In [1]:

```
import numpy as np
import pandas as pd
import os
import glob
pd.set_option('float_format', '{:f}'.format)

import warnings
warnings.filterwarnings('ignore')
```

In [2]:

```
df = pd.read_csv('./customer_data.csv', header=0, index_col=0)
```

In [5]:

```
df = df[['vint_dt', 'pr_enrll_any',
         'rwd_tier_dt', 'bl_3am_svm',
         'cr_bl_3am_svm', 'mled_acc_ct_svm',
         'meac_acc_ct_svm', 'mesd_acc_ct_svm',
        'fsvc_acc_ct_svm','cred_oacc_ct_svm',
      'opn_acc_ct_svm', 'rev_am_svm',
         'pfee_amt_svm','dep_oacc_ct_svm','ira_oacc_ct_svm',
         'mtg_oacc_ct_svm']]
```

In [6]:

```
df['vint_dt'] = pd.to_datetime(df['vint_dt'])
df['vint_dt_year'] = df['vint_dt'].dt.year
df['vint_dt_month'] = df['vint_dt'].dt.month
df['vint_dt_day'] = df['vint_dt'].dt.day
```

In [7]:

```
month_year = df.groupby(['vint_dt_month','vint_dt_year']).size().reset_index(n
ame='counts')
month_year.sort_values('counts', ascending=False).head()
```

Out[7]:

	vint_dt_month	vint_dt_year	counts
16	9	2016	6631
12	7	2016	6288
10	6	2016	5997
14	8	2016	5991
18	10	2015	5808

In [19]:

```
df = df[(df.vint_dt_month == 9) & (df.vint_dt_year == 2016)]
df['rwd_tier_dt'] = pd.to_datetime(df['rwd_tier_dt'])
df['rwd_tier_dt_year'] = pd.to_numeric(df['rwd_tier_dt'].dt.year, downcast='integer', errors='ignore')
df['rwd_tier_dt_month'] = pd.to_numeric(df['rwd_tier_dt'].dt.month, downcast='integer', errors='ignore')
df['rwd_tier_dt_day'] = pd.to_numeric(df['rwd_tier_dt'].dt.day, downcast='integer', errors='ignore')

df['rwd_tier_dt_year'] = df['rwd_tier_dt_year'].fillna(-1)
df['rwd_tier_dt_year'] = df['rwd_tier_dt_year'].astype(int)

df['rwd_tier_dt_month'] = df['rwd_tier_dt_month'].fillna(-1)
df['rwd_tier_dt_month'] = df['rwd_tier_dt_month'].astype(int)

df['rwd_tier_dt_day'] = df['rwd_tier_dt_day'].fillna(-1)
df['rwd_tier_dt_day'] = df['rwd_tier_dt_day'].astype(int)

df = df[((df.rwd_tier_dt_month == 9) | (df.rwd_tier_dt_month == -1)) & \
        (df.rwd_tier_dt_year == 2016) | (df.rwd_tier_dt_year == -1)]
df = df.reset_index(drop=True)
# df
df_cont=df[['vint_dt',
         'rwd_tier_dt', 'bl_3am_svm',
         'cr_bl_3am_svm', 'rev_am_svm']]
df_cat=df[[ 'pr_enrll_any',
         'mled_acc_ct_svm',
         'meac_acc_ct_svm', 'mesd_acc_ct_svm',
        'fsvc_acc_ct_svm','cred_oacc_ct_svm',
      'opn_acc_ct_svm',
         'pfee_amt_svm','dep_oacc_ct_svm','ira_oacc_ct_svm',
         'mtg_oacc_ct_svm']]
# df_cat
```

In [9]:

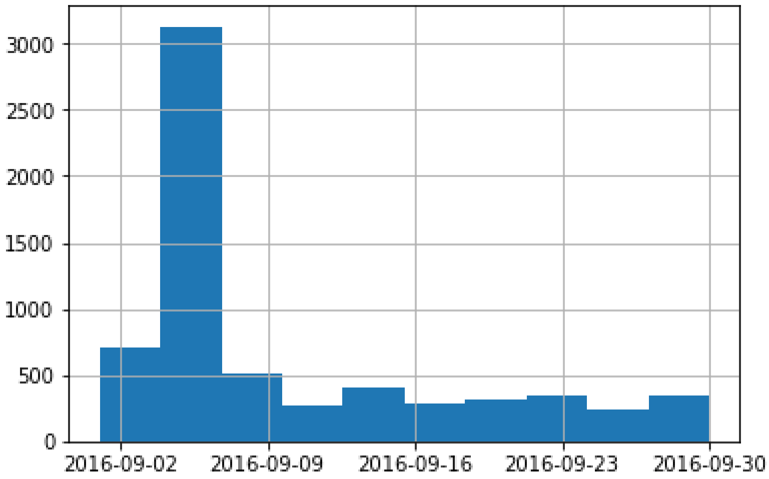
```
df.groupby(['pr_enrll_any']).size().reset_index(name='counts')
```

Out[9]:

	pr_enrll_any	counts
0	N	2883
1	Y	3675

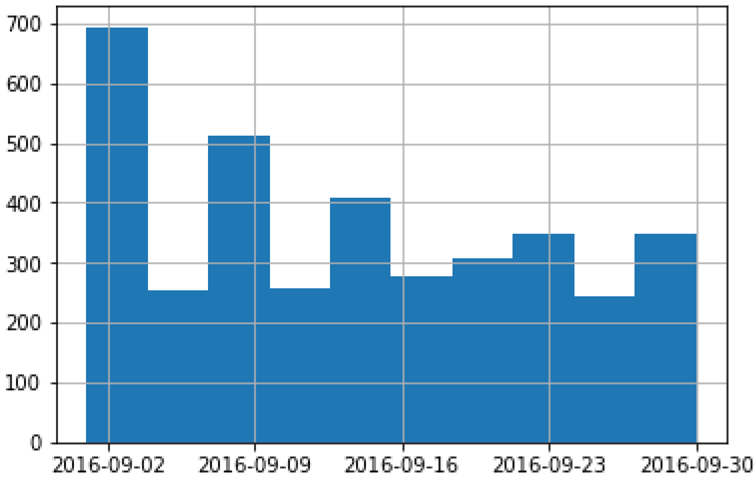
```
In [24]: df_cont['vint_dt'].hist()

Out[24]: <matplotlib.axes._subplots.AxesSubplot at 0xf03c0e1a58>
```



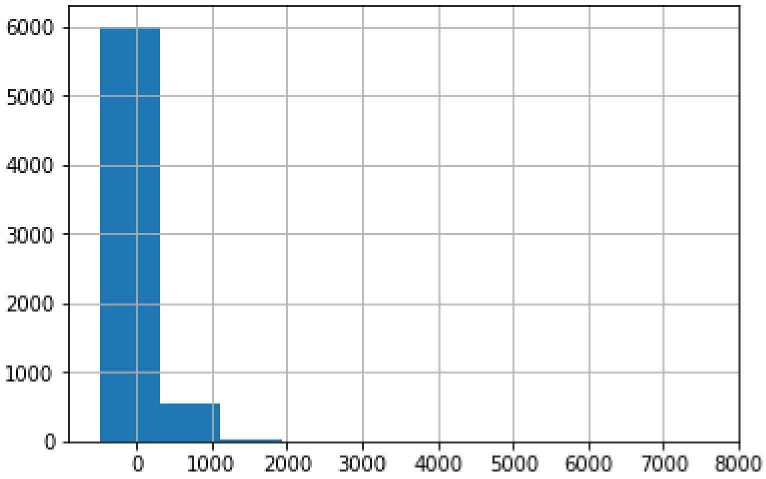
```
In [26]: df_cont['rwd_tier_dt'].hist()

Out[26]: <matplotlib.axes._subplots.AxesSubplot at 0xf03c16d4e0>
```

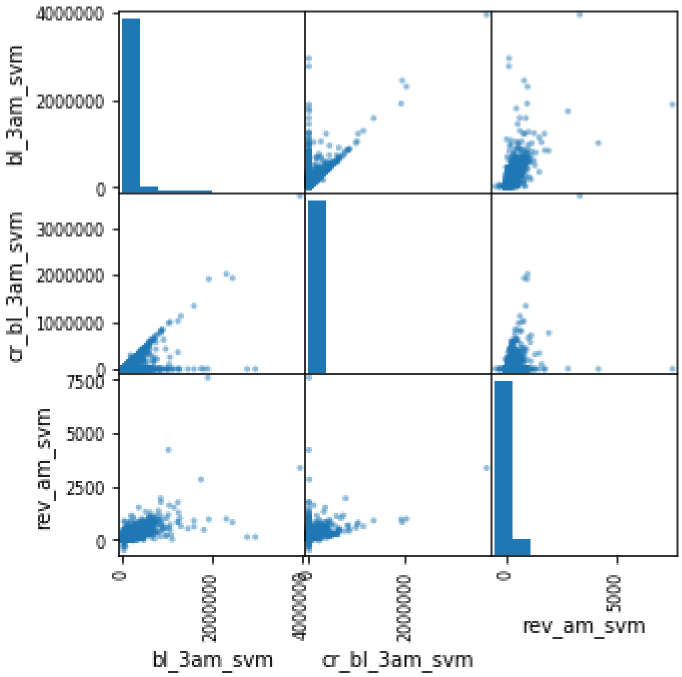


```
In [29]: df_cont['rev_am_svm'].hist()
```

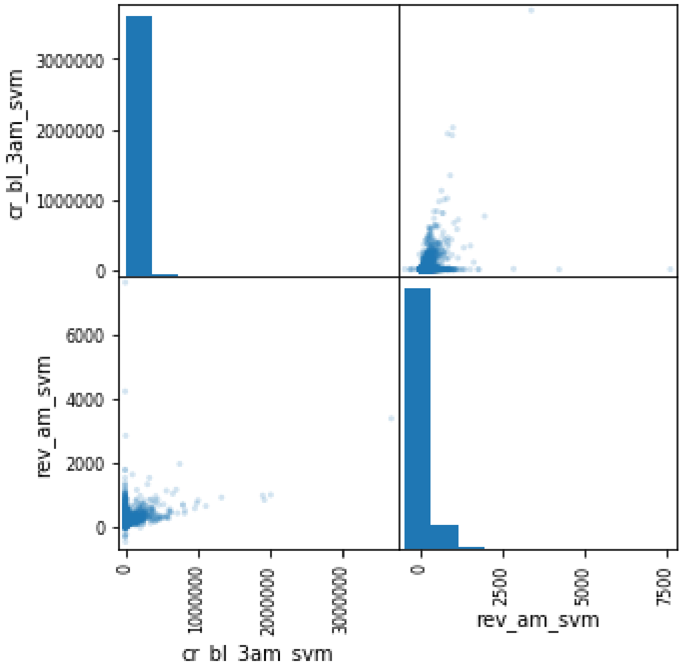
```
Out[29]: <matplotlib.axes._subplots.AxesSubplot at 0xf03c24cfd0>
```



```
In [32]: plot = pd.plotting.scatter_matrix(df_cont, figsize=(5,5))
```



In [31]:
```
plot = pd.plotting.scatter_matrix(df_cont[['cr_bl_3am_svm','rev_am_svm']], alpha = 0.2, figsize=(5,5))
```



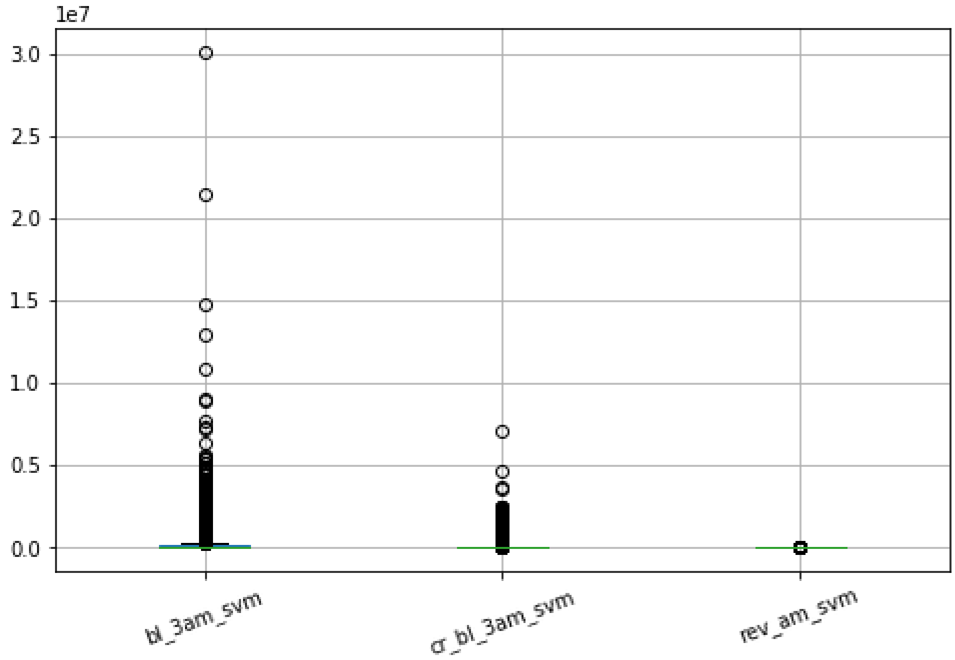
In [ ]:
```
# We observe that perhaps _____________ have exponential distributions, and ___________ have Gaussian distributions
# We observe ___________________are skewed therefore
#convert skewed preditors to log transformations
# we observe ________________ follow a pattern
# df[''] = np.log1p(df[''])



# plot = pd.plotting.scatter_matrix(df[['pr_enrll_any',' ',' ']], alpha = 0.2, figsize=(5,5))
```

In [10]:

```
#plotting outliers
ax = df_cont.boxplot(figsize=(8,5),rot=20)
```



In [23]:

```
#We observe the presence of a few outliers in our dataset lying far from the calculated mean
#in predictors such as bl_3am_svm and cr_bl_3am_svm
# We observe predictors like bl_3am_svm and cr_bl_3am_svm are skewed towards smaller values
#Analysing missing values
for col in df:
    print(col,': \tTrain:',df[col].isnull().values.any())
```

```
vint_dt :       Train: False
pr_enrll_any :  Train: False
rwd_tier_dt :   Train: True
bl_3am_svm :    Train: False
cr_bl_3am_svm :         Train: False
mled_acc_ct_svm :       Train: False
meac_acc_ct_svm :       Train: False
mesd_acc_ct_svm :       Train: False
fsvc_acc_ct_svm :       Train: False
cred_oacc_ct_svm :      Train: False
opn_acc_ct_svm :        Train: False
rev_am_svm :    Train: False
pfee_amt_svm :  Train: False
dep_oacc_ct_svm :       Train: False
ira_oacc_ct_svm :       Train: False
mtg_oacc_ct_svm :       Train: False
vint_dt_year :  Train: False
vint_dt_month :         Train: False
vint_dt_day :   Train: False
rwd_tier_dt_year :      Train: False
rwd_tier_dt_month :     Train: False
rwd_tier_dt_day :       Train: False
```

In [ ]:

```
#in order to check missing values in our dataset, we apply the function isnull
and find rwd_tier_dt has some null values which makes sense as
```

In [ ]:

```
#Through explanatory intital analysis, we were able to handle skewness in the
 data.
# We found out ______________ are highly correlated  with Enrollment status.
#Missing values-
#
```