Marketing Analytics Professor Swagato Chatterjee Vinod Gupta School of Management, Indian Institute of Technology, Kharagpur Lecture 55 Text Mining and Sentiment Analytics (Contd.)

Hello everybody, welcome to Marketing Analytics course. I am Doctor Swagato Chatterjee from VGSOM, IIT, Kharagpur who is taking this course for you. We are in week 10, session 5 and I will discuss about how to use R programming to apply the Naive Bayes algorithm in a data set.

(Refer Slide Time: 00:37)

	Project (None)
Environment History Connections	= 0
💣 🔒 🖙 Import Dataset 🔹 🔏	🗏 List + 🛛 🕥
* 🍈 Global Environment +	Q,
Environment is emp	pty
Files Plots Packages Help Viewer	-0
0 0 2 2 5mm 2 5port - 0	
:	
2	
·)	
	Tentecoment History Connections

A data set is given to you, it is the SMS spam data, so we will first put the set working directory and then I will read the data. The data set looks like this.

(Refer Slide Time: 00:46)

4	A B	С	D	E	F	G	н	1	J	K	L	٨
1	type	text										
2	1 ham	Hope you	are having	a good we	ek. Just ch	ecking in						
3	2 ham	Kgive ba	ick my than	ks.								
4	3 ham	Am also d	loing in cbe	only. But I	have to pa	y.						
5	4 spam	complime	entary 4 STA	AR Ibiza Ho	liday or Âf	10,000 cash	n needs yo	ur URGEN	Collection	. 0906636	4349 NOW	from
6	5 spam	okmail: D	ear Dave th	his is your fi	inal notice	to collect y	our 4* Ter	nerife Holio	lay or #500	0 CASH aw	ard! Call 0	90617
7	6 ham	Aiya we d	liscuss later	lar Pick	u up at 4 is	it?						
8	7 ham	Are you t	his much bu	uzy								
9	8 ham	Please as	k mummy t	o call fathe	r							
0	9 spam	Marvel M	lobile Play t	the official	Ultimate S	pider-man g	game (£4	.50) on ur	mobile righ	nt now. Te:	t SPIDER t	to 833
1	10 ham	fyi I'm at	usf now, sw	ing by the	room whe	never				-		
2	11 ham	Sure thin	g big man. i	have hock	ey election	ns at 6, shou	Idnå,Ëœ	t go on lon	ger than an	h h	h	
3	12 ham	I anything	g lor									
14	13 ham	By march	ending, i sh	hould be re	ady. But v	vill call you t	for sure. T	he problen	n is that my	carer	compl	ete. H
15	14 ham	Hmm wel	ll, night nigh	ht						17	2h	
6	15 ham	K I'll be su	ure to get u	p before n	oon and se	e what's wi	hat		-	11	4/10	
7	16 ham	Ha ha coo	ol cool chikk	u chikku:-)	:-DB-)				-	17V		1
18	17 ham	Darren w	as saying da	at if u meet	ing da ge	den we dun	meet 4 di	nner. Cos l	ater	the F	184	r

Where if you see it carefully there are the type whether it is spam or ham, and the text is written. For example, complementary 4-star Ibiza holiday, blah-blah-blah, this is spam. But are you this much busy or please ask mummy to call father, this is ham. So, we will be playing with that to see that whether.

(Refer Slide Time: 01:14)



• 🍕 🐨 • 🖶 🔛 📾 1 🔺 Go to the function 🔢 🔯 • Addins •				I Project (None)
a nbr =	-0 "	Invironment Hist	ory Connections	-0
Source on Save Q. / . + Run ++	+ Source + 2	r 🕞 📑 Impo	rt Dataset 🔹 🧃	🗉 18t • 🛛 🕥
<pre>1 sms_raw <- read.csv("sms_spam.csv", header=TRUE) 2 names(sms_raw)=c("slNo","type","text")</pre>	1.	Global Environme	nt •	9,
3 str(sms_raw) 5 sms_rawitype ← factor(sms_rawitype) 6 table(sms_rawitype) 7 liberry(tm) 9 streaments ← Communications and factor	0	sms_raw	5559 obs. of 3 variables	
<pre>3 am_corpus ~ corpus(recorpus(recorpus(recorpus)) 1 inspect(sms_corpus)1:10) 12 corpus_clean ~ tm_map(sorpus, content_transformers) 14 corpus_clean ~ tm_map(corpus_clean, removenumbers) 15 corpus_clean / corpus_clean, removenumbers) 16 corpus_clean / corpus_clean, removenumbers) 17 corpus_clean / corpus_clean, removenumbers) 18 corpus_clean / corpus_clean, removenumbers) 19 corpus_clean / corpus_clean / corpus_clean, removenumbers) 19 corpus_clean / corpus_clean /</pre>	r(tolower))	illes Plots Pack	ages Help Viewer	-0
81 (kp.Level) :	R Script ±			
console Yerminal =				
C/Users/NPTEL Studio-01/Desktop/Week10/Session 4/naive bayes/ 🖉				
<pre>\$ text: Factor w/ 5156 levels "'An Amazing Quote'' - Somet its difficult to decide whats wrong!! a lie that brings a ne " truncated,: 1651 2557 257 626 3308 190 357 3397</pre>	imes in lif smile or t 2726 1079		he	Ker
table(sms_raw\$type)			1	E
ham spam 812 747 4812/559 1] 0.8656;33				E!

So I read the data first and then I put the names properly. The structure of the data looks like this, there are 3 columns, serial number, the type which is factor variable spam and ham, there are two factors and then the text. And then what we do is, if I just check the table that there are 4812 spams so to get an idea there are 4812 / 5559 that means around 86 % ham and then 14 % spam or 86.5 % ham and 13.5 % ham is there, fair enough.

(Refer Slide Time: 01:57)



These In Life Cale Van Rets Senar July Dates Rolls Sena Rela			- 8 X
🕽 • 🤫 🕼 😥 😓 🖌 🚓 Go to The function 👘 🔛 • Addins •			Project (None) •
o nor =	Environment Histo	ry Connections	-0
🖉 🖓 Source on Save 🔍 🎢 🖓 🖓 🖓 🖓 🖓 🖓	൙ 🕞 🖙 import	t Dataset + 🧃	11 Gat • 1 🕄
<pre>2 names(sms_raw)=c("slvo","type","text") *</pre>	Giobal Environmen		0
3 A str(sms raw)	Data		
5 sms_rawStype <- factor(sms_rawStype)	O SMS_COPPUS	Large SimpleCorpus (5	559 elements, 75_ Q
6 table(sms_rawStype)	O sms_raw	5559 obs. of 3 variab	ales 🔲
8 library(tm)			
<pre>9 sms_corpus <- Corpus(VectorSource(sms_rawStext))</pre>			
10 print(sms_corpus)			
12 Inspect(sms_corpus(1:10))			
<pre>13 korpus_clean <- tm_map(sms_corpus, content_transformer(tolower))</pre>	Files Plots Packa	iges Help Viewer	= 0
14 corpus_clean <- tm_map(corpus_clean, removeNumbers)		Deport • 0	
<pre>15 16 corpus_clean <- tm_map(corpus_clean, removewords, stopwords()) .</pre>			
17			
Un (top Level) : Riscopt :			
Console Terminal = m			-
C/Users/NPTEL Studio-01/Desktop/Week10/Session 4/naive bayes/ 🕸			
> inspect(sms_corpus(1:10)) .		1	100
Metadata: corpus specific: 1.[document level (indexed): 0			-0
Content: documents: 10			1 TON
[1] time you are builded a could made that should be in			1 Martin
[1] Hope you are having a good week. Just checking in			E
		KLAT	
[2] Kgive back my thanks.		C10/	FIL
		ALVA	1
[3] Am also doing in cbe only. But have to pay.			in I
		113	

So, now as usual, Library tn, we are calling it changing it to corpus, we have done this before. And then, I will print the corpus at least the, print the corpus says that it has a corpus specific bond, document level zero and that documents are 5559 and if I inspect the first 10 documents, it looks like this, which is basically the first 10 observations.

(Refer Slide Time: 02:26)





Now, with this what I do is I transform it to their lower, I change, remove the numbers, I remove the stop words, I remove punctuation, we have done this before, right in session 2, if we are not wrong, session 2 or 3 I have done this. So, strip white space, so we are removing this and now with this thing I am creating a document on matrix.

(Refer Slide Time: 02:57)

. 0	I 🐨 • I 🔄 🖃 I 🗰 I 🔿 Go to flie function							Project (None
nbr		Environ	ment	History	Connec	tions			-0
	🔊 🔄 Source on Save 🔍 🖉 - 📔 🚽 Run 😁 📑 Source - 📚	C 8	1.00	import Data	set +	(III List	
18	corpus_clean <- tm_map(corpus_clean, stripwhitespace) *	Glob	bal Enviro	• treme				Q	
19	<pre>sms_dtm <- DocumentTermMatrix(corpus_clean)</pre>	Data							
21	sms_raw_train <- sms_raw[1:4169,]	0 core	pus_c]	lean i	Large	SimpleCorpus	(5559 e	lements, 61	0
22	sms_raw_test <- sms_raw[4170:5559,]	O sms	corpu	is i	Large	SimpleCorpus	(5559 e	lements, 75_	a
23	ane den ensin o ene den[1:4160]	O sms	dtm		Large	DocumentTerm	Matrix (6 elements	a
25	sms_dtm_test <- sms_dtm[4170;5559,]	0 585	raw		5559	obs. of 3 vari	ables		
26									
27	<pre>sms_corpus_train <- corpus_clean[1:4169] sms_corpus_test <- corpus_clean[4170.5550]</pre>								
29	sms_corpus_test <- corpus_clean(41/0:5559)	Files I	Plots	Packages	Help	Viewer			
30	prop.table(table(sms_raw_trainStype))			ton 1,7	Export	. 0 1			
31	<pre>prop.table(table(sms_raw_testStype))</pre>								
32	* marganization								
211	(Top Level) : R Script &								
onsole	Terminal =						-		
C/Users	/NPTEL Studio-01/Desktop/Week10/Session 4/naive bayes/ 🕫 🧹						1		
sms_	dtm <- DocumentTermMatrix(Corpus_clean)					A			P



So, SMS DTM is a large matrix. Now, next what I am doing here is there are some purpose of doing this. So, there are 3 things, you check, there is a raw data, from there we created the corpus, we clean the corpus and created data documental matrix. Now I will break this raw data into training raw and testing raw. Training corpus the same, exactly same row numbers, its training corpus and testing corpus. And training DTM and testing DTM, this is something that I am going to create.

Why? Because some of this raw one will be used to create word cloud. Corpus might be used to something else. DTM might be used to create something else so there is certain purpose. But there should be some matching so that you see, these three guys should be related to each other and these three guys should be related to each other.

(Refer Slide Time: 04:09)

Las Carl	ten Ret Jessen Reld Selves Rolle Note Help					0
. 0	🛿 📽 • 🔛 🗐 📥 🛛 🔺 Go to Behancton 🔢 🔯 • Add	ins -			🕔 Project i	(Nore
nbr		-0	Environment History	Connections		-
	Source on Save Q / - - Run	🍽 🕐 Source + ≥	💣 🔒 🔝 Import D	ataset • 🧃	🗏 List	•
22 23	sms_raw_test <- sms_raw[4170:5559,]%	1	Global Environment		٩,	
24 25 26	<pre>sms_dtm_train <- sms_dtm[1:4169,] sms_dtm_test <- sms_dtm[4170:5559,]</pre>		Ocorpus_clean Osms_corpus	Large SimpleCorpus Large SimpleCorpus	(5559 elements, 61. (5559 elements, 75.	q
20	ses corous train co corous clean[1:4169]		osms_corpus_tes	tList of 1390		q
28	propitable(k, margin = NULL) compus_clean[4170:5559]		0 sms_corpus_tra	List of 4169		0
29			O sms_dtm	Large DocumentTermM	atrix (6 elements	0
<pre>30 prop.table(table(sms_raw_trainitype)) 31 prop.table(table(sms_raw_test)type)) 32 prop.table(table(sms_raw_test)type)) 33 prop.table(table(sms_raw_test)type)) 34 prop.table(table(sms_raw_test)type)) 35 prop.table(table(sms_raw_test)type)) 36 prop.table(table(sms_raw_test)type)) 37 prop.table(table(sms_raw_test)type)) 38 prop.table(table(sms_raw_test)type)) 39 prop.table(table(sms_raw_test)type)) 30 prop.table(table(sms_raw_test)type)) 31 prop.table(table(sms_raw_test)type)) 32 prop.table(table(sms_raw_test)type)) 33 prop.table(table(sms_raw_test)type)) 34 prop.table(table(sms_raw_test)type)) 35 prop.table(table(sms_raw_test)type)) 35 prop.table(table(sms_raw_test)type)) 36 prop.table(table(sms_raw_test)type)) 37 prop.table(table(sms_raw_test)type)) 38 prop.table(table(sms_raw_test)type)) 39 prop.table(table(sms_raw_test)type))</pre>	_	osms_dtm_test	Large DocumentTermM	atrix (6 elements,_	9	
32	1 Shearsy (words loud)		Files Plots Package	is Help Viewer		-
34 35	<pre>wordcloud(sms_corpus_train, min.freq = 30) wordcloud(sms_corpus_train, min.freq = 30, rand</pre>	om.order = FALSE)		Baport - 0		
36 37 301	(top Level) :	R Script ±				
onsole	Terminal ×	-0			-	
/User	/NPTEL Studio-01/Desktop/Week10/Session 4/naive bayes/ 🧠				and the second s	
sms, sms,	raw_train <- sms_raw[1:4169,] .raw_test <- sms_raw[4170:5559,]					
sms,	dtm_train <- sms_dtm[1:4169,]				N K	
sms,	dtm_test <- sms_dtm[4170:5559,]				Tar	
	corner train as corner class[1:4160]				NEX III	
Sms	corpus test <- corpus clean[4170:5559]					1
1				A.	17	

So, I am using the same column numbers, row numbers 1 to 4169, and 4170 to 5599 so, I am just, so if I just do it 4169 / 5559. That means around 75 % goes to my train and 25 % goes to my test. So, I am breaking this thing. So, the first one is the raw data, next is DTM and third is corpus. Fair enough. I have broken them. Why have I broken them? So, after breaking them I have to just see that whether I have not done randomness, see. So, I am just checking that whether even the training and testing data the I would say distribution of spams and hams are similar or not.

(Refer Slide Time: 04:51)

• 👒 💣 • 🔛 🖓 👘 🔺 Go to file/function 👘 🔯 • Addins •	🐌 Proje	tt (None
P nbr =	Environment History Connections	-0
Source on Save 🔍 🖉 - 📔 - Run 🐤 - 🔤 Source - 📚	💣 🔒 📑 Import Dataset 🔹 🦸 📰	ist + (
22 sms_raw_test <- sms_raw[4170:5559,] -	Global Environment • Q	
<pre>23 sms_dtm_train <> sms_dtm[114169,] 25 sms_dtm_test <> sms_dtm[4170:5559,] 26 sms_corpus_train <> corpus_clean[4170:5559] 27 sms_corpus_test <> corpus_clean[4170:5559] 28 sms_corpus_test <> corpus_clean[4170:5559] 29 prop.table(table(sms_ram_train(type)) 21 prop.table(table(sms_ram_test:type)) 22 library(wordcloud) 23 wordcloud(sms_corpus_train, min.freq = 30) 25 wordcloud(sms_corpus_train, min.freq = 30, random.order = FALSE) 26 sms_corpus_table(table(sms_ram_test:type)) 27 sms_corpus_train, min.freq = 30, random.order = FALSE) 28 sms_corpus_table(table(sms_corpus_train, min.freq = 30, random.order = FALSE) 29 sms_corpus_table(table(table(ta</pre>	O corpus_clean Large SimpleCorpus (5559 elements, 6 O sms_corpus Large SimpleCorpus (5559 elements, 7 O sms_corpus_tra.List of 1390 O sms_corpus_tra.List of 4169 O sms_dtm_test Large DocumentTermMatrix (6 elements O sms_dtm_test Large DocumentTermMatrix (6 elements FMes Pode Packages Help Vewer	1_ Q 5_ Q 0_ - Q Q
101 (lop Level) : R Script :		
Colliser/WETD: Studio 01/Deckton/Week10/Lassion 4/make haves/		
> sms_corpus_train <- corpus_clean[1:4169] sms_corpus_test <- corpus_clean[4170:3559] prop.table(table(sms_rmm_trainStype)) ham spam .6647158 0.1332642 prop.table(table(sms_rmm_testStype)) ham spam .6683453 0.1316547		A

If you remember, earlier it was 86.5 % ham in the training data it is 86.47 similar. Here is it 86.8 one and the similar. So, I can deal with this thing.

(Refer Slide Time: 05:05)



So, now with this dataset, what I do is next is I call the library, the library is word cloud. I am calling the library and with the corpus, I am trying to draw a, so you see this is the corpus of my training data. This is how a word cloud looks like.

This is a word cloud this is how exactly it looks like. So, word cloud, let us say the, this guy is the bigger one, the bigger the one, the bigger the word in word cloud the more frequency of that word is there. Now, if I just random one is equal to false if I do that, what that will do, so before you plot the next plot, it is better to clean this plotting area. So, what it will do is, it will put the important ones at the middle.

So, you see call, now, just, will, can, these are the. Now what is my purpose? My purpose is using words to analyse whether it is span or ham. So by that logic exploratorily or visually if I see that the words which are coming common in spam and the words which are coming common in ham if that words are different in other words, the word cloud of a spam message and word cloud of all the ham message, if the prominent words are different then only I can say that words are the important variable to predict whether it is spam or ham.

Understand carefully what I am saying once more, I am saying that I can use words as my predicting variable for spam or ham only when the word cloud is giving me different sets of words for spam or ham that is the only case.

(Refer Slide Time: 07:06)



So, what I do next is, I divide the dataset into spam and ham. Okay, so subset the raw data into spam and ham and now with once I will draw it with spam and then I will draw it with ham.

So, I have drawn it with spam, okay so let me draw it once more, it did not come properly, so I will draw it with spam and spam if you see carefully, the words are coming up call, free, now, mobile So, some words just remember the claim, just remember this words and for ham, the words that are coming up are just, will, do not, can, so call and now are same but claim or let us say mobile, these words are going away.

So, there is some difference between the words that are common. So, that suggest that words can be one variable to use that. I have also shown how to create very beautiful word clouds which can be used for your reporting purpose. Now, imagine how many words are there?

(Refer Slide Time: 08:31)



So if I just find out, find frequent words and then put it one here it will find out all those words which has a frequency of at least 1. And there are 5738 words are there. So, sorry not six thousand, thousand are printed and omitted are 5738 so total 6738 words are there. I mean there is no point on playing with so many words. If they are occurring only once, why will have an importance on them. So, what I do is, I find out only those words which have up to 5 frequency and those words are 1 2 1 6, 1216 words.

So, what I do next is put these words in a dictionary and then again create this document from matrix training and testing, whichever I have created here, this document on matrix DTM, training and testing. I am recreating them but now with the dictionary of this data, not the whole set of words.

So, I am creating a document on matrix, if you remember document on matrix was the terms in one side, the document numbers and their corresponding counts, Counts okay, not whether it occurred or not. How many times it occurs was there? Now, remember in my Naive Bayes algorithm when I taught Naïve and Bayes equation it was just telling Viagra was there or not? Yes or no.

Grocery was there or not? Yes or no. Unsubscribe word was there or not? Yes or no. it was not talking about whether that word is occurring 5 times in a particular row or 2 times in a particular row. The frequency was not there. The only thing that was a variable was the occurrence whether it occurred or not. So, I have to also do something to change this document on practice which is frequency oriented to a occurrence oriented.

So, I am creating a function called convert counts which takes a value of x, if the x's value is > 0, any positive value, it will make it 1, otherwise 0. And then 0 and 1, it is leveling at as no and yes.

(Refer Slide Time: 10:42)





So, that is how I am writing a function and then I will use this function on my training and testing DTM document on matrix, to convert them to this kind of a thing. So, if I just run this, it will take little bit of time. If I just run this one and convert ten to ES no basically instead of the counts, the training one has been done but testing one also has been done, the summary of the training one looks like this.

So, this are the word, first 5 words, first 5 randomly chosen, no, no it is not like the first 5 words based on the frequency or something. So, this word is occurring, not occurring for 162 times, occurring 7 times. Would is occurring 163 times and not occurring 4006 n times and so on. So that is how I have created the summary.

(Refer Slide Time: 11:46)

• 👒 💣 • 🕞 🗐 💼 🗛 Go to file/function 🔤 🖾 • Addins •			Project	t (Nore
nbr =	Environment Histor	y Connections		-
🖉 🔄 Source on Save 🔍 🎢 - 📋 💛 Run 🐤 🕀 Source - 📚	🔄 🔒 🖙 import i	Dataset • 🧹	🗉 List	
55 secold a secold secold secold secold secold second seco	🐴 Global Environment	· · · · · · · · · · · · · · · · · · ·	9	
<pre>50 # apply() convert_counts() to columns or train/test data 57 sms_train <- apply(sms_train, 2, convert_counts)</pre>	o sms_dtm_train	Large DocumentTermMatrix (6	elements.	_ Q
<pre>58 sms_test <- apply(sms_test, 2, convert_counts)</pre>	O sms_raw	5559 obs. of 3 variables		
60 summary(sms train[, 1:5])	osms_raw_test	1390 obs. of 3 variables		
61	osms_raw_train	4169 obs. of 3 variables		
62 - ## step 3: Training a model on the data 63 library(e1071) 64 sms_classifier <- naiveBayes(sms_train, sms_raw_trainStype)	<pre>0 sms_test</pre>	Large matrix (1690240 elemen	ts, 13.1 .	- 0
	Osms_train	Large matrix (5069504 elemen	its, 39 Mb)
65 names(sms_classifier)	Files Plots Packag	pes Help Viewer		-
67 sms_classifierStables[1:2]	O Install			-
68	Name	Description	Version	
09+ ## Step 4: Evaluating model performance ,	Hear Library			
641 🚺 Step 3: Training a model on the data 1 R Script 1	over clurary	Minima Association Bulley and Economic		1
nsole Terminal ×	arues	Mining Association Rules and Frequent Itemsets	1.0-4	0
/Users/NPTEL Studio-01/Desktop/Week10/Session 4/naive bayes/ 🤌 🏮 🃝	askpass	Safe Password Entry for R, Git, and SSH	1.1	0
## Step 3: Training a model on the data	assertthat	Easy Pre and Post Assertions	0.2.1	0
<pre>inbnary(e10/1) ror in library(e10/1) : there is no package called 'e10/1' install.packages("e10/1")</pre>	backports	Reimplementations of Functions Introduce Since R-3.0.0	d 1.1.5	0
stalling package into 'C:/Users/NPTEL Studio-D1/Documents/R/win-librar	base64enc	Tools for base64 encoding	0.1-3	0
3.6'	ВН	Boost C++ Header Files	1.72.0-3	0
s 'no is unspecified)	D broom	Convert Statistical Analysis Objects into Ti Tibbles	dy 0.5.4	0
	C call	Call R from R	3.4.1	0
	colleannas	Translate Spreadsheet Cell Ranges to Row	5 110	0



Now what will I do? I will run a Naive Bayes now, the library is E 1071, there is no such package, so I have to install the package. So, E 1071 is the package name. I am installing it. Okay, it has been installed I am calling the library now. It has been called, there is a warning sign. Well do not care the warning sign probably and the function is Naive Bayes but the DTM is by x variable and the y variable is the SMS raw train, the raw data, raw training data, type variable is my basically the y variable.

So, now if I just run this thing, Naive Bayes, it will take some time to train, it is training right now. It will take some time. Okay, that is trained. And if I just want to see the tables.

(Refer Slide Time: 12:51)



So, the first two tables, it will create, if there are 4 words, if you remember, there were 4 tables. For Viagra word there was 1 table, and there was Viagra, Unsubscribe, Grocery and so Money, there were 4 tables.

Here I have taken 1216 words, top words which has at least 5 frequency. So, there will be 1216 these things. And I have taken only first 2. You can you can choose which word you want to choose. So, if you see carefully, that when there is ham, given it is ham, the probability of checking occurring is 0.99 and checking occurring is 0.002 and checking not occurring is 0.998.

And given it is spam, its always is it is 0 and it is 1. Similarly given it is spam, the probability often is a numerator, so P of n given s, remember. So, given it is spam, the probability of this occurring is 0.1 and 0.98. So, there is some distance here, 0.01 and 0.04, some distance here but there is probably very lower distance here.

If the 0.000 and 0.002, so we have to check for different, so if you remember there was a word called, there was a word called mobile or there was a word called, claim. So, let us find out what the word thing is? Okay, so I will just quickly find out the position of the word called claim. So, how will I find out that?

(Refer Slide Time: 14:32)

db/ imicondent Hilling Connection wordcloud(sms_corpus_train, min.freq = 30) imicondent Hilling Connection 35 wordcloud(sms_corpus_train, min.freq = 30) imicondent Hilling Connection 36 span <- subset(sms_ram_train, type == "span") imicondent Hilling Connection 36 span <- subset(sms_ram_train, type == "span") imicondent Hilling Connection 36 span <- subset(sms_ram_train, type == "span") imicondent Linge Sims_corpus_tatist of 4169 37 smc/cloud(spanitext, max.words = 40, scale = c(3, 0.5)) conse.tist of 4169 38 smc/cloud(spanitext, max.words = 40, scale = c(3, 0.5)) conse.tist of 4169 38 smc/cloud(spanitext, max.words = 40, scale = c(3, 0.5)) conse.tist of 4169 39 sms_test <- DocumentTernMatrix(sms_corpus_test, list(dictionary sms_test <- DocumentTernMatrix(sms_corpus_test <- Matrix(sms_corpus_test <-	List
<pre>wordcloud(sss_corpus_train, sin.freq = 30) wordcloud(sss_corpus_train, sin.freq = 30) wordcloud(sss_corpus_train, sin.freq = 30, random.order = FALSE) } spam <- subset(sss_ram_train, type == "spam") ham <- subset(sss_ram_train, type == "spam") ham <- subset(sss_ram_train, type == "ham") ham <- subset(sss_ram_train, type == "tam") ham <- subset(sss_ram_train, type == 'tam") ham <- subset(sss_ram_train, type == 'tam 'tam') ham <- subset(sss_ram_train, type == 'tam') ham <- subset(sss_ram_train, typ</pre>	ements, Ntches to y differ in th
34 wordcloud(sms_corpus_train, min.freq = 30) 35 wordcloud(sms_corpus_train, min.freq = 30, random.order = FALSC) 36 Obsms_corpus_train 37 sman <- subset(sms_raw_train, type = "spam")	1.7 Mb) nts, 75 ements, ements, witches to y differ in th
<pre>35 wordcloud(sms_corpus_train, min.freq = 30, random.order = PALSC) 35 spam <- subset(sms_raw_train, type == "spam") 36 ma <- subset(sms_raw_train, type == "ham") 37 sms_train, type == "ham") 38 wordcloud(pamitext, max.words = 40, scale = c(3, 0.5)) 40 wordcloud(pamitext, max.words = 40, scale = c(3, 0.5)) 41 wordcloud(pamitext, max.words = 40, scale = c(3, 0.5)) 52 sms_ditext - clist of 1450 53 sms_ditext - Large DocumentTermMatrix(6 elem 54 sms_ditext - ObcumentTermMatrix(sms_corpus_train, list(dictionary 54 sms_text - DocumentTermMatrix(sms_corpus_train, list(dictionary 55 sms_text - DocumentTermMatrix(sms_corpus_train, list(dictionary 54 (hptiven):</pre>	1.7 Mb) nts, 75. ements, ements, itches to y differ in th
<pre>spam <- subset(sms_ram_train, type == "spam") hm <- subset(sms_ram_train, type == "ham") hm <- subset(sms_ram_train, type == "ham") wordcloud(spamitext, max.words = 40, scale = c(3, 0.5)) wordcloud(spamitext, max.words = 40, scale = c(3, 0.5), colors = t findfreqTerms(sms_dtm_train, 5) sms_train <- DocumentTermMatrix(sms_corpus_test, list(dictionary sms_train <- DocumentTermMatrix(sms_corpus_test, list(dictionary) sms_train <- DocumentTermMatrix(sms_corpus_test)</pre>	ements, ements, Mches to y differ in th
spam <~ subset(sms_ram_train, type == "spam") hm << subset(sms_ram_train, type == "ham") hm << subset(sms_ram_train, type == "ham") u wordcloud(spamitext, max.words = 40, scale = c(3, 0.5)) u wordcloud(spamitext, max.words = 40, scale = c(3, 0.5)) second = comparison = co	ements, ements, Mches to y differ in th
<pre>9 ham <- subset(sms_raw_train, type == "ham") 1 wordcloud(spanitext, max.words = 40, scale = c(3, 0.5)) 2 wordcloud(hamitext, max.words = 40, scale = c(3, 0.5), colors = t 4 find/reqTerms(sms_dtm_train, 5)) 5 sms_train <- pocumentTermMatrix(sms_corpus_test, list(dictionary 5 sms_tst <- pocumentTermMatrix(sms_corpus_test, list(dictionary 1 (bplown) : Ksopt : 1</pre>	ements, ements, Mches to y differ in th
0 wordcloud(spanitext, max.words = 40, scale = c(3, 0.5)) 2 wordcloud(spanitext, max.words = 40, scale = c(3, 0.5)) 2 wordcloud(spanitext, max.words = 40, scale = c(3, 0.5)), colors = t 4 findFreqTerms(sms_dtm_train, 5)) 5 sms_ditt < c(findfreqTerms(sms_dtm_train, 5))	ements, ements, Mches to y differ in th
1. wordcloud(spanitext, max.words = 40, scale = c(3, 0.5)) 2. wordcloud(spanitext, max.words = 40, scale = c(3, 0.5), colors = t 3. direction (sm_dtm_train, 5)) 5. sms_train - DocumentTermMatrix(sms_corpus_train, 1)st(dictionary 5. sms_train - DocumentTermMatrix(sms_train - DocumentTermMatrix(sms	ements,_ ements,_ stches to y differ in th
2 wordcloud(hamitext, max.words = 40, scale = c(3, 0.5), colors = t 4 findfreqTerms(sms_dtm_train, 5)) 5 sms_train < DocumentTermMatrix(sms_corpus_train, 1)st(dictionary 5 sms_tstar <- DocumentTermMatrix(sms_corpus_test, 1)st(dictionary 1 (bp:wn) : RSorpt : 1 (bp:wn) : RSorpt :	ements,
4 findFreqTerms(sms_dtm_train, 5) 5 sms_dict <- c(findFreqTerms(sms_dtm_train, 5))	Itches to y differ in th
File Pile	Itches to y differ in th
6 sms_dict <- c(findFreqTerms(sms_dtm_train, 5)) 7 sms_train <- DocumentTermMatrix(sms_corpus_train, list(dictionary 9 (top Lewn) : R Sorpt : 1 (Nches to y differ in th
7 sms_train ~ DocumentTermMatrix(sms_corpus_train, list(dictionary 8 sms_test <- DocumentTermMatrix(sms_corpus_test, list(dictionary 1 (hplown) : 8 South: 1	Nches to y differ in th
<pre>& sms_test < DocumentTermMatrix(sms_corpus_test) ist(dictionary -</pre>	Nches to y differ in th
1 (Replayed) : R Script : grep, grep1, regexpr, gregexpr and regexec search for matc	atches to y differ in th
grep, grepl, regexpr, gregexpr and regexec search for match	atches to y differ in th
The second se	y differ in th
some terminal agument pattern within each element of a character vector; they d	
Iners/NPTEL Studio-01/Desktop/Week10/Session 4/nalve bayes/ 🖗 🖌 format of and amount of detail in the results.	
851 "water" "tonight" "freephone" aub and grup perform replacement of the first and	pectively
88] "drug" "regards" "recently"	accred.
91] "south" "hard" "workin" Usage	,
94] "vikky" "Imao" "bother"	1
J] "medical" "track" "glad" grep(pattern, x, ignore.case = FALL	Angal
fixed = FALSE, useBytes	Var
reached getoption("max.print") omitted 216 entries]	51/1
grepi(pattern, x, ignore.c	-
187 Séd	1
sub(pattern, replacement	
	1
Can line had leng hole ten mp	-
Cat ite Na Sues hat play hat be no Q 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	Project (
Con ten fan ten ber ber pen fan ten ne Con ten fan ten fan ten fan ten ne Con ten fan ten fan ten fan ten ten ten ten ten ten ten ten ten te	Project (
Con two has twee bar here har here twee Con two has twee bar here har here twee Con two har here har here twee here here here here here here here h	Project (
Con Yes, Nos Sues Ant May Nes Sus rep Q	Project (
Con two fee tw	Project (
Gas ten Nes been her bes ner Importante	Project (List + 1.7 Mb)
Con ten han been han ben hen	Project (List + 1.7 Mb) nts, 75_
Con ten has has has has has as Con ten has	Project (List + 1.7 Mb) nts, 75_
Con ten Non bar hen ne Con ten Non bar hen ne dr * Constitution Constitution Constitution Constitution dr * Constitution Constitution Constitution dr * Constitution Constitution dr * Constitution Constitution dr * Constitution Constitution dr * Constitu	Project (List + 1.7 Mb) nts, 75_
Cox tex has	Project (List - 1.7 Mb) nts, 75_
Com ten Nan base har jang han ban me Com ten Nan base har jang han ban me Of a ten Nan base har jang han ban me of apply() convert_counts() to columns of train/test data sms_train < apply(sms_train, 2, convert_counts) sms_test <	Project (List - 1.7 Mb) nts, 75_ ements,
Com ten Ann Jean Ann Jean Ann Ann Q	Project (List - 1.7 Mb) nts, 75_ ements,_ ements,_
Con ten her	Project (List - 1.7 Mb) nts, 75_ ements, ements,
Cas we has been her her her her her Cas we has been her her her her Cas we has been her her her Cas we have her her her her her her Cas we have her her her her her her her Cas we have her her her her her her her her her he	Project (List - 1.7 Mb) nts, 75_ ements,_
Can ten fan fan fan fan fan in Can ten fan fan fan fan fan fan fan fan fan fa	Project (List - 1.7 Mb) nts, 75_ ements,
Con ten has have have have have me Con ten has have have have have have me Con ten has have have have have have have have have	Project (List - 1.7 Mb) nts, 75_ ements, ements,
Com ten Ann Jean Ann Ann Ann Ann Ann Ann Ann Ann Ann A	Project (List - 1.7 Mb) nts, 75_ ements,_
Com Ten New	List
Con the law has has me Con the law has has me Con the law has has me Control to conserve the law has has me constrained to constrain the law has have constrained to constrained to c	List of the solution of the so
Com ten fan fan fan fan fan in Com ten fan fan fan fan fan fan fan fan fan fa	Lut v Lut v 1.7 Mb) nts, 75_ ements,
Ga we has been bar here here we have here we have here here here here here here here he	List - ements stches to y differ in th
Come Mon Nume Not New York States Address - Come Mon Nume Not New York States Address - Come Mon Nume Not New York States Address - Come Mon Nume Nume Nume Nume Nume Nume Nume Nume	Unt view of the second
Come the base has been her bas and Come the base has been her base and Come the base her base her base and Come the base her base her base and Come the base her base	Project () Ust - Ints, 75- ereents, stitles to y differ in th
Come Mon Name Name Name Come Mon Name Name Name Come Mon Name Name Come Mon Name Name Come Mon Name Name Come Name Name Come Name Name Come Name Name Come	Units in the spectively.
Come Non None None None None None Come None None None None None Come None None None None Come None None Come None None Come None None Come None Com	Project (List - List - List - List - ThD) nts, 75 ements, witches to y differ in th spectively.
Cas we has been her here here we Cas we has been her here here here here here here he	I project of List - List - 1.7 Mb) nts, 75 essents, wither in the spectively ya2
Ga to No. None Not None Not None Not	Project (List
Gate to Name Name Name Name Name Addims - Gate to Name Name Name Name Name Addims - Gate to Name Name Name Addims - Gate to Name Addims - </td <td>Intervention</td>	Intervention
Ga to No. None Not Not None Not	B Project () B List
Come None None None None None None None Non	Lut via and the second

So, find frequent terms or SMS DICT. So claim, wait, grep, grep function is written like this, grep so the help will tell me that how pattern and then x, okay so grep, then the pattern is claimed and what I am searching SMS underscore dict. So, if it is 187 okay so 187 and 566. So, if I just check 566 word, 566 is claimed to claim, so that is not so important word, but 187, 187 is the word called claim.

So, I will just check this particular table whatever I created for 187, word number 187 let us see, and then you will understand why I am saying this.

(Refer Slide Time: 15:46)



So, if I just write 187, this is the claim word, summary of SMS train, see 187. Oh no, not sorry, no, no, sorry-sorry, not summary, this one, not summary, this one and if I just write 187 here, you see that for claim, when it is spam, the probability is 0.5, that the claim word will be there.

Ham is 0.84 and when it is ham the probability is 0 and 1 so there is a huge distance 0.00 and 0.15, 15 % distance is a very high distance. So, that is something that we are trying to find out here.

(Refer Slide Time: 16:42)

Console Terminal =

C:/Users/NPTEL Studio-01/Desktop/Week10/Session 4/naive bayes/

Content type 'application/zip' length 114408 bytes (111 KB) downloaded 111 KB

package 'gtools' successfully unpacked and MDS sums checked package 'gdata' successfully unpacked and MDS sums checked package 'gmodels' successfully unpacked and MDS sums checked

The downloaded binary packages are in C:\Users\NPTEL studio-01\AppData\Local\Temp\RtmpuivSwv\downloa ded_packages

• On Pr . D Cl A combatering Adding .		B Insiert @	
		Project (-
nbr =	Environment History Connections	,	=
Source on Save Q V · [-+ Run ++ -+ Source ·	🤏 🔐 🔐 Import Dataset • 🔏	Elst •	
60 summary(sms_train[, 1:5])	Global Environment •	9,	
61	<pre>osms_test Large matrix (1690240 eleme</pre>	nts, 13.1 _	
62 ## Step 3: Training a model on the data	Osms_train Large matrix (5069504 eleme	nts, 39 Mb)	
54 sms_classifier <- naiveBayes(sms_train, sms_raw_trainStype)	O spam 564 obs. of 3 variables		
65 names(sms_classifier)	Values		
66 57 sms_classifierStables[1:2] 58	sms_dict chr [1:1216] "checking" "go sms_test_pred Factor w/ 2 levels "ham","s	od" "hope" " pam": 1 1 1	
69 - ## Step 4: Evaluating model performance	Files Plots Packages Help Viewer		j
<pre>// Sms_test_pred <- predict(sms_classifier, sms_test) // library(omodels)</pre>	D Install () Undate		
72 CrossTable(sms_test_pred,sms_raw_testStype,prop.chisg = FALSE,	Name Description	Version	
<pre>/3 dnn = c('predicted', 'actual'))</pre>	a consequent	Paratan	
14) 🖸 Step & Evaluating model performance : R Scr	User Library		
sole Terminal =	arules Mining Association Rules and Frequent Itemsets	1.6-4	
Users/NPTEL Studio-01/Desktop/Week10/Session 4/naive bayes/ 🔅	askpass Safe Password Entry for R, Git, and SSH	1.1	
## Step 4: Evaluating model performance	assertthat Easy Pre and Post Assertions	0.2.1	
sms_test_pred <- predict(sms_classifier, sms_test) library(gmodels) nor in library(amodels) : there is no package called 'amodels'	backports Reimplementations of Functions Introduce Since R-3.0.0	ed 1.1.5	
CrossTable(sms_test_pred,sms_raw_testStype,prop.chisq = FALSE,	base64enc Tools for base64 encoding	0.1-3	
<pre>dnn = c('predicted', 'actual'))</pre>	BH Boost C++ Header Files	1.72.0-3	ï
ror in CrossTable(sms_test_pred, sms_raw_testStype, prop.chisq = F/ : could not find function "CrossTable"	ALSE broom Convert Statistical Analysis Objects into Tibbles	idy 0.5.4	1
	Call R from R	3.4.1	ï
	cellranger Translate Spreadsheet Cell Ranges to Row	is 1.1.0	
	101	-	i
Con two Rets bases Ault Datas Aults Nate Nate • ♥♥ ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● ● On Interfunction ● ● ● Addins +		🚺 Project Ø	N
nbr =	Environment History Connections		j
Source on Save Q. Z +] - Bun ** # Source of	2 🔐 🕞 📴 Import Dataset + 🖌	T par -	ĺ
9	Global Environment •	0.	
50 summary(sms_train[, 1:5])			
52 - ## step 3: Training a model on the data	Osms_test Large matrix (1690240 eleme	nts, 13.1 -	
33 library(e1071)	Ornam 564 obs of 3 variables	nrs, 33 MD)	
<pre>sms_classifier <- naiveBayes(sms_train, sms_raw_trainStype) is names(sms_classifier)</pre>	Values		
66 67 sms_classifierStables[1:2]	sms_dict chr [1:1216] "checking" "go sms_test_pred Factor w/ 2 levels "ham"."s	od" "hope" " pam": 1 1 1	
0. ## Step 4: Evaluating model performance			
<pre>70 sms_test_pred <- predict(sms_classifier, sms_test)</pre>	Files Plots Packages Help Viewer		ŝ
71 library(gmodels)	🖸 Install 👔 Update 🔍 🔍		
<pre>// crosstable(sms_test_pred,sms_raw_test)type,prop.chisg = FALSE, // dom = c('predicted', 'actual'))</pre>	Name Description	Version	
Contract in a second second to the second se	User Library		
143 🖬 Step 4: Evaluating model performance 4 R Scr	4F 2		

Mining Association Rules and Frequent

Boost C++ Header Files

Itemsets Safe Password Entry for R, Git, and SSH 1.1 Easy Pre and Post Assertions 0.2.1

 Reimplementations of Functions Introduced
 1.1.5

 Since R-3.0.0
 Tools for base64 encoding
 0.1-3

 Boost C++ Header Files
 1.72.0-3

Convert Statistical Analysis Objects into Tidy 0.5.4 Tibbles Call R from R 3.4.1

Call R from R 34.1 © Translate Spreadsheet Cell Ranges to Rows 1.1.0 ©

User Library arules

askpass assertthat

backports

base64enc

BH

broom

callr cellranger

-0

1.6-4 0

0.1-3

1.72.0-3 0

0

0

0

0

• 9	G 1	CHI BI A	Go to De/tun	tion	Addins •						Project (Non
nbr =						Enviro	nment H	tistory C	onnections			-
	0180	Source on Save	97.1	🖂 🗃 Ri	un 🗣 📑 Source 🔹 📚	01	a IPin	port Datase	n • 💰		II List	
59					*	🚯 Gle	obal Environ	ment +		Q,		
61	summary (si	ns_traint, 1	:91)			O sm	test	L	arge matrix (1690	240 elements.	13.1	
62 -	## step 3	Training a	model on	the data		O sms	train	- u	arge matrix (5069	504 elements.	39 Mb)	
64	library(e	1071) ifier - naiv	veRaves (sm	s train sms	raw train(type)	O spa	Am	50	64 obs. of 3 vari	ables		
65	names (sms,	classifier)	repayestan	a	ran_cramscype/	value	15					
66	66 57 sms_classifieritables[1:2] 58					smi	_dict	cl	hr [1:1216] "chec	king" "good"	"hope"	٩.
68						smt	s_test_p	ored Fa	actor w/ 2 levels	evels "ham","spam"		
69 -	## Step 4	: Evaluating	model per	formance		Files	Plots P	ackapes	Help Viewer			
70 1	sms_test_	pred <- pred	ict(sms_cl	assifier, sms,	_test)	-		Indate	and there	á		7
72	crossTable	e(sms_test_p	red, sms_ra	testStype,p	rop.chisq = FALSE,	Name		De	scription			
73		dnn = c('p	redicted',	'actual'))								
73:43	🚺 Step 4 Eva	luating model perfo	rmance \$		R Script ±	User Library						
Console	Terminal =					9.9	rules	M	ining Association Rules a	nd Frequent	1.6-4	1
C/Users/	NPTEL Studio	01/Desktop/Week	10/Seision 4/na	ive bayes/ 10	1	0.	iskpass	Sa	ife Password Entry for P	100	1.1	1
pred	dicted	han	span	Row Total	1	0.	ssertthat	Ea	isy Pre and Post Asser		0.2.1	1
	han	1203	32	1235		0	oackports	Re	implementations of Function of Function	GR	1.1.5	1
	i	0.997	0.175			0.8	base64enc	b	ols for base64 encoding	190 L	0.1-3	1
		0.865	0.023			0	3H	Bo	oost C++ Header Flor	1 mg/X	10-3	1
	span	4	151 0.974	155 0.112		0.1	moom	Co	sovert State	12-		1
	1	0.003	0.825			0.	alr	Ca	all R from	1	1	1
	1	0.003	0.109	1		0.	olleannor	Sec.	andata	100	101	

So, now based on these, I will predict with my SMS test as my testing data, I will predict and then in the testing data, we will try to see that whether prediction is working well or not. So, let it predict a little bit, it might take for only another 1minute time. So, we are handling pretty large dataset, 5000 data set, even if it is taking 30 second, 40 second, that is okay, good enough to deal with. So, let it predict. So, right now it is predicting for the testing data only and SMS test is around 1390 observations.

But there are 1216 variables that are there. So, based on that, it is predicting, yes, the predictions has been done and now if I just plot them, okay, so G model's package is not there I will quickly do G models. Install the package. Now if I run this you see out of 1390 observations, I am correct 1203 times and 151 times. So, 1303 plus 151 comes up to be how much? 1354?

(Refer Slide Time: 18:37)



So, 1354 / 1390, I am correct almost 97 % times on this data set. Okay and then if I just think about spam detection there was this is actual, this is predicted, so actually there was spam 183 and I am correctly predicting 151 out of them. So, 151 / 183, I am also correct 82 % times in the spam detection part. So, no correct in the ham detection part, there is correctness spam detection part but still is giving pretty good result. So that is how we can use Naive Bayes algorithm in spam detection.

So, that is all for this particular video, in the next video or probably in the next week, we will use all whatever we have learnt in further more advance application like sentiment mining and emotion mining to find out newer insights. So, thank you very much for being with me in this particular video and I will see you in the next week. Thank you.