Topic 11 d: Probability: Normal Distribution -- Part 2

We left off with the statement below. We had been able to use pnorm() and qnorm() to solve various problems, but only in the case where we had a normal distribution with $\mu = 0$ and $\sigma = 1$.

This is all well and good but it only takes care of the situation where we have a normal distribution with $\mu=0$ and $\sigma=1$. What do we do in the case where we have a normal distribution with $\mu=76.5303$ and $\sigma=12.04146$? We have only one Normal Distribution table and, at least so far, both pnorm() and qnorm() deal only with our special case.

The solution, as far as using the table, is to"normalize" a non-standard normal distribution into one that has $\mu=0$ and $\sigma=1$.

Consider the following population.

gnrnd4(1774459804, 12500743)

91.9	81.9	64.7	69.5	88.7	70.2	72.4	77.5	64.9	76.5	74.0	82.1	65.5	80.9	72.7	80.1	49.7	77.1	81.2	66.2
54.4	98.6	49.5	82.7	56.6	95.4	87.5	75.6	88.8	54.5	96.0	72.9	85.4	92.7	88.0	75.5	67.7	80.0	61.8	92.0
79.9	88.9	54.0	82.8	72.5	85.5	95.3	68.5	58.1	83.5	78.9	73.5	60.6	69.6	90.8	74.4	81.0	72.1	90.9	76.8
73.7	46.4	85.4	80.4	79.7	60.4	89.5	73.9	92.2	92.0	68.9	59.0	86.2	77.7	80.0	58.7	99.6	79.9	66.0	83.7
72.7	80.2	75.0	69.5	70.5	54.6	70.7	90.2	75.2	101.8	76.9	92.0	77.3	65.7	73.2	80.3	88.0	78.4	70.1	

We will generate that population, verify the values, and find the mean and standard deviation.

```
source( "../gnrnd4.R")
gnrnd4( 1774459804, 12500743)
head(L1)
tail(L1)
mu <- mean(L1)
mu
source("../pop_sd.R")
sigma <- pop_sd( L1 )
sigma
```

```
> source( "../gnrnd4.R")
> gnrnd4( 1774459804, 12500743)
style= 4   size= 99    seed= 77445   num digits= 1
 alt_sign= 1
[1] "DONE "
> head(L1)
[1] 91.9 81.9 64.7 69.5 88.7 70.2
> tail(L1)
[1] 65.7 73.2 80.3 88.0 78.4 70.1
> mu <- mean(L1)
> mu
[1] 76.5303
> source("../pop_sd.R")
> sigma <- pop_sd( L1 )
> sigma
[1] 12.04146
```

Then we will create a new population by subtracting the mean from each value in L1. Look at the new mean and standard deviation.

```
      # Create L2 by subtracting mu from L1
L2 <- L1 - mu
mean( L2 )
pop_sd(L2)
```

```
>      # Create L2 by subtracting mu from L1
> L2 <- L1 - mu
> mean( L2 )
[1] 2.155263e-16
> pop_sd(L2)
[1] 12.04146
```

Note that the standard deviation is unchanged and that the mean is essentially 0. Create a new population by dividing each value in L2 by sigma, the standard deviation.

```
     # Create L3 by dividing L2 by sigma
L3 <- L2 / sigma
mean( L3 )
pop_sd( L3 )
```

```
>      # Create L3 by dividing L2 by sigma
> L3 <- L2 / sigma
> mean( L3 )
[1] 1.787619e-17
> pop_sd( L3 )
[1] 1
```

What we see is that L3 is now normal with μ=0 and σ=1. It was interesting to do that transformation for all 99 values in L1 but we really do not need to do this. Instead, if we have the problem "For a normal distribution with μ=76.5303 and σ=12.04146 find P(X < 88)", we could change this to a standard normal distribution problem by finding

$$z = \frac{(88 - \mu)}{\sigma}$$

and then solving the problem find P(X < z).

In our particular case we get z = 0.9525168. Then we can use the standard normal table or we can use pnorm(z).

```
     #  for a special population find P( X < 88 )
z <- ( 88 - mu ) / sigma
z
pnorm( z )
```

```
>      #  for a special population find P( X < 88 )
> z <- ( 88 - mu ) / sigma
> z
[1] 0.9525168
> pnorm( z )
[1] 0.8295825
```

We could do that same transformation, z = (x-μ)/σ, for each value in the following problems and then we could use our standard normal table, or pnorm(), to solve the problem based on using the z values that we generate from the x values in the problems.

1) For a normal distribution with μ=18.4 and σ=4.6 find P(X < 12)?

2) For a normal distribution with μ=-9.7 and σ=6.1 find P(X > 5)?

3) For a normal distribution with μ=134 and σ=15 find P(120 < X < 145)?

```
     # problem 1
z <- (12 -18.4) / 4.6
z
pnorm( z )
     # problem 2
z <- (5 - -9.7) / 6.1
z
pnorm( z, lower.tail=FALSE )
     # problem 3
z_high <- (145-134) / 15
z_low <- (120 - 134) /15
z_high
z_low
pnorm( z_high ) - pnorm( z_low )
```

```
>      # problem 1
> z <- (12 -18.4) / 4.6
> z
[1] -1.391304
> pnorm( z )
[1] 0.08206658
>      # problem 2
> z <- (5 - -9.7) / 6.1
> z
[1] 2.409836
> pnorm( z, lower.tail=FALSE )
[1] 0.007979845
>      # problem 3
> z_high <- (145-134) / 15
> z_low <- (120 - 134) /15
> z_high
[1] 0.7333333
> z_low
[1] -0.9333333
> pnorm( z_high ) - pnorm( z_low )
[1] 0.5929985
```

For problems such as

4) For a normal distribution with $\mu=244$ and $\sigma=17$ find y such that $P(X < y) = 0.23$?

5) For a normal distribution with $\mu=47.2$ and $\sigma=5.8$ find y such that $P(X > y) =0.05$?

we need to read the table backwards to get a z value or we need to use qnorm() to get a z value and then we get our y value by using the inverse transformation, $y = z*\sigma+\mu$.

```
     # problem 4
z <- qnorm(0.23)
z
y <- z*17 + 244
y
     # problem 5
z <- qnorm(0.05, lower.tail=FALSE)
z
y <- z*5.8 + 47.2
y
```

```
[1] 0.5929985
>      # problem 4
> z <- qnorm(0.23)
> z
[1] -0.7388468
> y <- z*17 + 244
> y
[1] 231.4396
>      # problem 5
> z <- qnorm(0.05, lower.tail=FALSE)
> z
[1] 1.644854
> y <- z*5.8 + 47.2
> y
[1] 56.74015
```

Just because we could do all of this work for these non-standard problems does not mean that we are happy to do it. These non-standard problems happen so often that pnorm() and qnorm() have forms that allow us to do those problems without resorting to "normalizing" the values in the problem (for pnorm() problems) or doing the inverse computation (for qnorm() problems).

Both pnorm() and qnorm() allow you to specify the mean and standard deviation of the normal distribution you are using. If you do not specify these values then pnorm() and qnorm() assume that they are 0 and 1, respectively. That is why pnorm() and qnorm(), as we have used them work when $\mu=0$ and $\sigma=1$.

Restate problems 1-3, and solve them with the expanded pnorm() statement.

1) For a normal distribution with $\mu=18.4$ and $\sigma=4.6$ find $P(X < 12)$?

2) For a normal distribution with $\mu=-9.7$ and $\sigma=6.1$ find $P(X > 5)$?

3) For a normal distribution with $\mu=134$ and $\sigma=15$ find $P(120 < X < 145)$?

```
     # redo problem 1
pnorm( 12, mean=18.4, sd=4.6)
     # redo problem 2
pnorm( 5, mean=-9.7, sd=6.1, lower.tail=FALSE)
     # redo problem 3
pnorm( 145, mean=134, sd=15) -
  pnorm( 120, mean=134, sd=15 )
```

```
>      # redo problem 1
> pnorm( 12, mean=18.4, sd=4.6)
[1] 0.08206658
>      # redo problem 2
> pnorm( 5, mean=-9.7, sd=6.1, lower.tail=FALSE)
[1] 0.007979845
>      # redo problem 3
> pnorm( 145, mean=134, sd=15) -
+   pnorm( 120, mean=134, sd=15 )
[1] 0.5929985
```

Restate problems 4 and 5 and solve them with the expanded qnorm() statement.

4) For a normal distribution with $\mu=244$ and $\sigma=17$ find y such that $P(X < y) = 0.23$?
5) For a normal distribution with $\mu=47.2$ and $\sigma=5.8$ find y such that $P(X > y) =0.05$?

```
     # redo problem 4
qnorm( 0.23, mean=244, sd=17 )
     # redo problem 5
qnorm( 0.05, mean=47.2, sd=5.8, lower.tail=FALSE)
```

```
>      # redo problem 4
> qnorm( 0.23, mean=244, sd=17 )
[1] 231.4396
>      # redo problem 5
> qnorm( 0.05, mean=47.2, sd=5.8, lower.tail=FALSE)
[1] 56.74015
```

Finally, we have one more item to cover. We get tired of saying that a distribution is normal with mean = 5.7 and standard deviation = 3.4. We want a much shorter way to at least write that statement. The shorter way is to write

N(5.7, 3.4)

Using this notation, our standard normal distribution is N(0, 1).

Given all of this, one might ask "For a distribution that is N(500, 100), what is the 95th percentile?" That would be the value that has 95% of the area under the curve to its left. We can get that via qnorm(0.95, mean=500, sd=100).

```
     # For N( 500, 100 ) find the 95th percentile
qnorm( 0.95, mean=500, sd=100 )
```

```
>      # For N( 500, 100 ) find the 95th percentile
> qnorm( 0.95, mean=500, sd=100 )
[1] 664.4854
```