

Levent Kayın Final Project Report

For my Final Project I have analyzed Eddie's marketing data and created a basic mixed marketing model to provide recommendations and dueTo's for their marketing channels. All my results can be seen in the attached csv file titled "LeventKayın_Final_Recommendations".

Modeling

After cleaning the data I used regression to estimate dueTo's using the formula: $\text{Revenue} \sim \text{costNews} + \text{costTV} + \text{costRadio} + \text{costSearch} + \text{costBanner} + \text{costFlyer} + \text{costDiscount}$.

This formula helped me relate the revenue to the associated costs of different marketing channels. I've tried using reach or transformed variables but did not see significant changes in model performance and decided to use the non-transformed variables for simplicity. Below you can see model performance.

Call:

```
lm(formula = Revenue ~ costNews + costTV + costRadio + costSearch +  
    costBanner + costFlyer + costDiscount, data = clean)
```

Residuals:

Min	1Q	Median	3Q	Max
-4424988	-322136	-113206	-9838	323279154

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-79065.4411	69318.2975	-1.141	0.2540
costNews	136.0409	30.9481	4.396	0.0000110742 ***
costTV	4339.5881	814.3777	5.329	0.0000000996 ***
costRadio	3888.3670	5741.9175	0.677	0.4983
costSearch	1640.6261	109.7732	14.946	< 0.0000000000000002 ***
costBanner	72015.9805	30798.5903	2.338	0.0194 *
costFlyer	0.5085	2.1198	0.240	0.8104
costDiscount	652.4297	402.2357	1.622	0.1048

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3584000 on 31387 degrees of freedom

Multiple R-squared: 0.0101, Adjusted R-squared: 0.009883

F-statistic: 45.77 on 7 and 31387 DF, p-value: < 0.00000000000000022

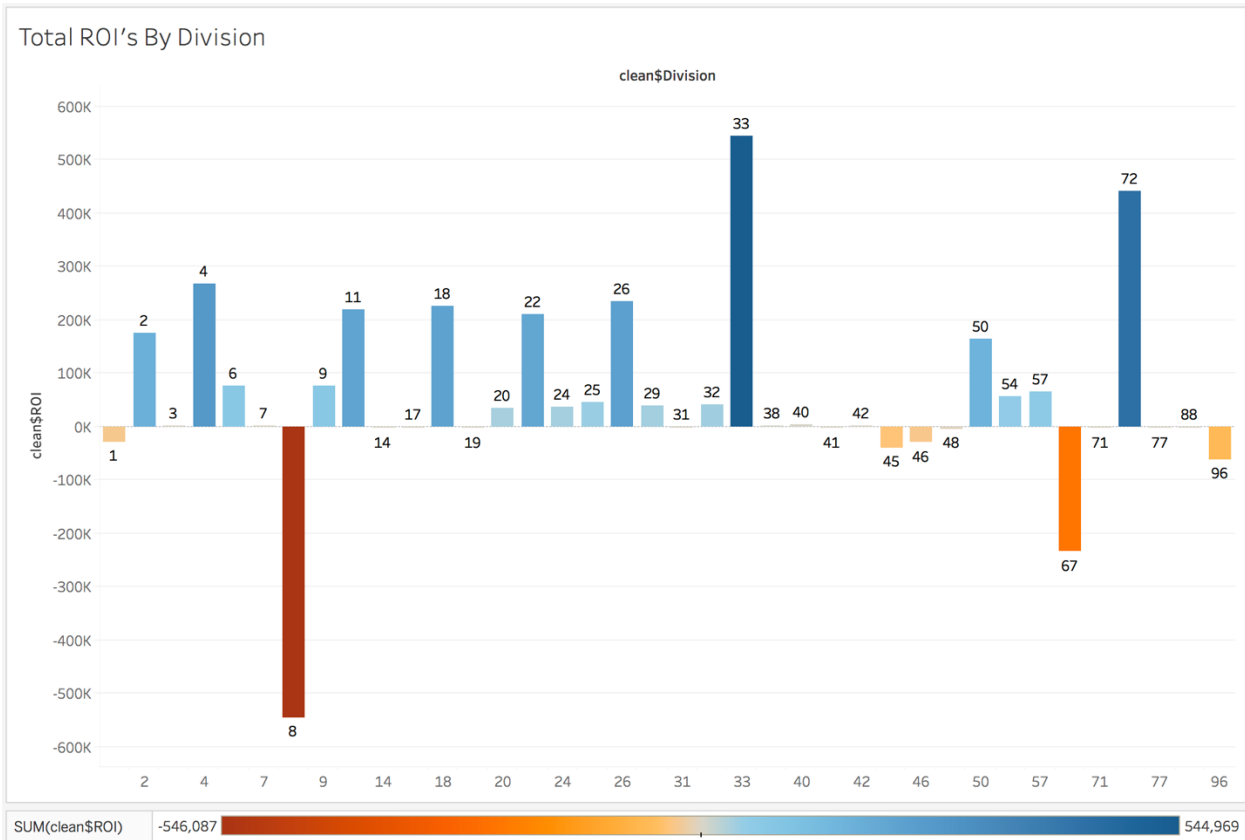
By looking at the coefficients below we can assign weights (dueTo's) to marketing channels. From the summary we see that Revenues are highly correlated with newspaper adds, TV ads and most of all paid search adds. These are all positively correlated with revenues (as expected).

Next, I've calculated the revenue (and ROI) based on the dueTo's from my model. The ROI's can be seen in the final_recommendation file by division and line.

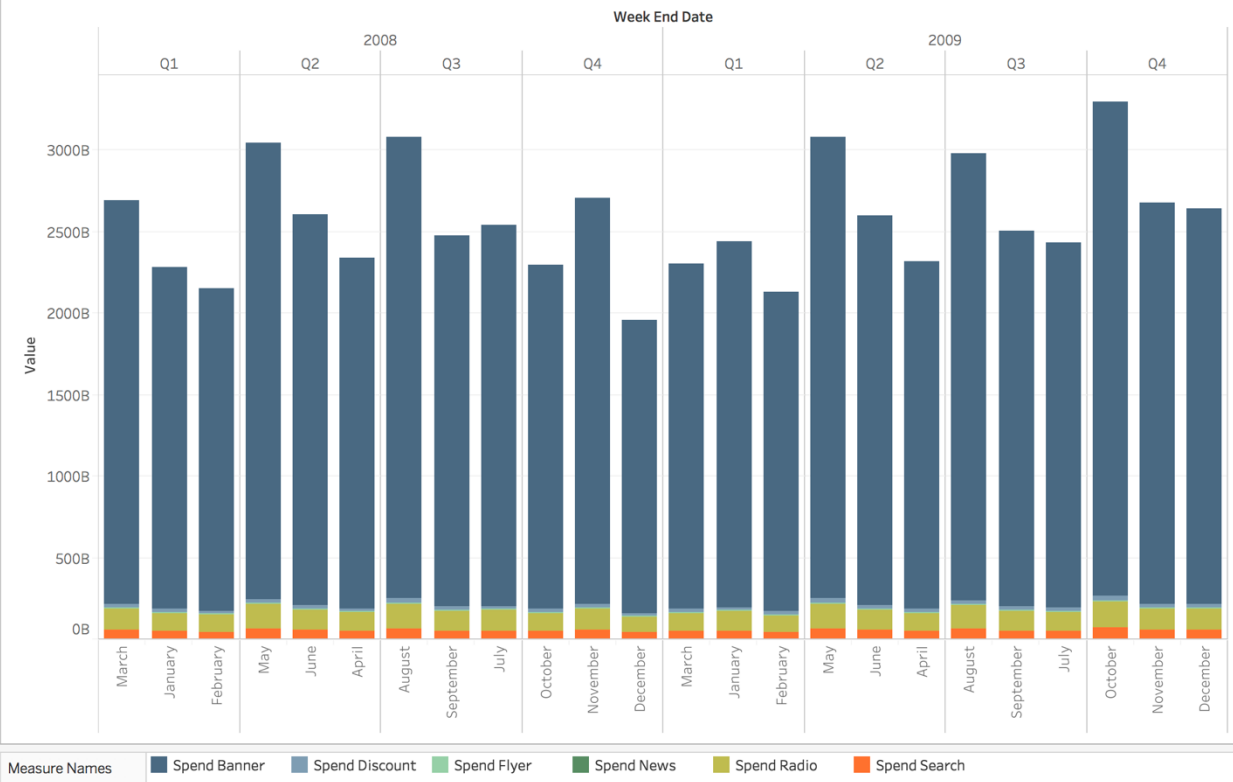
Finally, I used the model to assign new recommended spending levels based on the total marketing spend. Again, these values (called delta) on how much spending should be increased or decreased can be found on the output file. Also, all of the output file can be aggregated on division, line or date to get insights.

Given more time I would add seasonal variables to my model in order to capture holidays and other seasonal effects. I would also like to add addstocks into the model to improve predictive power.

Output Charts



Monthly Suggested Spending



Appendix - Analysis Code and Output

```
> setwd("/Users/leventkayin/Desktop/School/Marketing Analytics/Final Project/")
> options(scipen=999)
> ## Read in the data ----
> sales = read.csv("SalesDigitalPrint.csv")
>
> divisionRadio = read.csv("divisionRadio.csv")
> divisionTV = read.csv("divisionTV.csv")
>
> storeRadio = read.csv("storeRadio.csv")
> storeTV = read.csv("storeTV.csv")
>
> ## Introduce Cost Columns ----
> divisionRadio$radioReach = (0.65*(1-exp((-
divisionRadio$Promotional.Division.Radio..GRPs./100)*0.43)))
> divisionRadio$costRadio = divisionRadio$radioReach*300
>
> divisionTV$TVReach = (0.93*(1-exp((-divisionTV$Promotional.Division.TV..GRPs./100)*0.45)))
> divisionTV$costTV = divisionTV$TVReach*1000
>
> storeTV$StoreTvRech = (0.93*(1-exp((-storeTV$Branding.Store.TV..GRPs./100)*0.45)))
> storeTV$costStoreTv = storeTV$StoreTvRech*1000
>
> storeRadio$StoreRadioReach = (0.65*(1-exp((-
storeRadio$Branding.Store.Radio..GRPs./100)*0.43)))
> storeRadio$costStoreRadio = storeRadio$StoreRadioReach*300
>
> sales$costBanner = (sales$Banner.Ad..Impressions./1000)*2.5
> sales$costSearch = (sales$Paid.Search..Impressions./1000)*9.5
>
> ## Divide by division and by store costs into list items ----
> weeks = unique(sales$Week.Ending)
> divisions = unique(sales$Division)
>
> clean = data.frame()
>
> for (i in seq(length(weeks))) {
+
+   week = weeks[i]
+   weeklysales = subset(sales,sales$Week.Ending == week)
+
+   numflyers = length(subset(weeklysales,weeklysales$Flyer..Pages. != 0))
+   flyerpages = sum(weeklysales$Flyer..Pages.)
+   flyercost = ceiling(flyerpages/4)*50000
+
+   numpages = length(subset(weeklysales,weeklysales$Newspaper..Pages. != 0))
+   newspages = sum(weeklysales$Newspaper..Pages.)
+   newscost = ceiling(newspages*0.5)*40000
+
+   weeklysales$costFlyer = 0
+   weeklysales$costFlyer[weeklysales$Flyer..Pages.>0] = flyercost/numflyers
+
+   weeklysales$costNews = 0
+   weeklysales$costNews[weeklysales$Newspaper..Pages.>0] = newscost/numpages
+
+   weeklyDivTV = subset(divisionTV,divisionTV$Week.Ending == week)
```

```

+   weeklyDivRadio = subset(divisionRadio,divisionRadio$Week.Ending == week)
+
+   weeklysales$costTV = 0
+   weeklysales$costRadio = 0
+   weeklysales$reachTV = 0
+   weeklysales$reachRadio = 0
+
+   weeklysales$costStoreRadio = 0
+   weeklysales$costStoreTV = 0
+
+   if (week %in% storeRadio$Week) {
+     weeklyStoreRadio = subset(storeRadio,storeRadio$Week == week)
+     weeklysales$costStoreRadio =
weeklyStoreRadio$costStoreRadio/length(weeklysalses$Division)
+   }
+
+   if (week %in% storeTV$Week) {
+     weeklyStoreTV = subset(storeTV,storeTV$Week == week)
+     weeklysales$costStoreTV = weeklyStoreTV$costStoreTv/length(weeklysalses$Division)
+   }
+
+   for (j in seq(length(unique(sales$Division)))) {
+
+     div = unique(sales$Division)[j]
+
+     if (div %in% weeklyDivTV$Division) {
+       weeklyDivTvCost = weeklyDivTV$costTV[weeklyDivTV$Division == div]
+       weeklysales$costTV[weeklysalses$Division == div] =
weeklyDivTvCost/sum(weeklysalses$Division == div)
+
+       weeklyDivTvreach = weeklyDivTV$TVReach[weeklyDivTV$Division == div]
+       weeklysales$reachTV[weeklysalses$Division == div] =
weeklyDivTvreach/sum(weeklysalses$Division == div)
+     }
+
+     if (div %in% weeklyDivRadio$Division) {
+       weeklyDivRadioCost = weeklyDivRadio$costRadio[weeklyDivRadio$Division == div]
+       weeklysales$costRadio[weeklysalses$Division == div] =
weeklyDivRadioCost/sum(weeklysalses$Division == div)
+
+       weeklyDivRadioreach = weeklyDivRadio$radioReach[weeklyDivRadio$Division == div]
+       weeklysales$reachRadio[weeklysalses$Division == div] =
weeklyDivRadioreach/sum(weeklysalses$Division == div)
+     }
+   }
+ }
+
+ clean = rbind(clean,weeklysalses)
+ }
+
>
> write.csv(clean,"Marketing_Final_Clean_Data")
> ## Explarotory analysis ----
> library(ggplot2)
>
> str(clean)
'data.frame': 31395 obs. of  23 variables:
 $ Division      : int  1 1 1 1 1 1 1 1 1 1 ...
 $ Line          : int  1012 1015 1020 1022 1029 1060 1066 1071 1072 1073 ...
 $ Week.Ending   : int  39452 39452 39452 39452 39452 39452 39452 39452 39452 39452 ...

```

```

$ Cost.of.Goods      : num  0 5996 0 0 78797 ...
$ Revenue            : num  0 6662 0 0 66978 ...
$ Sales.Volume       : int  0 18 0 0 131 0 0 0 22 0 ...
$ Unit.Cost          : num  332 333 472 587 602 ...
$ Unit.Price         : num  449 411 593 641 682 ...
$ Discounted.Unit.Price : num  359 370 593 448 511 ...
$ Banner.Ad..Impressions. : int 1022 951 1092 1032 1022 920 1022 1082 1022 1052 ...
$ Paid.Search..Impressions.: int 3465 3498 210 16218 24360 2 24 2 2 0 ...
$ Flyer..Pages.      : num  0.000414 0.127003 0.006243 0.000148 0.000422 ...
$ Newspaper..Pages.  : num  0 0 0 0.000292 0.00676 ...
$ costBanner         : num  2.56 2.38 2.73 2.58 2.56 ...
$ costSearch         : num  32.92 33.23 1.99 154.07 231.42 ...
$ costFlyer          : num  26667 26667 26667 26667 26667 ...
$ costNews           : num  0 0 0 2667 2667 ...
$ costTV             : num  0 0 0 0 0 0 0 0 0 ...
$ costRadio          : num  0 0 0 0 0 0 0 0 0 ...
$ reachTV            : num  0 0 0 0 0 0 0 0 0 ...
$ reachRadio         : num  0 0 0 0 0 0 0 0 0 ...
$ costStoreRadio     : num  0 0 0 0 0 0 0 0 0 ...
$ costStoreTV        : num  0 0 0 0 0 0 0 0 0 ...
> #clean$Date = as.Date(clean$Week.Ending, origin = "1899-12-30")
>
> clean$costDiscount = clean$Unit.Price-clean$Discounted.Unit.Price
>
> clean$TotalMarketingCost =clean$costBanner +
+                           clean$costSearch +
+                           clean$costFlyer +
+                           clean$costNews +
+                           clean$costTV +
+                           clean$costRadio +
+                           clean$costStoreRadio +
+                           clean$costStoreTV
>
> clean$profits = clean$Revenue-clean$Cost.of.Goods
>
> clean$reachBanner = sqrt(clean$costBanner)
> clean$reachSearch = sqrt(clean$costSearch)
>
> clean$ROI = (clean$profits-clean$TotalMarketingCost)/clean$TotalMarketingCost
>
> division_summary = data.frame()
>
> for (l in seq(length(weeks))) {
+
+   week = weeks[l]
+
+   for (k in seq(length(divisions))) {
+
+     div = divisions[k]
+     weekDat = subset(clean,clean$Week.Ending == week)
+     divDat = subset(weekDat,weekDat$Division == div)
+     division_roll = as.data.frame(t(apply(divDat, 2, sum)))
+     division_roll$Division = div
+     division_roll$Week.Ending = week
+     division_roll$Line = NULL
+
+     division_summary = rbind(division_summary,division_roll)
+   }
+ }
>

```

```

> division_summary$ROI = (division_summary$profits-
division_summary$TotalMarketingCost)/division_summary$TotalMarketingCost
>
>
> model = lm(Revenue ~ costNews + costTV + costRadio + costSearch + costBanner +costFlyer +
costDiscount, data = clean)
> coefs = model$coefficients
> coefs
      (Intercept)      costNews      costTV      costRadio      costSearch      costBanner
costFlyer
-79065.4410586    136.0408866   4339.5880881   3888.3670110   1640.6260787   72015.9804671
0.5085133
costDiscount
  652.4297450
> coefs_opt = coefs[2:8]
> summary(model)

```

Call:

```
lm(formula = Revenue ~ costNews + costTV + costRadio + costSearch +
costBanner + costFlyer + costDiscount, data = clean)
```

Residuals:

Min	1Q	Median	3Q	Max
-4424988	-322136	-113206	-9838	323279154

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-79065.4411	69318.2975	-1.141	0.2540
costNews	136.0409	30.9481	4.396	0.0000110742 ***
costTV	4339.5881	814.3777	5.329	0.0000000996 ***
costRadio	3888.3670	5741.9175	0.677	0.4983
costSearch	1640.6261	109.7732	14.946	< 0.0000000000000002 ***
costBanner	72015.9805	30798.5903	2.338	0.0194 *
costFlyer	0.5085	2.1198	0.240	0.8104
costDiscount	652.4297	402.2357	1.622	0.1048

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3584000 on 31387 degrees of freedom

Multiple R-squared: 0.0101, Adjusted R-squared: 0.009883

F-statistic: 45.77 on 7 and 31387 DF, p-value: < 0.00000000000000022

```

>
> clean$revNews = clean$Revenue*coefs[2]
> clean$revTV = clean$Revenue*coefs[3]
> clean$revRadio = clean$Revenue*coefs[4]
> clean$revSearch = clean$Revenue*coefs[5]
> clean$revBanner = clean$Revenue*coefs[6]
> clean$revFlyer = clean$Revenue*coefs[7]
> clean$revDiscount = clean$Revenue*coefs[8]
>
> clean$ROINews = clean$revNews/clean$costNews
> clean$ROINews[clean$ROINews == Inf] = 0
> clean$ROINews[is.nan(clean$ROINews)] = 0
>
> clean$ROITV = clean$revTV/clean$costTV
> clean$ROITV[clean$ROITV == Inf] = 0
> clean$ROITV[is.nan(clean$ROITV)] = 0
>
> clean$ROIRadio = clean$revRadio/clean$costRadio
> clean$ROIRadio[clean$ROIRadio == Inf] = 0

```

```

> clean$ROIRadio[is.nan(clean$ROIRadio)] = 0
>
> clean$ROIsearch = clean$revSearch/clean$costSearch
> clean$ROIsearch[clean$ROIsearch == Inf] = 0
> clean$ROIsearch[is.nan(clean$ROIsearch)] = 0
>
> clean$ROIBanner = clean$revBanner/clean$costBanner
> clean$ROIBanner[clean$ROIBanner == Inf] = 0
> clean$ROIBanner[is.nan(clean$ROIBanner)] = 0
>
> clean$ROIFlyer = clean$revFlyer/clean$costFlyer
> clean$ROIFlyer[clean$ROIFlyer == Inf] = 0
> clean$ROIFlyer[is.nan(clean$ROIFlyer)] = 0
>
> clean$ROIDiscount = clean$revDiscount/clean$costDiscount
> clean$ROIDiscount[clean$ROIDiscount == Inf] = 0
> clean$ROIDiscount[is.nan(clean$ROIDiscount)] = 0
>
> current_spending = as.data.frame(cbind(clean$costBanner,
+                                       clean$costSearch,
+                                       clean$costFlyer,
+                                       clean$costNews,
+                                       clean$costTV,
+                                       clean$costRadio,
+                                       clean$costDiscount))
> colnames(current_spending) =
c("costBanner", "costSearch", "costFlyer", "costNews", "costTV", "costRadio", "costDiscount")
>
> proposed_spending = data.frame(spendTotal = clean$TotalMarketingCost)
> proposed_spending$spendNews = clean$TotalMarketingCost*coefs_opt[1]
> proposed_spending$spendTV = clean$TotalMarketingCost*coefs_opt[2]
> proposed_spending$spendRadio = clean$TotalMarketingCost*coefs_opt[3]
> proposed_spending$spendSearch = clean$TotalMarketingCost*coefs_opt[4]
> proposed_spending$spendBanner = clean$TotalMarketingCost*coefs_opt[5]
> proposed_spending$spendFlyer = clean$TotalMarketingCost*coefs_opt[6]
> proposed_spending$spendDiscount = clean$TotalMarketingCost*coefs_opt[7]
>
> suggestion = as.data.frame(cbind(current_spending, proposed_spending))
> suggestion$deltaNews = suggestion$spendNews-suggestion$costNews
> suggestion$deltaTV = suggestion$spendTV-suggestion$costTV
> suggestion$deltaRadio = suggestion$spendRadio-suggestion$costRadio
> suggestion$deltaSearch = suggestion$spendSearch-suggestion$costSearch
> suggestion$deltaBanner = suggestion$spendBanner-suggestion$costBanner
> suggestion$deltaFlyer = suggestion$spendFlyer-suggestion$costFlyer
> suggestion$deltaDiscount = suggestion$spendDiscount-suggestion$costDiscount
>
> final_suggestions = cbind.data.frame(clean$Division, clean$Line, as.Date(clean$Week.Ending,
origin = "1899-12-30"),
+
clean$ROI, clean$ROIBanner, clean$ROIDiscount, clean$ROIFlyer, clean$ROInews,
+
clean$ROIRadio, clean$ROIsearch, clean$ROITV, suggestion)
> colnames(final_suggestions)[3] = c("WeekEndDate")
>
> summary(final_suggestions)
clean$Division    clean$Line      WeekEndDate      clean$ROI      clean$ROIBanner
Min.   : 1.00    Min.   : 1012    Min.   :2008-01-05    Min.   : -228733.80    Min.   :
0
1st Qu.: 8.00    1st Qu.: 8026    1st Qu.:2008-07-05    1st Qu.:   -1.00    1st Qu.:
2805684
Median :24.00    Median :24091    Median :2009-01-03    Median :   -1.00    Median :
152990509

```


Mean :30.86	Mean :30905	Mean :2009-01-03	Mean : 64.22	Mean :
26034905443				
3rd Qu.:46.00	3rd Qu.:46040	3rd Qu.:2009-07-04	3rd Qu.: -0.95	3rd Qu.:
3517163405				
Max. :96.00	Max. :96091	Max. :2010-01-02	Max. : 128270.14	Max.:
:40201802203500				
clean\$ROIDiscount	clean\$ROIFlyer	clean\$ROINews	clean\$ROIRadio	
clean\$ROISearch				
Min. : 0	Min. : 0.000	Min. : 0	Min. : 0	Min. :
0				
1st Qu.: 0	1st Qu.: 0.000	1st Qu.: 0	1st Qu.: 0	1st Qu.:
0				
Median : 41103	Median : 0.015	Median : 0	Median : 0	Median :
43452				
Mean : 60068359	Mean : 4.933	Mean : 1964	Mean : 137197053	Mean :
415648914				
3rd Qu.: 2517074	3rd Qu.: 0.583	3rd Qu.: 0	3rd Qu.: 0	3rd Qu.:
2617830				
Max. :99720700075	Max. :5491.727	Max. :16528316	Max. :300548867323	Max.:
:1472415249550				
clean\$ROITV	costBanner	costSearch	costFlyer	costNews
costTV				
Min. : 0	Min. :0.2775	Min. : 0.0000	Min. : 0	Min. : 0.0
Min. : 0.00				
1st Qu.: 0	1st Qu.:0.6375	1st Qu.: 0.5035	1st Qu.:26667	1st Qu.: 0.0
1st Qu.: 0.00				
Median : 0	Median :1.0250	Median : 23.3130	Median :26667	Median : 0.0
Median : 0.00				
Mean : 20176104	Mean :1.1981	Mean : 101.9689	Mean :25017	Mean : 169.4
Mean : 12.54				
3rd Qu.: 0	3rd Qu.:1.6875	3rd Qu.: 115.9712	3rd Qu.:30000	3rd Qu.: 0.0
3rd Qu.: 0.00				
Max. :31626288102	Max. :3.2725	Max. :2745.0915	Max. :33333	Max. :5333.3
Max. :628.07				
costRadio	costDicount	spendTotal	spendNews	spendTV
spendRadio				
Min. : 0.000	Min. : 0.000	Min. : 0.28	Min. : 38	Min. : 1226
Min. : 1098				
1st Qu.: 0.000	1st Qu.: 1.996	1st Qu.:26669.39	1st Qu.:3628128	1st Qu.:115734187
1st Qu.:103700394				
Median : 0.000	Median : 8.229	Median :29350.32	Median :3992843	Median :127368292
Median :114124810				
Mean : 1.666	Mean : 27.906	Mean :25303.65	Mean :3442331	Mean :109807422
Mean : 98389881				
3rd Qu.: 0.000	3rd Qu.: 22.247	3rd Qu.:30096.67	3rd Qu.:4094378	3rd Qu.:130607156
3rd Qu.:117026904				
Max. :92.691	Max. :320.359	Max. :39594.17	Max. :5386426	Max. :171822391
Max. :153956667				
spendSearch	spendBanner	spendFlyer	spendDiscount	deltaNews
deltaTV				
Min. : 463	Min. : 20345	Min. : 0.144	Min. : 184	Min. :
38 Min. : 1226				
1st Qu.:43754504	1st Qu.:1920622605	1st Qu.:13561.741	1st Qu.:17399906	1st
Qu.:3628128	1st Qu.:115734187			
Median :48152898	Median :2113691952	Median :14925.026	Median :19149021	Median
:3990177	Median :127368292			
Mean :41513830	Mean :1822267229	Mean :12867.242	Mean :16508855	Mean
:3442162	Mean :109807410			
3rd Qu.:49377384	3rd Qu.:2167441289	3rd Qu.:15304.557	3rd Qu.:19635964	3rd
Qu.:4094373	3rd Qu.:130607156			

```

Max.      :64959229   Max.      :2851413014   Max.      :20134.161   Max.      :25832415   Max.
:5381093   Max.      :171822345
  deltaRadio      deltaSearch      deltaBanner      deltaFlyer      deltaDiscount
Min.      :    1098   Min.      :    463   Min.      :   20344   Min.      : -16383   Min.      :    1
1st Qu.:103700394   1st Qu.:43754503   1st Qu.:1920622604   1st Qu.: -14728   1st Qu.:17399872
Median :114124810   Median :48152881   Median :2113691951   Median : -13106   Median :19149021
Mean    : 98389880   Mean     :41513728   Mean     :1822267228   Mean     : -12149   Mean     :16508827
3rd Qu.:117026904   3rd Qu.:49377288   3rd Qu.:2167441288   3rd Qu.: -12884   3rd Qu.:19635941
Max.     :153956663   Max.     :64958351   Max.     :2851413013   Max.      : 2712   Max.     :25832413
> write.csv(final_suggestions,"LeventKayın_Final_Recommendations")

```