nb

December 23, 2020

[138]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()
plt.rcParams['font.family'] = ['Arial Unicode MS']
```

[104]:
```
ords = pd.read_csv('data/
↪cb7dcb64e554e04f617275e422935500_be7a5fbd7e37e48cacdadd6d46d432ba_8.csv')
```

[105]:
```
ords.shape, ords.columns
```

[105]:
```
((9959, 25),
 Index(['     ', '     ', '    ', '   ', '    ', '  ', '  ',
        '  ', '  ', '  ', '  ', '  ', '  ', '  ', '  ', '  ',
        '  ', ' /  ', '  ', '  ', '  ', '  ID', '  ', '  ', '  '],
       dtype='object'))
```

[106]:
```
ords.head()
```

[106]:

0	2	3	(61)	170	130	1	-47%	
1	4	6	43	164	125	1	34%	
2	4	6	4	42	32	1	13%	
3	4	6	(27)	421	321	1	-8%	
4	2	3	550	1,803	1,376	1	40%	

0	Fiskars ,	(61)	…	40%	2	
1	GlobeWeis ,	43	…	0%	2	
2	Cardinal ,	4	…	40%	2	
3	Kleencut ,	(27)	…	40%	4	
4	KitchenAid ,	550	…	0%	3	

0	US-2018-1357144	2018/4/27	130
1	CN-2018-1973789	2018/6/15	125

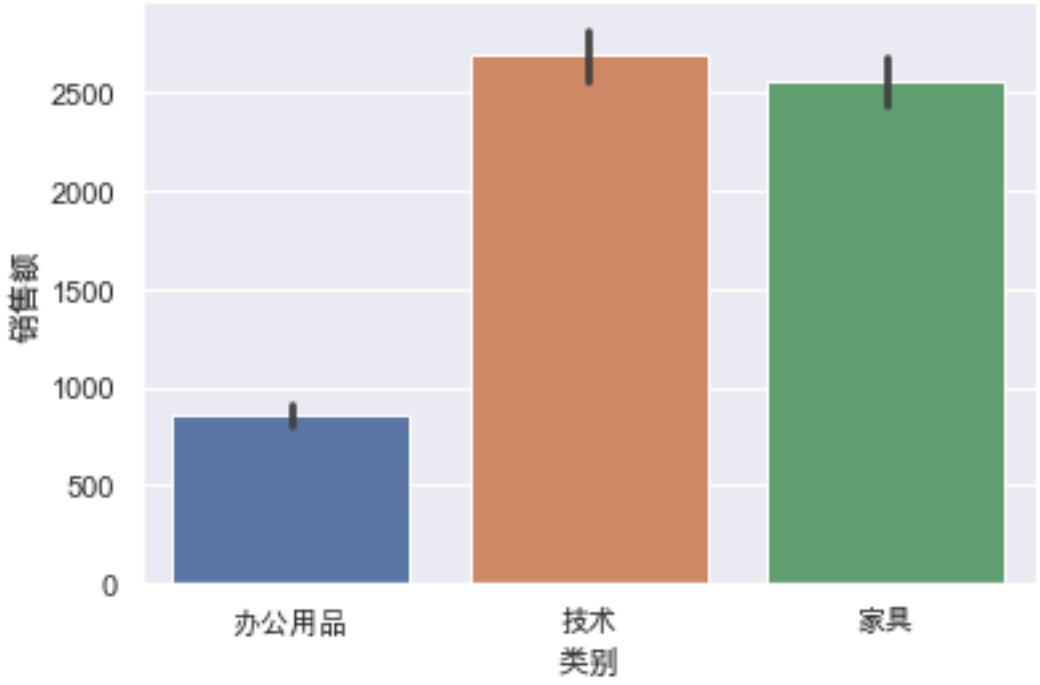
```
2  CN-2018-1973789  2018/6/15         32
3  US-2018-3017568  2018/12/9        321
4  CN-2017-2975416  2017/5/31       1,376

[5 rows x 25 columns]
```

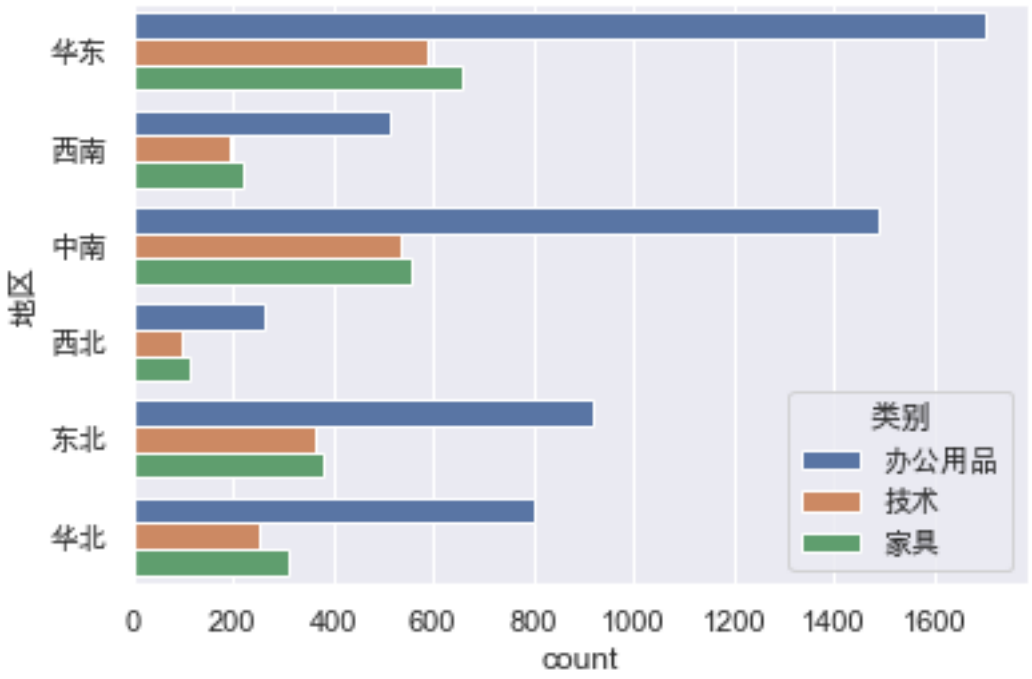
[107]:
```
def remove_par_and_unit_from_int_str(s: str):
    s = s.replace(',', '')
    if s[0] == '(':
        return -int(s[2:-1])
    else:
        return int(s[1:])
```

[108]:
```
cols = ['     ', '    ', '   ', '    ',' ', '  ']
for col in cols:
    ords[col] = ords[col].apply(remove_par_and_unit_from_int_str)
```

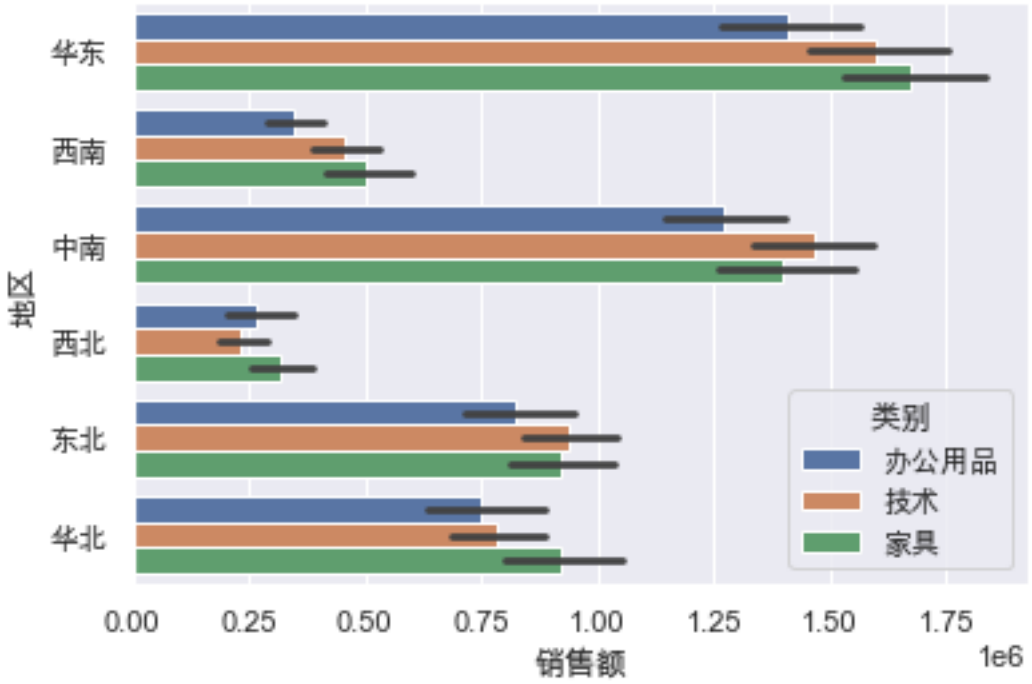
[109]:
```
sns.barplot(x=' ', y='  ', data=ords, estimator=np.mean);
```



[110]:
```
sns.countplot(y=' ',hue=' ', data=ords);
```



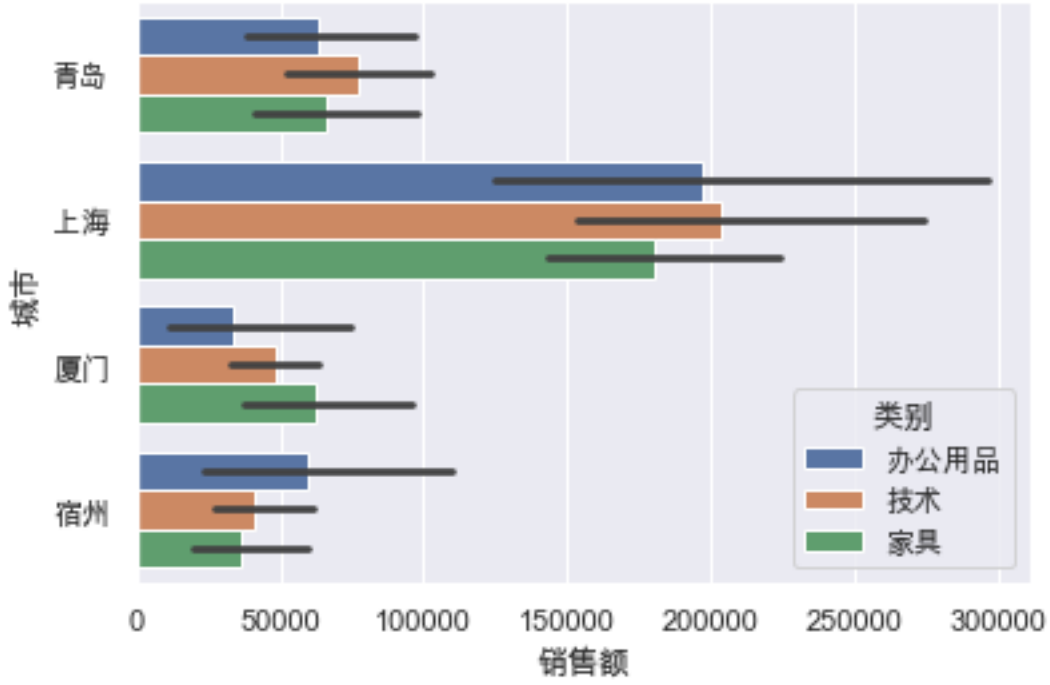
[111]:
```
sns.barplot(x='  ', y=' ',hue=' ', data=ords, estimator=np.sum);
```



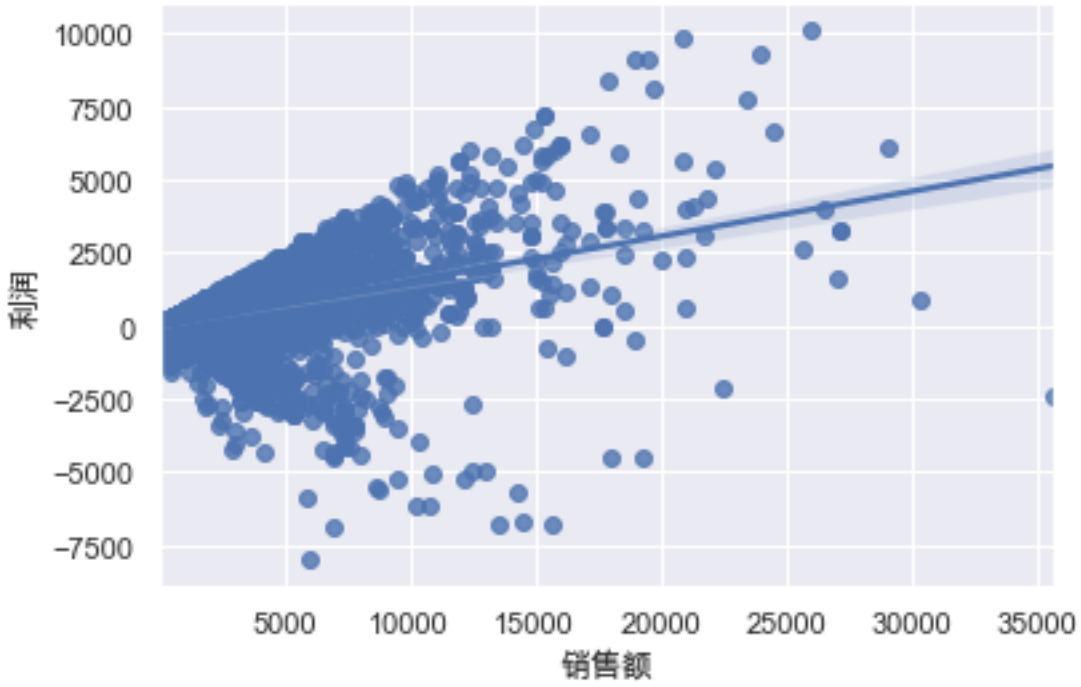
[112]:
```
hd_ords = ords[ords[' ']==' ']
top4_cities = hd_ords.groupby(' ')['  '].sum().sort_values(ascending=False)[:4].
 ↪index
top4_cities
```

[112]:
```
Index([' ', ' ', ' ', ' '], dtype='object', name=' ')
```

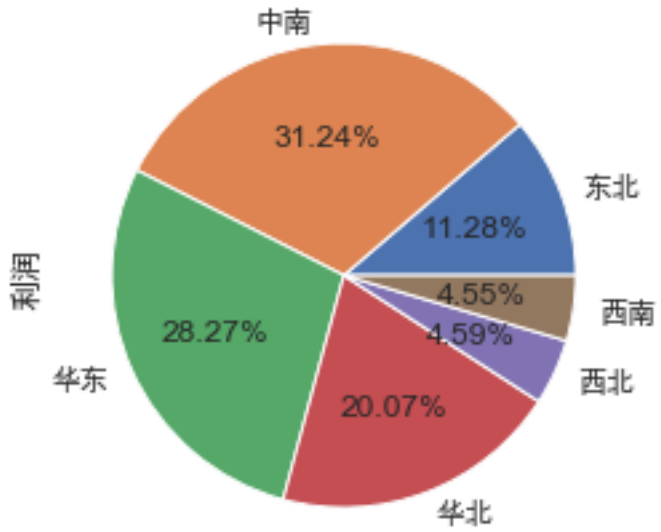
[113]:
```
sns.barplot(x='  ', y=' ',hue=' ', data=hd_ords[hd_ords[' '].
 ↪isin(list(top4_cities))], estimator=sum);
```



[114]:
```
sns.regplot(x='  ', y=' ', data=ords);
```

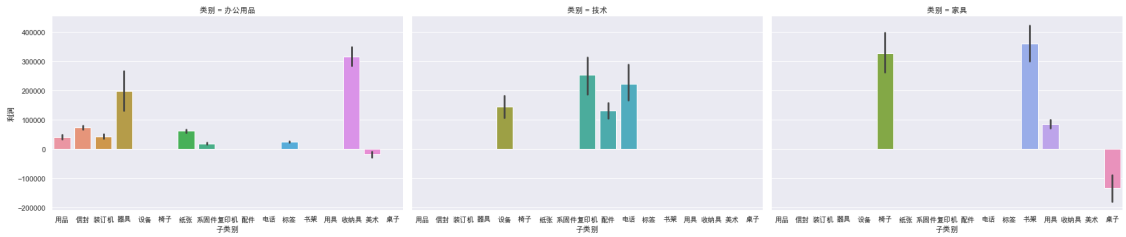


[115]:
```
ords.groupby(' ')[' '].sum().plot.pie(autopct='%1.2f%%');
```



[179]:
```
clazz = ords.groupby([' ', '  ']).sum()
clazz.index = clazz.index.to_flat_index()
```

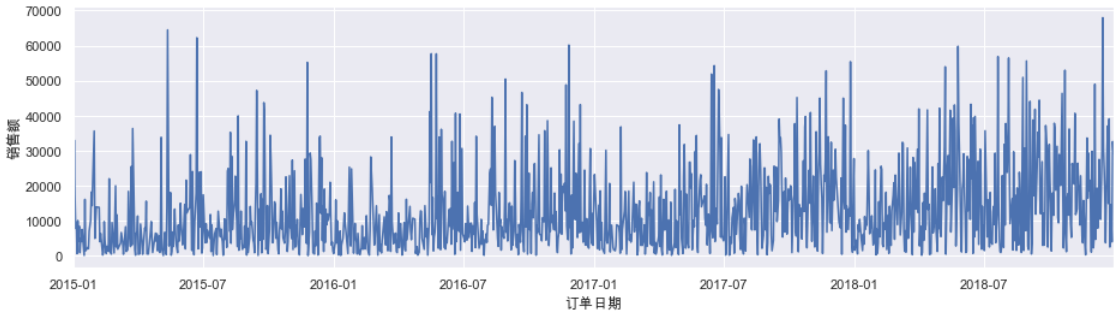
[184]:
```
sns.catplot(x="  ", y=" ",
            col=" ", aspect=1.6,
            kind="bar", estimator=np.sum, data=ords);
```



[128]:
```
total_by_date = ords.groupby('   ').sum()
total_by_date.index = pd.DatetimeIndex(total_by_date.index)
```

[151]:
```
plt.gca().margins(x=0)
plt.gcf().canvas.draw()
tl = plt.gca().get_xticklabels()
# maxsize = max([t.get_window_extent().width for t in tl])
maxsize = 30
m = 0.2  # inch margin
s = maxsize / plt.gcf().dpi * 30 + 2 * m
margin = m / plt.gcf().get_size_inches()[0]

plt.gcf().subplots_adjust(left=margin, right=1. - margin)
plt.gcf().set_size_inches(s, plt.gcf().get_size_inches()[1])
sns.lineplot(data=total_by_date, x='   ', y='  ');
```



[ ]:
```

```