PyCon 2015 大会 上海 - Python大数据分析与可视化

丁来强

- Email: wjo1212 at 163.com
- Wechat: LaiQiangDing

In [1]:

```
%matplotlib inline
%run ./env.py
```

分析500W客户数据

AdultFriendFinder.com

In [2]:

```
!ls a/*
```

```
a/aff0.txt  a/aff11.csv a/aff14.csv a/aff3.csv  a/aff6.csv  a/aff
9.csv
a/aff1.csv  a/aff12.csv a/aff15.csv a/aff4.csv  a/aff7.csv
a/aff10.csv a/aff13.csv a/aff2.csv  a/aff5.csv  a/aff8.csv
```

In [3]:

```
#!head -n 10 a/aff1.csv
```

In [4]:

```
def parse(x):
    try:
        year = int(x.split('-')[0])
        x = '20' + x if year < 10 else '19' + x
        return datetime.strptime(x, '%Y-%m-%d')
    except Exception as ex:
        return pandas.NaT

def parse_csv(x):
    df = pd.read_csv('a/aff{}.csv'.format(x),
                 parse_dates = ['birthdate'],
                 usecols=['sex', 'birthdate'],
                 date_parser=parse).dropna()  #.set_index('birthdate')
    df['age'] = 2015 - df.birthdate.dt.year
    df.loc[(df.age >= 70) | (df.age < 15)] = np.nan
    return df.dropna()

def parse_csv2(x):
    df = pd.read_csv('a/aff{}.csv'.format(x), usecols=['sex', 'age'])
    df.loc[(df.age >= 70) | (df.age < 15)] = np.nan
    return df.dropna()
```

In [5]:

```
df1 = parse_csv(1)
df2 = parse_csv(11)
dfs = [df1, df2] + [parse_csv2(x)
       for x in range(12, 16)]
df = pd.concat(dfs)
df.sex = df.sex.map({'1':'Man', '2':'Woman', 'Man':'Man', 'Woman':'Woman'})
```

```
/Users/wjo1212/python_analysis/lib/python2.7/site-packages/panda
s/io/parsers.py:1170: DtypeWarning: Columns (4) have mixed types.
Specify dtype option on import or set low_memory=False.
  data = self._reader.read(nrows)
```

男女比例如何?

In [6]:

```
df[:10]
```

Out[6]:

	age	birthdate	sex
0	41	1974-09-20	Man
1	46	1969-03-03	Man
2	61	1954-04-05	Man
3	57	1958-01-07	Man
4	44	1971-09-12	Man
5	54	1961-09-02	Man
6	51	1964-06-16	Man
7	64	1951-07-31	Man
8	51	1964-04-04	Man
9	65	1950-06-21	Man

In [7]:

```
#df1.sex.resample('A-DEC', how='count').plot(kind='area', figsize=(15,10))
#dfs[2].age.plot(kind='hist')
```

In [8]:

```
sex_dist = df.sex.value_counts()
sex_dist
```

Out[8]:

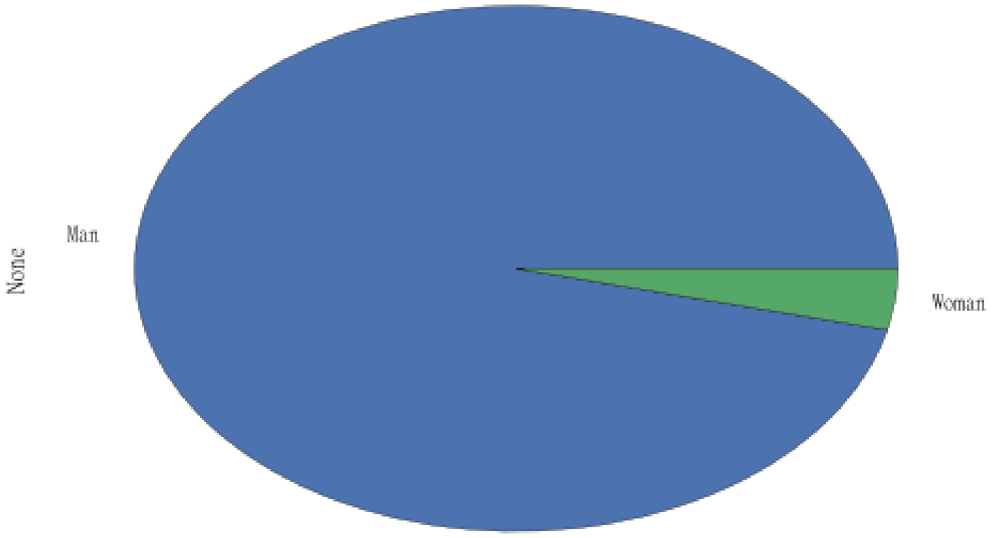
```
Man      1444520
Woman      56296
dtype: int64
```

In [9]:

```
sex_dist.plot(kind='pie')
```

Out[9]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x10742eed0>
```

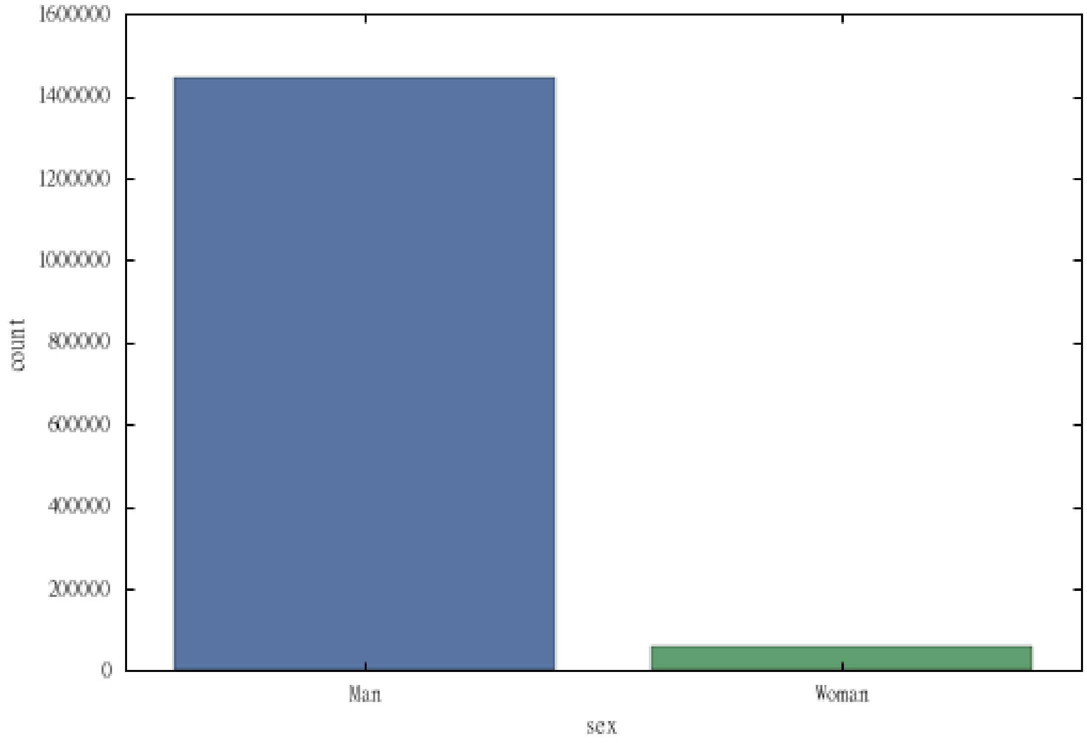


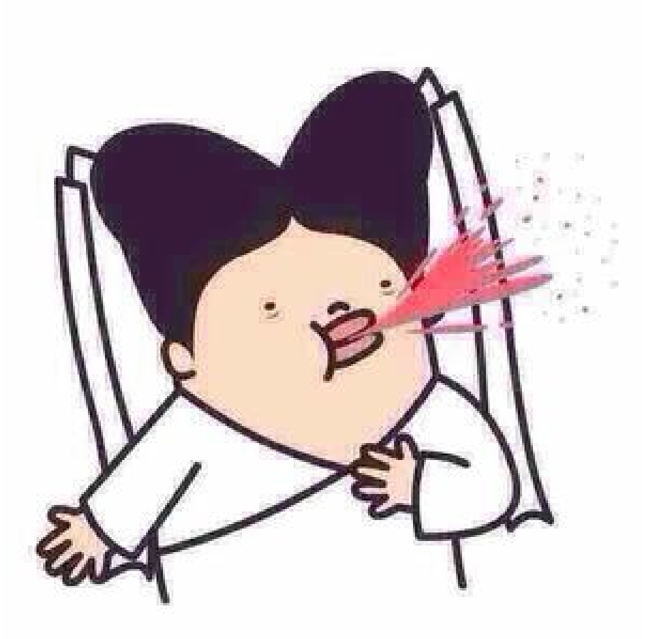
In [10]:

```
sns.countplot(x='sex', data=df)
```

Out[10]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1095a0990>
```





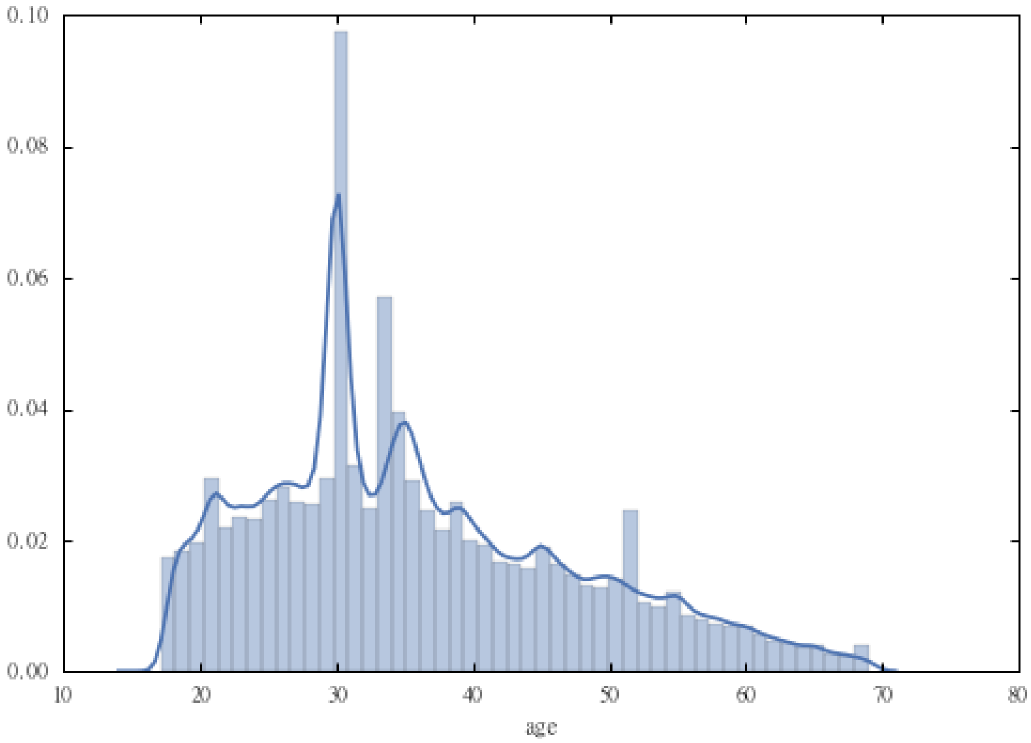
年龄分布如何?

In [11]:

```
sns.distplot(df.age)
```

Out[11]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x109585550>
```



In [12]:

```
from bokeh.charts import Histogram, output_notebook, show, Bar
output_notebook()
```

(http://bokeh.pydata.org)

BokehJS successfully loaded.

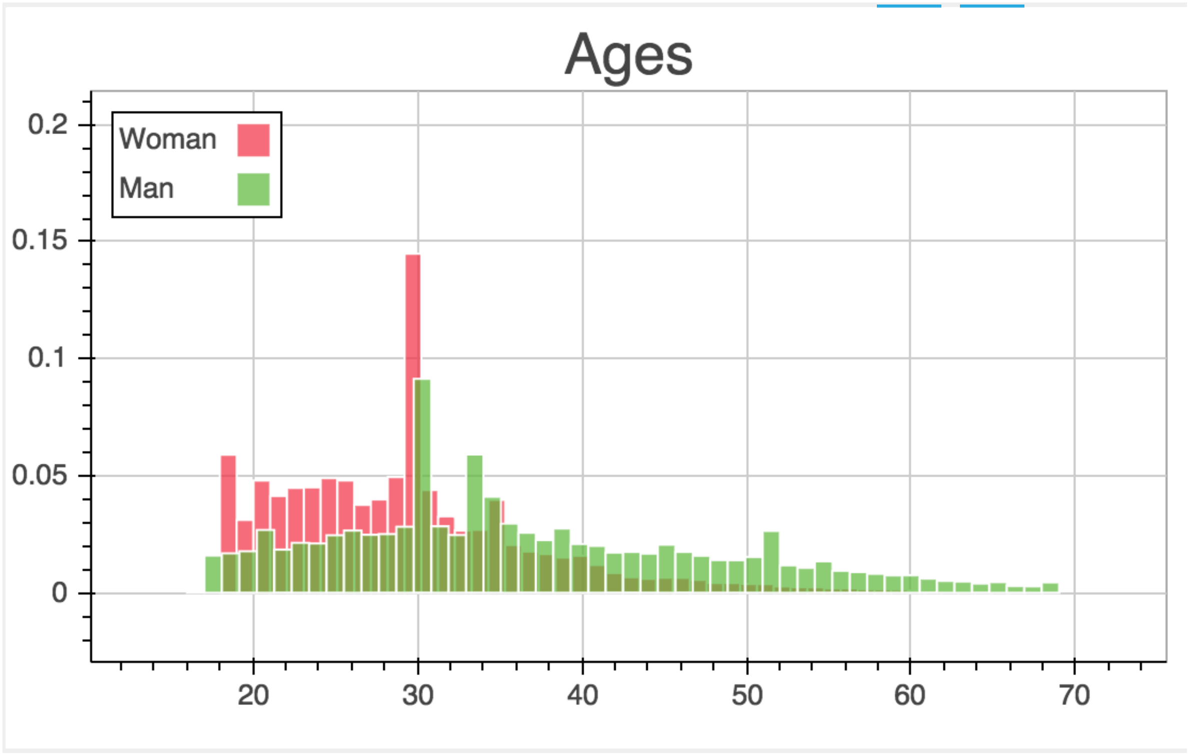
In [13]:

```
f = open("aff2.json", "w+")
def test(x):
    f.write("{" + '"age":{0}, "language":"{1}"'.format(
            int(x[0]), x[1]) + "}\n")
df.apply(test, axis=1)
f.close()
```

In [14]:

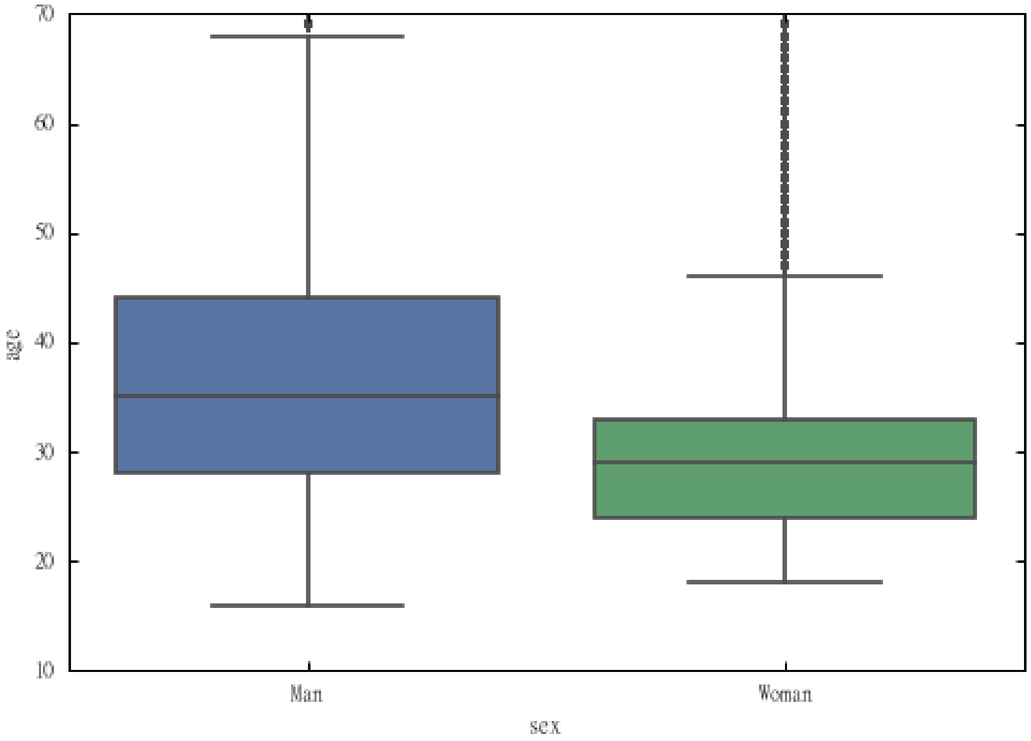
```
hm = Histogram({'Man': df[df.sex=='Man'].age,
                'Woman': df[df.sex=='Woman'].age},
               bins=50, title='Ages', legend=True)

show(hm)
```

(http://bokeh.pydata.org/)



In [15]:

```
ax = sns.boxplot(x="sex", y="age", data=df)
```



哪个星座星座最多?

In [16]:

```
df[:10]
```

Out[16]:

	age	birthdate	sex
0	41	1974-09-20	Man
1	46	1969-03-03	Man
2	61	1954-04-05	Man
3	57	1958-01-07	Man
4	44	1971-09-12	Man
5	54	1961-09-02	Man
6	51	1964-06-16	Man
7	64	1951-07-31	Man
8	51	1964-04-04	Man
9	65	1950-06-21	Man

In [17]:

```
df_xz = df[df.birthdate.notnull()]
```

In [18]:

```
df_xz.sex.value_counts()
```

Out[18]:

```
Man    811092
dtype: int64
```

In [19]:

```
df_xz = df_xz.assign(dayofyear=df_xz.birthdate.dt.dayofyear)
```

In [20]:

```
xz_labels = [u"水瓶", u"双鱼", u"白羊", u"金牛", u"双子", u"巨蟹",
             u"狮子", u"室女", u"天秤", u"天蝎", u"射手", u"摩羯"]
df_xz['XingZuo'] = pd.cut(df_xz.dayofyear.map(lambda x: x - 19
                                             if x > 19 else 366-19+x),
                          [0, 31, 62, 92, 123, 155, 186, 217, 248, 279,
                           309, 338, 367],
                          right=False, labels=xz_labels)
```

In [21]:

```
t = df_xz[:10]
t
```

Out[21]:

	age	birthdate	sex	dayofyear	XingZuo
0	41	1974-09-20	Man	263	室女
1	46	1969-03-03	Man	62	双鱼
2	61	1954-04-05	Man	95	白羊
3	57	1958-01-07	Man	7	摩羯
4	44	1971-09-12	Man	255	室女
5	54	1961-09-02	Man	245	室女
6	51	1964-06-16	Man	168	双子
7	64	1951-07-31	Man	212	狮子
8	51	1964-04-04	Man	95	白羊
9	65	1950-06-21	Man	172	双子

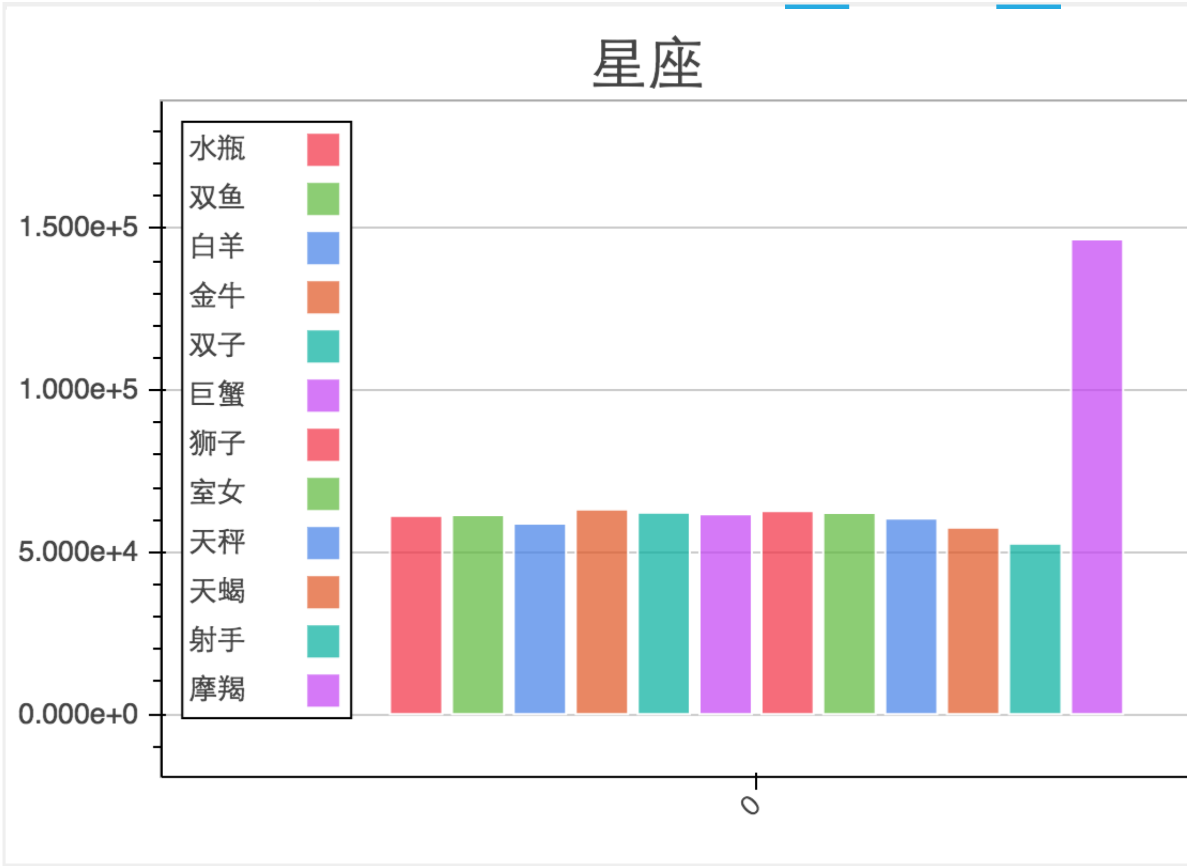
In [22]:

```
from collections import OrderedDict
data = OrderedDict()
for v,k in zip(df_xz.groupby('XingZuo').sex.count().values, xz_labels):
    data[k] = [v]
```

In [23]:

```
hm = Bar(data, title=u'星座', legend=True)

show(hm)
```

(http://bokeh.pydata.org/)

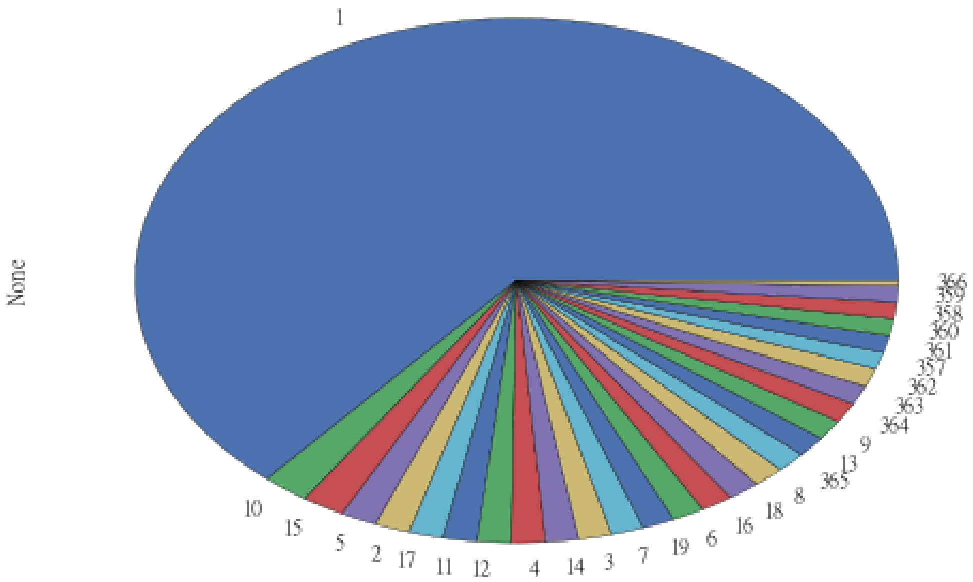


In [43]:

```
df_xz[df_xz.XingZuo==u"摩羯"].birthdate.dt.dayofyear \
            .value_counts().plot(kind='pie')
```

Out[43]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x116364b50>
```

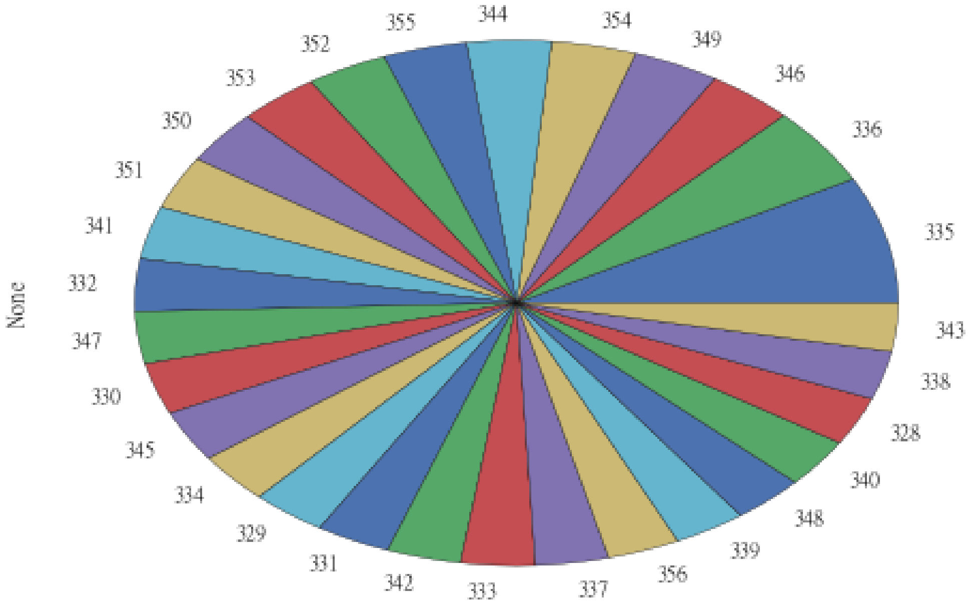


In [25]:

```
df_xz[df_xz.XingZuo==u"射手"].birthdate.dt.dayofyear \
            .value_counts().plot(kind='pie')
```

Out[25]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x120a35b90>
```

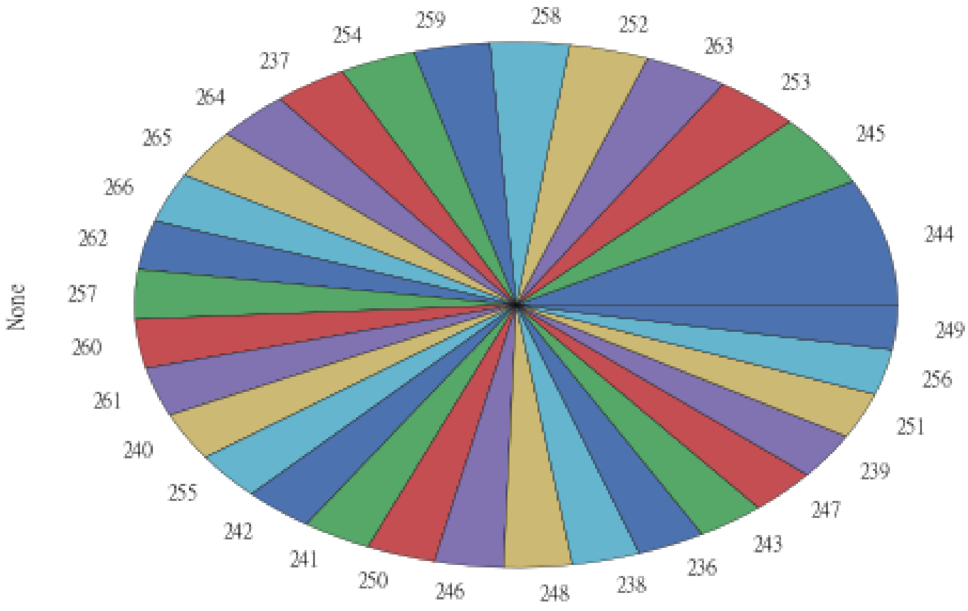


In [26]:

```
df_xz[df_xz.XingZuo==u"室女"].birthdate.dt.dayofyear \
            .value_counts().plot(kind='pie')
```

Out[26]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x10cd81c50>
```



In [27]:

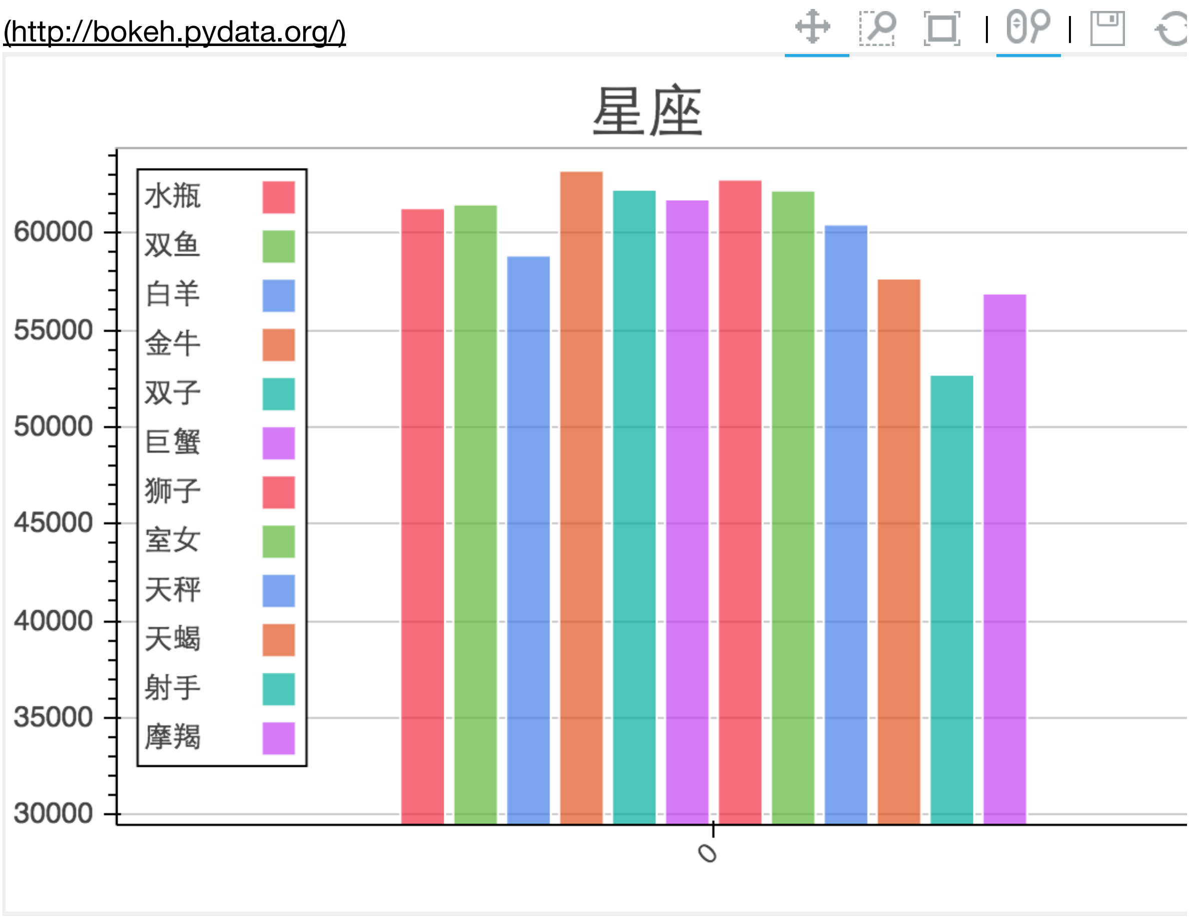
```
df_xz_mj = df_xz[df_xz.XingZuo==u"摩羯"]
pert = (df_xz_mj[(df_xz_mj.birthdate.dt.dayofyear==1)].count()
        / df_xz_mj.birthdate.count()).age
```

减去多余天数的摩羯(第一天只保留其他天数2倍)

In [28]:

```
mj_count = data[u'摩羯'][0]
mj_count = mj_count * (1-pert) * 16 / 15
data[u'摩羯'] = [mj_count]

hm = Bar(data, title=u'星座', legend=True)
show(hm)
```

(http://bokeh.pydata.org/)



语言分布

In [29]:

```
def parse_csv(x):
    df = pd.read_csv('a/aff{}.csv'.format(x),
                     parse_dates = ['birthdate'],
                     usecols=['birthdate', 'language'],
                     date_parser=parse).dropna()  #.set_index('birthdate')
    df['age'] = 2015 - df.birthdate.dt.year
    df.loc[(df.age >= 70) | (df.age < 15)] = np.nan
    return df.drop('birthdate', axis=1).dropna()
```

In [30]:

```
def parse_csv2(x):
    df = pd.read_csv('a/aff{}.csv'.format(x), usecols=['age', 'show_lang'])
    df.columns=['age', 'language']
    df.loc[(df.age >= 70) | (df.age < 15)] = np.nan
    return df.dropna()
```

In [31]:

```
df1 = parse_csv(1)
dfs = [df1] + [parse_csv2(x)
       for x in range(12, 16)]
```

In [32]:

```
df = pd.concat(dfs)
```

In [33]:

```
df[df.language=='0'] = np.nan
df = df.dropna()
df.count()
```

Out[33]:

```
age         1230500
language    1230500
dtype: int64
```

In [34]:

```
df['count'] = 1
data = df.groupby(['age', 'language']). count().reset_index()
data[:10]
```

Out[34]:

	age	language	count
0	17	english	9
1	18	chinese	395
2	18	dutch	336
3	18	english	16909
4	18	french	1225
5	18	german	907
6	18	italian	473
7	18	japanese	27
8	18	korean	12
9	18	portuguese	4328

In [35]:

```
df_lang = df.language.value_counts().to_frame()
df_lang.columns=['Count']
df_lang['Percent'] = df_lang.Count / df_lang.Count.sum() * 100
df_lang
```

Out[35]:

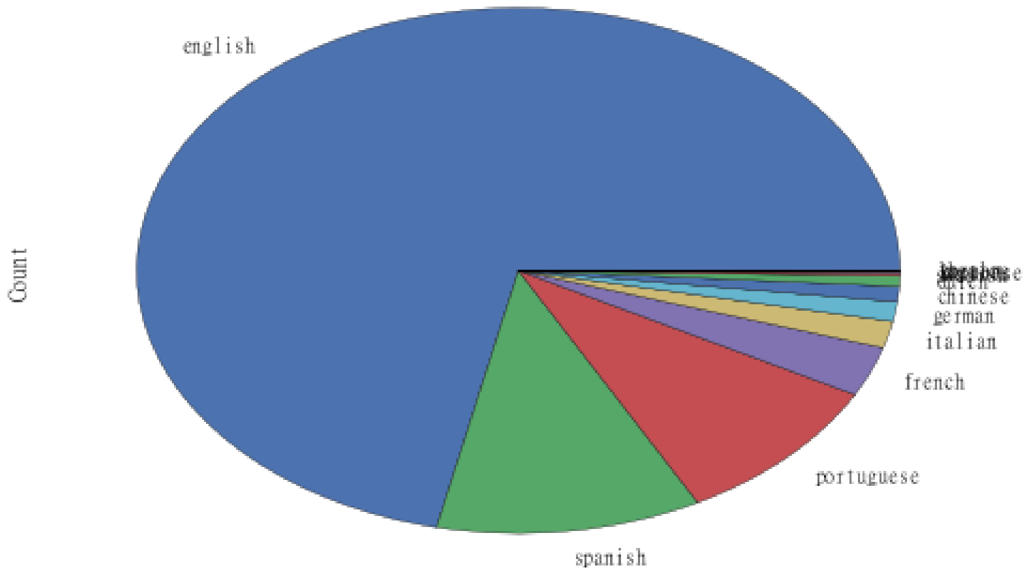
	Count	Percent
english	880519	71.557822
spanish	138174	11.229094
portuguese	115538	9.389516
french	37881	3.078505
italian	20261	1.646566
german	14886	1.209752
chinese	11528	0.936855
dutch	7725	0.627794
swedish	2267	0.184234
japanese	933	0.075823
korean	618	0.050223
gb	102	0.008289
tagalog	68	0.005526

In [36]:

```
df_lang.Count.plot(kind='pie')
```

Out[36]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x10d396350>
```

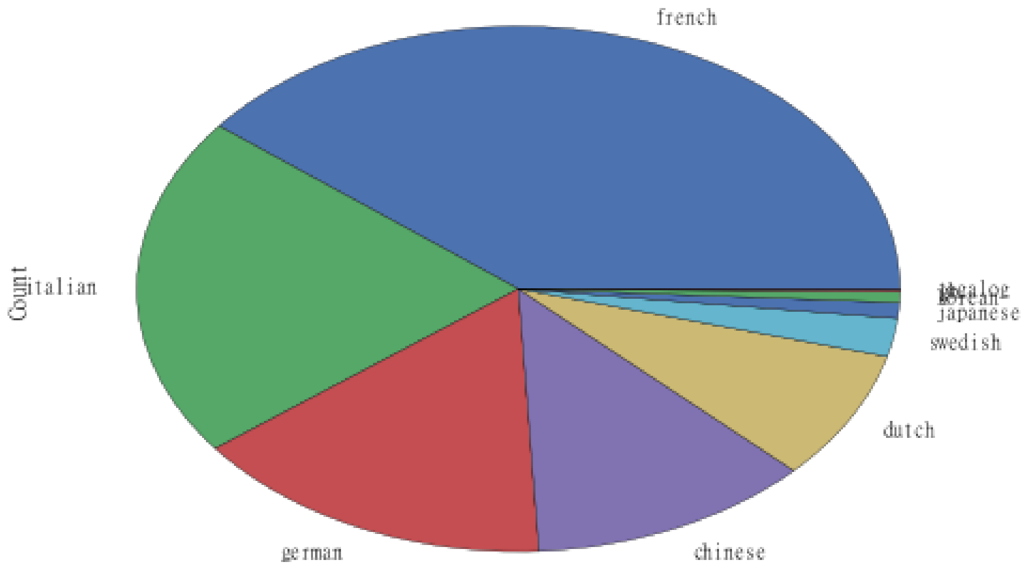


In [37]:

```
df_lang.Count[3:].plot(kind='pie')
```

Out[37]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x10d3fd6d0>
```



In [38]:

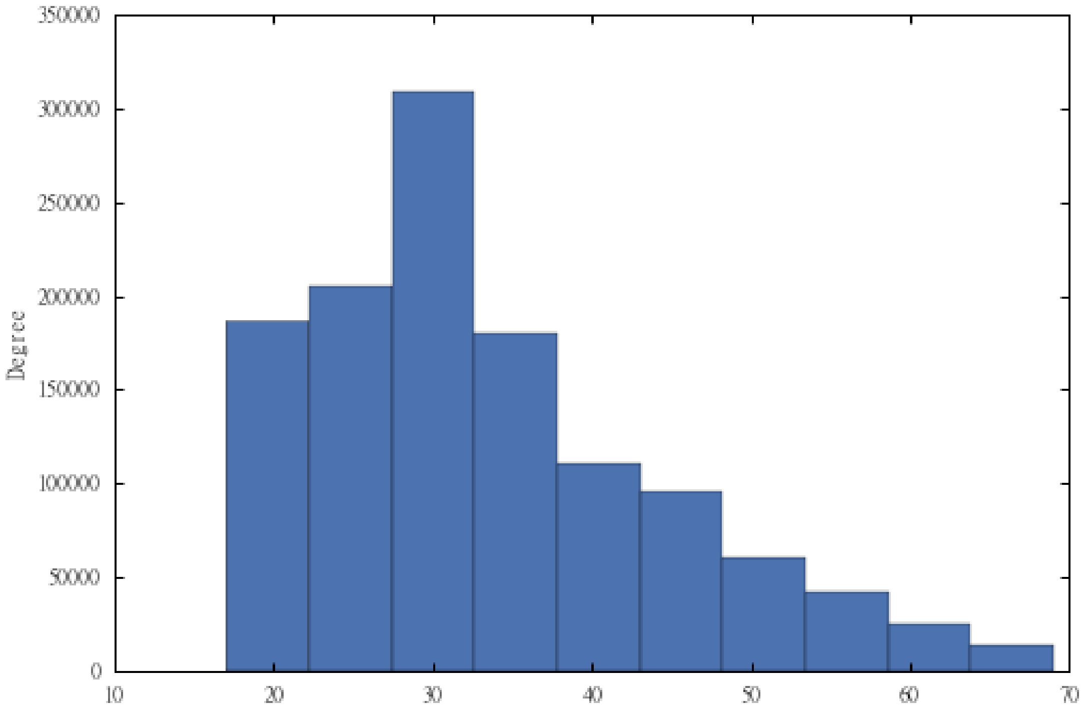
```
df_asia = df[df.language.isin(['chinese', 'japanese', 'korean'])]
df_chinese = df[df.language.isin(['chinese'])]
```

In [39]:

```
df.age.plot(kind='hist')
```

Out[39]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x11cf27d10>
```



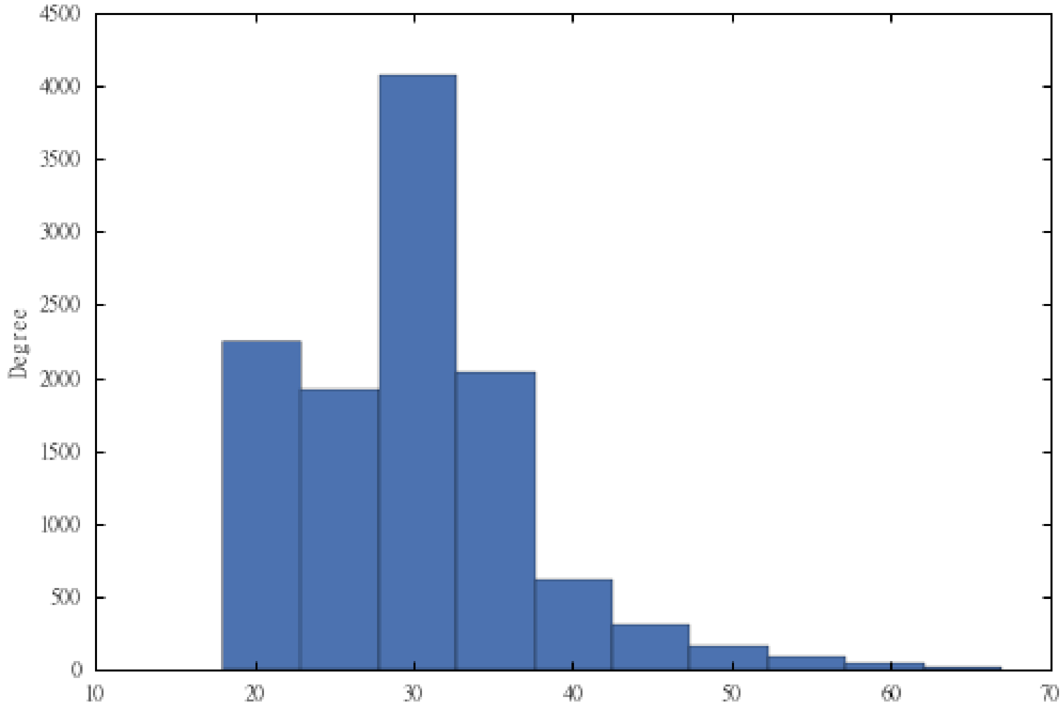
中文年龄分布

In [40]:

```
df_chinese.age.plot(kind='hist')
```

Out[40]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x119f79b90>
```



In [ ]:

```

```