MA_FOR_DS

Vahan Yeranosyan

2023-09-27

Product

The Aura Strap 2 is a smart strap for the Apple Watch that measures body fat, muscle, protein and hydration levels (among other things), adding functionality to your smartwatch. It also offers personalized diet and exercise advice via its iOS app. This product targets old people that have heath problems that need constant monitoring of those functions.

link

Market estimation

According to one of news **articles** We get that 30% of Iphone users use Apple Watch

30% of apple users use apple watch as of 2021 Q2 exceeding 100 million. considering that 15% of population are elderly. We can do the following calculation.

70% of apple users (230mil) don't use apple watch and 15% (34mil) of them are elderly, which would be our market potential.

Data

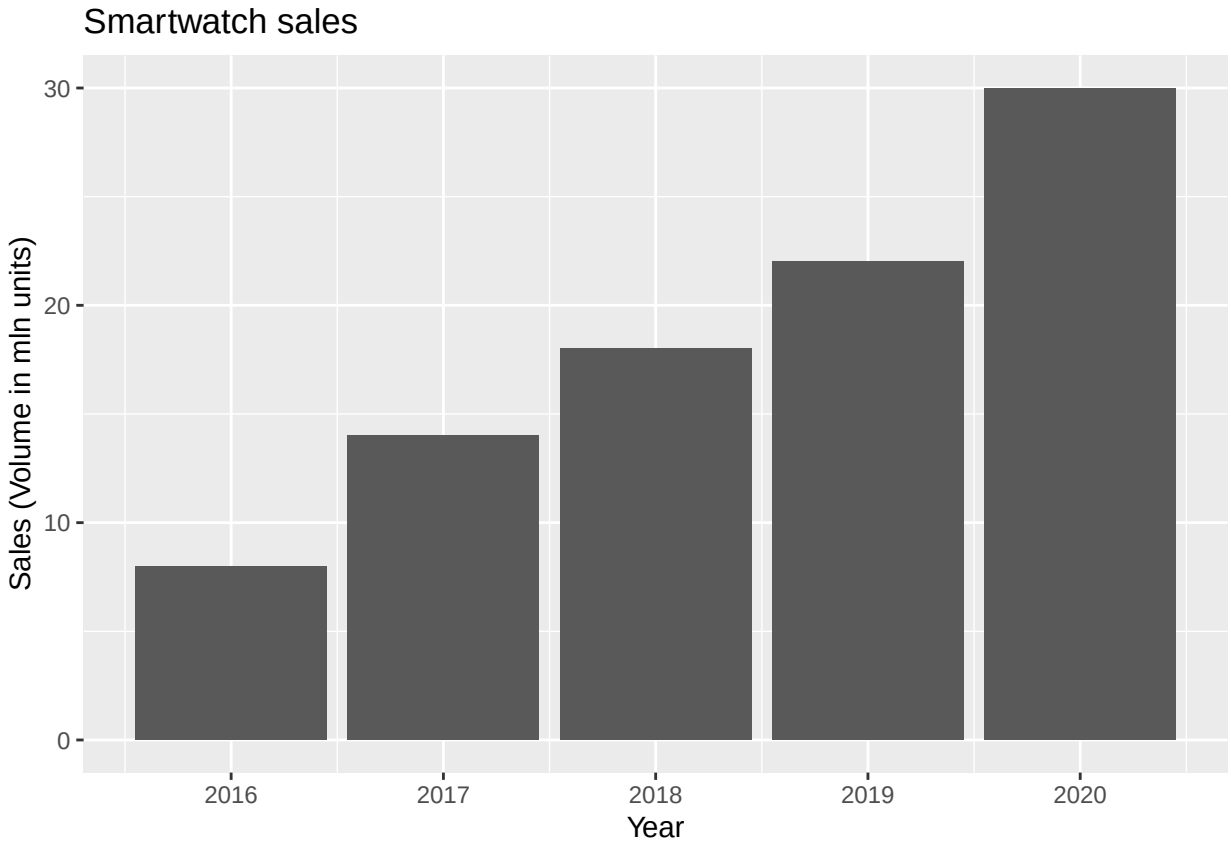
link: https://www.statista.com/statistics/1221051/apple-watch-users-worldwide/

This data is cumulative, so we need to make it to represent current year.

```
total = c(10,18,32,50,72,102)
sales = data.frame(cumulative = total, sales = c(total[1],diff(total)),
                   year = c(1:6) + 2014)[-1,] # remove the 1st row
sales$time = 1:5
sales
```

```
##   cumulative sales year time
## 2         18     8 2016    1
## 3         32    14 2017    2
## 4         50    18 2018    3
## 5         72    22 2019    4
## 6        102    30 2020    5
```

```
library(ggplot2)
library(ggpubr)


ggplot(data = sales, aes(x = year, y = sales)) +geom_bar(stat = 'identity')  +
ggtitle('Smartwatch sales') + xlab("Year") + ylab("Sales (Volume in mln units)")
```

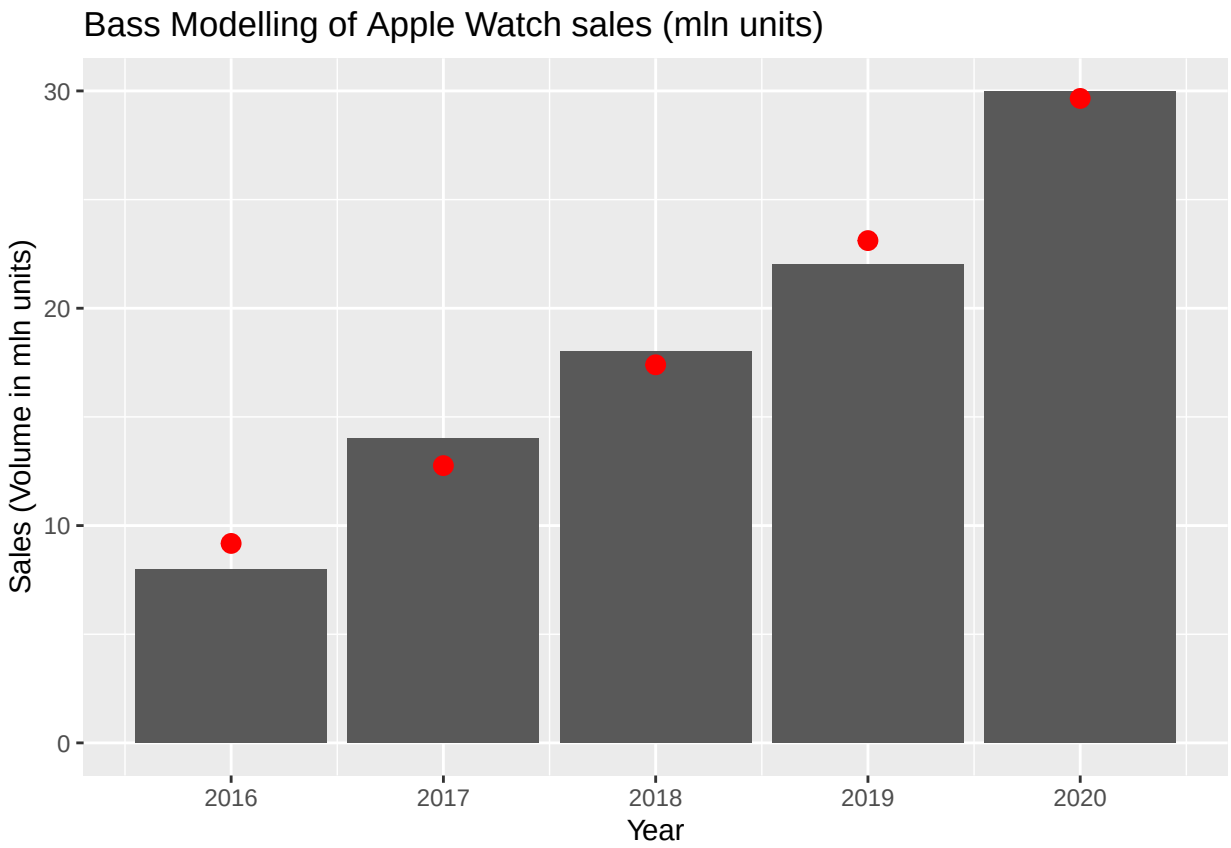



```
t = 1:(length(sales) + 1)
bass_m = nls(sales$sales ~ (m*(((p+q)^2/p)*exp(-(p+q)*t))) / (1+(q/p)*exp(-(p+q)*t))^2 ,
start=c(list(m=sum(sales$sales),p=0.02,q=0.4)))

p = coef(bass_m)["p"]
q =  coef(bass_m)["q"]
m =  coef(bass_m)["m"]
bass.f = function(t,p,q){((p+q)**2/p)*exp(-(p+q)*t)/ (1+(q/p)*exp(-(p+q)*t))**2}

summary(bass_m)
```

```
##
## Formula: sales$sales ~ (m * (((p + q)^2/p) * exp(-(p + q) * t)))/(1 +
##     (q/p) * exp(-(p + q) * t))^2
##
## Parameters:
##    Estimate Std. Error t value Pr(>|t|)
## m 483.62573  667.27537   0.725    0.544
## p   0.01346    0.01600   0.841    0.489
## q   0.36223    0.13188   2.747    0.111
##
## Residual standard error: 1.524 on 2 degrees of freedom
##
## Number of iterations to convergence: 7
## Achieved convergence tolerance: 4.173e-06
```

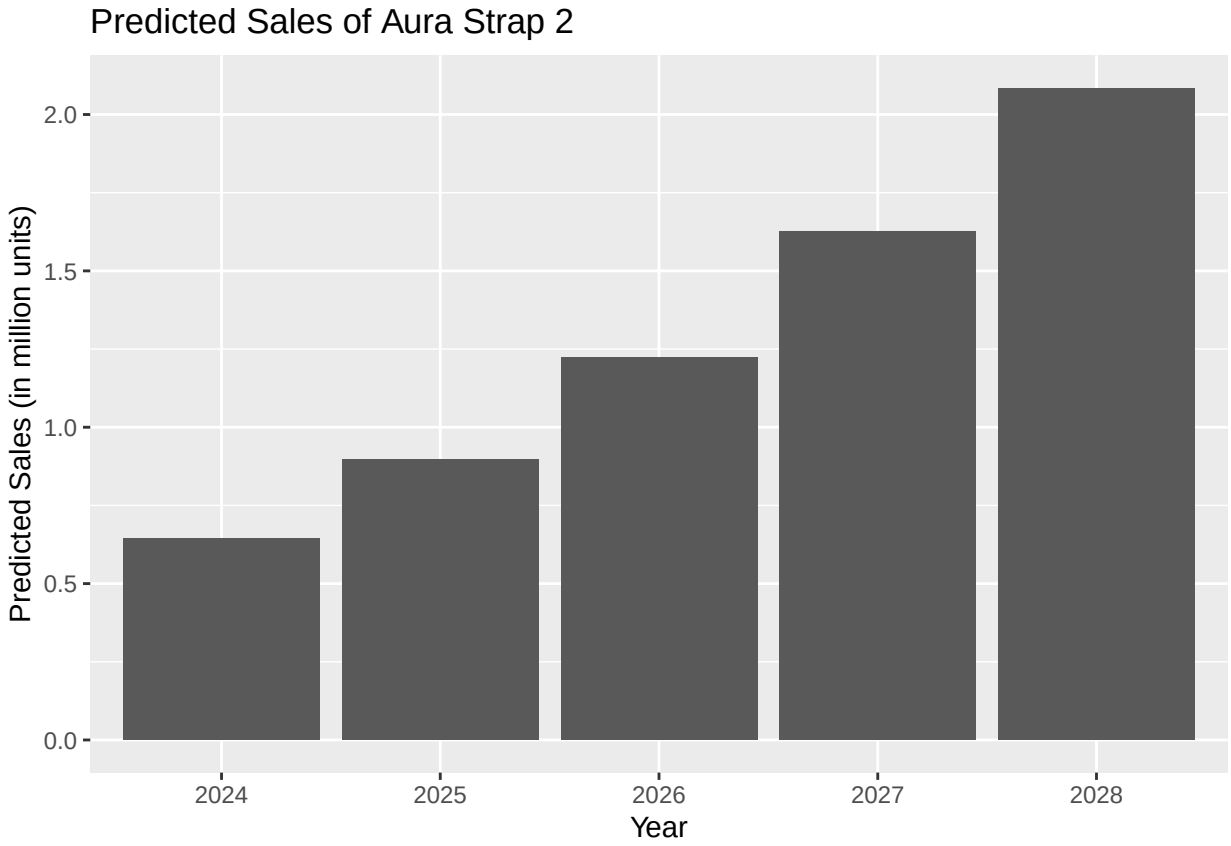
```
sales$pred = bass.f(sales$time, p, q) * m
```

```r
ggplot(data = sales, aes(x = year, y = sales)) +
  geom_bar(stat = 'identity') +
  xlab("Year") +
  ylab("Sales (Volume in mln units)")+
  ggtitle('Bass Modelling of Apple Watch sales (mln units)')+
  geom_point(mapping = aes(x=year, y=pred), color = 'red', size = 3)
```



Considering that our model predicts M = 483, which is above the number of iphone users. This score is above the market potential of apple watch. The score of market potential is unreliable so I will choose the M=34 as a Fermi Estimation stated above.

```r
m = 34
t = (1:5)
aura_sales = bass.f(t, p, q) * m
plot_data = data.frame(
  Year = 2024:2028,
  Predicted_Sales = aura_sales)

ggplot(data = plot_data, aes(x = as.factor(Year), y = aura_sales)) +
  geom_bar(stat = 'identity') +
  labs(title = 'Predicted Sales of Aura Strap 2',
       x = 'Year',
       y = 'Predicted Sales (in million units)')
```


Predicted Sales of Aura Strap 2
Predicted Sales (in million units)
2.0
1.5
1.0
0.5
0.0
2024
2025
2026
2027
2028
Year