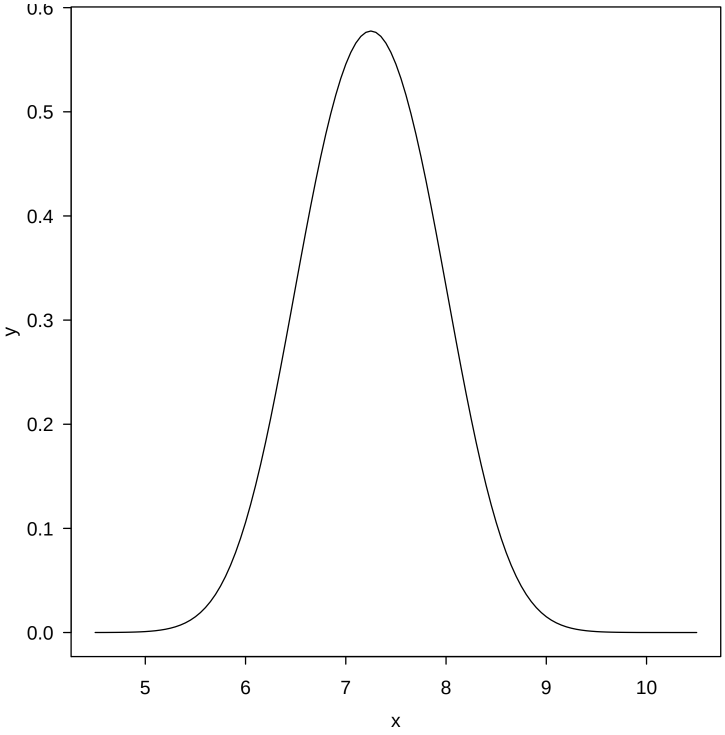
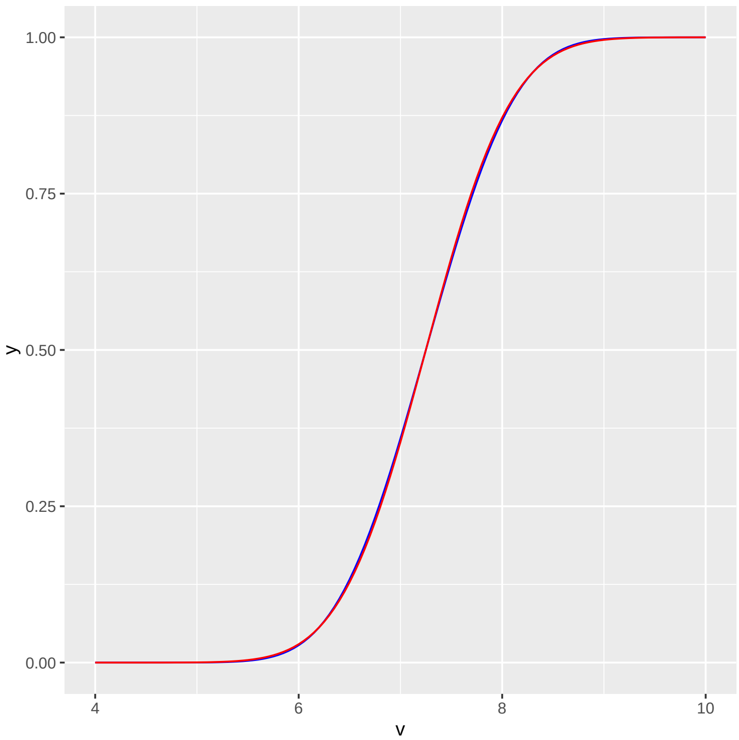
The results below are generated from an R script.

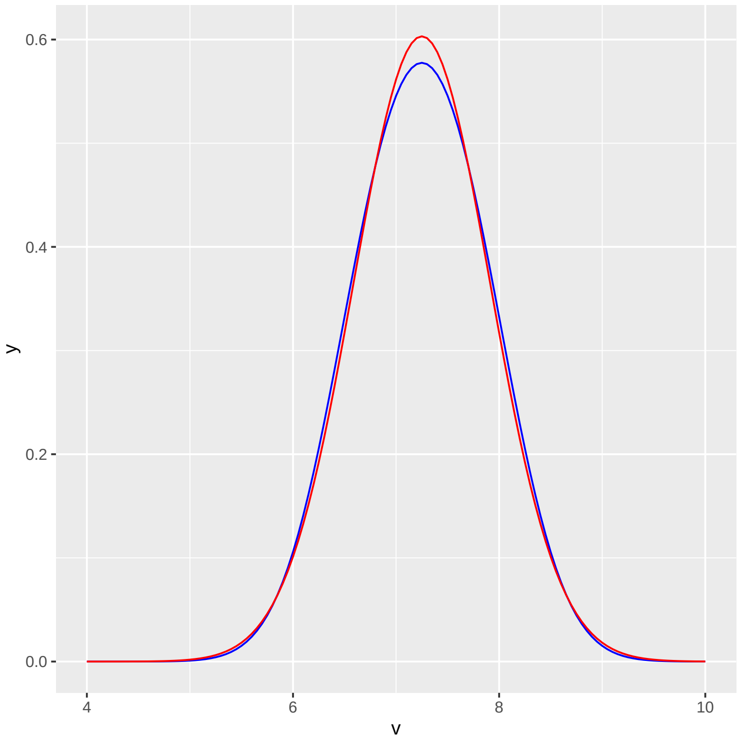
```
list.of.packages <- c("pracma")
new.packages <- list.of.packages[!(list.of.packages %in% installed.packages()[,"Package"])]
if(length(new.packages)) install.packages(new.packages)

library(pracma)
library(ggplot2)

## Formel für Dichtefunktion von V und Verteilungsfunktion von V bzw. P(Gesamtrendite \in  (l,h)). Herle

Density_fun <- function(v){

  term_1 = -13 + 2*v
  term_2 = -16 + 2*v

  sum_all = pnorm(term_1) - pnorm(term_2)
  return(2/3 * sum_all)

}

x <- seq(4.5,10.5,by=0.05)
y <- Density_fun_2(x)
plot(x, y, type ="l")
```



```
Prob_dist <- function(l=-1000,h){

  ah = -13+2*h
  al = -13+2*l
  bh = -16+2*h
  bl = -16+2*l

  term_ah = ah * pnorm(ah) + exp(-(ah^2)/2)/sqrt(2*pi)
  term_al = al * pnorm(al) + exp(-(al^2)/2)/sqrt(2*pi)
  term_bh = bh * pnorm(bh) + exp(-(bh^2)/2)/sqrt(2*pi)
  term_bl = bl * pnorm(bl) + exp(-(bl^2)/2)/sqrt(2*pi)

  sum_all = term_ah - term_al - term_bh + term_bl
  return(1/3 * sum_all)
}

# Plots zum veranschaulichen des Unterschieds zwischen Verteilung gemäß Faltung und Normalverteilung

x=seq(4,10,by=0.05)

df <- data.frame(v=x, y = Prob_dist(h=x), z =  pnorm(x,mean= 7.25,sd = sqrt(7/16)) )

ggplot(df, aes(x=v))+
  geom_line(aes(y=y),color = "blue")+ #Verteilungsfunktion von V
  geom_line(aes(y=z),color = "red")   #Verteilungsfunktion der Normalverteilung mit \mu = E(V) und \sig
```



```
df2 <- data.frame(v=x, y = Density_fun(v=x), z = dnorm(x,mean= 7.25,sd = sqrt(7/16)) )

ggplot(df2, aes(x=v))+
  geom_line(aes(y=y),color = "blue")+  #Dichtefunktion von V
  geom_line(aes(y=z),color = "red")    #Dichtefunktion der Normalverteilung mit \mu = E(V) und \sigma^2
```



```
## Gleiche Verteilungen/Dichten, bloss mit erf-function berechnet.

Density_fun_erf <- function(v){

  term_1 = sqrt(2)*(-6.5 + v)
  term_2 = sqrt(2)*(-8 + v)

  sum_all = erf(term_1)- erf(term_2)
  return(1/3 * sum_all)

}



Prob_dist_erf <- function(l=-1000,h){

  ah = sqrt(2)*(-6.5 + h)
  al = sqrt(2)*(-6.5 + l)
  bh = sqrt(2)*(-8 + h)
  bl = sqrt(2)*(-8 + l)

  term_ah = ah * erf(ah) + exp(-ah^2)/sqrt(pi)
  term_al = al * erf(al) + exp(-al^2)/sqrt(pi)
  term_bh = bh * erf(bh) + exp(-bh^2)/sqrt(pi)
  term_bl = bl * erf(bl) + exp(-bl^2)/sqrt(pi)

  sum_all = term_ah - term_al - term_bh + term_bl
  return(1/3 * 1/sqrt(2) * sum_all)
}
```

The R session information (including the OS info, R version and all packages used):

```
sessionInfo()

## R version 4.0.4 (2021-02-15)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS 12.1
##
## Matrix products: default
## LAPACK: /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
##
## locale:
## [1] de_CH.UTF-8/de_CH.UTF-8/de_CH.UTF-8/C/de_CH.UTF-8/de_CH.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] ggplot2_3.3.5 pracma_2.3.6
##
## loaded via a namespace (and not attached):
##  [1] knitr_1.36       magrittr_2.0.1   tidyselect_1.1.1 munsell_0.5.0    colorspace_2.0-2
##  [6] R6_2.5.1         rlang_0.4.12     fansi_0.5.0      highr_0.9        stringr_1.4.0
## [11] dplyr_1.0.7      tools_4.0.4      grid_4.0.4       gtable_0.3.0     xfun_0.29
## [16] utf8_1.2.2       DBI_1.1.1        withr_2.4.3      ellipsis_0.3.2   digest_0.6.29
## [21] assertthat_0.2.1 tibble_3.1.6     lifecycle_1.0.1  crayon_1.4.2     farver_2.1.0
## [26] purrr_0.3.4      vctrs_0.3.8      evaluate_0.14    glue_1.6.0       labeling_0.4.2
## [31] stringi_1.7.6    compiler_4.0.4   pillar_1.6.4     generics_0.1.1   scales_1.1.1
## [36] pkgconfig_2.0.3

Sys.time()

## [1] "2022-01-29 18:13:50 CET"
```