In [ ]:

```
#Please read the comments in this notebook
#Just try to play with the data
#If you are new to all these python tools, take this week in just learning them
#Learn pandas, matplotlib, seaborn basics from the resources in the document Python-
```

In [135]:

```
#Learn pandas, matplotlib, seaborn basics from the resources in the document Python-
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
news_popularity = pd.read_csv('OnlineNewsPopularity.csv') #read your data
pd.options.display.max_columns = None #This line of code just helps in displaying a
```

In [136]:

```
news_popularity.head() #head gives you the first 5 results from your dataframe
```

Out[136]:

	url	n_tokens_title	n_tokens_content	n_unique_tokens	n_
0	http://mashable.com/2013/01/07/amazon-instant-...	12	219	0.663594	
1	http://mashable.com/2013/01/07/ap-samsung-spon...	9	255	0.604743	
2	http://mashable.com/2013/01/07/apple-40-billio...	9	211	0.575130	
3	http://mashable.com/2013/01/07/astronaut-notre...	9	531	0.503788	
4	http://mashable.com/2013/01/07/att-u-verse-apps/	13	1072	0.415646	

In [137]:

```
news_popularity.columns #print the columns of dataframe
#if you see the column names below there is a space at the start of the column names
#For example: ' n_tokens_title' maybe confusing so you might convert it to 'n_tokens
#This might create problems for you, so news_popularity['n_tokens_title'] will not w
#news_popularity[' n_tokens_title'] would work

#So whats the fix? You can strip the spaces
```

Out[137]:

```
Index(['url', ' n_tokens_title', ' n_tokens_content', ' n_unique_token
s',
       ' n_non_stop_words', ' n_non_stop_unique_tokens', ' num_hrefs',
       ' num_self_hrefs', ' num_imgs', ' num_videos', ' average_token_
length',
       ' num_keywords', ' data_channel_is_lifestyle',
       ' data_channel_is_entertainment', ' data_channel_is_bus',
       ' data_channel_is_socmed', ' data_channel_is_tech',
       ' data_channel_is_world', ' weekday_is_monday', ' weekday_is_tu
esday',
       ' weekday_is_wednesday', ' weekday_is_thursday', ' weekday_is_f
riday',
       ' weekday_is_saturday', ' weekday_is_sunday', ' is_weekend', '
LDA_00',
       ' LDA_01', ' LDA_02', ' LDA_03', ' LDA_04',
       ' global_sentiment_polarity', ' global_rate_positive_words',
       ' global_rate_negative_words', ' avg_positive_polarity',
       ' avg_negative_polarity', ' title_sentiment_polarity', ' share
s'],
      dtype='object')
```

In [138]:

```
#Thats how you strip the spaces
news_popularity.columns = news_popularity.columns.str.lstrip()
```

In [127]:

```
#print the column again and see there are no spaces
news_popularity.columns
```

Out[127]:

```
Index(['url', 'n_tokens_title', 'n_tokens_content', 'n_unique_tokens',
       'n_non_stop_words', 'n_non_stop_unique_tokens', 'num_hrefs',
       'num_self_hrefs', 'num_imgs', 'num_videos', 'average_token_leng
th',
       'num_keywords', 'data_channel_is_lifestyle',
       'data_channel_is_entertainment', 'data_channel_is_bus',
       'data_channel_is_socmed', 'data_channel_is_tech',
       'data_channel_is_world', 'weekday_is_monday', 'weekday_is_tuesd
ay',
       'weekday_is_wednesday', 'weekday_is_thursday', 'weekday_is_frid
ay',
       'weekday_is_saturday', 'weekday_is_sunday', 'is_weekend', 'LDA_
00',
       'LDA_01', 'LDA_02', 'LDA_03', 'LDA_04', 'global_sentiment_polar
ity',
       'global_rate_positive_words', 'global_rate_negative_words',
       'avg_positive_polarity', 'avg_negative_polarity',
       'title_sentiment_polarity', 'shares'],
      dtype='object')
```

In [139]:

```
#Describe the basic statistics of the data (min, max values, mean, count etc.)
news_popularity.describe()

#Just check you shares variable (which is your target) do you see some anomaly or ou
```

Out[139]:

	n_tokens_title	n_tokens_content	n_unique_tokens	n_non_stop_words	n_non_stop_unique_
count	39644.000000	39644.000000	39644.000000	39644.000000	39644.
mean	10.398749	546.514731	0.548216	0.996469	0.
std	2.114037	471.107508	3.520708	5.231231	3.
min	2.000000	0.000000	0.000000	0.000000	0.
25%	9.000000	246.000000	0.470870	1.000000	0.
50%	10.000000	409.000000	0.539226	1.000000	0.
75%	12.000000	716.000000	0.608696	1.000000	0.
max	23.000000	8474.000000	701.000000	1042.000000	650.

In [140]:

```
#Lets just remove these outliers manually for simplicity (however the correct way mi

news_popularity = news_popularity[news_popularity['shares'] < 3500]
```

In [ ]:

```
#Read about Box plot and see how it works and what are its fundamental principles
#IGNORE THIS CODE FOR NOW, HOWEVER THIS IS ONE OF THE CORRECT WAYS OF REMOVING OUTL
# BUT FOR SIMPLICITY WE USE THE ABOVE CODE
'''

sns.boxplot(x=news_popularity['shares'])

def Remove_Outlier_Indices(news_popularity):
    Q1 = news_popularity[' shares'].quantile(0.25)
    Q3 = news_popularity[' shares'].quantile(0.75)
    IQR = Q3 - Q1
    trueList = ~((news_popularity[' shares'] < (Q1 - 1.5 * IQR)) |(news_popularity[
    return trueList

index_news_outlier = Remove_Outlier_Indices(news_popularity)
news_popularity = news_popularity[index_news_outlier]
'''
```

In [142]:

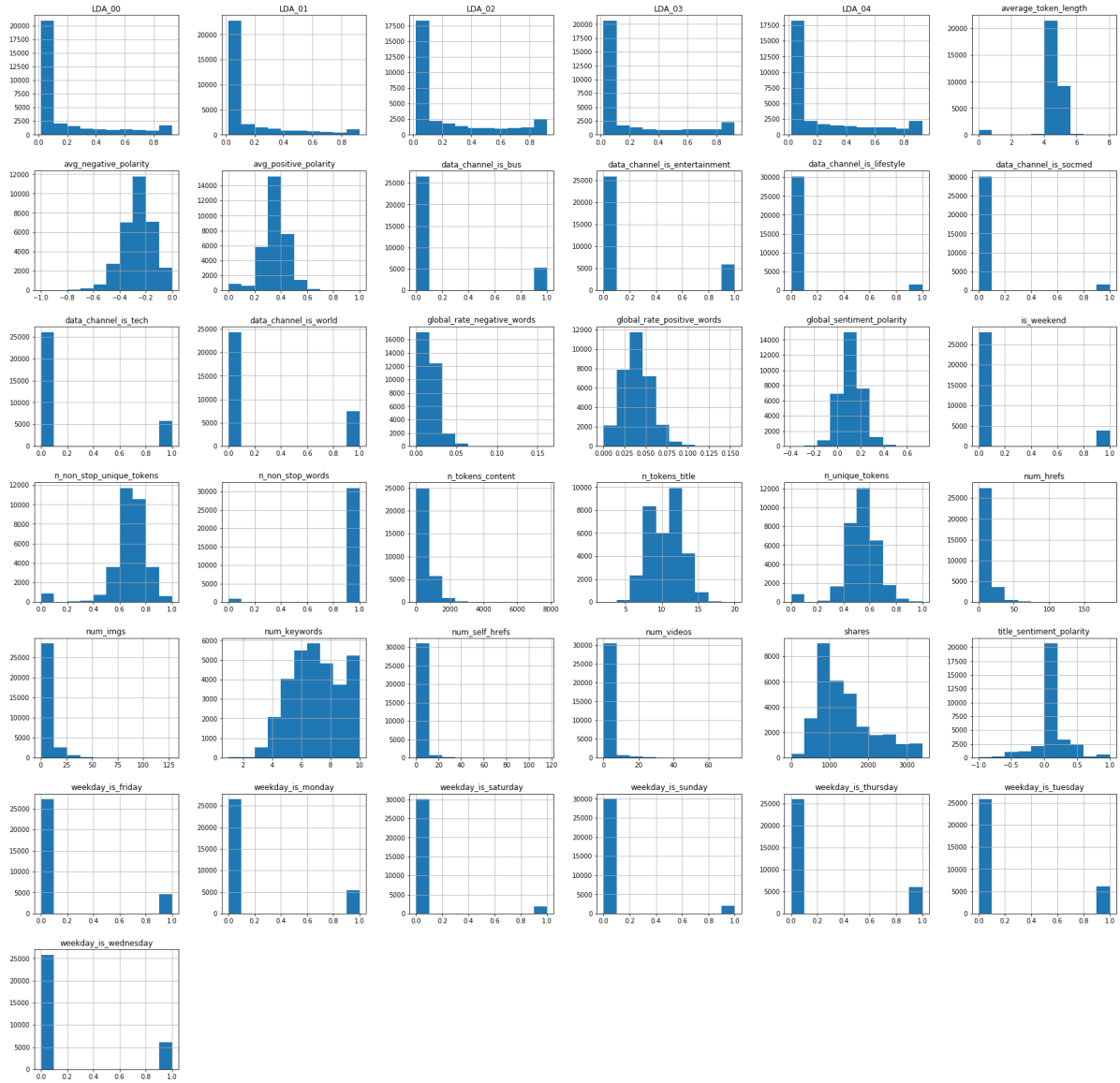
```
news_popularity.describe() #What do you see here? Are there extreme anomalies in the
```

Out[142]:

global_sentiment_polarity	global_rate_positive_words	global_rate_negative_words	avg_positive_polari
31836.000000	31836.000000	31836.000000	31836.0000
0.118151	0.039410	0.016584	0.3526
0.096016	0.017306	0.010728	0.1020
-0.393750	0.000000	0.000000	0.0000
0.056538	0.028087	0.009615	0.3050
0.117400	0.038647	0.015337	0.3569
0.176598	0.050000	0.021739	0.4090
0.727841	0.155488	0.162037	1.0000

In [166]:

```
#This takes some time to execute
news_popularity.hist(figsize=(30, 30));

#What does these historgrams represent?
```

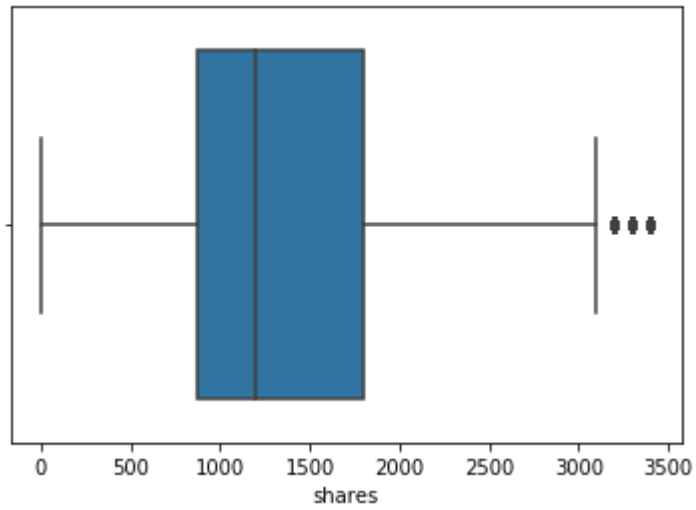


In [144]:

```
sns.boxplot(x=news_popularity['shares']) #Study about Box Plots #Whats an anomaly? W
```

Out[144]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1a4181ddd8>
```

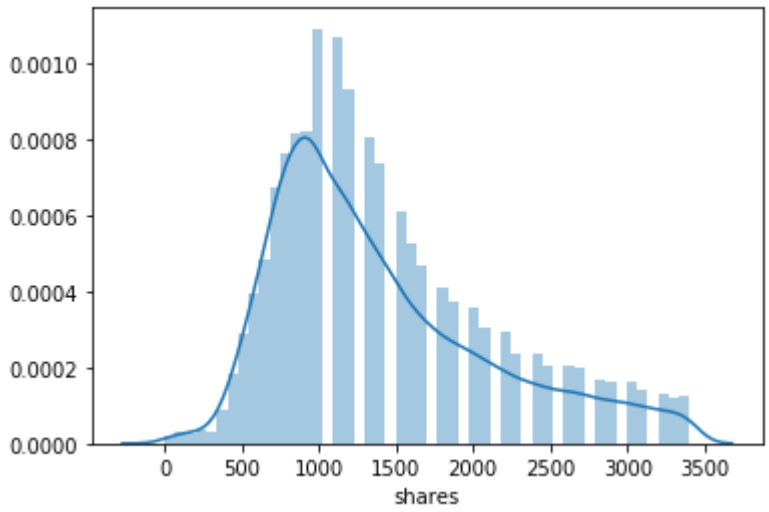


In [145]:

```
#Check the distribution of number of shares #What do you notice?
sns.distplot(news_popularity['shares'])
```

Out[145]:

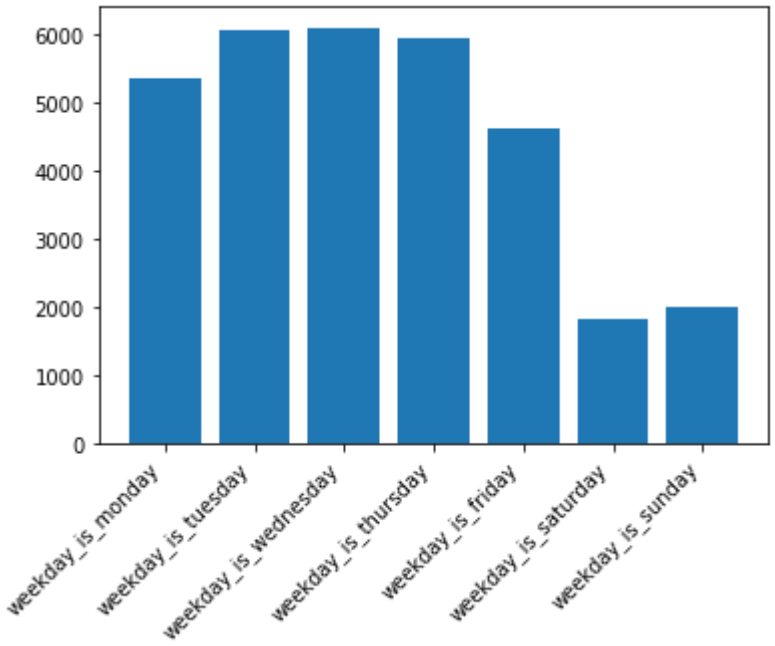
```
<matplotlib.axes._subplots.AxesSubplot at 0x1a40ff3390>
```



In [172]:

```
#This is a quick and dirty/bad code for plotting bar charts or count plots
#Try to improve this code if you can else just play around with the data, with the p

day_wise_count = []
day_of_week = ['weekday_is_monday', 'weekday_is_tuesday',
       'weekday_is_wednesday', 'weekday_is_thursday', 'weekday_is_friday',
       'weekday_is_saturday', 'weekday_is_sunday']

for item in day_of_week:
    day_wise_count.append(news_popularity[item].sum())
    print (item, ": ", news_popularity[item].sum())

plt.bar(day_of_week,height=day_wise_count,orientation='vertical')
plt.xticks(rotation=45, horizontalalignment='right',fontweight='light')

plt.show()

#So what do you infer from the chart below??
```

```
weekday_is_monday :  5354
weekday_is_tuesday :  6060
weekday_is_wednesday :  6093
weekday_is_thursday :  5936
weekday_is_friday :  4601
weekday_is_saturday :  1801
weekday_is_sunday :  1991
```

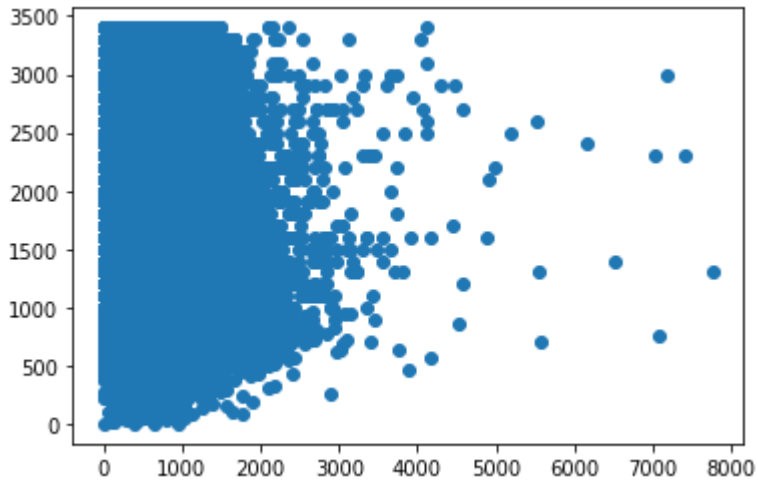


In [173]:

```
#Plot scatterplots for continuous variables, check for some patterns
#Lets plot a scatter plot of number of tokens in content v/s shares
plt.scatter(news_popularity['n_tokens_content'], news_popularity['shares'])
#What does the plot below tell you?
```

Out[173]:

```
<matplotlib.collections.PathCollection at 0x1a58f1b208>
```



In [ ]:

```
# Now try to create pie charts or again bar charts for other features like data_cha
#Plot a pie chart for tech, business, entertainment, socmed etc. (we have these in
#https://towardsdatascience.com/creating-a-basic-pie-chart-using-matplotlib-16dd3bf
```