ProbComputations-Normal.R

```
# Basic probability computations using the normal distribution

# The time required to assemble an electronic component is normally
distributed,
# with a mean of 12 minutes and a standard deviation of 1 1/2 minutes.

# Before getting into specific computations, I want to draw the normal
curve to visualize
# the problems.

# Create a variable representing possible times.  Extend the range +/-
4 standard deviations
# from the mean.
the.mean = 12
the.std.dev = 1.5
times = seq(the.mean-4*the.std.dev,the.mean+4*the.std.dev,0.01)

# Create a variable representing densities for each of the possible
times
time.density = dnorm(times,the.mean,the.std.dev)

# Plot the times and densities
plot(times, time.density, type="l", yaxs="i")

#####################################################################
#
# Find probabilities that a particular assembly will require the
following lengths of time:
# More than 14 minutes

# First, let's visualize the area to be found on the normal curve.
This isn't necessary for
# computing the answer, but it helps to have the picture!

# set up a subset of times and densities for the range of interest
times.sub=seq(14,the.mean+4*the.std.dev,0.01)
times.dens.sub=dnorm(times.sub,the.mean,the.std.dev)

# create vectors of horizontal and vertical coordinates for the
polygon
cord.h=c(14,14,times.sub,the.mean+4*the.std.dev)
cord.v=c(0,dnorm(14,the.mean,the.std.dev),times.dens.sub,0)

# use the polygon function to draw on the existing curve
polygon(cord.h,cord.v,col="skyblue")
```

```
# compute the cumulative probability of 14 minutes or less
cum.prob=pnorm(14,the.mean,the.std.dev)

# the probability of 14 or more is the total area minus the cumulative
probability
prob.more.14 = 1-cum.prob

# display probability on the plot
text(15,.025,round(prob.more.14,4))

##############################################
# Find the probability of assembly time between 8 and 10 minutes

# set up a subset of times and densities for the range of interest
times.sub=seq(8,10,0.01)
times.dens.sub=dnorm(times.sub,the.mean,the.std.dev)

# create vectors of horizontal and vertical coordinates for the
polygon
cord.h=c(8,8,times.sub,10,10)
cord.v=c(0,dnorm(8,the.mean,the.std.dev),times.dens.sub,dnorm(10,the.m
ean,the.std.dev),0)

# use the polygon function to draw on the existing curve
polygon(cord.h,cord.v,col="yellow")

# compute the probability
prob.less.10 = pnorm(10,the.mean,the.std.dev)
prob.less.8 = pnorm(8,the.mean,the.std.dev)
prob.8.to.10 = prob.less.10 - prob.less.8

# display probability on the plot
text(9,.025,round(prob.8.to.10,4))

################################################
# find time such that 25% of all times are lower
# for this problem, we have to determine the time before we can draw
it on the plot
time.25=qnorm(.25,the.mean,the.std.dev)

# draw a line on the plot to indicate the position of this time
segments(time.25,0,time.25,dnorm(time.25,the.mean,the.std.dev),col="re
d",lwd=2)
text(time.25,0.15,paste(round(time.25,2),"min"))
```