TEXT BASED SIGNALS

RUSTEM SHINKARUK

UCLA Anderson
THINK IN THE NEXT

CONTENT

UCLA Anderson
THINK IN THE NEXT

INTRODUCTION

1. INTRODUCTION

- Evaluate, clean and analyze the data(comments and reviews) from Amazon
- Goal:
  - Predict the behavior of returns of individual companies based on the general sentiment which people exhibit through their reviews
- General assumption and intuition:
  - Company expected to perform better if the consensus over a product is relatively positive
  - Supply and demand(Intended to buy products that have good quality reviews more often than that have negative or no reviews)
  - Increase in future sales lead to increasing in earnings and stock prices

UCLAAnderson
THINK IN THE NEXT

PREMILINARY ANALYSIS

3A. DATA OVERVIEW

- Data: “Reviews and Products Database” of Amazon for the period from May 1996 to July 2014
- Goal: Create a large “Technology” database consisting of a number of subcategories and products
- Used reduced data in the form of 5-core
  - Products with at least 5 reviews
    - ✓ Cut products with no reviews
    - ✓ Make sure sentiment could be extracted
  - Users that have done at least 5 reviews
    - ✓ Exclude potential fake accounts and reviews

UCLA Anderson
THINK IN THE NEXT

3A. DATA OVERVIEW

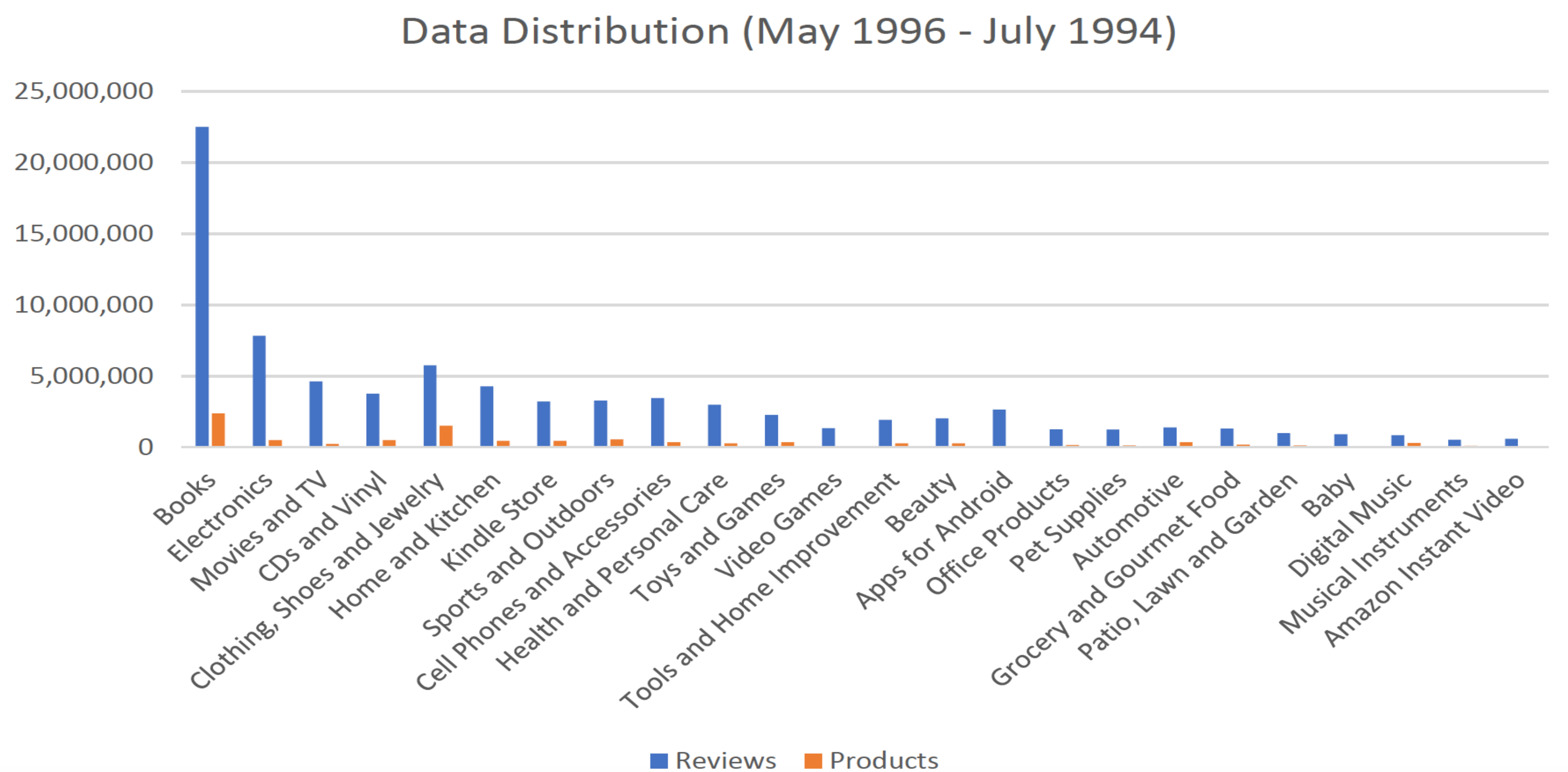
» Combined subsets:

Books	5-core (8,898,041 reviews)	ratings only (22,507,155 ratings)
Electronics	5-core (1,689,188 reviews)	ratings only (7,824,482 ratings)
Movies and TV	5-core (1,697,533 reviews)	ratings only (4,607,047 ratings)
CDs and Vinyl	5-core (1,097,592 reviews)	ratings only (3,749,004 ratings)
Clothing, Shoes and Jewelry	5-core (278,677 reviews)	ratings only (5,748,920 ratings)
Home and Kitchen	5-core (551,682 reviews)	ratings only (4,253,926 ratings)
Kindle Store	5-core (982,619 reviews)	ratings only (3,205,467 ratings)
Sports and Outdoors	5-core (296,337 reviews)	ratings only (3,268,695 ratings)
Cell Phones and Accessories	5-core (194,439 reviews)	ratings only (3,447,249 ratings)
Health and Personal Care	5-core (346,355 reviews)	ratings only (2,982,326 ratings)
Toys and Games	5-core (167,597 reviews)	ratings only (2,252,771 ratings)
Video Games	5-core (231,780 reviews)	ratings only (1,324,753 ratings)
Tools and Home Improvement	5-core (134,476 reviews)	ratings only (1,926,047 ratings)
Beauty	5-core (198,502 reviews)	ratings only (2,023,070 ratings)
Apps for Android	5-core (752,937 reviews)	ratings only (2,638,172 ratings)
Office Products	5-core (53,258 reviews)	ratings only (1,243,186 ratings)
Pet Supplies	5-core (157,836 reviews)	ratings only (1,235,316 ratings)
Automotive	5-core (20,473 reviews)	ratings only (1,373,768 ratings)
Grocery and Gourmet Food	5-core (151,254 reviews)	ratings only (1,297,156 ratings)
Patio, Lawn and Garden	5-core (13,272 reviews)	ratings only (993,490 ratings)
Baby	5-core (160,792 reviews)	ratings only (915,446 ratings)
Digital Music	5-core (64,706 reviews)	ratings only (836,006 ratings)
Musical Instruments	5-core (10,261 reviews)	ratings only (500,176 ratings)
Amazon Instant Video	5-core (37,126 reviews)	ratings only (583,933 ratings)

UCLA Anderson
THINK IN THE NEXT

3A. DATA OVERVIEW

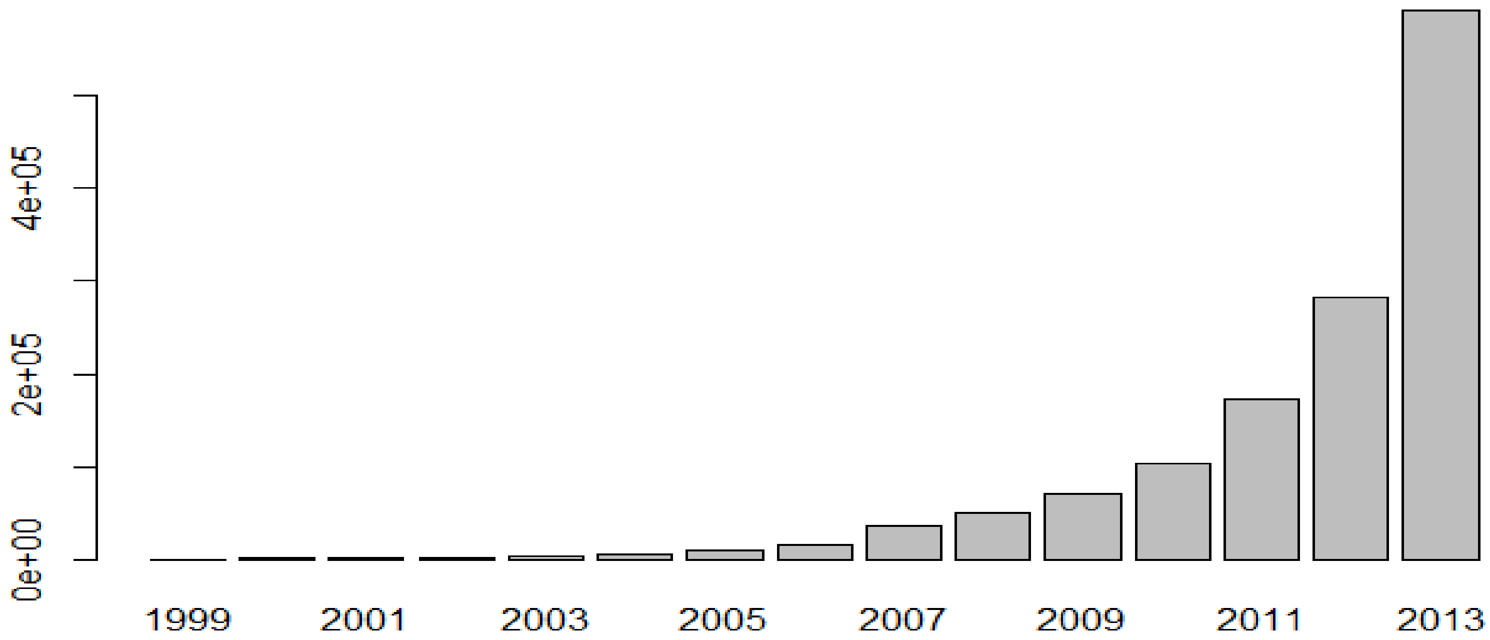
The distribution of number of reviews and products available under each category



UCLAAnderson
THINK IN THE NEXT

3A. DATA OVERVIEW

The number of reviews available each year under electronic category



UCLA Anderson
THINK IN THE NEXT

3B. DATA DESCRIPTION

Product Database

Name	Description
Asin	ID of the product
imURL	URL of the product image
Description	Description of the product
Categories	Category the product belongs to
title	The title of the product
Price	Price of the product
Related	Related products (also viewed, also bought, buy together)
salesRank	Sales Rank information
Brand	Brand name

3B. DATA DESCRIPTION

Review Database

Name	Description
Asin	ID of the product
ReviewerID	Id of the reviewer
ReviewerName	Name of reviewer
helpful	Helpfulness rating of the review
Reviewtext	Text of the review
Overall	Rating of the product
Summary	Summary of the review
unixReviewTime	Time of the review (unix time)
reviewTime	Time of the review (raw)

3C. METHODOLOGY

- Match Company names to products
- List of publicly traded companies
  - 644 electronics companies
  - Included all companies that could relate to electronics production
- List of the unique products traded under Electronics category for the sample period
  - Subset this list to products that have at least 1 review
  - Around 63000 distinct products traded in the dataset with at least 1 review

UCLAAnderson
THINK IN THE NEXT

3D. THE MATCHING PROCESS

» Modifications to the data
  - ❑ Remove all punctuations from titles, description and company names column
  - ❑ Transmute all columns to lower case letters

» Process
1. Find the full name of the company in the tiles
   - ❑ Example company name: 21 Vianet Group, Inc.
   - ❑ After modifications: 21 vianet group inc
   - ❑ Title: kelby training dvd mastering blend modes in 21vianet group inc adobe photoshop cs5 by corey barker
   - ❑ Not a match if any part of the company name is missing
   - ❑ Only 8 matches in total

UCLAAnderson
THINK IN THE NEXT

3D. THE MATCHING PROCESS

» Process(cont'd)

2. Find the full name of the company in descriptions
   - 337 total matches
3. Remove the most repetitive words from company name

```
 [1] "inc"           "corporation"   "ltd"       "holdings"   "technologies"
 [6] "international" "limited"       "group"     "systems"    "corp"
[11] "technology"    "incorporated"  "software"  "solutions"  "networks"
[16] "plc"           "holding"       "n.v"       "company"    "communications"
```

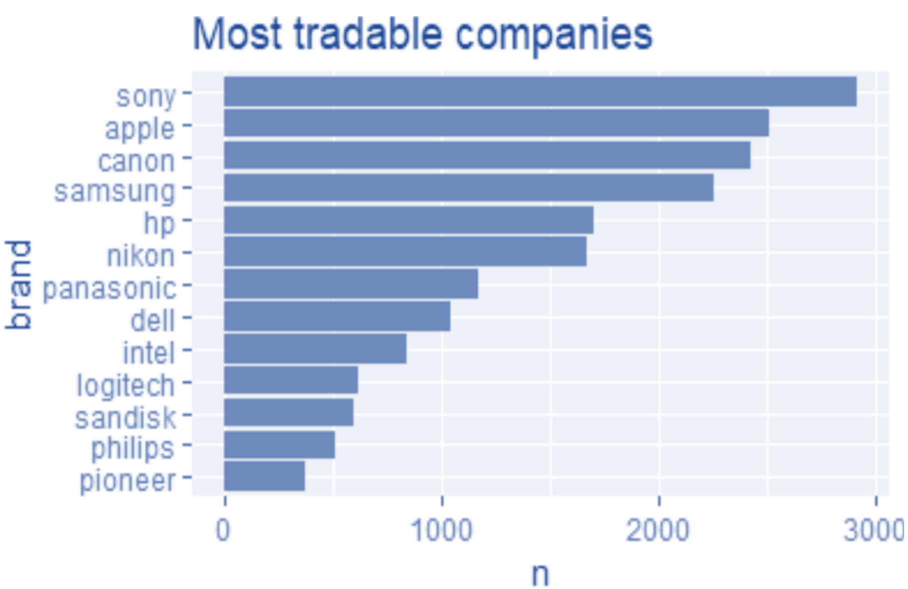
   - 21 vianet group inc → 21 vianet group
   - Problematic for some companies: Box inc → Box
   - Manually change these companies' names back
   - 22842 matches. 40% of products data
4. Do the same for description columns
   - 96707 matches. 180% of data due to some companies share the same products

UCLAAnderson
THINK IN THE NEXT

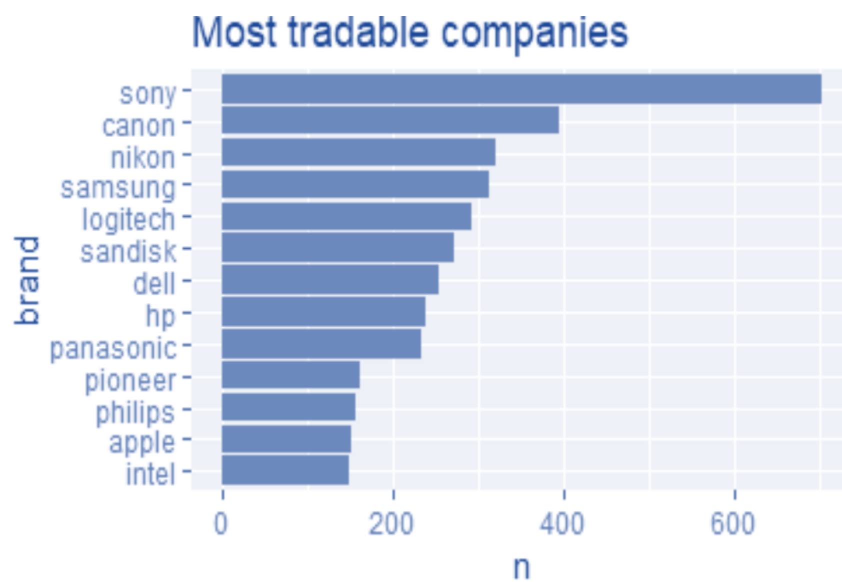
3D. THE MATCHING PROCESS

» Pick 13 biggest companies from the list

Based on Title column search



Based on Brand column search



» The accuracy of the model is 92.2%

» 13 companies cover 18789 products out of 22842 products

UCLA Anderson
THINK IN THE NEXT

3E. SENTIMENT ANALYSIS

» Identify sentiment for each review

» Merge only the numeric sentiment score for the product and the date when the review was written.

UCLA Anderson
THINK IN THE NEXT

3E. SENTIMENT ANALYSIS

Step1.

- Remove reviews of the products that weren't match to any of the product that is included in our 13 firms
- Left with around 500 000 reviews

Step2.

- Make it lower case
- Remove punctuations
- Remove special characters and extra spaces
- Remove numbers
- Don't remove stopwords dictionary
- Remove empty strings
- Will not perform stemming

UCLAAnderson
THINK IN THE NEXT

3E. SENTIMENT ANALYSIS

Step3.

» Use bing dataset to identify the sentiment

	word	sentiment
1	2-faces	negative
2	abnormal	negative
3	abolish	negative
4	abominable	negative
5	abominably	negative
6	abominate	negative
7	abomination	negative
8	abort	negative
9	aborted	negative
10	aborts	negative
11	abound	positive
12	abounds	positive
13	abrade	negative
14	abrasive	negative
15	abrupt	negative
16	abruptly	negative

UCLAAnderson
THINK IN THE NEXT

3E. SENTIMENT ANALYSIS

Step4.

- Use polarity function from R

$$\delta = \frac{x_i^T}{\sqrt{n}}$$

$$x_i^T = \sum((1 + c(x_i^A - x_i^D)) \cdot w(-1)^{\sum x_i^N})$$

$$x_i^A = \sum(w_{neg} \cdot x_i^a)$$

$$x_i^D = \max(x_i^{D'}, -1)$$

$$x_i^{D'} = \sum(-w_{neg} \cdot x_i^a + x_i^d)$$

$$w_{neg} = \left(\sum x_i^N\right) \bmod 2$$

	asin	text	score	time	polarity
1	0528881469	well can say ive unit truck four days now prior garmin t n...	3	2010-09-09	0.51784389
2	0528881469	going write long review even thought unit deserves one i...	2	2010-11-24	-0.06851887
3	0528881469	im professional otr truck driver bought tnd truck stop ho...	1	2010-11-25	-0.06950480
4	0528881469	ive mine year heres got tries route non truck routes tellin...	1	2011-09-29	-0.39605902
5	0528881469	got gps husband otr road trucker impressed shipping tim...	5	2013-06-02	1.38564065
6	8862936826	read many reviews folio cases ipad will find best ever revi...	5	2010-11-29	1.20021366
7	8862936826	ive waiting cover everything thought moleskine quality s...	1	2010-11-30	0.17614097
8	8862936826	im starting review changing stars like stars dont like first ...	2	2010-11-30	1.03091175
9	8862936826	product appeared high quality well made also feature tur...	4	2011-12-17	0.40824829
10	8862936826	owned month hoping use meeting situations ipad typing...	3	2013-02-16	0.57370973

UCLA Anderson
THINK IN THE NEXT

3E. SENTIMENT ANALYSIS

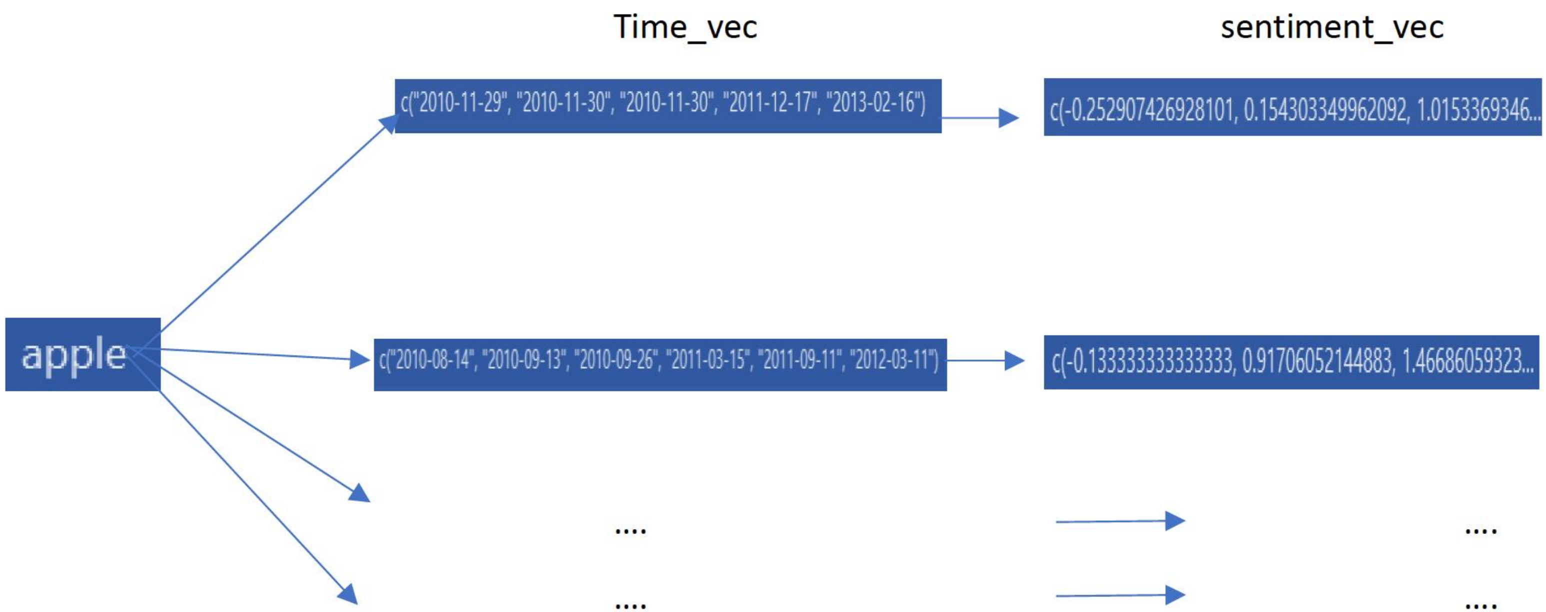
Step5.

- Merge firms, products and sentiments in one data table

	company_name	time_vec	sentiment_vec
1	sony	c("1999-07-23", "1999-10-13", "1999-10-20", "1999-12-09...	c(-0.252907426928101, 0.154303349962092, 1.0153369346...
2	canon	c("1999-12-31", "2000-05-02", "2000-06-08", "2000-06-17...	c(-0.133333333333333, 0.91706052144883, 1.46686059323...
3	nikon	c("1999-10-26", "1999-11-03", "1999-11-15", "1999-12-02...	c(-0.158943882847805, 0.126491106406735, -0.474341649...
4	samsung	c("2000-12-31", "2001-05-04", "2001-09-05", "2001-12-04...	c(0.784398431204706, 1.29875199971232, 1.05065722146...
5	logitech	c("2000-05-23", "2000-06-24", "2000-08-13", "2000-08-14...	c(0.832050294337844, 0, 1.18594462200587, 0.998687663...
6	sandisk	c("2000-06-16", "2000-06-17", "2000-08-06", "2000-09-07...	c(1.10858717169259, 0.351123441588392, 0.11396057645...
7	dell	c("2002-12-25", "2003-01-04", "2003-01-13", "2003-03-01...	c(0.539359889970594, 0.92689738158054, -0.14744195615...
8	hp	c("2000-06-02", "2000-08-06", "2000-08-20", "2000-09-06...	c(0.0333333333333333, 0.514614016722141, 0.991647658...
9	panasonic	c("1999-07-08", "1999-09-02", "1999-11-23", "1999-12-05...	c(0.366508333068916, 0, 0.859246812473437, 1.05531237...
10	philips	c("1999-11-17", "1999-11-27", "1999-12-01", "1999-12-08...	c(0.7184212081071, 0.567698757436222, 1.502637680879...
11	apple	c("2001-09-14", "2001-09-22", "2001-09-26", "2001-10-19...	c(0.612372435695795, -0.171498585142509, 0.7483314773...
12	pioneer	c("2000-05-16", "2000-08-02", "2000-11-15", "2000-12-19...	c(0.896179253777926, 1.24151489048701, 0.83445538495...
13	intel	c("2000-04-24", "2000-04-28", "2000-04-30", "2000-05-02...	c(0.758011398851084, 1.95544352835329, 1.21065445546...

UCLA Anderson
THINK IN THE NEXT

3E. SENTIMENT ANALYSIS

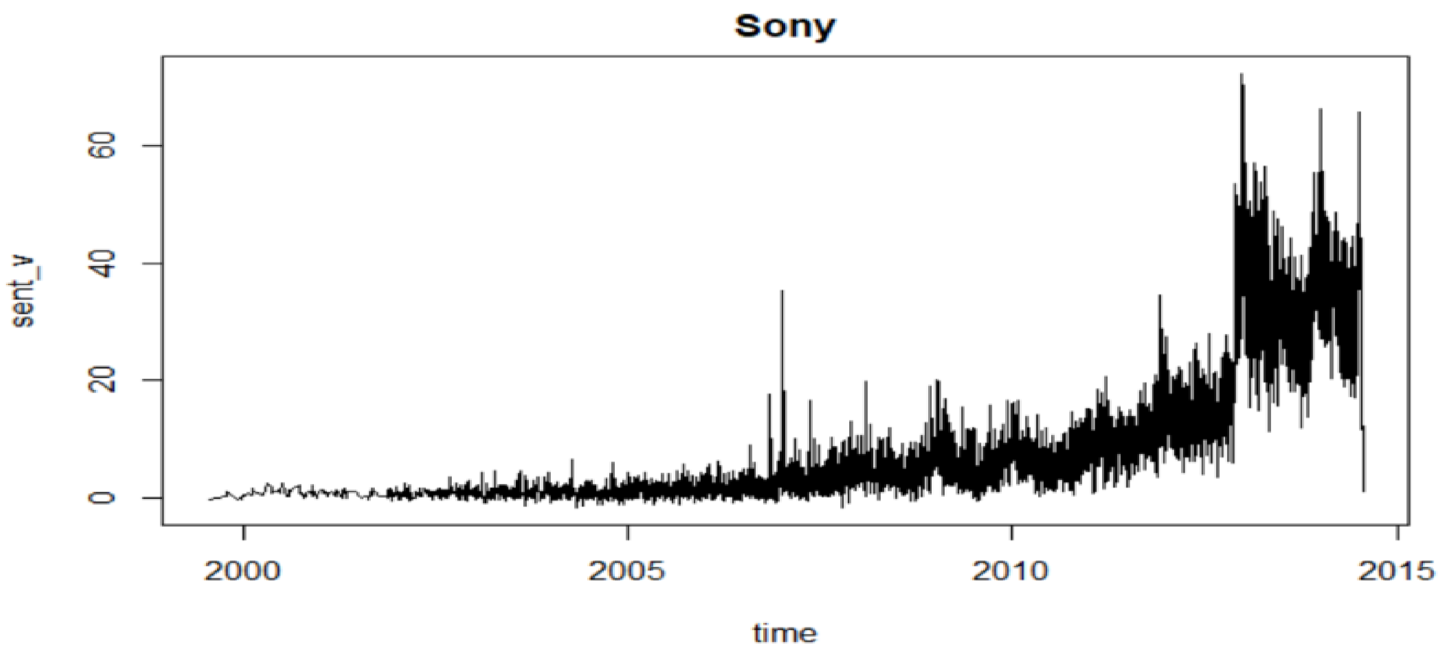


UCLA Anderson
THINK IN THE NEXT

3E. SENTIMENT ANALYSIS

Two methods for combining vectors:

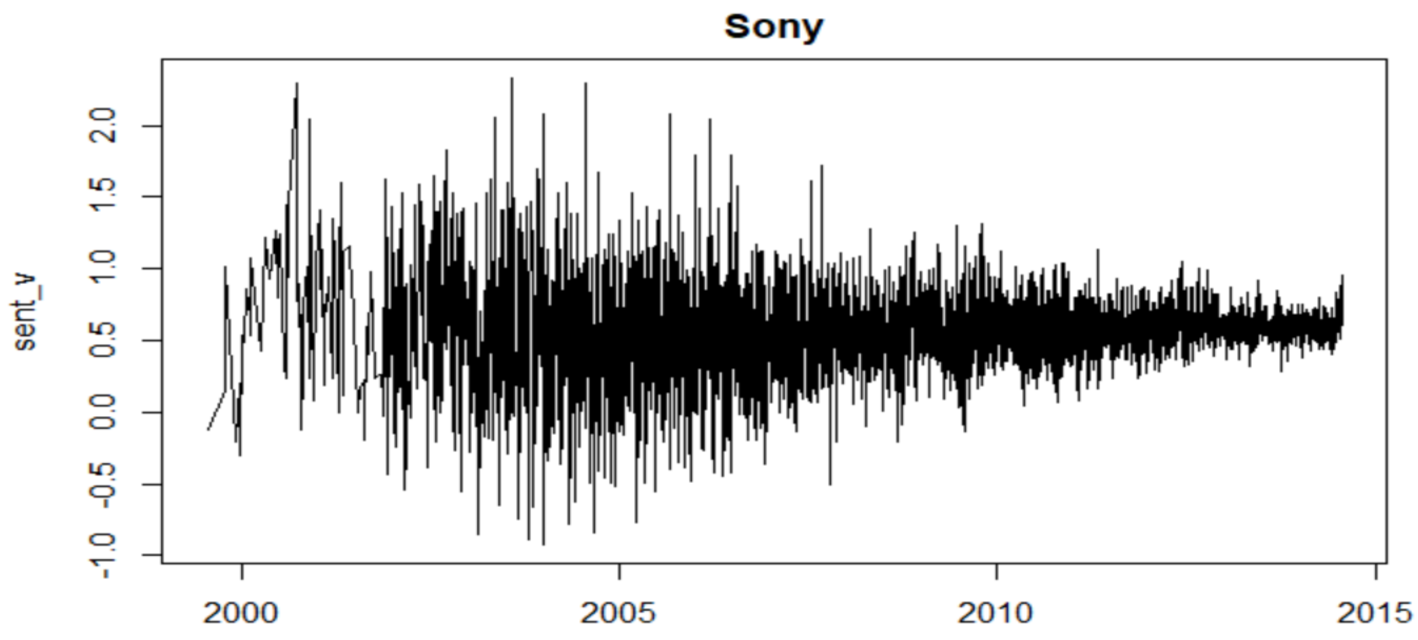
1. Simple method



UCLA Anderson
THINK IN THE NEXT

3E. SENTIMENT ANALYSIS

2. Weighted method



UCLAAnderson
THINK IN THE NEXT

3F. FUNDAMENTALS EXPLAINED BY SENTIMENT

- Construct quarterly sentiment index
- Calculated the Pearson Correlation of total sentiment score over the quarter with next quarter's fundamentals

	sony	canon	logitech	sandisk	hp	panasonic	apple	intel	max	min	median	median/sd
Cost of Goods Sold	0.125	0.372	0.495	0.026	0.468	0.482	-0.021	0.353	0.495	-0.021	0.362	1.827
Total Assets	0.294	0.387	0.348	-0.061	0.387	0.598	-0.293	0.357	0.598	-0.293	0.353	1.318
Current Liabilities	0.430	0.427	0.346	-0.209	0.147	0.615	-0.245	0.527	0.615	-0.245	0.386	1.264
Revenue	0.100	0.284	0.471	-0.243	0.412	0.474	-0.019	0.247	0.474	-0.243	0.265	1.113
Liabilities	0.284	0.395	0.321	-0.151	0.277	0.663	-0.423	0.485	0.663	-0.423	0.303	0.920
Common Equity	-0.029	0.349	0.288	0.018	0.260	0.463	-0.178	0.082	0.463	-0.178	0.171	0.841
Current Assets	0.159	0.341	0.157	-0.060	0.115	0.530	-0.242	0.211	0.530	-0.242	0.158	0.720
Cash	0.087	0.340	-0.237	0.259	-0.361	0.356	-0.241	0.210	0.356	-0.361	0.148	0.548
Cash/ST Investments	0.013	0.300	-0.237	0.127	-0.355	0.372	-0.294	-0.015	0.372	-0.355	-0.001	-0.003
Retained Earnings	-0.187	-0.067	0.358	0.059	0.175	0.420	-0.148	-0.137	0.420	-0.187	-0.004	-0.018
EPS 12MM	-0.353	0.071	-0.075	-0.399	0.026	-0.198	0.480	0.120	0.480	-0.399	-0.024	-0.092
Common Shares	-0.418	0.618	0.203	0.425	0.287	-0.349	-0.916	-0.513	0.618	-0.916	-0.073	-0.144
EPS Operations	-0.017	-0.085	0.208	-0.264	-0.103	-0.174	0.491	-0.205	0.491	-0.264	-0.094	-0.400
Net Income	-0.018	0.025	0.072	-0.181	-0.151	-0.189	0.108	-0.328	0.108	-0.328	-0.084	-0.594

UCLA Anderson
THINK IN THE NEXT

CONCLUSION

4. CONCLUSION

	sony	canon	logitech	sandisk	hp	panasonic	apple	intel	max	min	median	median/sd
Cost of Goods Sold	0.125	0.372	0.495	0.026	0.468	0.482	-0.021	0.353	0.495	-0.021	0.362	1.827
Total Assets	0.294	0.387	0.348	-0.061	0.387	0.598	-0.293	0.357	0.598	-0.293	0.353	1.318
Current Liabilities	0.430	0.427	0.346	-0.209	0.147	0.615	-0.245	0.527	0.615	-0.245	0.386	1.264
Revenue	0.100	0.284	0.471	-0.243	0.412	0.474	-0.019	0.247	0.474	-0.243	0.265	1.113
Liabilities	0.284	0.395	0.321	-0.151	0.277	0.663	-0.423	0.485	0.663	-0.423	0.303	0.920
Common Equity	-0.029	0.349	0.288	0.018	0.260	0.463	-0.178	0.082	0.463	-0.178	0.171	0.841
Current Assets	0.159	0.341	0.157	-0.060	0.115	0.530	-0.242	0.211	0.530	-0.242	0.158	0.720
Cash	0.087	0.340	-0.237	0.259	-0.361	0.356	-0.241	0.210	0.356	-0.361	0.148	0.548
Cash/ST Investments	0.013	0.300	-0.237	0.127	-0.355	0.372	-0.294	-0.015	0.372	-0.355	-0.001	-0.003
Retained Earnings	-0.187	-0.067	0.358	0.059	0.175	0.420	-0.148	-0.137	0.420	-0.187	-0.004	-0.018
EPS 12MM	-0.353	0.071	-0.075	-0.399	0.026	-0.198	0.480	0.120	0.480	-0.399	-0.024	-0.092
Common Shares	-0.418	0.618	0.203	0.425	0.287	-0.349	-0.916	-0.513	0.618	-0.916	-0.073	-0.144
EPS Operations	-0.017	-0.085	0.208	-0.264	-0.103	-0.174	0.491	-0.205	0.491	-0.264	-0.094	-0.400
Net Income	-0.018	0.025	0.072	-0.181	-0.151	-0.189	0.108	-0.328	0.108	-0.328	-0.084	-0.594

- COGS and Total Assets have highest median correlation
- Four of the fundaments have correlation higher than 30% which is decent in terms of predicting the fundamentals so far
- Problems with analyzing smaller companies in the future
  - 13 companies cover 90% of the data

UCLA Anderson
THINK IN THE NEXT

THANK YOU

UCLA Anderson
THINK IN THE NEXT

APPENDIX

5 APPENDIX

```
library(data.table)
library(jsonlite)
library(tidytext)
library(dplyr)
library(tidyverse)
library(tm)
library(stringr)
library(readxl)
library(DT)
library(textdata)
library(class)
library(gmodels)
library(ggplot2)
rev=as.data.table(fread("C:\\Users\\Rustem\\Desktop\\afp 2\\data.csv"))#reviews
products=as.data.table(fread("C:\\Users\\Rustem\\Desktop\\afp 2\\products.csv"))#products
comp=as.data.table(fread("C:\\Users\\Rustem\\Desktop\\afp 2\\companylist1.csv"))#company list

products=products[products$asin %in% unique(rev$asin),]#choose products only for which we have reviews
```

```
matched_brands=as.data.table(fread("C:\\Users\\Rustem\\Desktop\\afp 2\\brands matched.csv"))
comp=matched_brands$brand[1:20]
comp=comp[-2]
comp=comp[c(1:9,15,16,18,19)]
```

UCLA Anderson
THINK IN THE NEXT

5 APPENDIX

Match only by titles

```
titles<-removePunctuation(products$title)#create vector of titles
titles <- tolower(titles)
complementary=!is.na(titles)
titles=titles[complementary]#remove na titles
data3=comp

#create vector with products asin numbers
prodAsin=products$asin
prodAsin=prodAsin[complementary]

#create datatable to fill in later
df=data.table(data3)
setnames(df,c("name"))

#create vector to hold number of products traded for each company
num_products<- vector("numeric",length(data3))

#
index_place=list()
asin_list=list()

for(i in 1:length(data3)){
```

UCLAAnderson
THINK IN THE NEXT

5 APPENDIX

Match only by titles(cont'd)

```
   num_products[i]=sum(str_detect(titles,pattern=data3[i]))
   dd=which(str_detect(titles,pattern=data3[i])==TRUE)
   if(length(dd)!=0){
     index_place[[i]]=dd
     asin_list[[i]]=prodAsin[dd]
   }else{
     index_place[[i]]=0
     asin_list[[i]]=0
   }
}

df[,num_products:=num_products,]
df[,index_place:=index_place,]
df[,asin_list:=asin_list,]

main=df[num_products!=0]
main=main[,.(name,asin_list)]
#vectorize asin_list column, make it one string separated by space
main[,asin:=0,]
for(i in 1:length(main$name)){
   main$asin[i]=paste(unlist(main$asin_list[i]),collapse=" ")
}

main[,asin_list:=NULL,]
setnames(main,c("name","asin_vec"))

#now use unnest_token to make the data tidy

tidy_data <- main%>%
   unnest_tokens(asin_num,asin_vec)

tidy_data$asin_num=toupper(tidy_data$asin_num)
```

5 APPENDIX

Show the number of products for each company matched only by titles

```
x=tidy_data%>%
  count(name)%>%
  arrange(desc(n))


x=as.data.table(x)

words_count <- x%>%
  mutate(brand=fct_reorder(name,n))
ggplot(words_count,aes(x=brand,y=n))+geom_col()+coord_flip()+ggtitle("Most tradable companies")
```

Based only on brand column (not titles column)

```
brand <- removePunctuation(products$brand)
brand <- tolower(brand)
temp_dt=data.table(brand)
complementary=!is.na(temp_dt$brand)
temp_dt <- na.omit(temp_dt)

titles_brand=temp_dt$brand
```

UCLA Anderson
THINK IN THE NEXT

5 APPENDIX

Based only on brand column (not titles column)(cont'd)

```r
#create vector with products asin numbers
prodAsin=products$asin
prodAsin=prodAsin[complementary]

#create datatable to fill in later
df=data.table(data3)
setnames(df,c("name"))

#create vector to hold number of products traded for each company
num_products<- vector("numeric",length(data3))

#
index_place=list()
asin_list=list()

for(i in 1:length(data3)){
  num_products[i]=sum(str_detect(titles_brand,pattern=fixed(data3[i])))
  dd=which(str_detect(titles_brand,pattern=data3[i])==TRUE)
  if(length(dd)!=0){
    index_place[[i]]=dd
    asin_list[[i]]=prodAsin[dd]
  }else{
    index_place[[i]]=0
    asin_list[[i]]=0
  }
  #print(i)
}
#==========================================================================
#==========================================================================
df[,num_products:=num_products,]
df[,index_place:=index_place,]
df[,asin_list:=asin_list,]

main=df[num_products!=0]
main=main[,.(name,asin_list)]
#vectorize asin_list column, make it one string separated by space
```

```r
for(i in 1:length(main$name)){
  main$asin[i]=paste(unlist(main$asin_list[i]),collapse=" ")
}

main[,asin_list:=NULL,]
setnames(main,c("name","asin_vec"))

#now use unnest_token to make the data tidy

tidy_data2 <- main%>%
  unnest_tokens(asin_num,asin_vec)

tidy_data2$asin_num=toupper(tidy_data2$asin_num)
```

```r
x=tidy_data2%>%
  count(name)%>%
  arrange(desc(n))

x=as.data.table(x)

words_count <- x%>%
  mutate(brand=fct_reorder(name,n))
ggplot(words_count,aes(x=brand,y=n))+geom_col()+coord_flip()+ggtitle("Most tradable companies")
```

UCLA Anderson
THINK IN THE NEXT

5 APPENDIX

shows how many unque brands are there on amazon database and shows a number of products for each brand

```
y=brand
y=na.omit(y)
y=data.table(y)
setnames(y,c("name"))
y=y%>%
  count(name)%>%
  arrange(desc(n))


y=as.data.table(y)
```

Combine brands and titles results into one data table

```
# brand <- removePunctuation(products$brand)
# brand <- tolower(brand)
# titles<-removePunctuation(products$title)#create vector of titles
# titles <- tolower(titles)
#
# temp_dt=data.table(titles,brand)
#
# #temp_dt=temp_dt[-which(is.na(titles) & is.na(brand)),]
#
# temp_dt[,title_brand:=ifelse((!is.na(titles) & !is.na(brand)),paste(titles,brand),ifelse(!is.na(title
#
#
# titles_brand=temp_dt$title_brand
# complementary=!is.na(titles_brand)
# titles_brand=na.omit(titles_brand)
# #create vector with products asin numbers
# prodAsin=products$asin
# prodAsin=prodAsin[complementary]
#
# #create datatable to fill in later
# df=data.table(data3)
# setnames(df,c("name"))
#
# #create vector to hold number of products traded for each company
# num_products<- vector("numeric",length(data3))
#
# #
```

UCLAAnderson
THINK IN THE NEXT

5 APPENDIX

Combine brands and titles results into one data table(cont'd)

```
# index_place=list()
# asin_list=list()
#
# for(i in 1:length(data3)){
#   num_products[i]=sum(str_detect(titles_brand,pattern=fixed(data3[i])))
#   dd=which(str_detect(titles_brand,pattern=data3[i])==TRUE)
#   if(length(dd)!=0){
#     index_place[[i]]=dd
#     asin_list[[i]]=prodAsin[dd]
#   }else{
#     index_place[[i]]=0
#     asin_list[[i]]=0
#   }
#   #print(i)
# }
# #=====================================================================
# #=====================================================================
# df[,num_products:=num_products,]
# df[,index_place:=index_place,]
# df[,asin_list:=asin_list,]
#
#
# main=main[,.(name,asin_list)]
# #vectorize asin_list column, make it one string separated by space
# main[,asin:=0,]
# for(i in 1:length(main$name)){
#   main$asin[i]=paste(unlist(main$asin_list[i]),collapse=" ")
# }
#
# main[,asin_list:=NULL,]
# setnames(main,c("name","asin_vec"))
#
# #now use unnest_token to make the data tidy
#
# tidy_data3 <- main%>%
#   unnest_tokens(asin_num,asin_vec)
#
# tidy_data3$asin_num=toupper(tidy_data3$asin_num)


#write.csv(tidy_data3,"C:\\Users\\Rustem\\Desktop\\afp 2\\tidy_data3.csv", row.names = FALSE)
tidy_data3=as.data.table(fread("C:\\Users\\Rustem\\Desktop\\afp 2\\tidy_data3.csv"))
```

UCLA Anderson THINK IN THE NEXT

5 APPENDIX

Cleaning reviews for sentiment analysis

```
# cleaned_reviews=rev[rev$asin %in% unique(tidy_data3$asin_num),]
# cleaned_reviews[,reviewerID:=NULL,]
# cleaned_reviews[,reviewerName:=NULL,]
# cleaned_reviews[,unixReviewTime:=NULL,]
# cleaned_reviews$reviewTime=as.Date(cleaned_reviews$reviewTime,format="%m %d, %Y")
#
#
# setnames(cleaned_reviews,c("asin","text","score","summary","time"))
# setorderv(cleaned_reviews,c("asin","time"))
# #==============================================================
# #firsly let's remove all digits and punctuations from our text and make all lower letters
# #==============================================================
# cleaned_reviews$text=tolower(cleaned_reviews$text)
# cleaned_reviews$text=removePunctuation(cleaned_reviews$text)
# cleaned_reviews$text=removeNumbers(cleaned_reviews$text)
# cleaned_reviews$text=removeWords(cleaned_reviews$text,stopwords("english"))
# cleaned_reviews=cleaned_reviews[-which(cleaned_reviews$text==""),]
# cleaned_reviews$text=stripWhitespace(cleaned_reviews$text)
# cleaned_reviews[,summary:=NULL]
# #====================================================================
# #WE WILL NOT PERFORM STEMMING SINCE IT DOES NOT MAKE SENSE IN THIS CASE
# #====================================================================
# library(textdata)
# a=get_sentiments("bing")
# #write.csv(cleaned_reviews,"C:\\Users\\Rustem\\Desktop\\afp 2\\cleaned_reviews3.csv", row.names = FAL
cleaned_reviews=as.data.table(fread("C:\\Users\\Rustem\\Desktop\\afp 2\\cleaned_reviews3.csv"))
```

5 APPENDIX

Sentiment Analysis: (polarity approach)

```
#
# install.packages("qdap")
# library(qdapRegex)
# library(qdapDictionaries)
# install.packages("rJava")
# library(rJava)
# #Sys.setenv(JAVA_HOME="C:\\Program Files\\Java\\jre1.8.0_221") # for 32-bit version
# library(qdap)


#write.csv(pol_df,"C:\\Users\\Rustem\\Desktop\\afp 2\\pol_df2.csv", row.names = FALSE)
pol_df=as.data.table(fread("C:\\Users\\Rustem\\Desktop\\afp 2\\pol_df2.csv"))
```

check correlation with score

```
yy=data.table(pol_df$score,pol_df$polarity)
yy=na.omit(yy)
cor(yy$V1,yy$V2)
```

Accuracy of the model

```
accuracy=length(unique(tidy_data3$asin_num))/length(tidy_data3$asin_num)

accuracy
```

UCLAAnderson
THINK IN THE NEXT

5 APPENDIX

Group polarity score and time by firm

```r
tidy_data3[,time_vec:=0,]
tidy_data3[,sentiment_vec:=0,]

for(i in 1:length(tidy_data3$asin_num)){
  aa=pol_df[asin==tidy_data3$asin_num[i],]
  l1=list(aa$time,aa$polarity)
  tidy_data3$time_vec[i]=l1[1]
  tidy_data3$sentiment_vec[i]=l1[2]
  if(i%%250==0){
   print(i)
  }
}
```

```r
final=data.table()
final[,company_name:=unique(tidy_data3$name),]
final[,time_vec:=0,]
final[,sentiment_vec:=0,]
for(i in 1:length(final$company_name)){
  aa=tidy_data3[name==final$company_name[i],]
  master_dt=data.table()
  for(j in 1:length(which(tidy_data3$name==final$company_name[i]))){
    dt1=data.table(unlist(aa$time_vec[j]),unlist(aa$sentiment_vec[j]))
    master_dt=rbind(master_dt,dt1)
  }
  x=master_dt%>%
    group_by(V1)%>%
    summarize(sum(V2))
  l1=list(x$V1,x$`sum(V2)`)
  final$time_vec[i]=l1[1]
  final$sentiment_vec[i]=l1[2]

  print(i)
}
```

```r
time=as.Date(unlist(final$time_vec[1]),format="%Y-%m-%d")
sent_v=unlist(final$sentiment_vec[1])
plot(time,sent_v,type="l",main="Sony")
```

```r
final_complex=data.table()
final_complex[,company_name:=unique(tidy_data3$name),]
final_complex[,time_vec:=0,]
final_complex[,sentiment_vec:=0,]
for(i in 1:length(final_complex$company_name)){
  aa=tidy_data3[name==final_complex$company_name[i],]
  master_dt=data.table()
  for(j in 1:length(which(tidy_data3$name==final_complex$company_name[i]))){
    dt1=data.table(rep(aa$asin_num[j],length(unlist(aa$time_vec[j]))),unlist(aa$time_vec[j]),unlist(aa$
    master_dt=rbind(master_dt,dt1)
  }
  master_dt[,weight:=1/(.N),by=.(V1,V2)]
  master_dt[,weighted_sent:=V3*weight]

  master_dt[,num_reviews:=(.N),by=.(V1,V2)]
  master_dt[,total_reviews:=(.N),by=.(V2)]
  master_dt[,day_weight:=num_reviews/total_reviews]
  x=master_dt[,list(col1=sum(weighted_sent),col2=day_weight),by=.(V1,V2)]
  x=unique(x)


  setnames(x,c("asin","time","lambda","weights"))
```

5 APPENDIX

```
  x[,weighted_sent_day:=weights*lambda]
  x=x[,sum(weighted_sent_day),by=c("time")]
  setorderv(x,c("time"))

  l1=list(x$time,x$V1)
  final_complex$time_vec[i]=l1[1]
  final_complex$sentiment_vec[i]=l1[2]

  print(i)
}
```

```
time=as.Date(unlist(final_complex$time_vec[1]),format="%Y-%m-%d")
sent_v=unlist(final_complex$sentiment_vec[1])
plot(time,sent_v,type="l",main="Sony")
```

```
fund=as.data.table(fread("C:\\Users\\Rustem\\Desktop\\afp 2\\fundamentals.csv"))
sony=fund[189:226,]
sony$datadate=as.Date(as.character(sony$datadate),format="%Y%m%d")
```

For sony we have data for each data through 4,5 years

```
dt_sony=data.table(time=as.Date(unlist(final_complex$time_vec[1]),format="%Y-%m-%d"),sent_v=unlist(fina

dt_sony[,year:=year(time)]
dt_sony[,month:=month(time)]
dt_sony[,quarter:=ifelse(month<=3,1,ifelse(month<=6,2,ifelse(month<=9,3,ifelse(month<=12,4,4))))]
dt_sony[,period:=paste(year,"Q",quarter,sep="")]
dt_sony[,year:=NULL]
dt_sony[,month:=NULL]
dt_sony[,quarter:=NULL]
dt_sony=dt_sony[time>="2010-01-01" & time<="2014-06-01",]

dt_sony=na.omit(dt_sony)
sup=dt_sony%>%
  group_by(period)%>%
  summarise(total=sum(sent_v))

sup=as.data.table(sup)
sup[,normalized_sent:=(total-mean(total))/(sd(total))]


sony=sony[datadate>="2010-01-01" & datadate<="2014-06-30",]

cor(sup$normalized_sent,sony[,c(15:28)])
```

UCLA Anderson
THINK IN THE NEXT

5 APPENDIX

```
master_cor=data.table()
for(i in 1:length(final_complex$company_name)){
  time=as.Date(unlist(final_complex$time_vec[i]),format="%Y-%m-%d")
  sentiment=unlist(final_complex$sentiment_vec[i])
  dt=data.table(time,sentiment)[time>="2010-01-01" & time<="2014-06-01",]
  dt[,year:=year(time)]
  dt[,month:=month(time)]
  dt[,quarter:=ifelse(month<=3,1,ifelse(month<=6,2,ifelse(month<=9,3,ifelse(month<=12,4,4))))]
  dt[,period:=paste(year,"Q",quarter,sep="")]
  dt[,year:=NULL]
```

UCLA Anderson
THINK IN THE NEXT

5 APPENDIX

```
  dt[,month:=NULL]
  dt[,quarter:=NULL]

  dt=na.omit(dt)
  sup=dt%>%
    group_by(period)%>%
    summarise(total=sum(sentiment))
  sup=as.data.table(sup)
  sup[,normalized_sent:=(total-mean(total))/(sd(total))]



  firm=fund[which(conm==final_complex$company_name[i]),]
  firm$datadate=as.Date(as.character(firm$datadate),format="%Y%m%d")
  firm=firm[datadate>="2010-01-01" & datadate<="2014-06-30",]


  if(length(firm$gvkey)==0){
    next
  }

  if(final_complex$company_name[i]=="dell"){
    next
  }
  y=cor(sup$normalized_sent,firm[,c(15:28)])
  master_cor=rbind(master_cor,y)
 # eval(parse(text=paste("dt_",final_complex$company_name[i],"=dt",sep="")))

}



master_cor=cbind(data.table(final_complex$company_name[c(1,2,5,6,8,9,11,13)]),master_cor)
#write.csv(master_cor,"C:\\Users\\Rustem\\Desktop\\afp 2\\master_cor.csv", row.names = FALSE)
master_cor=as.data.table(fread("C:\\Users\\Rustem\\Desktop\\afp 2\\master_cor.csv"))
```

UCLA Anderson
THINK IN THE NEXT

5 APPENDIX

```
vec=data.table("max")
for(i in 2:15){
  vec=cbind(vec,max(master_cor[[i]]))
}
setnames(vec,colnames(master_cor))
vec1=data.table("min")
for(i in 2:15){
  vec1=cbind(vec1,min(master_cor[[i]]))
}
setnames(vec1,colnames(master_cor))
vec2=data.table("meadian")
for(i in 2:15){
  vec2=cbind(vec2,median(master_cor[[i]]))
}
setnames(vec2,colnames(master_cor))

master_cor=rbind(master_cor,vec,vec1,vec2)
```

UCLA Anderson
THINK IN THE NEXT