Probability Distributions

Logan Kelly, Ph.D.

Managerial Statistics - ECON 730

University of Wisconsin-River Falls

Objectives

- After this lecture you should be able to use probability distributions in managerial decision making
- Topics:
  - Mean
  - Standard Deviation
  - Variances
  - Proportions
  - Normal and t-distributions
  - Sampling distributions
  - Confidence intervals

Case Background

Double E (EE) Chain of Consumer Electronics Stores

- EE is a chain of stores selling consumer electronics in the US
- EE suspects that the main reasons for declining profits are the falling quality of service and growing competition
- EE is concerned that it is spending too much time with pseudo customers
- EE has collated data on time spent with customers and whether the customer was a pseudo customer.

Probability Distributions

Probability Distributions

- A probability distribution allows us to compute the chance that a variable lies within a given range
- Examples:
  - Probability sales are between 10,000 and 50,000
  - Probability that a customer buys 2 items

Types

- Probability distributions can be
  - **Discrete:** only taking on certain values
  - **Continuous:** taking on any value within a range or set of ranges
- Examples:
  - The number of items that a customer buys follows a **discrete** probability distribution
  - The daily sales at EE follows a **continuous** probability distribution

Important Attributes

- **Mean:** The average or 'expected value' of a distribution.
  - Denoted by μ (The Greek letter mu)
- **Variance:** A measure of dispersion and volatility.
  - Denoted by σ^2 (Sigma Squared)
- **Standard deviation:** A related measure of dispersion computed as the square root of the variance.
  - Denoted by σ (The Greek letter sigma)

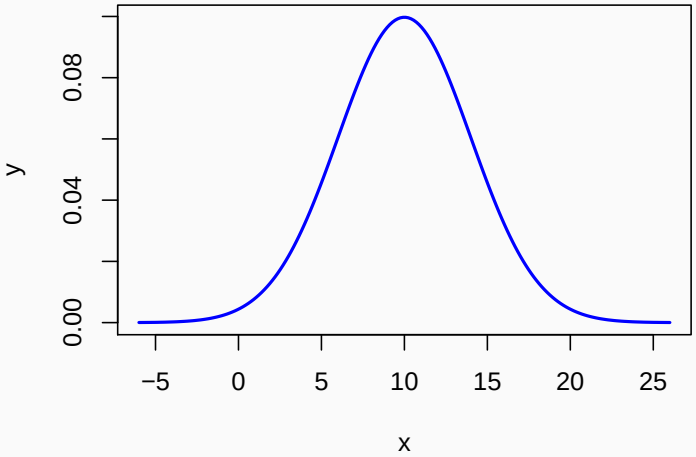
Proportions

- Some variables take on only two values
  - Employee is Male/Female
  - Voters Approve/Disapprove
  - Customer buys something/doesn't buy anything
- The **proportion** of customers who buy something at the store has special characteristics
- If a variable has a proportion, p, then we can show mathematically that its variance is $p(1-p)$ and the standard deviation is $\sqrt{p(1-p)}$

Normal Distribution

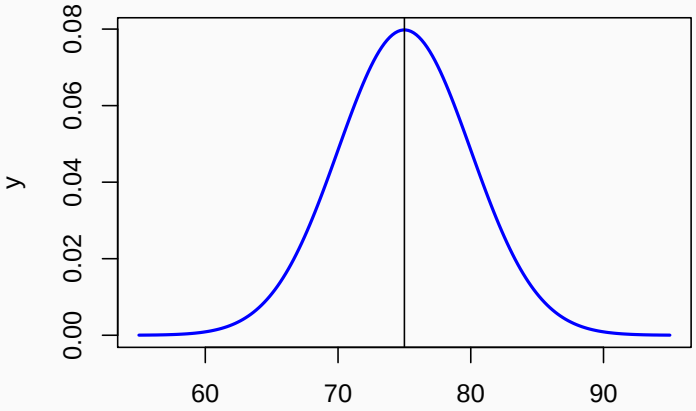
- One of the most common distributions in statistics is the normal distribution
- There are actually innumerable normal distributions, each characterized by two parameters:
    - The mean
    - The standard deviation
- The standard normal has a mean of zero and a standard deviation of one

The Normal Distribution: A Bell Curve



The Normal Distribution is Symmetric

The area under each half of the normal PDF is 0.5

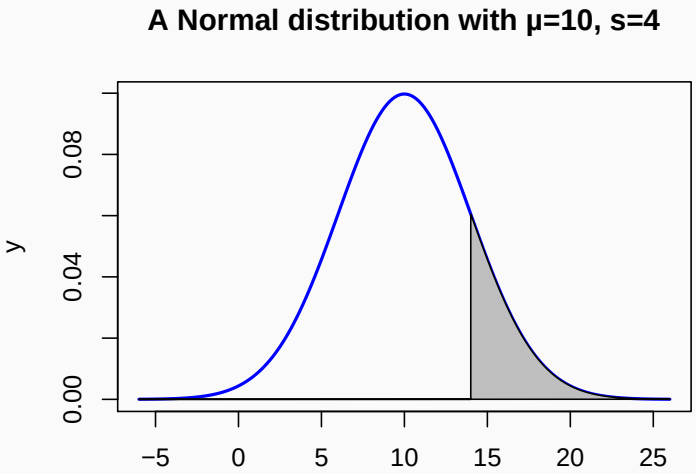


Calculating probabilities

- In this example, we will calculate the probability that a customer will purchase 14 items or more.
- We are told that the averge number of items purchased by a customer is a normally distributed random variable, the population mean is 10, $\mu = 10$, and the population standard deviation is 4, $\sigma = 4$.

The average number of items per customer

Shaded Area measures the probability that the average is greater than 14



Calulating the probability that the average is greater than 14

To calculate this probability, use the qnorm() command. See https://ljkelly3141.github.io/teaching/ECON226/practice/p07.html for practice.

```
pnorm(q=14, mean=10, sd=4, lower.tail=F)
```

```
[1] 0.1586553
```

The probability that the average is less than 14 can be calulated by seting **lower.tail=T**

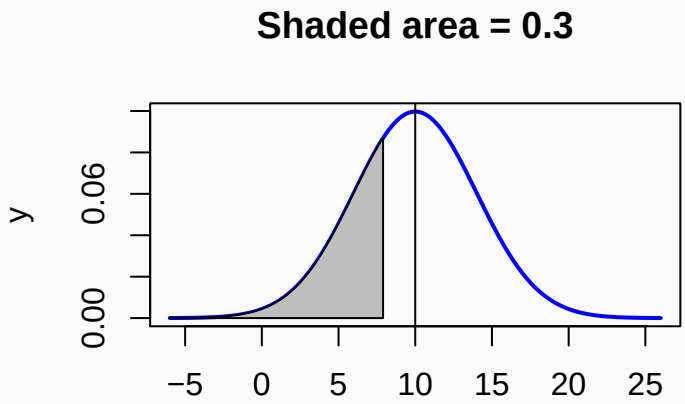
```
pnorm(q=14, mean=10, sd=4, lower.tail=T)
```

```
[1] 0.8413447
```

What value of X gives an area of 0.3 to its left ?

```
qnorm(p=0.3, mean=10, sd=4, lower.tail=T)
```

```
[1] 7.902398
```



The Double E case

Count the number of pseudo customers

- Load the Case Data

```
pseudo.cust <- read.csv("data/EE_PseudoCustomers.csv")
```

- Count the number of pseudo customers
  - To calculate a frequency table, we use the **table()** command

```
attach(pseudo.cust)
table(pseudo.customer)
```

```
pseudo.customer
 0  1
60 40
```

Calculate the mean service time for pesudo and true customers

- We need to install the “tidyverse” package
- The code `if (!require("tidyverse"))...` checks if the package has been installed and installs it if it is missing

```
if (!require("tidyverse")) install.packages("tidyverse")
```

Calculate the mean (Cont.)

- Now we need to load the “tidyverse” package and calculate the average service time by customer type

```
library(tidyverse)
pseudo.cust.avg <- pseudo.cust %>% group_by(pseudo.customer) %>%
  summarize(avg = mean(service.time))
```

- The pipe operator, **%<%**, can be interpreted as “with this do that.” So the command above means group the pseudo.cust data by customer type and then calculate the average of each group.

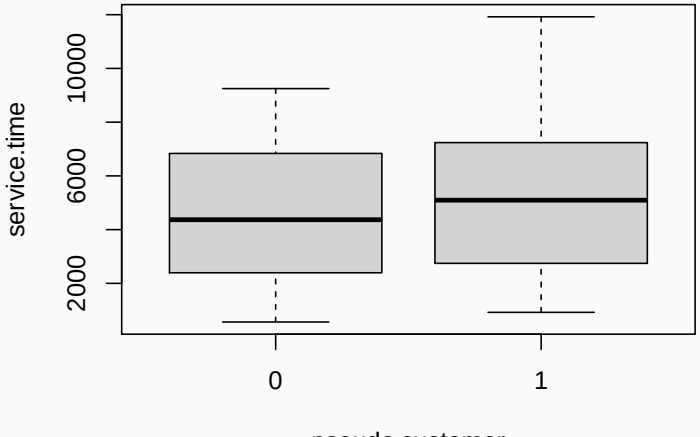
Let’s see the resulults

```
pseudo.cust.avg
```

```
# A tibble: 2 x 2
  pseudo.customer   avg
            <int> <dbl>
1               0 4584.
2               1 5324.
```

Comparing the two groups with boxplots

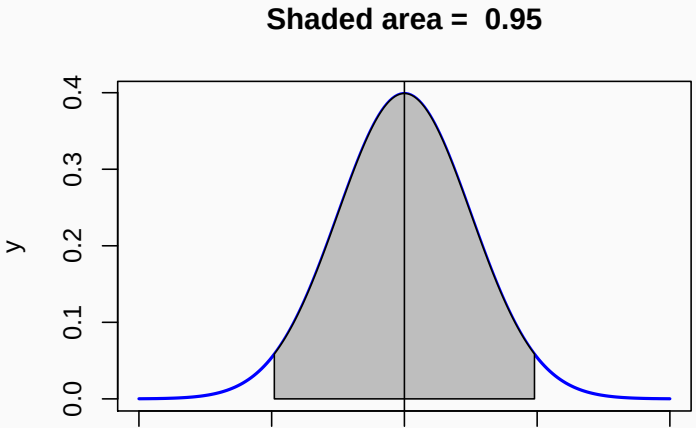
We can examine the difference in time spent with pseudo and true customers with side by side boxplots of the data: **`boxplot(service.time~pseudo.customer)`**



Calculate a 95% confidence interval

Confidence interval

- A confidence interval is an interval that contains the population parameter with probability $1 - \alpha$



Confidence interval math

- A confidence interval takes on the form:

$$\bar{X} \pm t_{\alpha/2,N-1} S_{\bar{X}}$$

  - $t_{\alpha/2,N-1}$ is the value needed to generate an area of α/2 in each tail of a t-distribution with n-1 degrees of freedom
  - $S_{\bar{X}} = \frac{s}{\sqrt{N}}$ is the standard error of the mean
- The confidence level is equal to $1 - \alpha$

Calculate a confidence interval

To calculate a confidence interval, we need the following steps

1. Calculate the mean
2. Calculate the standard error of the mean
3. Find the t-score that corresponds to the confidence level
4. Calculate the error bound and construct the confidence interval

Step 1: Calculate the mean

- Use the **group_by()** and **summarize()** commands to calculate the average service time spent with pseudo and true customers.

```
pseudo.cust.avg <- pseudo.cust %>% group_by(pseudo.customer) %>%
  summarize(avg = mean(service.time))
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

- Store the averages in a vector, or list of numbers

```
avg.service.time <- pseudo.cust.avg$avg
sample.mean <- avg.service.time[2]
```

Step 2: Calculate the standard error of the mean

- The formula for the standard error of the mean is $S_{\bar{X}} = \frac{\sigma}{\sqrt{N}}$
- And if we do not know the population standard deviation $S_{\bar{X}} = \frac{s}{\sqrt{N}}$
- The **sd()** command can be used to find the standard deviations of the two samples

```
pseudo.cust.se <- pseudo.cust %>% group_by(pseudo.customer) %>%
  summarize(se = sd(service.time)/(n())^.5)
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

```
se.service.time <- pseudo.cust.se$se
sample.se <- se.service.time[2]
```

Step 3: Find the t-score that corresponds to the confidence level

- We need to have $\alpha/2$ probability in the lower and upper tails, we divide by two because there are two tails.
- The `qt()` command will calculate the t-score, $t_{\alpha/2,N-1}$

Step 3: Find the t-score (Cont.)

```
# Set Alpha
alpha = 0.05

# find N, number of pseudo customers
count.type <- table(pseudo.customer)
N = count.type[2]

# Calculate t-score
t.score = qt(p=alpha/2,N-1,lower.tail=F)

print(t.score)
```

```
[1] 2.022691
```

Step 4. Calculate the error bound and construct the confidence interval

- The error bound is $t_{\alpha/2,N-1}S_{\bar{X}}$

```
error.bound <- t.score * sample.se
```

- Construct the confidence interval

```
lower.bound <- sample.mean - error.bound
upper.bound <- sample.mean + error.bound

print(c(lower.bound,upper.bound))
```

```
[1] 4434.669 6213.181
```

A much easier way

Let’s use linear regression as a short cut

```
# Calculate the mean and standard error
l.model <- lm(service.time ~ 1,
              data=subset(pseudo.cust, pseudo.customer==1))

# Calculate the confidence interval
confint(l.model, level=0.95)
```

```
                2.5 %   97.5 %
(Intercept) 4434.669 6213.181
```