In [772	on boo	ks' ratings and My module will	sion of a book of user numbers. receive the inorder the books	I used predict	tive analysi books that a	is, Linear a certain u	Regressic user read	on, and dat and rated,	a visuali use them	sation using to rate all
In [803	import import import	pandas <b>as</b> pd numpy <b>as</b> np seaborn <b>as</b> sns	ng all the neces		5					
In [804	#We ge	klearn.model_se  t the datasest  pd.read_csv("C:	by starting wir /Users/Home/Des	train_test_spl	data					
In [805 In [806	All1 =	check how our	st' data ::/Users/Home/De dataframe look:		3.CSV")					
Out[806	<b>book</b> l	D title	<b>authors avera</b> Ray Bradbury		s <b>bn isb</b>		<b>ge_code nur</b> eng	m_pages rati	ngs_count	text_reviews_cou
	<b>1</b> 89	Lord of the	John Steinbeck William		9.780000e		eng	103 182	1755253	255 261
	<b>3</b> 772	The Metamorphosis and Other Stories	Golding  Franz  Kafka/Jason  Baker/Donna  Freed		9.780000e		eng	224	38114	7
	<b>4</b> 10	Dune Messiah	Frank Herbert	3.88 441172		·+12	eng	331	97494	23
In [807 Out[807		scribe()  bookID average		n13 num_pages r						
		195.368421 3 568.391027 0	.918947 9.780526e+ .210816 2.294157e+	+12 306.210526 6	5.008104e+05 7.462868e+05	19.00 10743.31 11405.54	15789 2.84	00000 42105 14515		
	<b>50</b> % 1	546.000000 3 274.000000 3	.550000 9.780000e+ .825000 9.780000e+ .880000 9.780000e+	+12 196.500000 4 +12 259.000000 2		759.50 6734.00	2.00 2.00 2.00 3.00	00000		
In [808	max 7	723.000000 4	.995000 9.780000e+ .570000 9.790000e+	+12 704.000000 2	2.095690e+06	18873.00 35877.00	00000 4.00	00000 00000	the plot	of the heal
	#has reserved seabor	<pre>ead, and see if lplot(x = 'aver n.axisgrid.Face</pre>	the correlation the cage_rating', y etGrid at 0x16c	on is high = 'ratings_cou			ica books,		the proc	of the book
	2.0 -	•		•						
	1.5 -									
	atings count	•	•							
	0.0 -	• • • •	•••							
In [809		do the same wi	verage_rating  th our test data  tage_rating', y		unt', data :	= All1)				
Out[809	<seabor< td=""><td>n.axisgrid.Face</td><td>etGrid at 0x16c</td><td>cb9bc1c0&gt;</td><td></td><td></td><td></td><td></td><td></td><td></td></seabor<>	n.axisgrid.Face	etGrid at 0x16c	cb9bc1c0>						
	4 -									
	ratings_count									
	1 -									
	0-	1 2 ave	3 4 erage_rating	1 1 1 4 5 5						
In [834 Out[834	array([	[ <axessubplot:t <axessubplot:t< td=""><td>st(bins = 58, restitle={'center' title={'center' title={'center'</td><td><pre>:'bookID'}&gt;, :'average_rating</pre></td><td>ng'}&gt;],</td><td>ize = (9,9)</td><td>))</td><td></td><td></td><td></td></axessubplot:t<></axessubplot:t 	st(bins = 58, restitle={'center' title={'center' title={'center'	<pre>:'bookID'}&gt;, :'average_rating</pre>	ng'}>],	ize = (9,9)	))			
		<pre><axessubplot:t [<axessubplot:t<="" pre=""></axessubplot:t></pre>	<pre>citle={'center' citle={'center' citle={'center'</pre>	<pre>:'ratings_coun' :'text_reviews :'user_rating'</pre>	t'}>], count'}>,					
	1000			8000 6000 4000						
	0 0	500001000001500001 num_pa			0000a5000@0000 ratings_count					
	10000 — 8000 — 6000 —			8000 6000 4000						
	2000	500001000001500000 text reviews		2000 0 5000010	00000150000200000 user_rating	002500003000000				
	10000 - 8000 - 6000 -	text_reviews	Count	10000 8000 6000	user_raung					
	2000	500001000001500000	200000250000300000	2000 0 5000010	0000015000020000	002500008000000				
In [810	#For t		eract the books ang to get the .						are not go	ping to recor
Out[810	Arr = Arr array([		24, 7723, 106, 40, 1845, 2122,			599, 968	,			
In [811	#We are	e not going to All1.drop(labe	use the unique els = 'isbn', as els = 'isbn13',	books identifixis = 1)		will get r.	id of thos	se columns		
In [812	<pre>i = 0 while</pre>	i < len(Arr): l1 = All1.drop(	All1[All1["bool			vith the A.	rr array.			
In [813	#We ge All1 =	t rid of the nu All1.dropna(ax	xis = 1)							
In [814	DSS.dr	op("bookID", ax	ating", "isbn",							
In [815 In [816	X_trai: #And c	<pre>e model validat n, X_test, y_tr reate our modul = LinearRegres</pre>	rain, y_test = 1	train_test_spli	it(X, y, tes	st_size = (	0.3, rando	om_state =	3)	
	module	<pre>.fit(X_train, y egression()</pre>								
	predic	3.72242818, 5.4	oredict(X_test)	20168, 2.17261	593, 2.5259	1383,				
In [819	#Now w # It i	2.36527998]) e check tha val	lidity of our mo	odule.			ntain mucl	n data		
Out[819 In [820	#We pr	524178851816 epare our test All1.drop(["ti	dataset tle", "authors'	<b>", "</b> language_co	ode <b>", "</b> publ:	ication_da	te"], axis	s = 1)		
In [821 In [822	Test =	Test.drop_dupl	icates()  f our predicted							
In [823	#We tu pd.Data	<pre>module.predic  rn it into a cc aFrame({ ser_rating" : F mns=["user_rati"]</pre>	olumn 'inal							
Out[823		mns=["user_rati	9 11							
	1 2 3	0.645494 1.445715 -3.554119								
	4  11103	3.161369  6.757283								
	11104 11105 11106	6.531840 7.215365 7.988110								
	<b>11107</b> 11108 rov	7.567700 vs × 1 columns								
In [824 In [825	#To av		ing over 5 and		we give ou.	r Final co.	lumn limit	s from top	and bott	:om
In [826 In [829	All1 = All1.merge(Final.rename('user_rating'), left_index=True, right_index=True)  #Then we sort the books by the rating, which will show the order of books appearing at the beginning of the well									
Out[829	#This All1.s	is how our Test ort_values('use	er_rating', asce	like now	)					
	11107	45572 Fantastic Mr. Fox (Cover to Cover)	Roald Dahl	4.05	eng	90	7		0	9/15/1995
	7770	Science Fiction 29979 Treasury: Six Complete Novels	H.G. Wells/George Gesner	3.94	eng	688	17		1	9/19/1984
	7772	Notebooks 29990 of the Mind: Explorations of Thinkin	Vera John- Steiner	3.83	eng	288	17		1	1/23/1997
	7773	The Visible Hand: The 29993 Managerial Revolution in	Alfred D. Chandler Jr.	4.01	en-US	624	222		26	1/1/1977
	7775	29999 The Maltese Falcon	Dashiell	3.91 	eng 	213 	69254 	,	2997 	3/1/2005
	654 616	2107 Falconry & Hawking  The 2010 Santaroga	Frank Herbert	4.31 3.66	eng	352 256	51 1625		5	2/1/2006 9/16/2002
	615	Barrier  Harry Potter and the 2005 Half-Blood	J.K. Rowling	4.57	eng	768	1213		78	6/23/2006
	614	Prince (Harry J.K. Rowling's 2004 Harry Potter		3.58	eng	96	78		9	9/26/2001
	-	Novels: A Reader's  Emergence:	·		∵.3		.0			
		The								
	709		Steven Johnson	3.96	en-US	288	2769		200	9/10/2002