Levon Gevorgyan MA Homework 1

2025-09-25

Tasks

1. Choose an Innovation from the List.

For the link to the innovation from the "Innovations' List" *click here* (the first innovation on the list). For the actual product profile, *click here*.

2. Identify a Similar Innovation from the Past.

Apple Vision Pro and the look-alike Oculus/Meta Quest

In this work, I will be examining the innovation of the ***Apple Vision Pro.*** This headset lets users see and use apps, videos, and other digital tools in 3D space, mixing the real world with virtual screens and objects. The look-alike innovation from the past is the ***Oculus Quest,*** a virtual reality headset that also allowed users to step into immersive digital environments. While Oculus focused mainly on gaming and entertainment, Apple Vision Pro aims to combine work, communication, and media into a single device.

By bringing apps and digital content into the user's actual surroundings, Apple Vision Pro aims to make computing more natural and immersive, just as Oculus Quest helped make virtual reality more accessible to everyday users. Studying the diffusion of Oculus devices can provide useful insights into how Apple Vision Pro might spread in the market and what challenges it may face in achieving wider adoption.

Libraries

```r
library(readxl)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(ggplot2)
```

3. Find Historical Data.

Time Series data on Meta/Oculus Quest Headsets

```r
df <- read_excel("data.xlsx", sheet = "Data", skip = 4, col_names = FALSE)
```

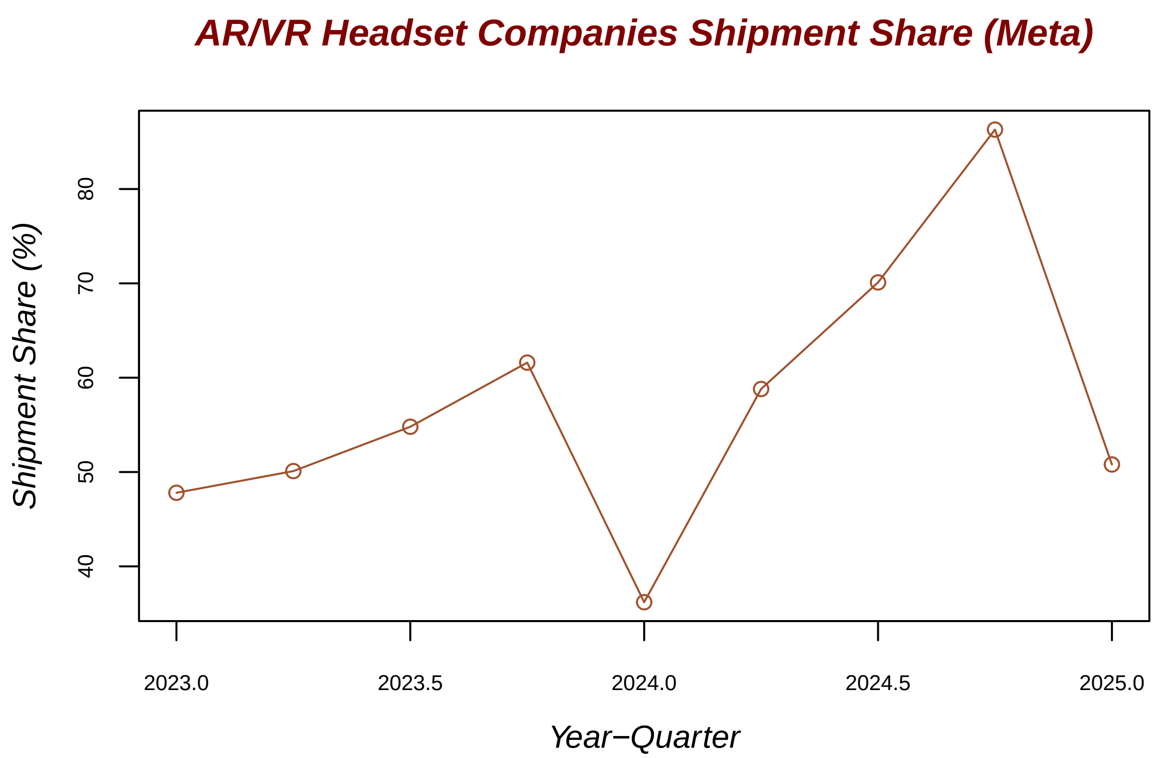
```
## New names:
## * `` -> `...1`
## * `` -> `...2`
## * `` -> `...3`
## * `` -> `...4`
## * `` -> `...5`
## * `` -> `...6`
## * `` -> `...7`
## * `` -> `...8`
## * `` -> `...9`
## * `` -> `...10`
## * `` -> `...11`
```

```r
head(df)
```

```
## # A tibble: 6 x 11
##   ...1    ...2  ...3  ...4  ...5       ...6  ...7   ...8  ...9   ...10 ...11
##   <chr>   <chr> <chr> <chr> <chr>      <chr> <chr>  <chr> <chr>  <chr> <chr>
## 1 <NA>    Meta  Sony  Apple ByteDance  Xreal Others HTC   Viture TLC   <NA>
## 2 Q1 2023 47.8  -     -     6.1        1.3   43.4   1.4   -      -     in %
## 3 Q2 2023 50.1  27.2  -     9.6        2.5   10.6   -     -      -     in %
## 4 Q3 2023 54.8  23.3  -     6.4        4.2   11.3   -     -      -     in %
## 5 Q4 2023 61.6  15.9  -     6.4        4.7   11.4   -     -      -     in %
## 6 Q1 2024 36.2  2.5   16.5  11.5       9.1   19.6   -     2      2.6   in %
```

```r
colnames(df) <- df[1, ]
df <- df[-1, ]

meta_vec <- df$Meta

meta_numeric <- as.numeric(gsub(",", ".", meta_vec))
meta_numeric <- meta_numeric[!is.na(meta_numeric)]

meta_ts <- ts(meta_numeric, start = c(2023,1), frequency = 4)

# ---------------------------------------------
# Plot of the Sales Volume on Historical Data
# ---------------------------------------------

plot(meta_ts, type = "o",
     col = "#A0522D",
     xlab = "",
```

```
     ylab = "",
     main = "AR/VR Headset Companies Shipment Share (Meta)",
     col.main = "#800000",
     font.main = 4,
     xaxt = "n", yaxt = "n")

axis(1, cex.axis = 0.7)
axis(2, cex.axis = 0.7)

mtext("Year-Quarter", side = 1, line = 2.5, font = 3)
mtext("Shipment Share (%)", side = 2, line = 2.5, font = 3)
```



The graph shows that Meta's headset shipment share fluctuated between 2023 and 2025, with a general upward trend and a sharp peak in late 2024, though periods of steep decline reveal that its dominance is unstable and strongly influenced by competition, product cycles, and seasonal demand.

4. Estimate Bass Model Parameters.

```
meta_frac <- meta_ts / 100

cum_adopt <- cumsum(meta_frac)

time <- 1:length(cum_adopt)
```

```r
start_vals <- list(
  p = 0.02,
  q = 0.4,
  M = max(cum_adopt)
)

bass_model <- nls(
  cum_adopt ~ M * ((1 - exp(-(p+q)*time)) / (1 + (q/p) * exp(-(p+q)*time))),
  start = start_vals,
  algorithm = "port",
  lower = c(0, 0, 0),
  upper = c(1, 1, 10),
  control = list(maxiter = 500)
)

params <- coef(bass_model)
p <- params["p"]
q <- params["q"]
M <- params["M"]

cat("Estimated Bass parameters:\n")
```

```
## Estimated Bass parameters:
```

```r
cat("p (innovation) =", p, "\n")
```

```
## p (innovation) = 0.04087809
```

```r
cat("q (imitation)  =", q, "\n")
```

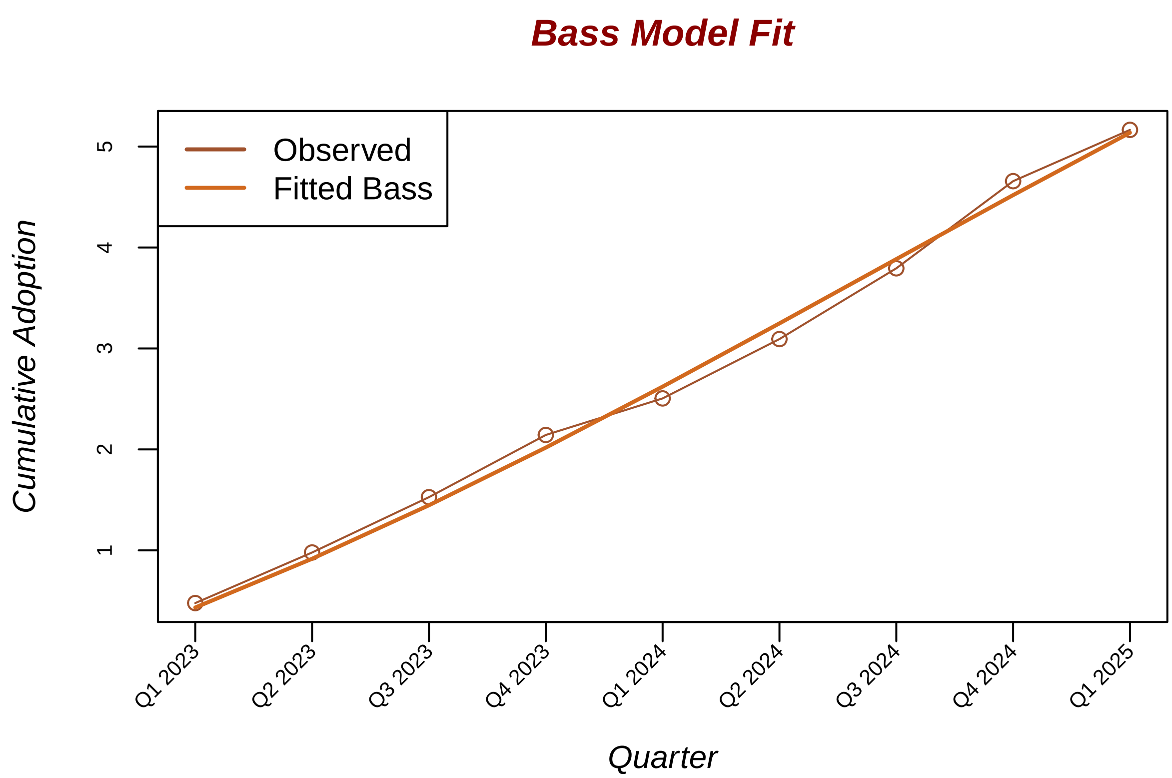
```
## q (imitation)  = 0.1631384
```

```r
cat("M (market potential) =", M, "\n")
```

```
## M (market potential) = 10
```

```r
# ----------------------------------------------
# Plot of Observed vs Fitted Cumulative Adoption
# ----------------------------------------------

fitted_cum <- predict(bass_model)

years <- rep(2023:2025, each = 4)
quarters <- rep(c("Q1","Q2","Q3","Q4"), times = 3)
labels <- paste0(quarters, " ", years)[1:length(cum_adopt)]

plot(time, cum_adopt, type = "o",
     col = "#A0522D",
     ylab = "Cumulative Adoption",
     xlab = "Quarter",
```

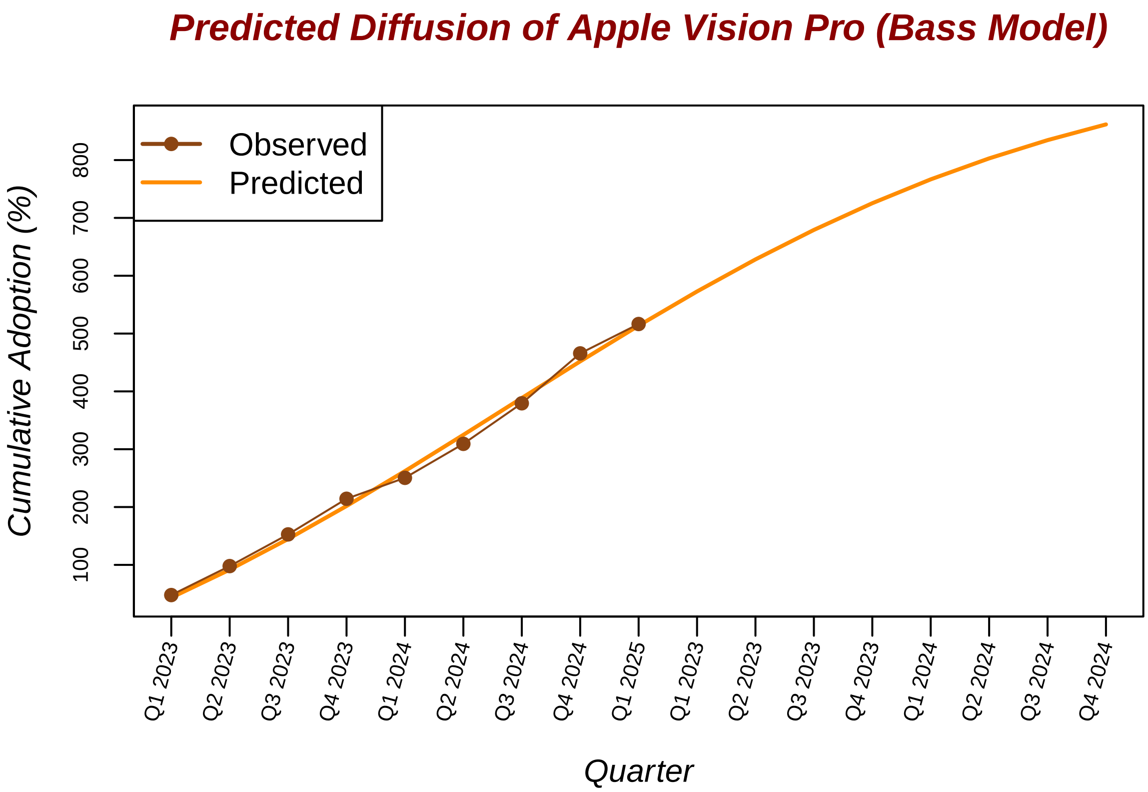
```
     xaxt = "n", yaxt = "n",
     main = "Bass Model Fit",
     col.main = "#8B0000",
     font.main = 4,
     font.lab = 3)

lines(time, fitted_cum, col = "#D2691E", lwd = 2)

axis(2, cex.axis = 0.7)

axis(1, at = time, labels = FALSE)
text(x = time, y = par("usr")[3] - 0.05*diff(par("usr")[3:4]),
     labels = labels, srt = 45, adj = 1, xpd = TRUE, cex = 0.7)

legend("topleft", legend = c("Observed", "Fitted Bass"),
       col = c("#A0522D", "#D2691E"), lty = 1, lwd = 2)
```



Here, the Bass model provides a very close fit to the observed cumulative adoption of Meta's headsets over time, with the fitted curve almost overlapping the actual data points, which suggests that the diffusion of adoption can be well explained by a mix of innovation and imitation effects, and that the estimated market potential captures the observed growth quite accurately.

5. Predict the Diffusion of the Innovation.

```
future_periods <- 8
total_periods <- length(cum_adopt) + future_periods
future_time <- 1:total_periods

bass_predict <- function(t, p, q, M){
  M * ((1 - exp(-(p + q) * t)) / (1 + (q / p) * exp(-(p + q) * t)))
}

predicted_cum <- bass_predict(future_time, p, q, M)

predicted_cum_percent <- predicted_cum * 100

# -------------------------------------
# Plot of Observed + Predicted Diffusion
# -------------------------------------

y_ticks <- pretty(predicted_cum_percent, n = 6)

plot(future_time, predicted_cum_percent, type="l", col="#FF8C00", lwd=2,
     ylab="", xlab="", xaxt="n", yaxt="n",
     main="Predicted Diffusion of Apple Vision Pro (Bass Model)",
     col.main="#8B0000", font.main=4)

points(time, cum_adopt*100, type="o", col="#8B4513", pch=16)

axis(2, at=y_ticks, labels=y_ticks, cex.axis=0.7)
axis(1, at=future_time, labels=FALSE)

text(x=future_time,
     y=par("usr")[3] - 0.05*diff(par("usr")[3:4]),
     labels=labels, srt=75, adj=1, xpd=TRUE, cex=0.7)

mtext("Quarter", side=1, line=3.5, font=3)
mtext("Cumulative Adoption (%)", side=2, line=2.5, font=3)

legend("topleft", legend=c("Observed", "Predicted"),
       col=c("#8B4513","#FF8C00"), lty=1, lwd=2, pch=c(16, NA))
```



Here we apply the Bass model to the adoption of Apple Vision Pro, where the observed values closely follow the fitted curve in the initial quarters, and the prediction suggests that cumulative adoption will continue increasing but at a decreasing rate as the market moves toward saturation.

6. Choose a Scope.

For the ***Apple Vision Pro*** diffusion analysis, we focus on a *global scope* because Apple sells its products *worldwide almost at the same time,* and early adopters are present in many regions. Looking only at one country may hide how adoption in one market can affect another. In the same way, ***Meta's Oculus*** has also spread on a *global level*, with users in North America, Europe, and Asia, not just in one place. Since both Apple and Meta work in very international markets, a ***global view*** gives a clearer picture of how these devices spread and compete.

7. Estimate the number of Adopters by Period.

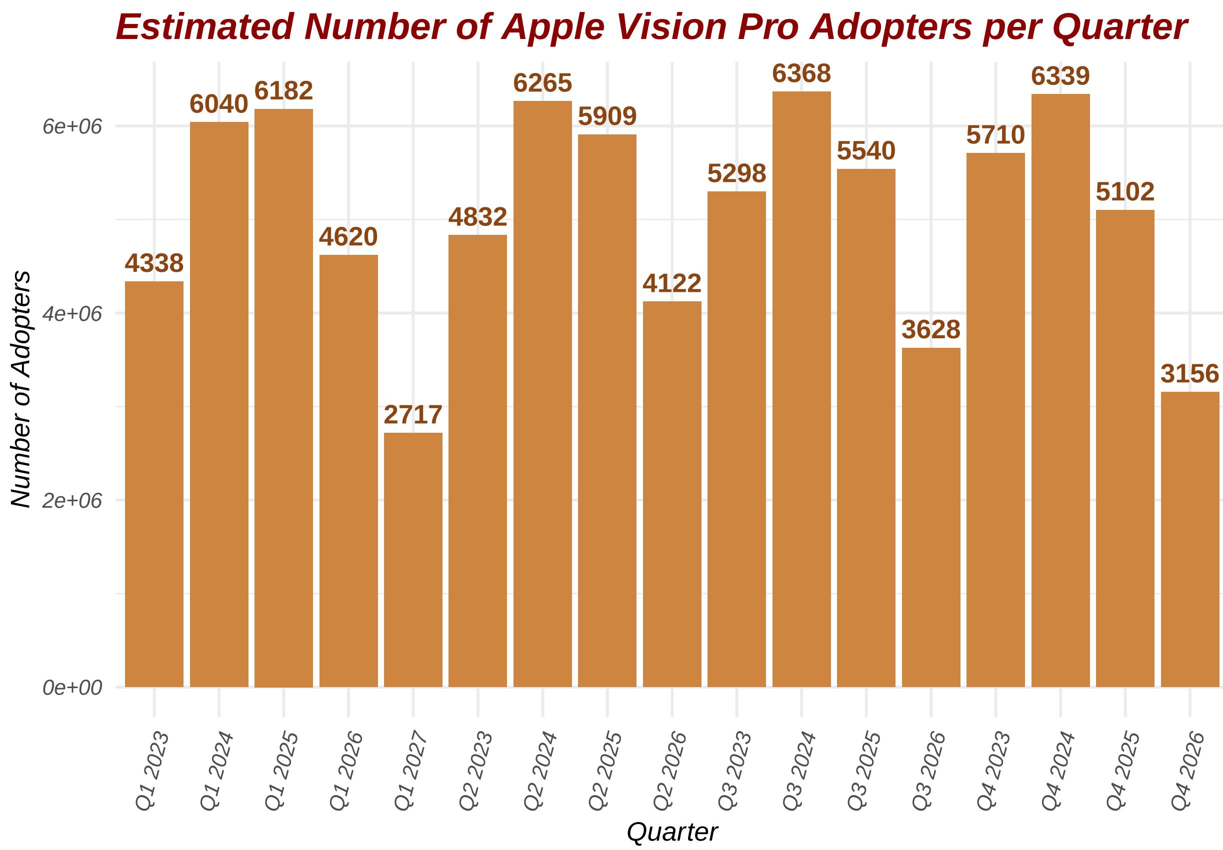
```
future_periods <- 8                        # number of future quarters to predict
                                           #(as we predict for 2025-2026 -> 2 years)
total_periods <- length(cum_adopt) + future_periods
future_time <- 1:total_periods

bass_predict <- function(t, p, q, M){
  M * ((1 - exp(-(p + q) * t)) / (1 + (q / p) * exp(-(p + q) * t)))
}
```

```r
predicted_cum <- bass_predict(future_time, p, q, M)

adopters_per_period <- c(predicted_cum[1], diff(predicted_cum))

adopters_percent <- adopters_per_period * 100

M_estimate <- 1e7                              # 10 million potential adopters
adopters_units <- adopters_per_period * M_estimate

data.frame(
  Quarter = paste0(rep(c("Q1","Q2","Q3","Q4"), times=5)[1:total_periods],
                   " ", rep(2023:2027, each=4)[1:total_periods]),
  Adopters_Percent = round(adopters_percent, 2),
  Adopters_Estimated_Units = round(adopters_units)
)
```

```
##    Quarter Adopters_Percent Adopters_Estimated_Units
## 1  Q1 2023            43.38                  4337952
## 2  Q2 2023            48.32                  4831901
## 3  Q3 2023            52.98                  5297988
## 4  Q4 2023            57.10                  5709651
## 5  Q1 2024            60.40                  6039998
## 6  Q2 2024            62.65                  6265132
## 7  Q3 2024            63.68                  6367588
## 8  Q4 2024            63.39                  6339104
## 9  Q1 2025            61.82                  6181996
## 10 Q2 2025            59.09                  5908770
## 11 Q3 2025            55.40                  5540071
## 12 Q4 2025            51.02                  5101523
## 13 Q1 2026            46.20                  4620251
## 14 Q2 2026            41.22                  4121814
## 15 Q3 2026            36.28                  3628018
## 16 Q4 2026            31.56                  3155763
## 17 Q1 2027            27.17                  2716813
```

```r
# --------------------------------------------
# Visualization of Predicted Number of Adopters
# --------------------------------------------

plot_data <- data.frame(
  Quarter = paste0(rep(c("Q1","Q2","Q3","Q4"), times=5)[1:total_periods],
                   " ", rep(2023:2027, each=4)[1:total_periods]),
  Adopters = adopters_units
)

ggplot(plot_data, aes(x=Quarter, y=Adopters)) +
  geom_bar(stat="identity", fill="#CD853F") +
  geom_text(aes(label=round(Adopters/1000)),
            vjust=-0.5, size=3.5, color="#8B4513", fontface="bold") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle=75, hjust=1, size=8, face="italic"),
    axis.text.y = element_text(size=8, face="italic"),
```

```
    axis.title.x = element_text(face="italic", size=10),
    axis.title.y = element_text(face="italic", size=10),
    plot.title = element_text(face="bold.italic", color="#8B0000", size=14)
  ) +
  labs(
    title = "Estimated Number of Apple Vision Pro Adopters per Quarter",
    x = "Quarter",
    y = "Number of Adopters"
  )
```



The bar chart above presents the ***estimated quarterly adoption of the Apple Vision Pro***, predicted using the Bass diffusion model with a market potential of 10 million adopters. The forecast indicates that adoption rises from about *4.34 million* in Q1 2023, reaching peaks of over *6.3 million* adopters per quarter by 2024. After this surge, adoption gradually tapers off through 2025–2026, ending near *3.16 million* in Q4 2026. This trend follows the typical ***S-shaped diffusion pattern***, where early growth is driven by innovators and imitators, before slowing down as the market approaches saturation.

Reference: Statista — *"AR/VR headset companies shipment share worldwide 2023–2025, by quarter." Click here for the link*