + fl_x1) / fl_r

# Closest point
fl_x2 <- (fl_y1 * fl_r + fl_x1) / (fl_r^2 + 1)
fl_y2 <- (fl_y1 * fl_r^2 + fl_x1 * fl_r) / (fl_r^2 + 1)

# Print state
print(paste("x1=", fl_x1, "x2=", fl_x2, "R=", fl_r, sep = " "))
```

```
## [1] "x1= 1.68041526339948 x2= 3.6609038475849 R= 1.92471175687388"
```

```r
print(paste("x2=", fl_x2, "y2=", fl_y2, sep = " "))
```

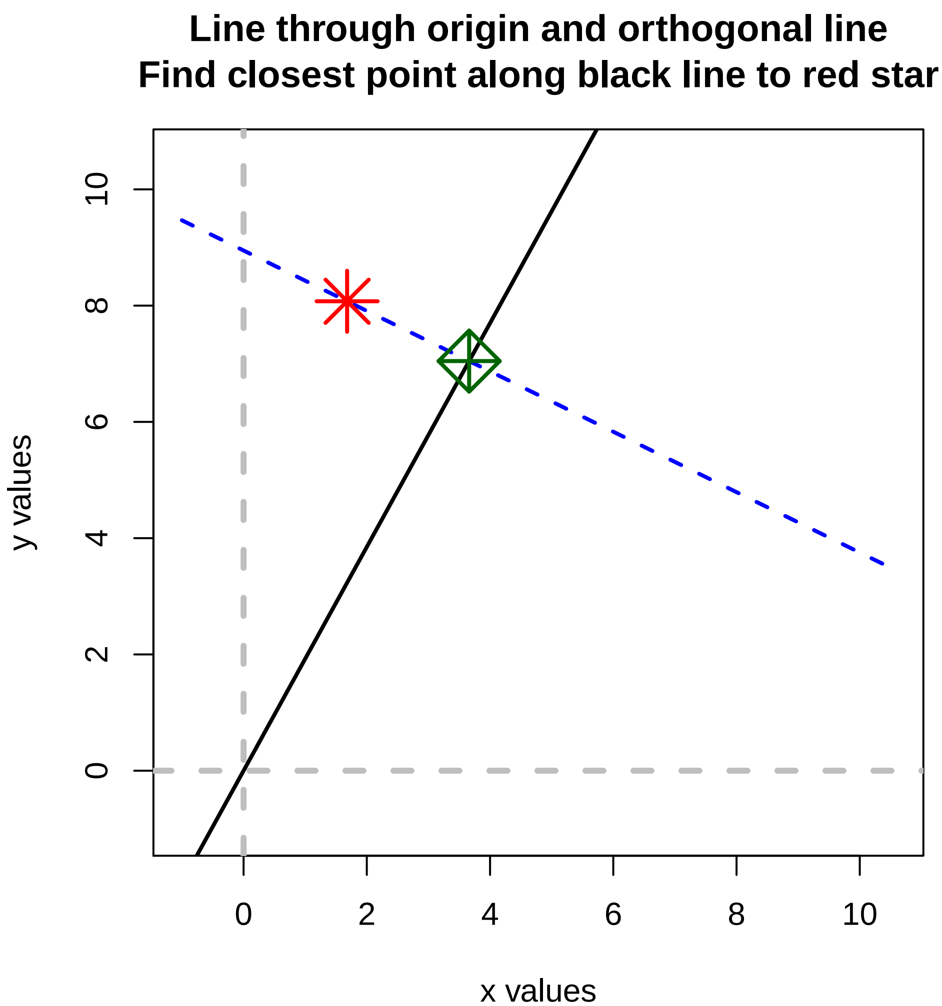
```
## [1] "x2= 3.6609038475849 y2= 7.04618467623146"
```

```r
# X and y lims
ar_xylim <- c(-1, max(fl_x1, fl_y2) * 1.5)

# Visualize
par(mfrow = c(1, 1))
# Line through origin
curve(0 + fl_r * x, ar_xylim[1], ar_xylim[2],
  col = "black", lwd = 2, lty = 1,
  ylim = ar_xylim,
  ylab = "", xlab = ""
)
# Diaganol line
```

```r
curve(fl_diag_yintercept + fl_diag_slope * x,
  add = TRUE,
  col = "blue", lwd = 2, lty = 2,
  ylim = ar_xylim,
  ylab = "", xlab = ""
)
# Point
points(fl_x1, fl_y1,
  add = TRUE,
  pch = 8, col = "red", cex = 3, lwd = 2,
  ylab = "", xlab = ""
)
points(fl_x2, fl_y2,
  add = TRUE,
  pch = 9, col = "darkgreen", cex = 3, lwd = 2,
  ylab = "", xlab = ""
)
# Origin lines
abline(
  v = 0, h = 0,
  col = "gray", lwd = 3, lty = 2
)

# Titles
title(
  main = paste0(
    "Line through origin and orthogonal line\n",
    "Find closest point along black line to red star"
  ),
  sub = paste0(
    "Black line goes through origin,",
    " blue line goes through (x1,y1) and (x2, y2)"
  ),
  xlab = "x values", ylab = "y values"
)
```



Black line goes through origin, blue line goes through (x1,y1) and (x2,