Regression Analysis for Open Pipeline & Revenue

Hilary Lai 2020-08-13

```
library(readr)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v dplyr   1.0.2
## v tibble  3.0.4     v stringr 1.4.0
## v tidyr   1.1.2     v forcats 0.5.0
## v purrr   0.3.4

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(knitr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

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

```
options(digits=2)
options(scipen = 1)

#get the open peak annual value file downloaded from QV
open.pav <- read_csv("open_pav.csv", col_types = cols(Snapshot_Month = col_date(format = "%m/%d/%Y")))[,-7] %>%
  mutate(Fiscal_Mth = year(Snapshot_Month) * 100 + month(Snapshot_Month))
```

```
## Warning: Missing column names filled in: 'X7' [7]
```

```
rev <- read_csv("rev_by_mth.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Fiscal_Mth = col_double(),
##   End_Mkt_Segment = col_character(),
##   BU = col_character(),
##   Region = col_character(),
##   Director = col_character(),
##   Revenue = col_character()
## )
```

```
#align director and region
rev <- rev %>%
  mutate(Director = case_when(
      rev$Director == "Hong, SK" ~ "SHONG",
      rev$Director == "Hsu, Eric" ~ "EHSU2",
      rev$Director == "JD Kang" ~ "JKANG2",
```

```
      rev$Director == "Ted Park" ~ "TPARK",
      rev$Director == "Wan, Daryl" ~ "MWAN"),
      Region = case_when(
        Region == "Asean" ~ "AS",
        Region == "Australia" ~ "AL",
        Region == "India" ~ "IN",
        Region == "Korea" ~ "KR",
        Region == "Taiwan" ~ "TA"))

#fix revenue format
rev$Revenue <- gsub("[(]", "-", rev$Revenue)
rev$Revenue <- gsub("[),$]", "", rev$Revenue)

#summarise revenue by variables
rev <-  rev %>%
  group_by(End_Mkt_Segment, Fiscal_Mth, Director, Region, BU) %>%
  dplyr::summarise(Revenue = sum(as.numeric(Revenue)))
```

```
## `summarise()` regrouping output by 'End_Mkt_Segment', 'Fiscal_Mth', 'Director', 'Region' (override with `.groups`
```

```
#align column names
names(open.pav)[c(3,5)] <- c("End_Mkt_Segment", "BU")
dat <- full_join(rev, open.pav[,-6]) %>%
  filter(Fiscal_Mth >= 201811 & Fiscal_Mth <= 202007)
```

```
## Joining, by = c("End_Mkt_Segment", "Fiscal_Mth", "Director", "Region", "BU")
```

```
dat[is.na(dat)] <- 0
dat <- dat %>%
  filter(log(Revenue)>0, log(PAV)>0)
```

```
## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced

## Warning in log(Revenue): NaNs produced
```

Dataset

The dataset I use combines each year's snapshot of the **Peak Annual Value (PAV)** for open pipeline, with each year's **revenue value**.

The dataset contains the following fields:

- Fiscal month
- Director
- PAV
- Revenue

A partial look of part of the dataset:

```
library(knitr)
#get the final table
options(knitr.kable.NA = '')
kable(head(dat))
```

End_Mkt_Segment	Fiscal_Mth	Director	Region	BU	Revenue	PAV
AEG	201811	EHSU2	TA	ADEFTG	144815	2980700
AEG	201811	EHSU2	TA	AEGTG	178021	1588304
AEG	201811	EHSU2	TA	AEITG	42278	3827447
AEG	201811	EHSU2	TA	CHNTG	371	86528
AEG	201811	EHSU2	TA	COMTG	42146	675178
AEG	201811	EHSU2	TA	CSTG	34458	1511320

Thus, we can assume there is a linear relationship between the 2 variables.

ANOVA (Analysis of Variance) with all Variables

Next, I run an ANOVA analysis with (PAV^{(0.1)}) as the dependent variable, and (ln{(Revenue)}^2), BU, Region, and End Customer Segment as independent variables. The goal is to see whether there are differences in the means of Revenue value between each variable.

Note:

1. To keep all values positive for transformation, I only included data with PAV & Revenue > 0. The purpose of keeping all values positive is to keep the distribution of variables normal, so that it meets model requirements for regression analysis.

2. The 0.1 comes from Box-Cox Transformation of data. The purpose is to make data normal in order to meet the requirements for linear regression analysis. We will use this value from now on.

3. The reason I use (ln{(PAV)}^2) instead of (ln{(PAV)}), is that the transformation makes the relationship look more linear. Hence, I suspect there is a relationship between the 2 variables.

```
plot_grid(plot.1, plot.2, plot.3, ncol = 3)
```

Results:

```
#show the summary results
summary.aov(dat.fit)
```

```
##                        Df Sum Sq Mean Sq F value Pr(>F)    
## I((log(Revenue))^2)     1   1705    1705  6184.4 <2e-16 ***
## End_Mkt_Segment         6    146      24    88.5 <2e-16 ***
## Director                4     53      13    48.1 <2e-16 ***
## BU                      9     93      10    37.6 <2e-16 ***
## Residuals            4734   1305       0                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As shown, all variables have significant F values (p-vale < 0.001). Therefore, we can say that **there is enough evidence that the means of ((Revenue)^{(0.1)}) differ between different market segments, directors, and PAV values.**

Linear Regression Model A: 4 independent and 1 dependent variables

Next, I run linear regression analysis for each variable, including director, BU, and end market segment. The reason I did not use Region is that it is correlated to Director, which would violate linear regression requirements.

The linaer model should look something like this:

((PAV)^{(0.1)}) = (\beta_0) + (\beta_1) * (\ln(Revenue)^2) + … + (\epsilon)

Note: (\epsilon) = Error variable, (\beta_0) = intercept.

Results (A):

```
summary(dat.fit)
```

```
##
## Call:
## lm(formula = PAV^lamb ~ I((log(Revenue))^2) + End_Mkt_Segment +
##     Director + BU, data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1220 -0.3244  0.0061  0.3104  3.0530
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.365103   0.039470   85.26  < 2e-16 ***
## I((log(Revenue))^2)  0.010773   0.000207   52.00  < 2e-16 ***
## End_Mkt_SegmentASD  -0.186009   0.030880   -6.02  1.8e-09 ***
## End_Mkt_SegmentAUT   0.053323   0.029810    1.79  0.07372 .
## End_Mkt_SegmentCOM  -0.149330   0.028803   -5.18  2.3e-07 ***
## End_Mkt_SegmentCON  -0.221650   0.029135   -7.61  3.3e-14 ***
## End_Mkt_SegmentDHC  -0.529754   0.030647  -17.29  < 2e-16 ***
## End_Mkt_SegmentINS  -0.273139   0.027261  -10.02  < 2e-16 ***
## DirectorJKANG2       0.223293   0.043948    5.08  3.9e-07 ***
## DirectorMWAN        -0.245482   0.020686  -11.87  < 2e-16 ***
## DirectorSHONG       -0.055403   0.024300   -2.28  0.02265 *
## DirectorTPARK       -0.054038   0.038287   -1.41  0.15819
## BUAEGTG             -0.159089   0.038982   -4.08  4.6e-05 ***
## BUAEITG             -0.056911   0.029849   -1.91  0.05663 .
## BUCHNTG             -0.549558   0.048777  -11.27  < 2e-16 ***
## BUCOMTG              0.049774   0.029417    1.69  0.09071 .
## BUCSTG              -0.213615   0.032178   -6.64  3.5e-11 ***
## BUDHCTG             -0.035023   0.047137   -0.74  0.45752
## BUOTH               -0.905467   0.526144   -1.72  0.08533 .
## BUPPGTG              0.173434   0.029419    5.90  4.0e-09 ***
## BUPTPTG             -0.109628   0.029970   -3.66  0.00026 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.52 on 4734 degrees of freedom
## Multiple R-squared:  0.605,  Adjusted R-squared:  0.603
## F-statistic:  362 on 20 and 4734 DF,  p-value: <2e-16
```

The model has an adjusted (R^2) value of 0.60, which means that it is quite useful (60% of data can be represented by the equation). Hence, we can use it for forecasting current PAV given a revenue value.

Hence, we can get the equation:

**((PAV)^{(0.1)}) = (3.37) + (0.01) * (\ln(Revenue)^2) + (-0.19) * (ASD) + (0.05) * (AUT) + (-0.15) * (COM) + (-0.22) * (CON) + (-0.53) * (DHC) + (-0.27) * (INS) +(0.22) * (JKANG2) + (-0.25) * (MWAN) +(-0.06) * (SHONG) +(-0.05) * (TPARK) + (-0.16) * (AEGTG) + (-0.06) * (AEITG) + (-0.55) * (CHNTG) + (0.05) * (COMTG) + (-0.21) * (CSTG)

- (-0.04) * (DHCTG) + (-0.91) * (OTH) + (0.17) * (PPGTG) + (-0.11) * (PTPTG)**

Note: For both director and segment categories, they are presented as dummy variables (either 0 or 1). For example: if Director = Daryl, then MWAN = 1, JKANG2 = SHONG = TPARK = 0.

Regression Plots (A)

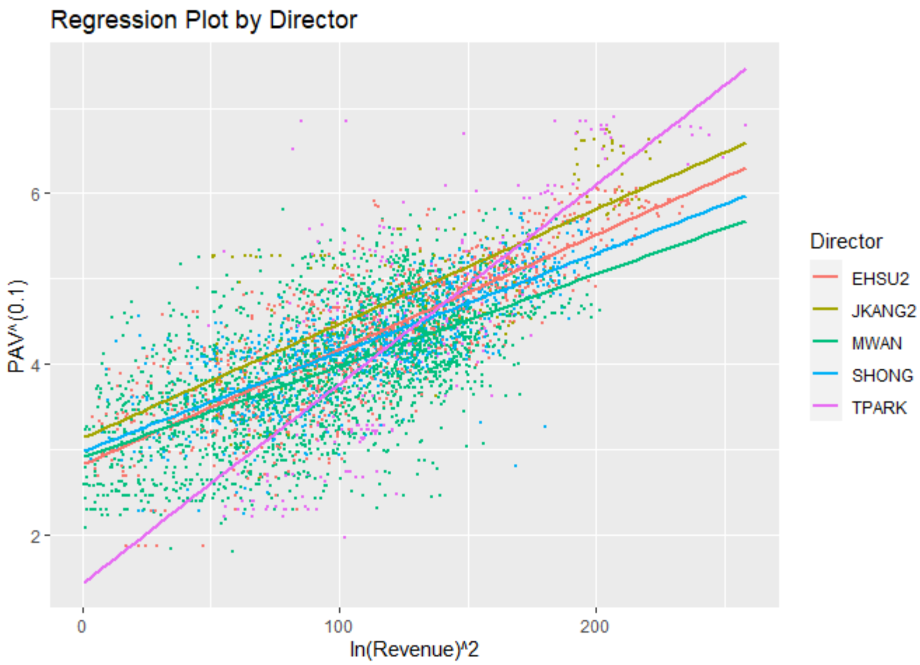
We can draw regression lines using the model above.

By Director

```
#plot by director
plot.dir <- dat %>%
  ggplot(aes(x = log(Revenue)^2 , y = PAV^lamb, color = Director)) +
  geom_point(size = 0.3) +
   xlab("ln(Revenue)^2")+
  ylab("PAV^(0.1)")+
  ggtitle("Regression Plot by Director") +
```

```
  stat_smooth(method = "lm", aes(fill = Director), se=FALSE, fullrange=TRUE)
plot.dir
```

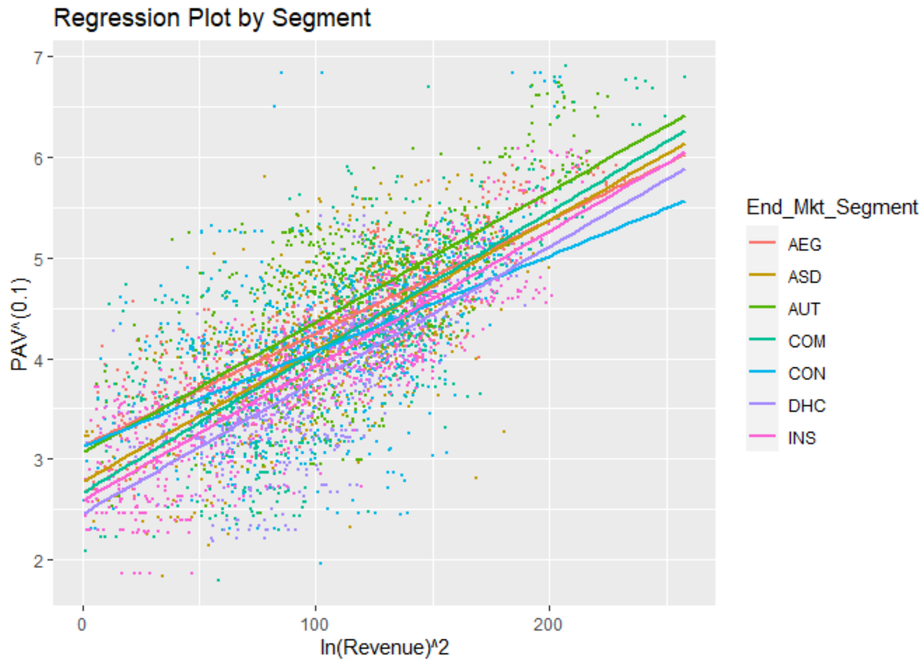
```
## `geom_smooth()` using formula 'y ~ x'
```



By Segment

```
#plot by segment
plot.seg <- dat %>%
  ggplot(aes(x = log(Revenue)^2 , y = PAV^lamb, color = End_Mkt_Segment)) +
  geom_point(size = 0.3) +
   xlab("ln(Revenue)^2")+
  ylab("PAV^(0.1)")+
    ggtitle("Regression Plot by Segment") +
  stat_smooth(method = "lm", aes(fill = End_Mkt_Segment), se=FALSE, fullrange=TRUE)
plot.seg
```

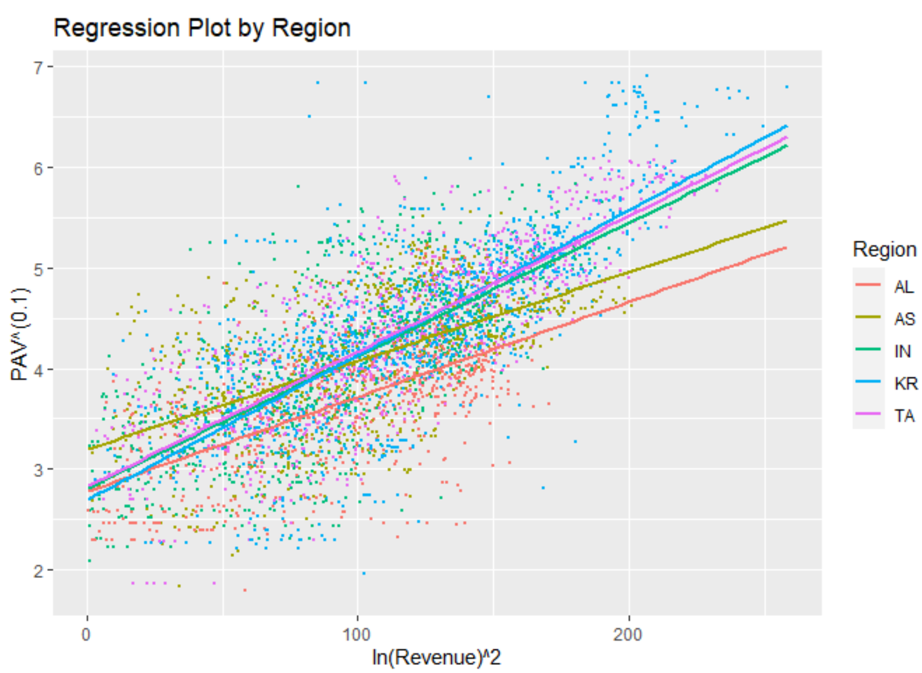
```
## `geom_smooth()` using formula 'y ~ x'
```



By Region

```
#plot by region
plot.reg <- dat %>%
  ggplot(aes(x = log(Revenue)^2, y = PAV^lamb, color = Region)) +
  geom_point(size = 0.3) +
   xlab("ln(Revenue)^2")+
  ylab("PAV^(0.1)")+
    ggtitle("Regression Plot by Region") +
  stat_smooth(method = "lm", aes(fill = Region), se=FALSE, fullrange=TRUE)
plot.reg
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Linear Regression Model B: 1 independent and 1 dependent variables

To simplify the model, let’s only use (\ln(Revenue)^2) and ((PAV)^{0.1}) as prediction.

Results (B):

```
goal.tbl <- read_csv("goal_seg_reg.csv") #read again
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   `Region (DAR)` = col_character(),
##   Segment = col_character(),
##   FY21E = col_double()
## )
```

```
goal <- goal.tbl[[3]]
options(scipen=1)

goal.fit <- lm(PAV^lamb ~ I((log(Revenue))^2), dat)
summary(goal.fit)
```

```
##
## Call:
## lm(formula = PAV^lamb ~ I((log(Revenue))^2), data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1666 -0.3697  0.0053  0.3599  2.9604
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         2.781902   0.020894   133.2   <2e-16 ***
## I((log(Revenue))^2) 0.012965   0.000182    71.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.58 on 4753 degrees of freedom
## Multiple R-squared:  0.516,  Adjusted R-squared:  0.516
## F-statistic: 5.07e+03 on 1 and 4753 DF,  p-value: <2e-16
```

As shown, there is strong evidence to show that there exists a linear relationship between (\ln(Revenue)^2) and (PAV^{0.1}). Also, about 52% of data can be explained by the model.

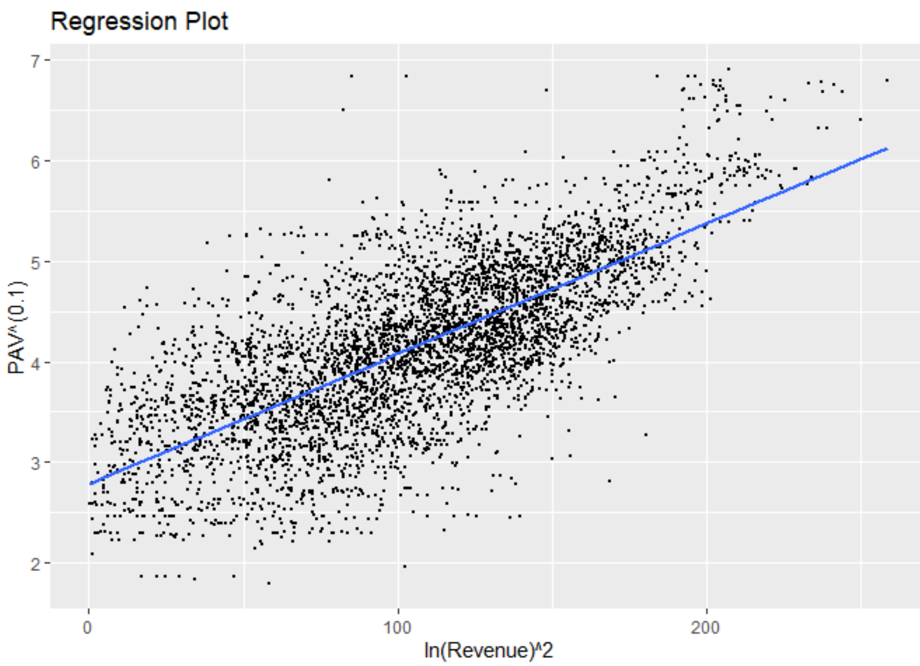
We get the equation: ((PAV)^{(0.1)}) = 2.78 + 0.01 * (\ln(Revenue)^2).

Regression Plot (B)

And the plot is shown below:

```
plot.goal <- dat %>%
  ggplot(aes(x = log(Revenue)^2 , y = PAV^lamb)) +
  geom_point(size = 0.3) +
   xlab("ln(Revenue)^2")+
  ylab("PAV^(0.1)")+
  ggtitle("Regression Plot") +
  stat_smooth(method = "lm", se=FALSE, fullrange=TRUE)
plot.goal
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Forecast and Predictions

1. Example A: Revenue Forecast for a given PAV with Multiple Settings

In March 2019, an open pipeline snapshot for SK as director, PTPTG as BU, and COM as segment has a total PAV snapshot value of $20M. How would the revenue be like?

To calculate the revenue, we can apply the formula from **Model A**:

((20000000)^{(0.1)}) = (3.37) + (0.01) * (\ln(Revenue)^2) + (-0.19) * 0 + (0.05) * 0 + (-0.15) * 1 + (-0.22) * 0 +(-0.53) * 0 + (-0.27) * 0 +(0.22) * 0 + (-0.25) * 0 +(-0.06) * 1 +(-0.05) * 0 + (-0.16) * 0 + (-0.06) * 0 + (-0.55) * 0 + (0.05) * 0 + (-0.21) * 0 + (-0.04) * 0 + (-0.91) * 0 + (0.17) * 1 + (-0.11) * 0

As a result, the revenue forecast would be **$1,202,604**.

2. Example B: Get Predicted PAV to achieve Revenue Goals.

For 2021, there is a revenue goal for each segment and region.

To calculate the PAV we need in order to achieve the goals, we use the equation from **Model B**:

((PAV)^{(0.1)}) = 2.78 + 0.01 * (\ln(Revenue)^2).

Next, we put in all the revenue goal numbers in the equation, and we can get the table below:

```
options(knitr.kable.NA = "")
pred.tbl <- pred.tbl %>%
  add_row(`Region (DAR)`= "Total", FY21E = sum(as.numeric(pred.tbl$FY21E)), Expected_PAV = as.numeric(sum(pred.tbl$Ex
pred.tbl$FY21E <-  pred.tbl$FY21E %>%
  format(big.mark=",")
```

```
pred.tbl$Expected_PAV <-  pred.tbl$Expected_PAV %>%
  format(big.mark=",")

kable(pred.tbl)
```

Region (DAR)	Segment	FY21E	Expected_PAV
AEG	AL	1,159,003	17,810,639
	AS	2,318,084	28,583,798
	IN	10,051,328	77,690,144
	KR	12,047,000	87,859,055
	TA	41,685,588	203,212,056
ASD	AL	3,975,351	41,304,604
	AS	1,531,602	21,540,945
	IN	11,315,675	84,201,720
	KR	4,744,000	46,597,188
	TA	15,000,000	101,945,235
AUT	AL	771,751	13,500,610
	AS	2,919,645	33,459,695
	IN	1,512,997	21,362,037
	KR	82,243,000	320,227,117
	TA	1,987,122	25,730,380
COM	AL	2,664,602	31,435,728
	AS	4,775,525	46,808,109
	IN	7,271,348	62,334,642
	KR	67,944,000	281,899,898
	TA	19,876,555	123,348,676
CON	AL	4,241,768	43,173,379
	AS	3,767,557	39,819,064
	IN	497,734	10,020,569
	KR	32,722,000	172,677,844
	TA	20,389,000	125,490,245
DHC	AL	3,492,903	37,814,613
	AS	140,997	4,283,674
	IN	2,827,465	32,734,930
	KR	16,097,000	106,939,282
	TA	6,741,945	59,207,580
INS	AL	3,974,625	41,299,458
	AS	22,338,594	133,479,886

Region (DAR)	Segment	FY21E	Expected_PAV
	IN	5,133,854	49,175,026
	KR	25,440,000	145,727,566
	TA	32,389,792	171,495,432
Total		475,989,410	2,844,190,825

In sum, there is a total of 2.8B for the revenue goals to be achieved.