

# GenbaDataView

July 2, 2019

```
In [8]: import pandas as pd
import time
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
sns.set(font='AppleMyungjo')
import matplotlib as mpl
```

```
In [9]: train_genba_path = "./Data/train/train_genba.tsv"
train_goto_path = "./Data/train/train_goto.tsv"
test_genba_path = "./Data/test/test_genba.tsv"
test_goto_path = "./Data/test/test_goto.tsv"

train_genba = pd.read_csv(train_genba_path, sep="\t")
train_goto = pd.read_csv(train_goto_path, sep="\t")
test_genba = pd.read_csv(test_genba_path, sep="\t")
test_goto = pd.read_csv(test_goto_path, sep="\t")
```

```
In [10]: train_genba["ave_pr"] = pd.Series(
    np.random.randn(len(train_genba)), index=train_genba.index
)
train_goto = train_goto.groupby('pj_no').mean()
train_goto.describe()
```

```
Out[10]:
```

	keiyaku_pr	tc_mseki	tt_mseki	fukuin	magutchi	\
count	2.781000e+03	2781.000000	2781.000000	2740.000000	2740.000000	
mean	2.789168e+07	139.414250	98.357139	5.396227	9.198886	
std	6.653501e+06	51.782006	17.621183	2.080451	4.013845	
min	6.750000e+06	39.160000	0.000000	1.900000	0.000000	
25%	2.330000e+07	106.532000	96.000000	4.000000	6.719167	
50%	2.715000e+07	128.455000	102.260000	4.723810	8.800000	
75%	3.175000e+07	158.020000	105.990000	6.000000	10.850000	
max	6.048000e+07	643.990000	132.490000	25.100000	48.300000	

	niwasaki	garage
count	2781.000000	2781.000000
mean	4.082410	1.788458
std	18.463374	0.714755

min	0.000000	0.000000
25%	2.000000	1.000000
50%	3.000000	2.000000
75%	5.000000	2.000000
max	967.000000	9.000000

```
In [11]: # add keiyaku price
start = time.time()
ave_pr_col_no = 140
for index,row in train_genba.iterrows():
    train_goto_row = train_goto.iloc[index]
    train_genba.iloc[index,ave_pr_col_no] = train_goto_row["keiyaku_pr"]
end = time.time()
print("excute time: {}".format(end-start))
```

excute time: 1.6823759078979492

```
In [12]: # data in number
train_genba_number = train_genba.select_dtypes(exclude=["object"])
# for category in train_genba_number.columns:
#     if train_genba[category].isnull().sum()>0:
#         train_genba[category].fillna(0, inplace=True)

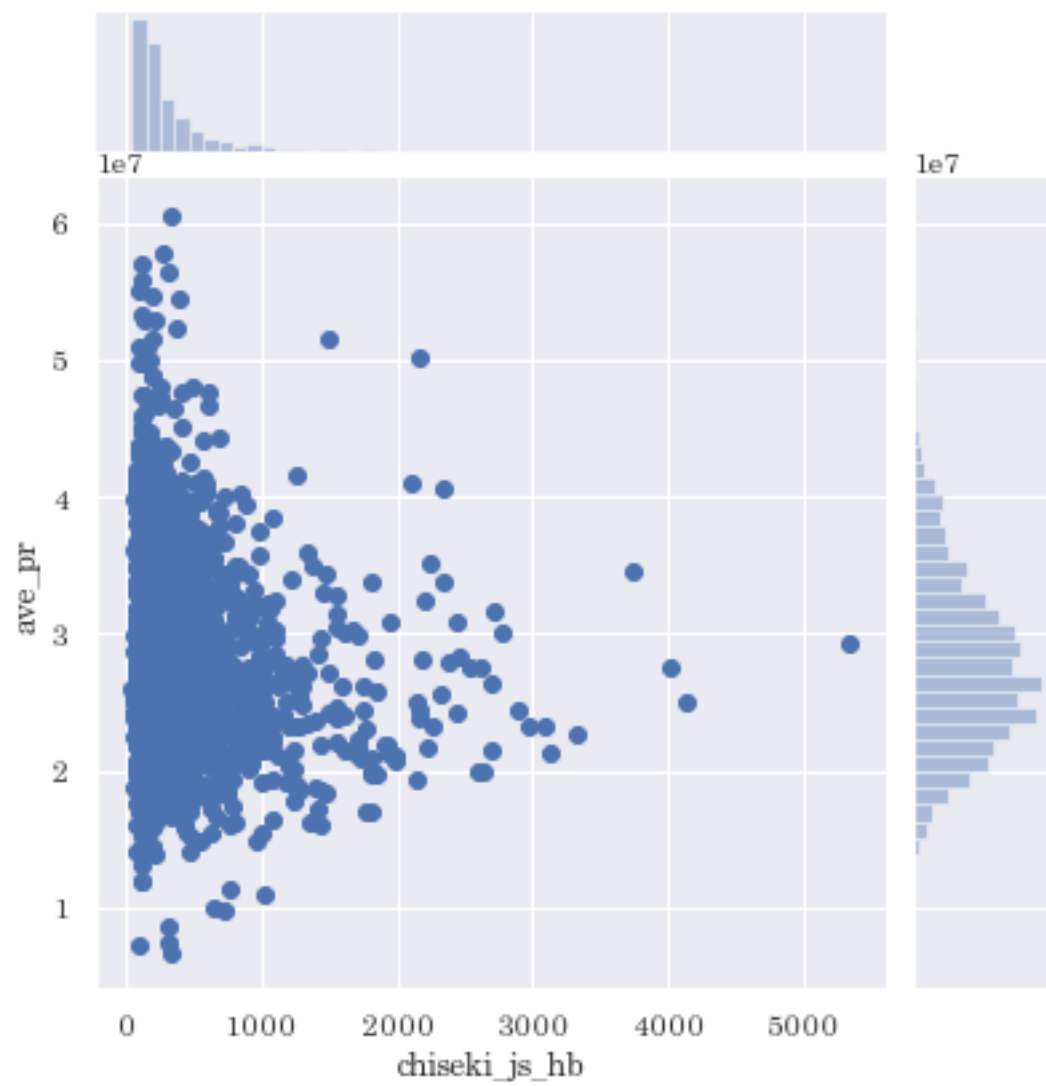
# # train_genba.isnull().any()
```

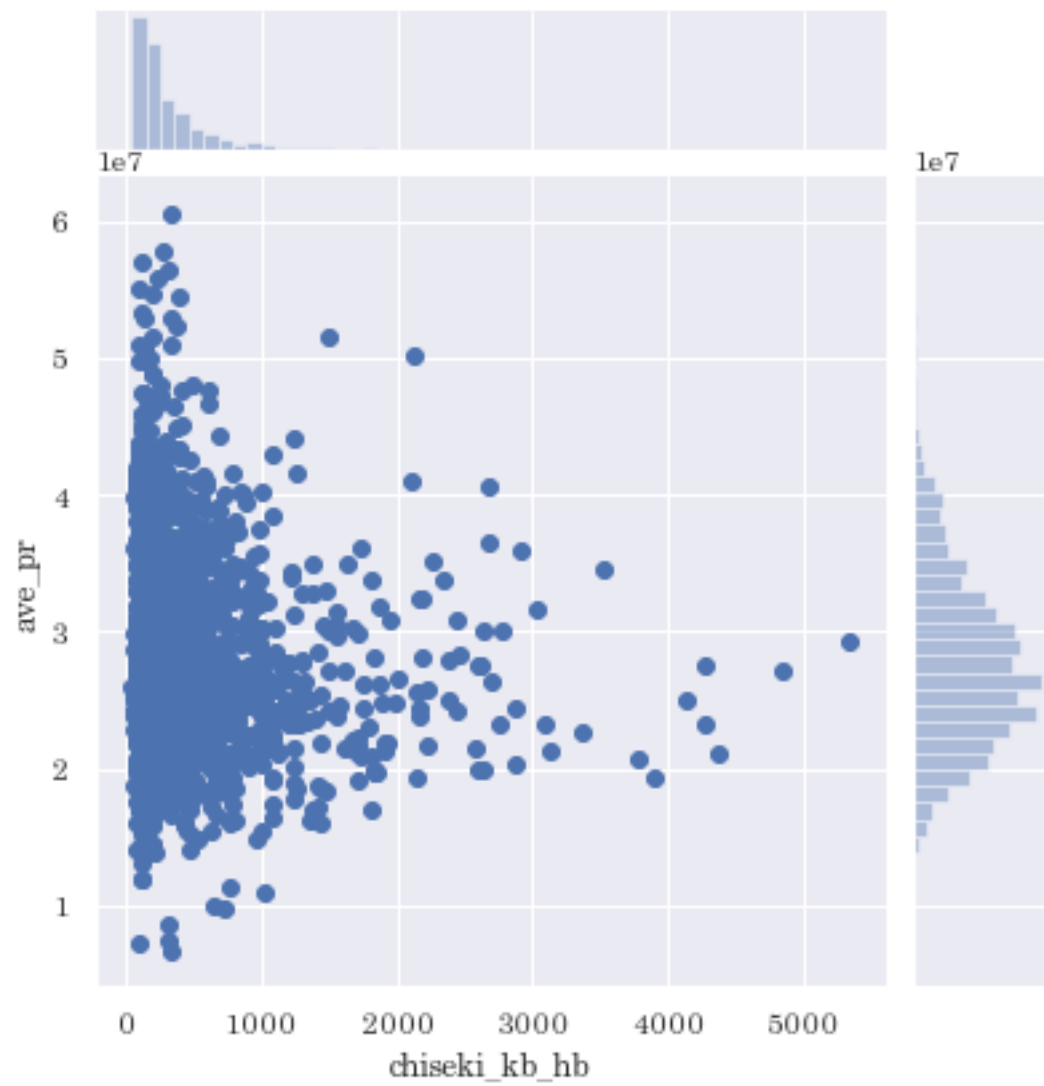
```
In [13]: # define feature to plot
features = ['chiseki_js_hb', 'chiseki_kb_hb', 'yoto1', 'yoto2', 'road1_sb', 'road1_fi', 'ro
            'kaoku_um', 'kaoku_hb', 'gk_sho_kyori', 'gk_chu_kyori', 'toshikuiki1', 'toshikuil
            'sho_conv', 'sho_super', 'sho_shoten', 'sho_market', 'shu_jutaku', 'shu_park', 'sl
            'shu_sogi', 'shu_zoki', 'shu_kokyo', 'shu_highway', 'shu_kaido', 'shu_line_ari',
# len(features)    # 46
```

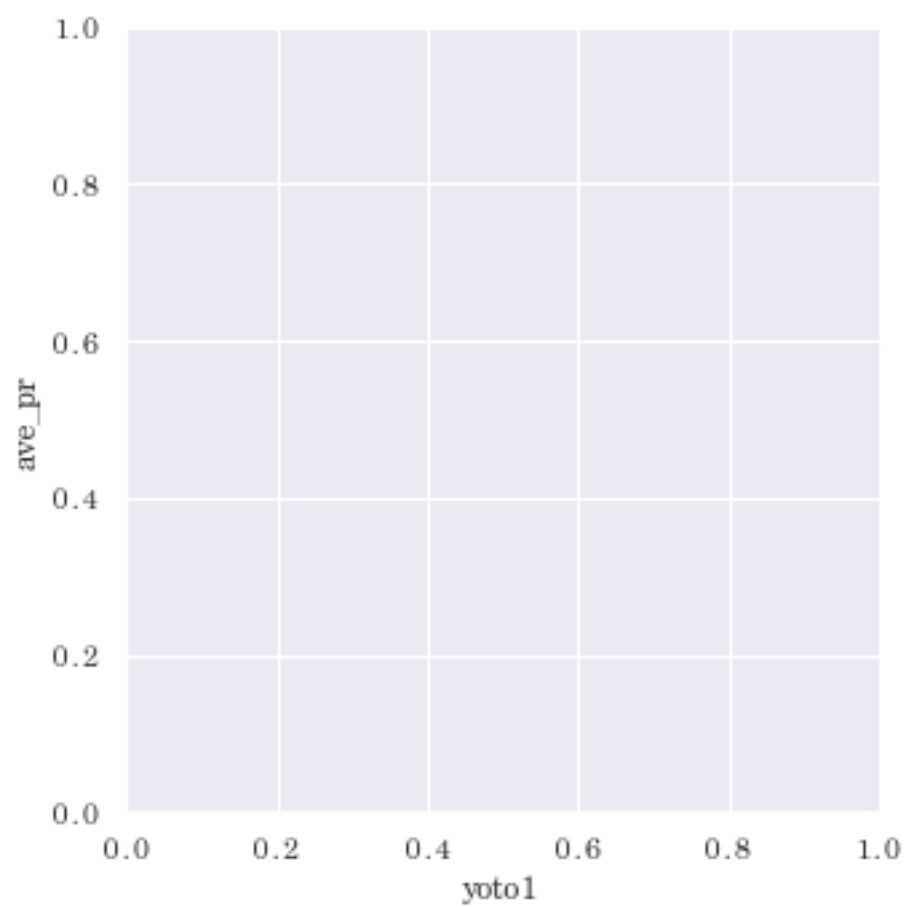
```
In [14]: fig1 = plt.figure(figsize=(40,50))
fig2 = plt.figure(figsize=(13,8))
a = 1
b = 0
for feature in features:
    if feature in train_genba_number.columns:
        sns.jointplot(x=feature, y="ave_pr", data=train_genba)
    else:
        axs = fig1.add_subplot(10,4,a)
        sns.catplot(x=feature, y="ave_pr", data=train_genba,ax = axs)
        a+=1
```

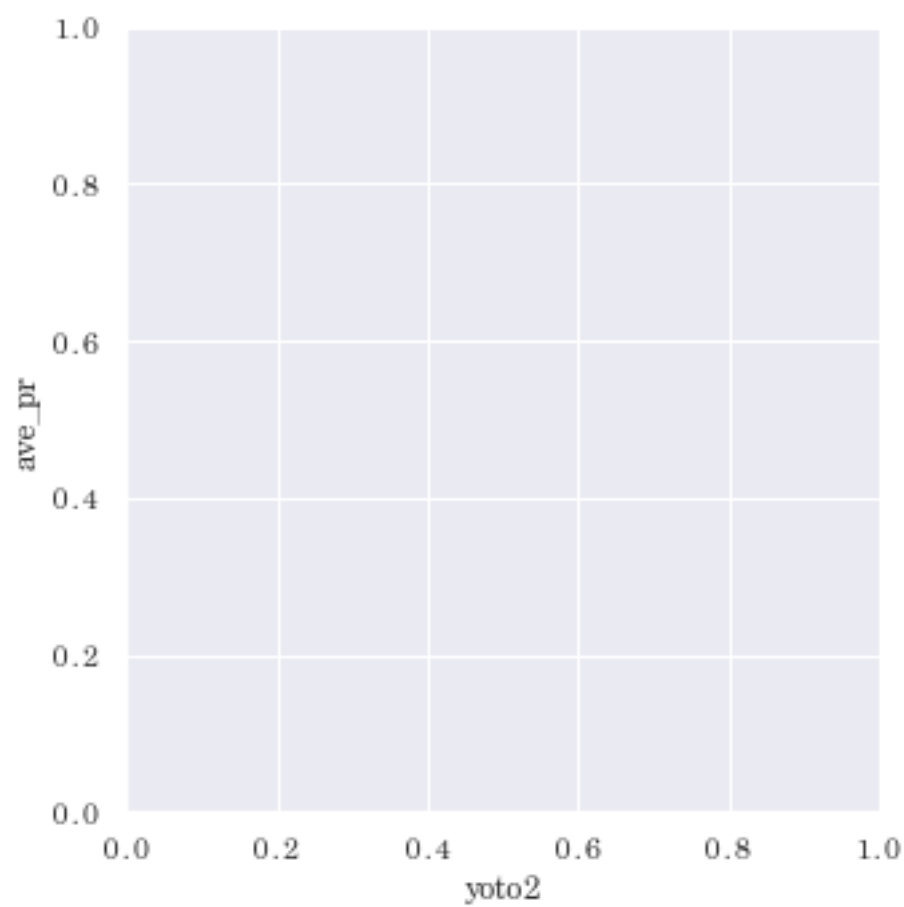


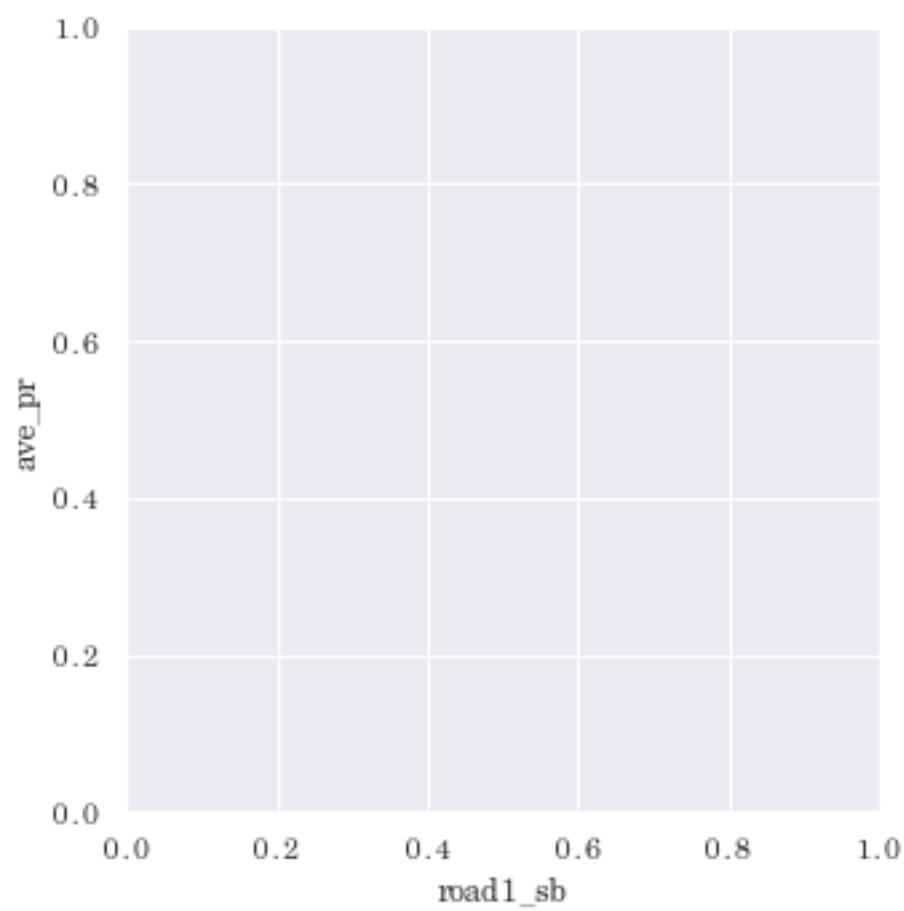
<Figure size 936x576 with 0 Axes>



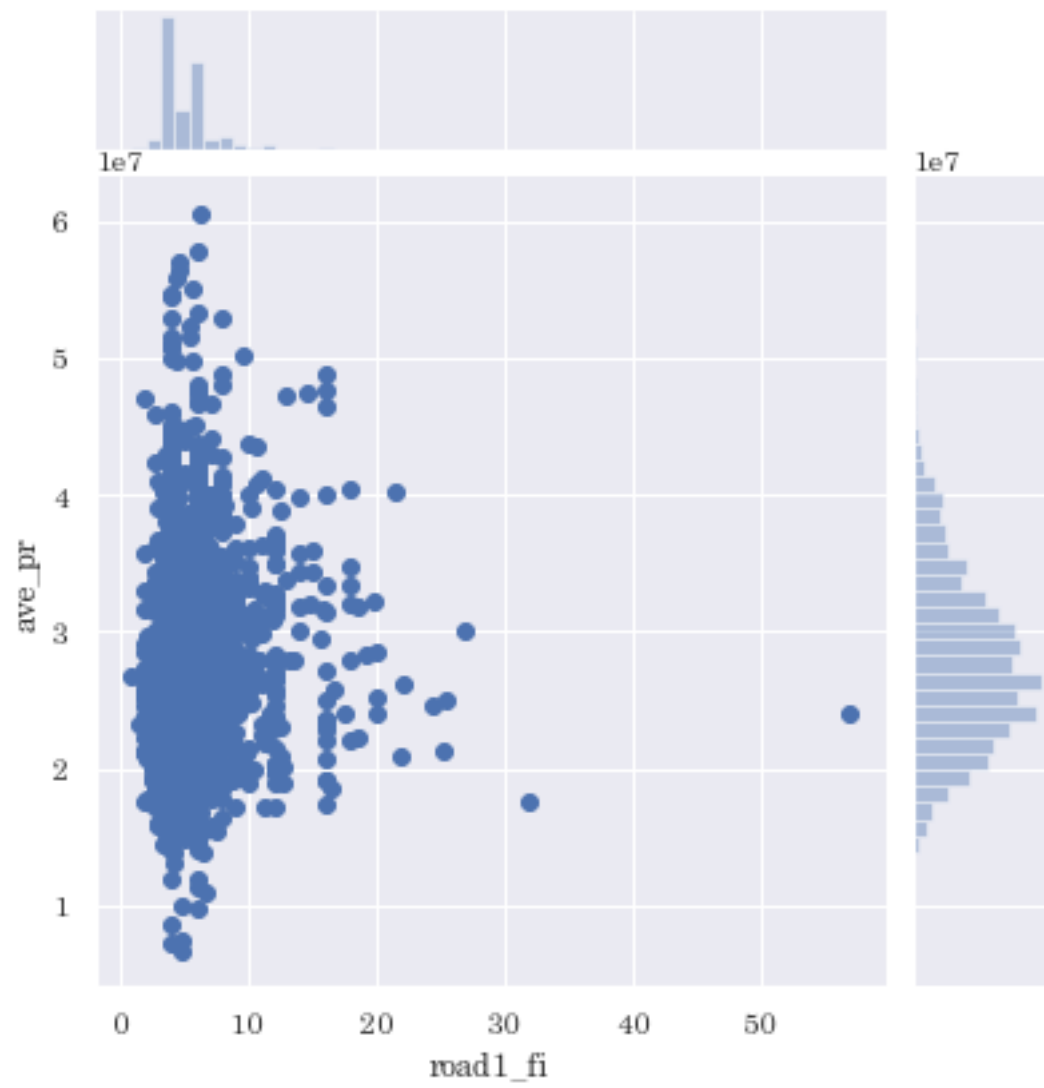


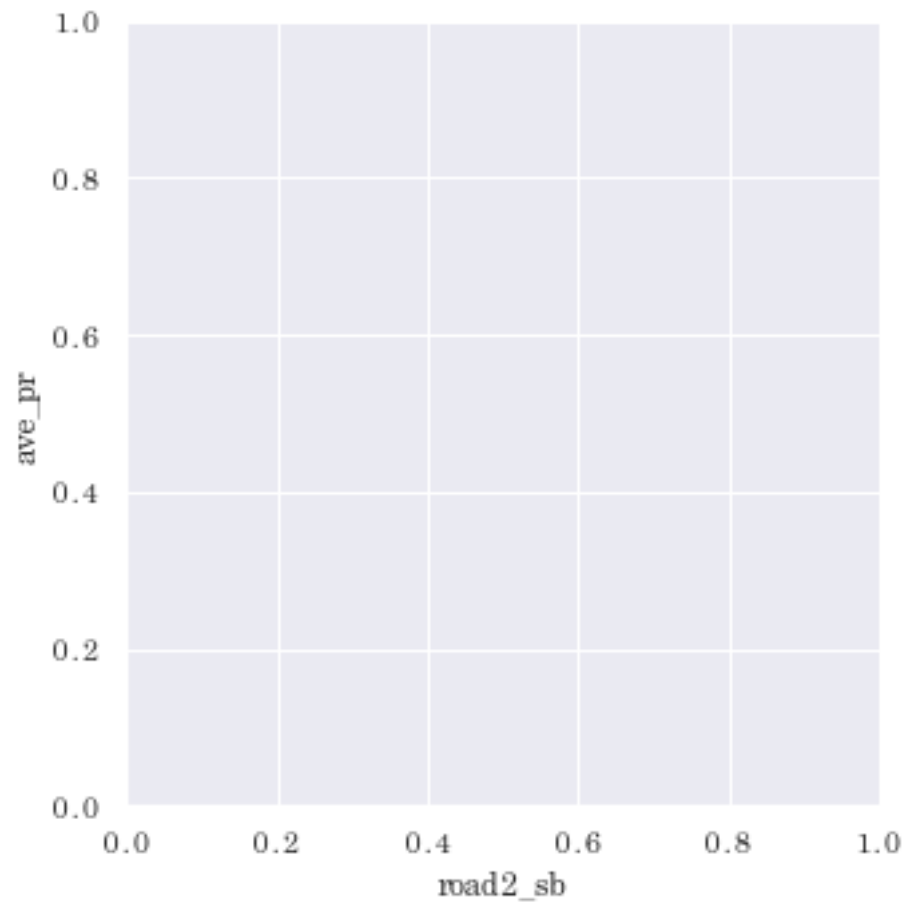


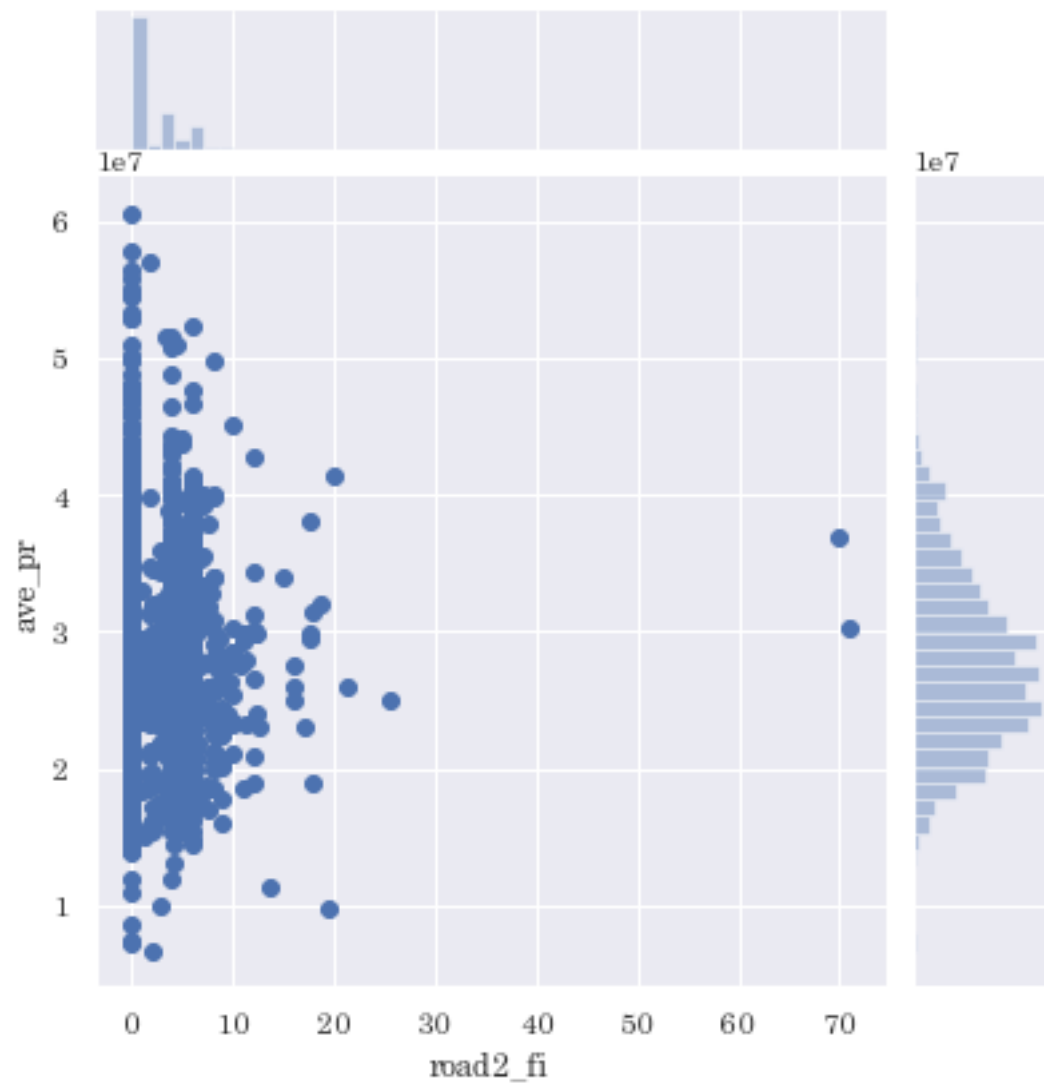


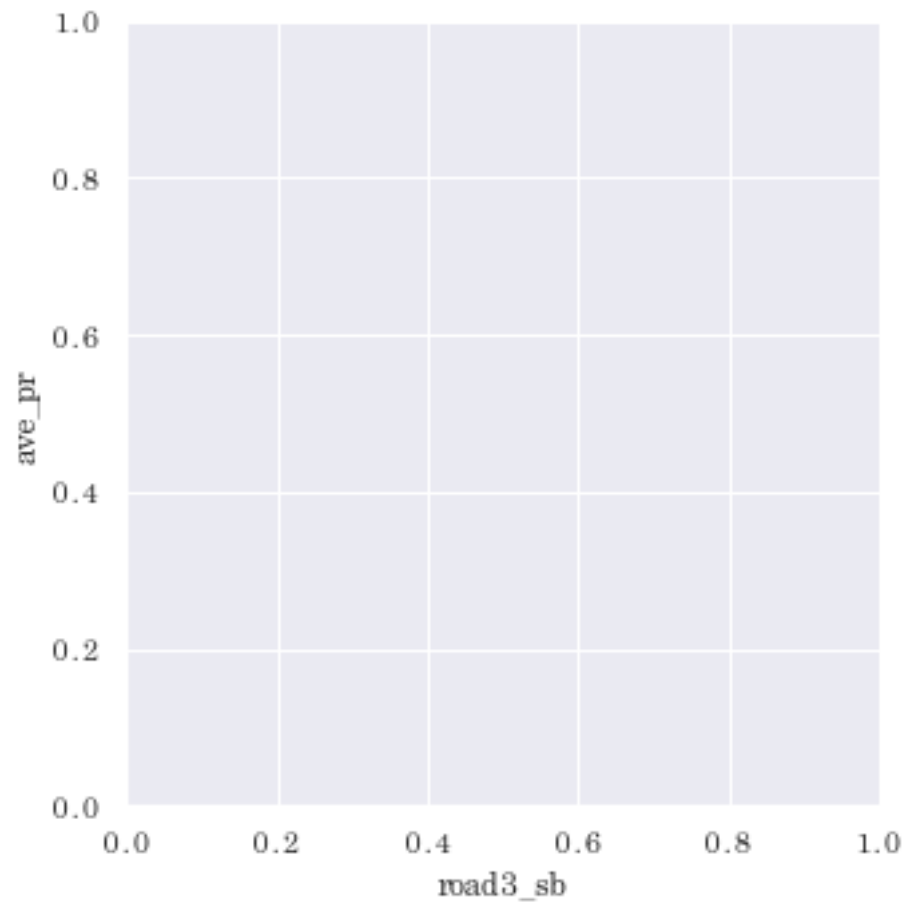


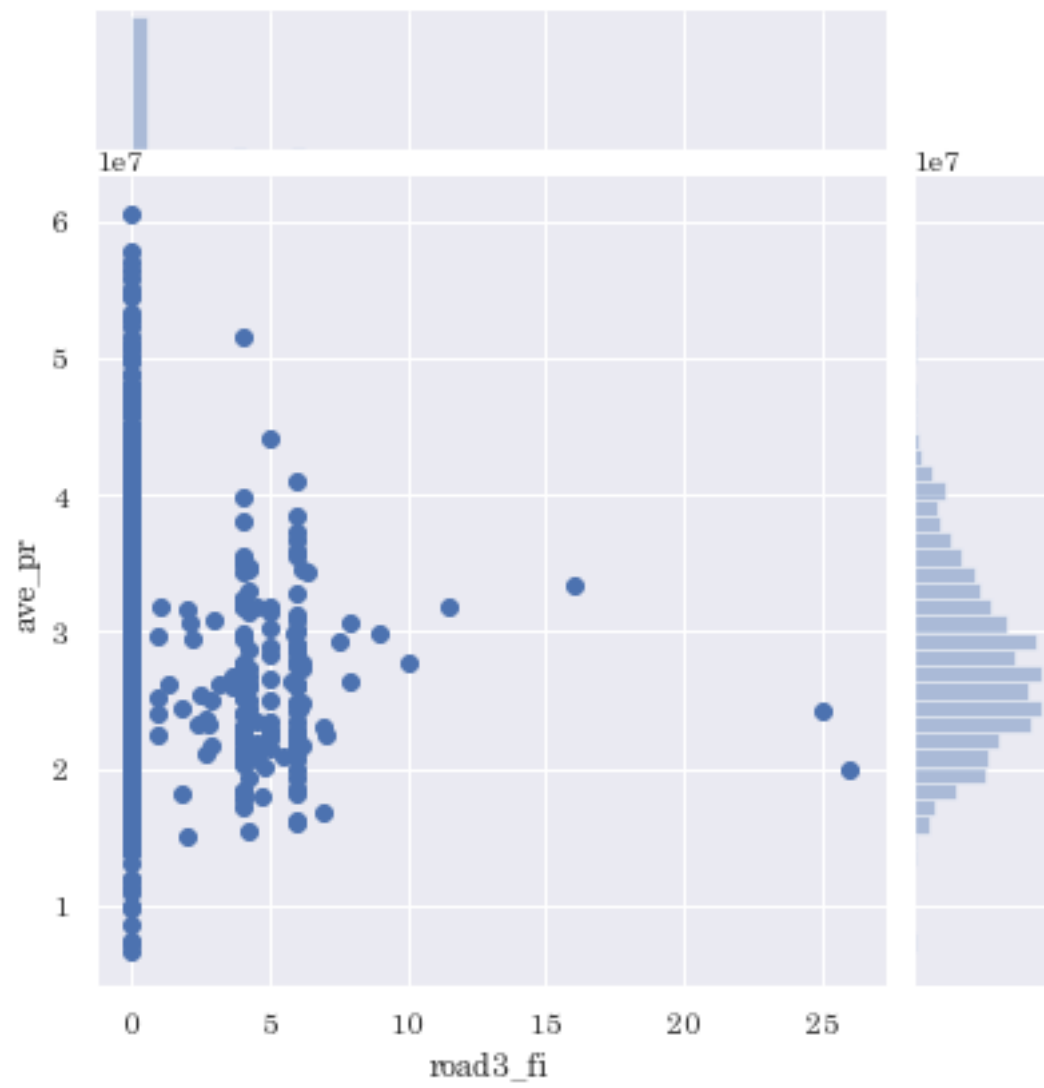


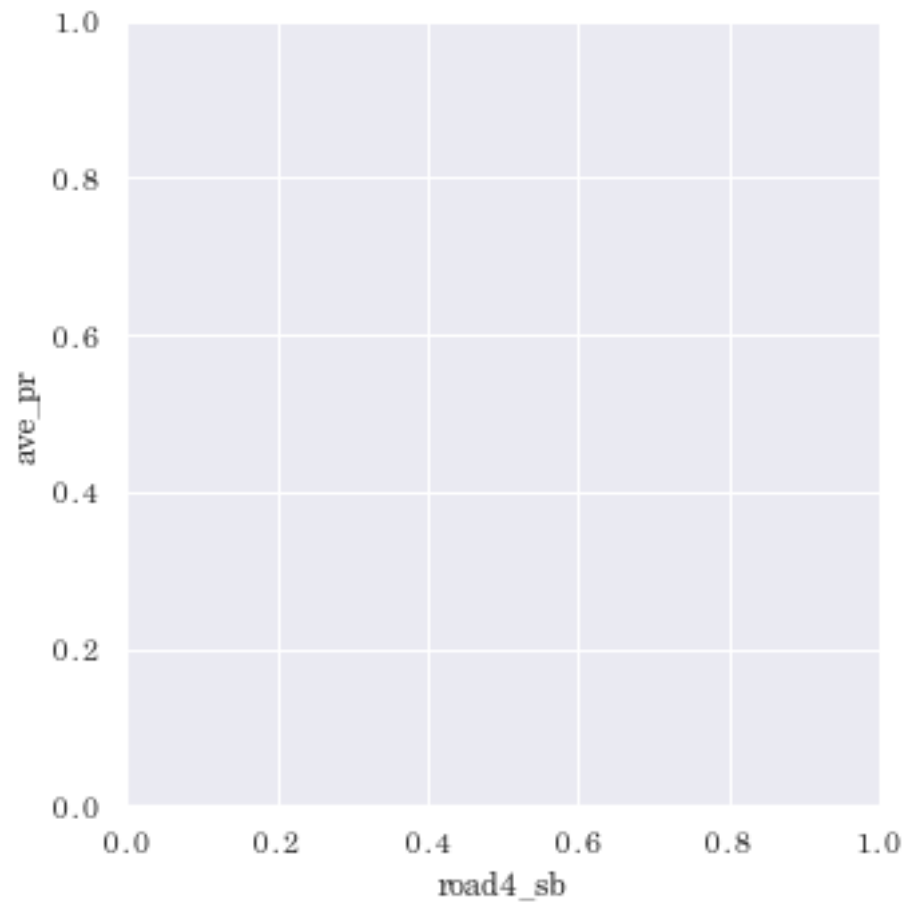


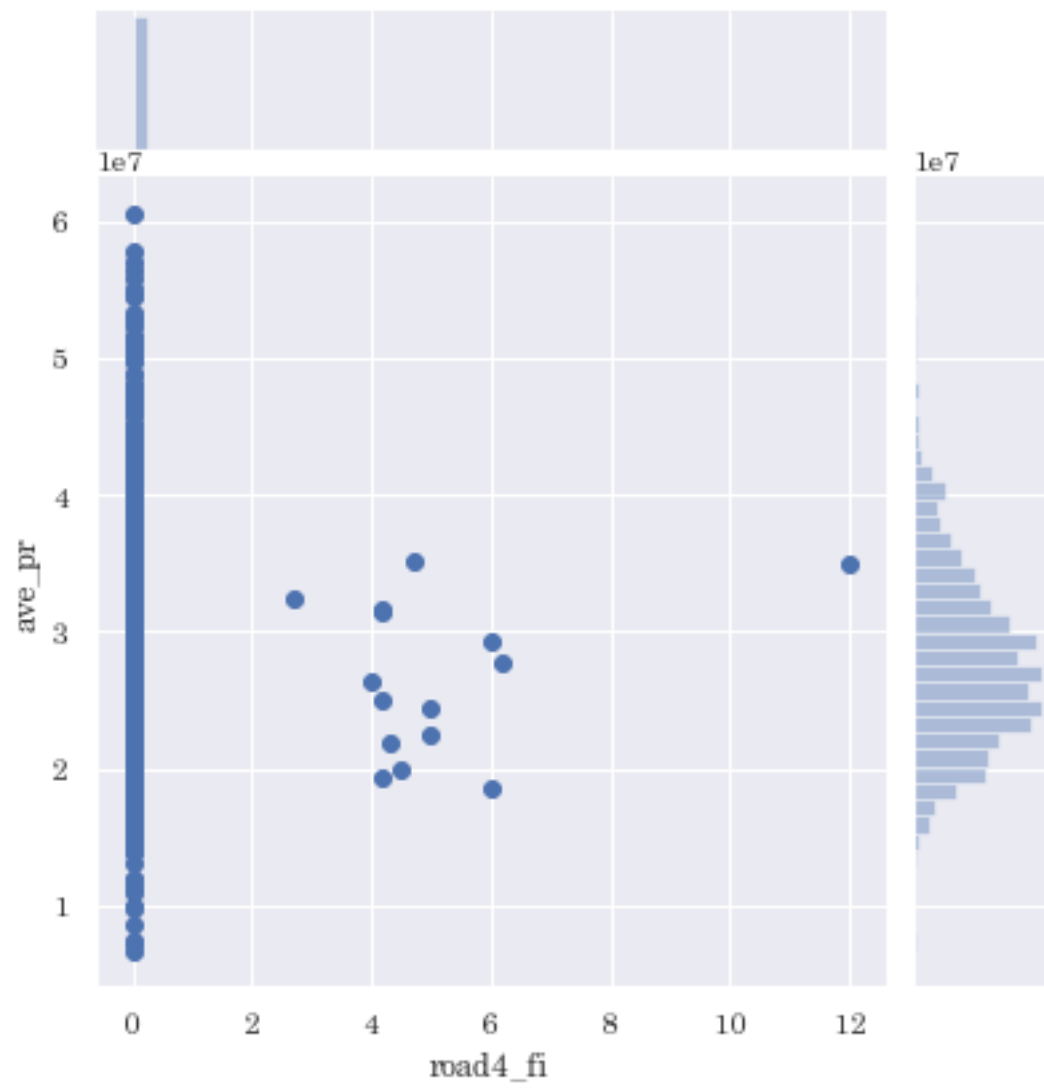


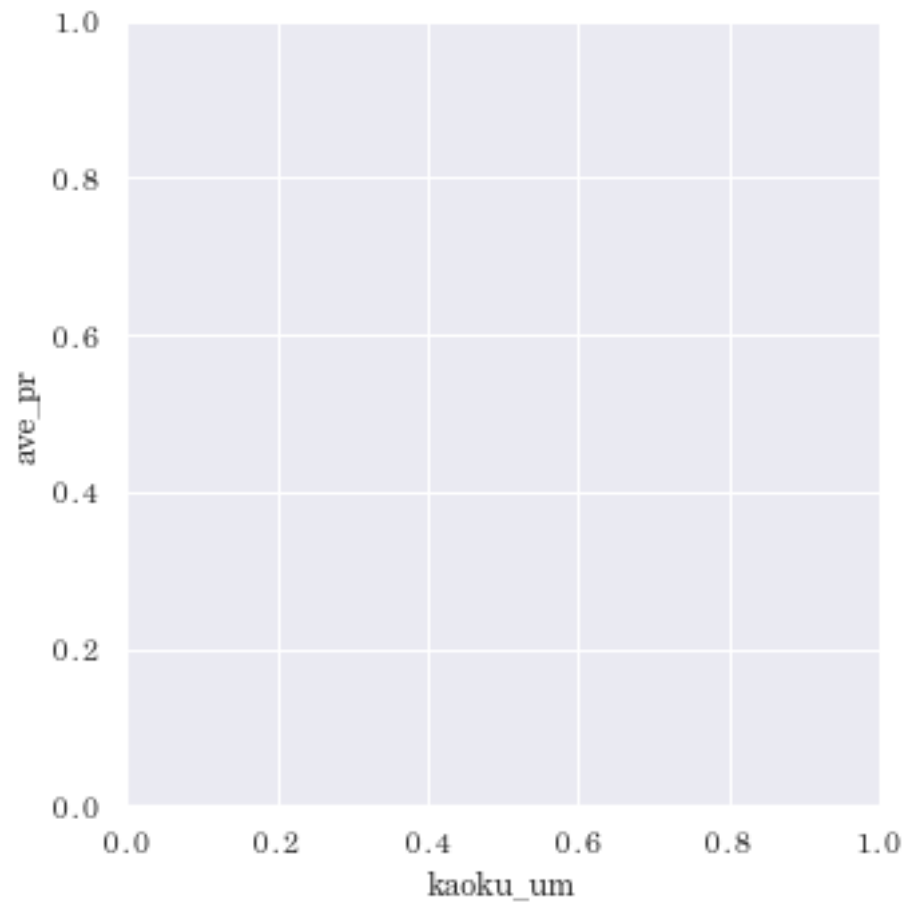




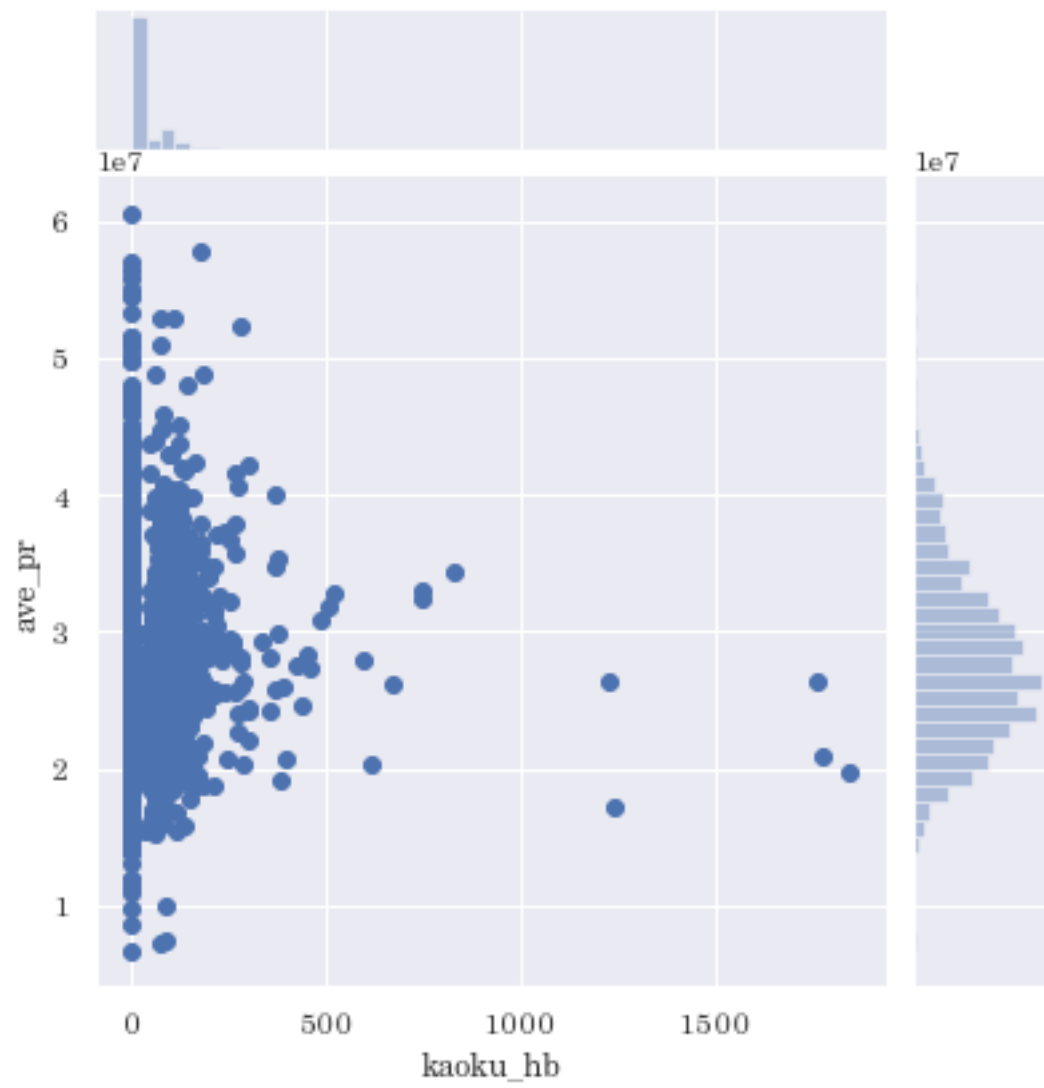


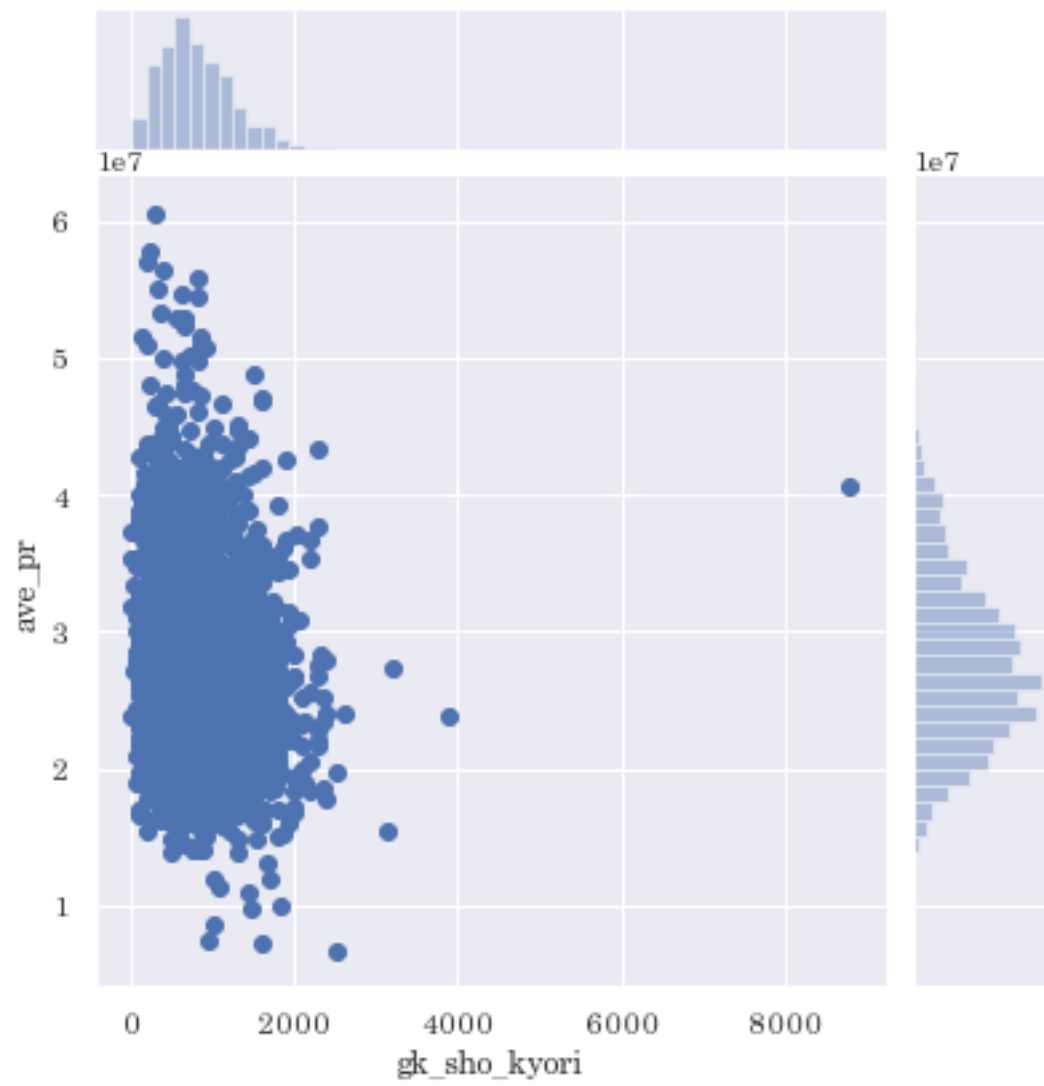


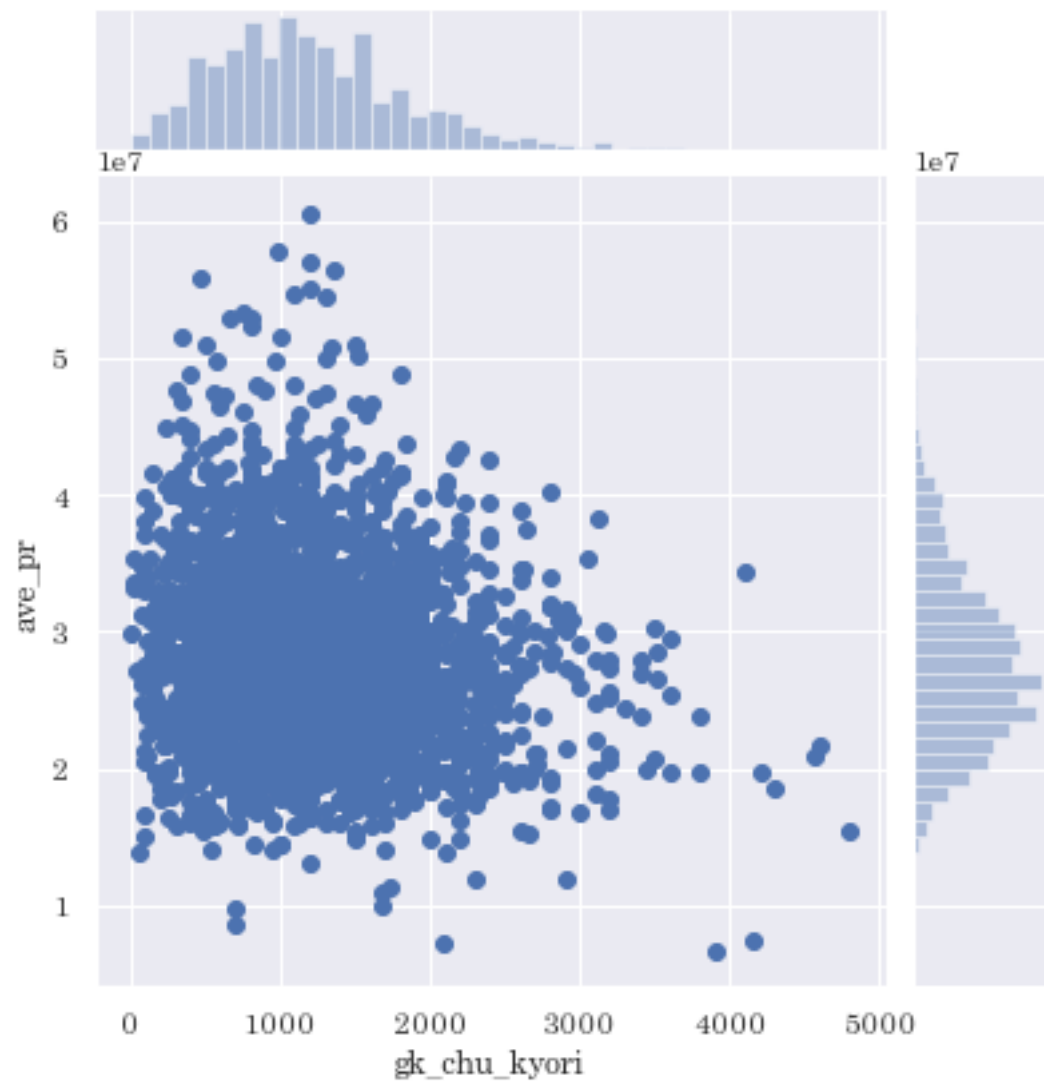


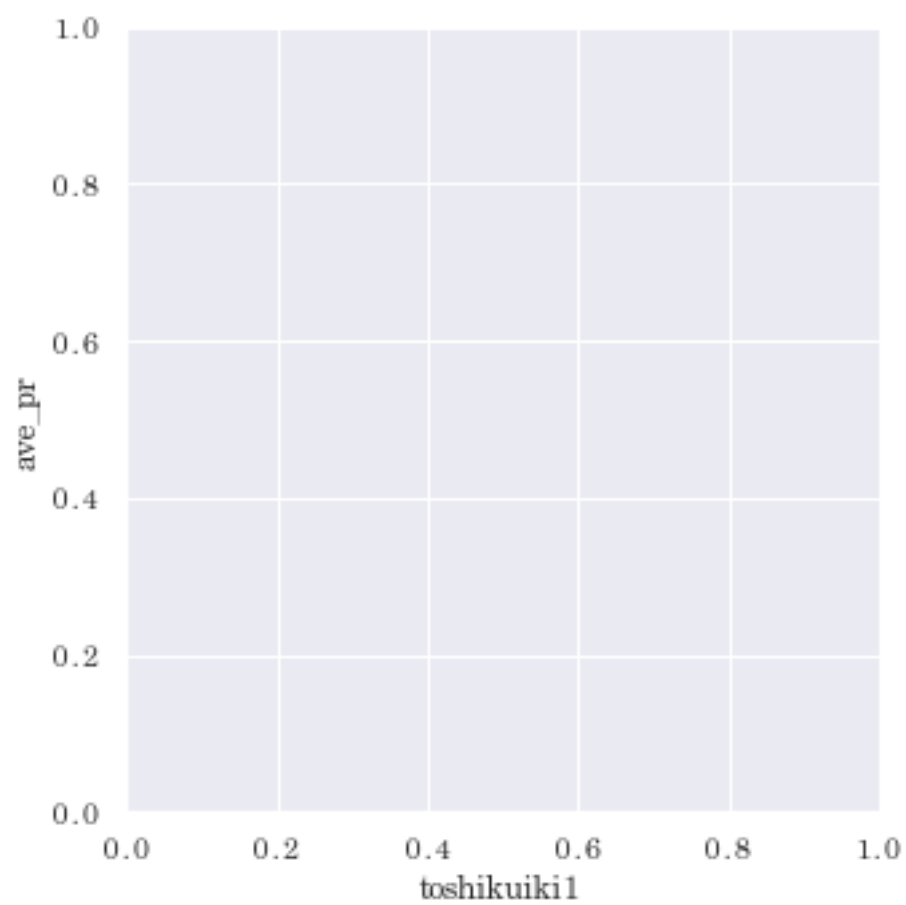


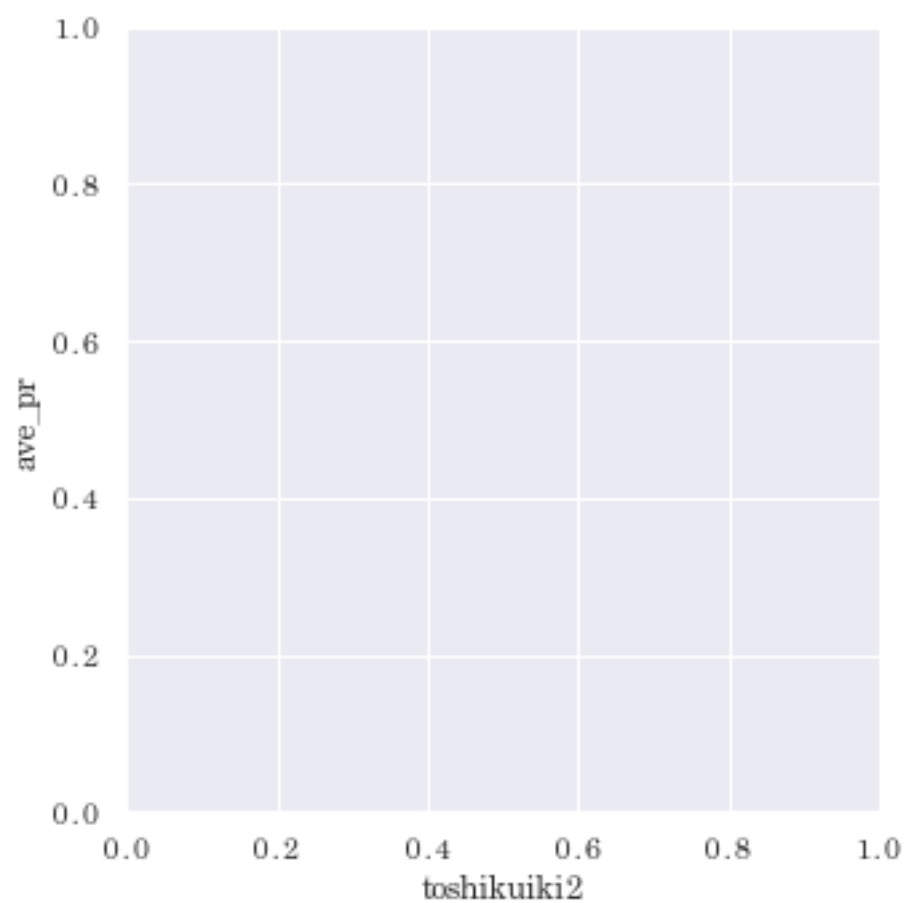


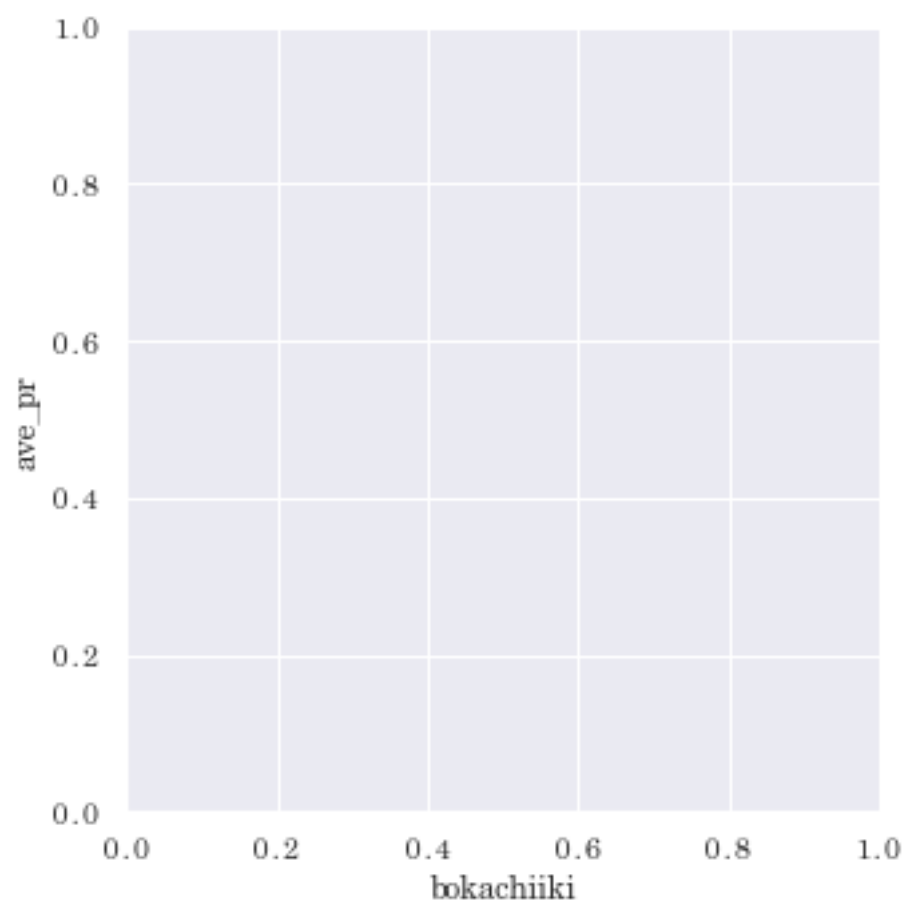


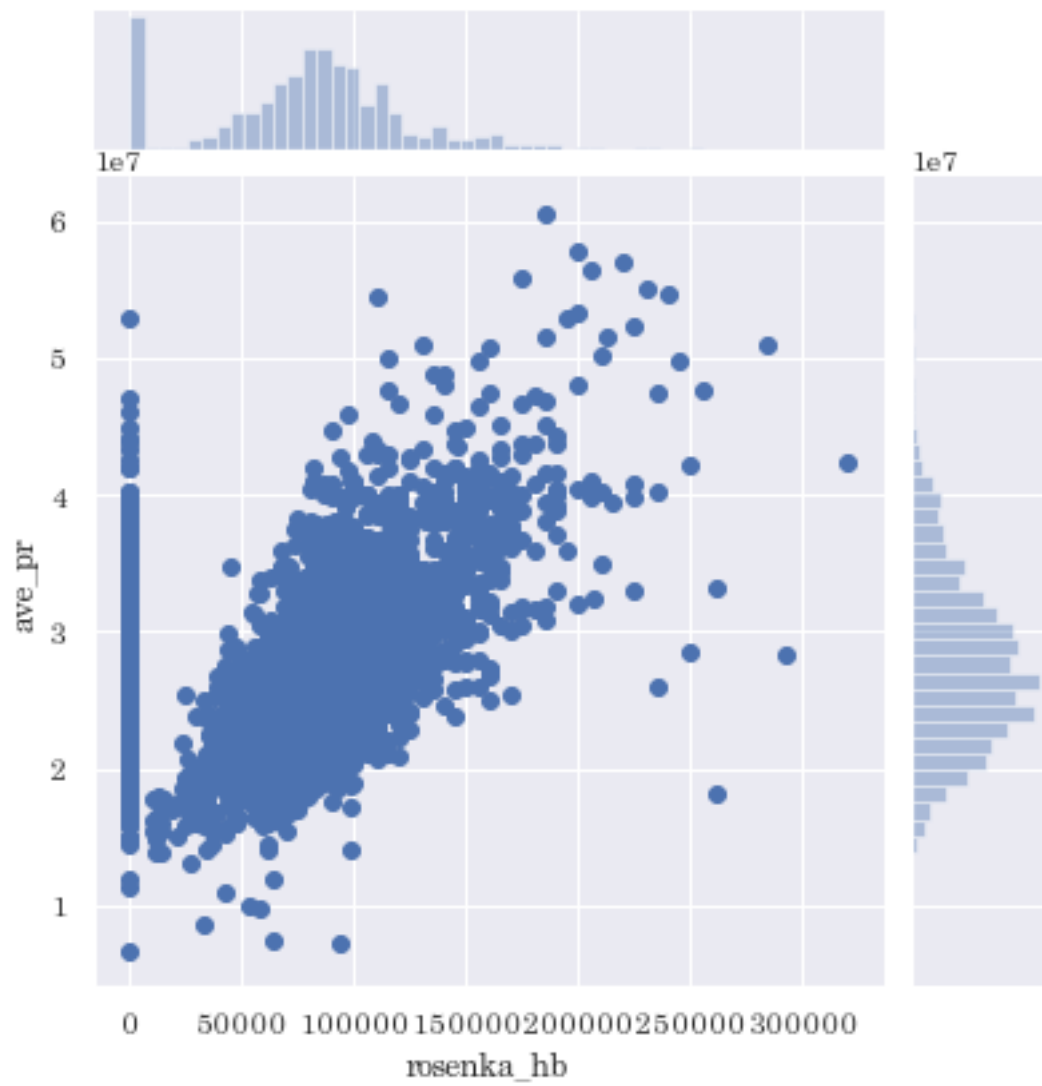


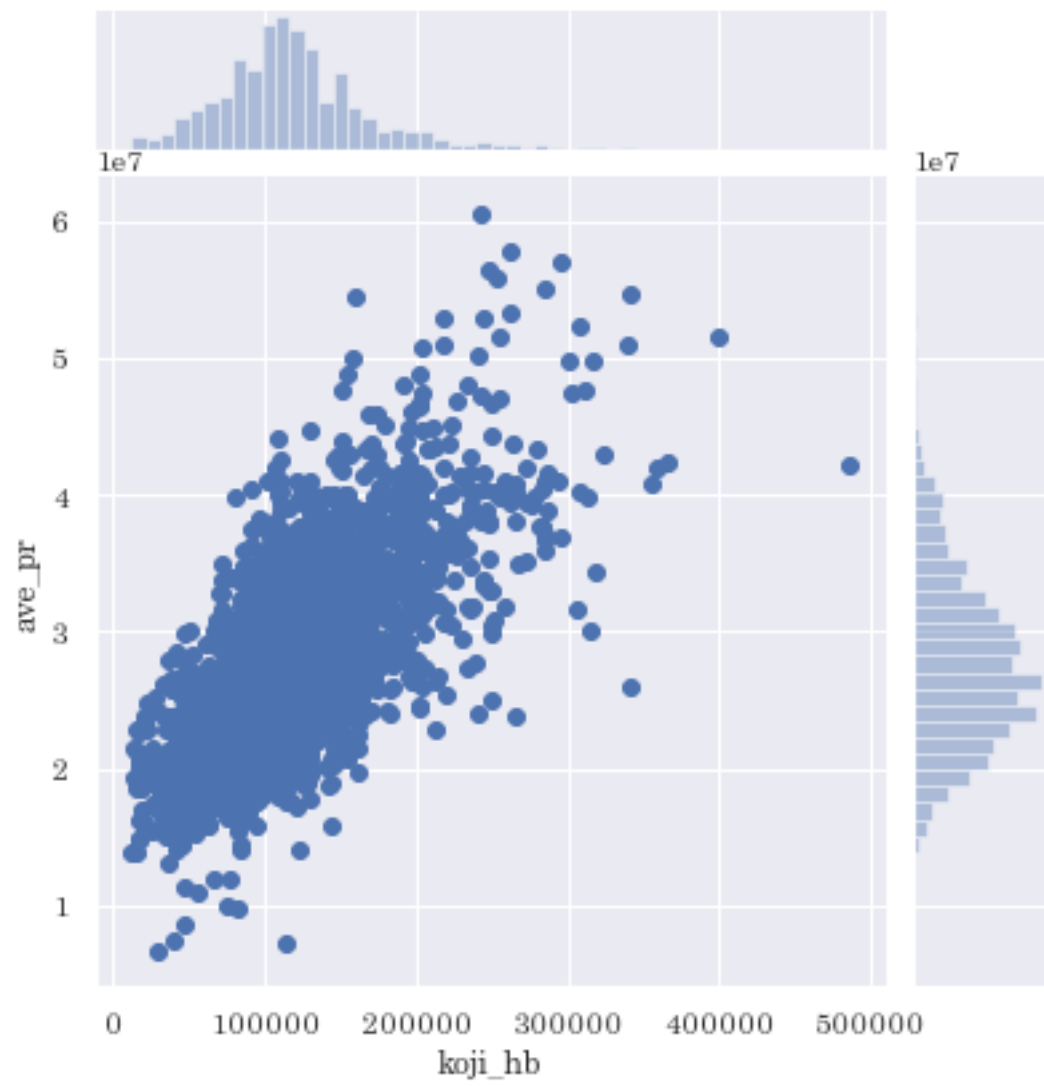




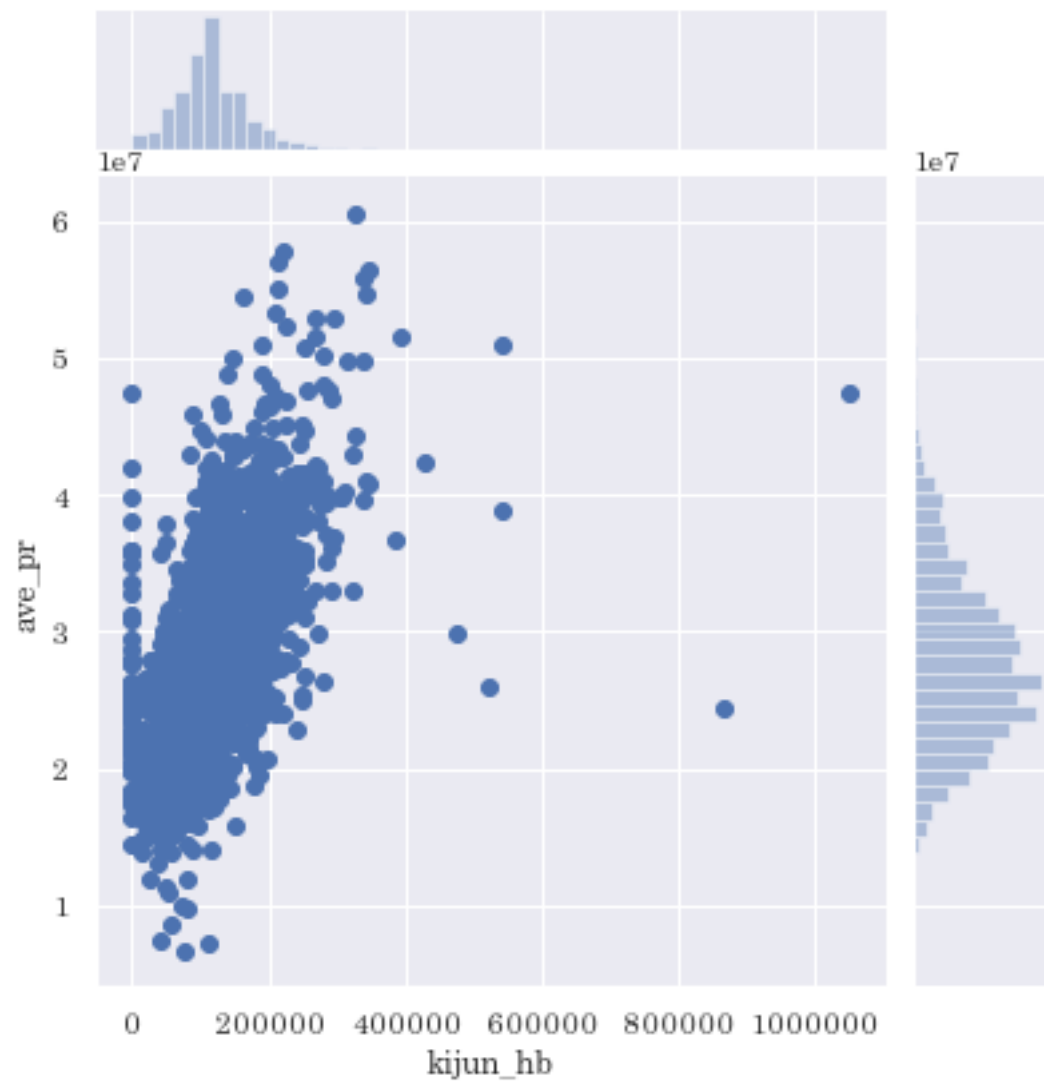


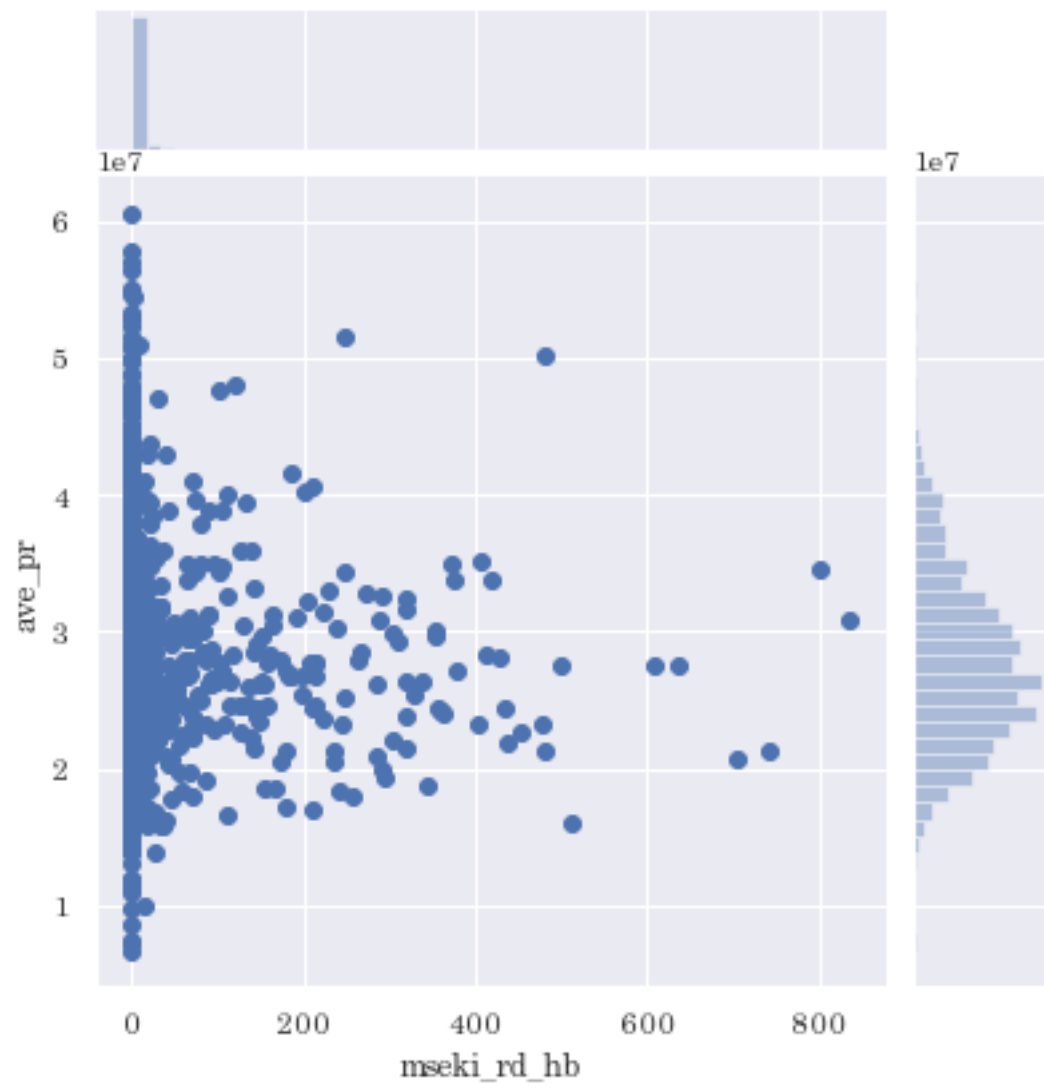


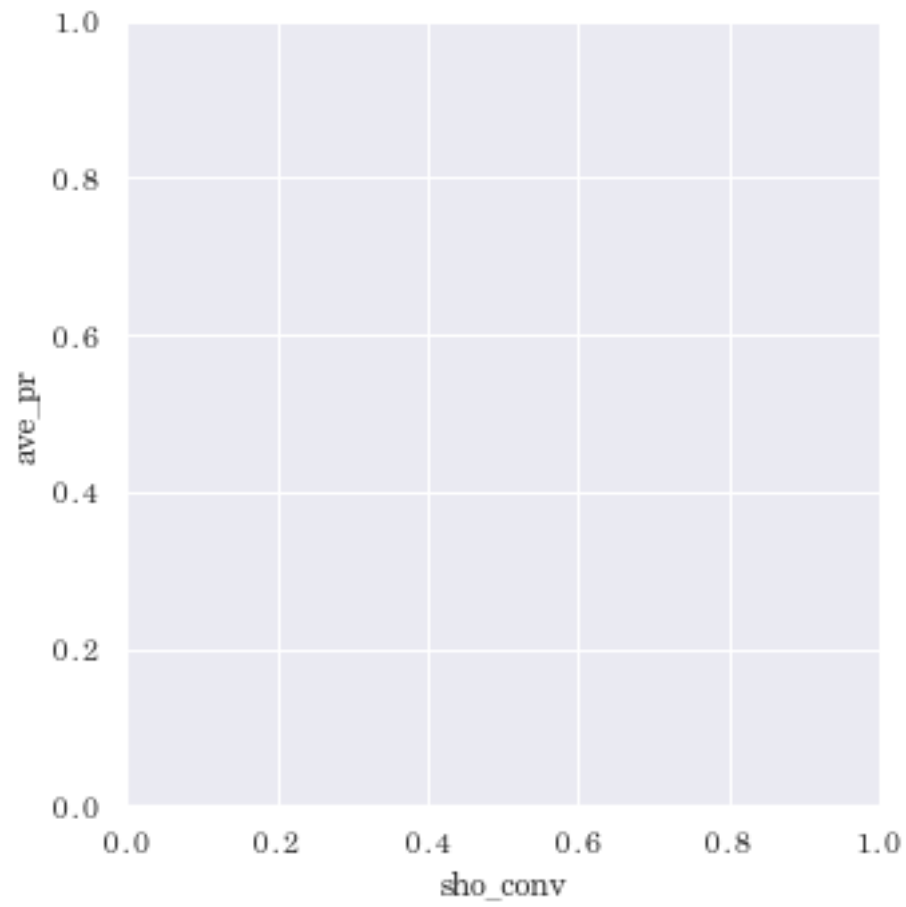


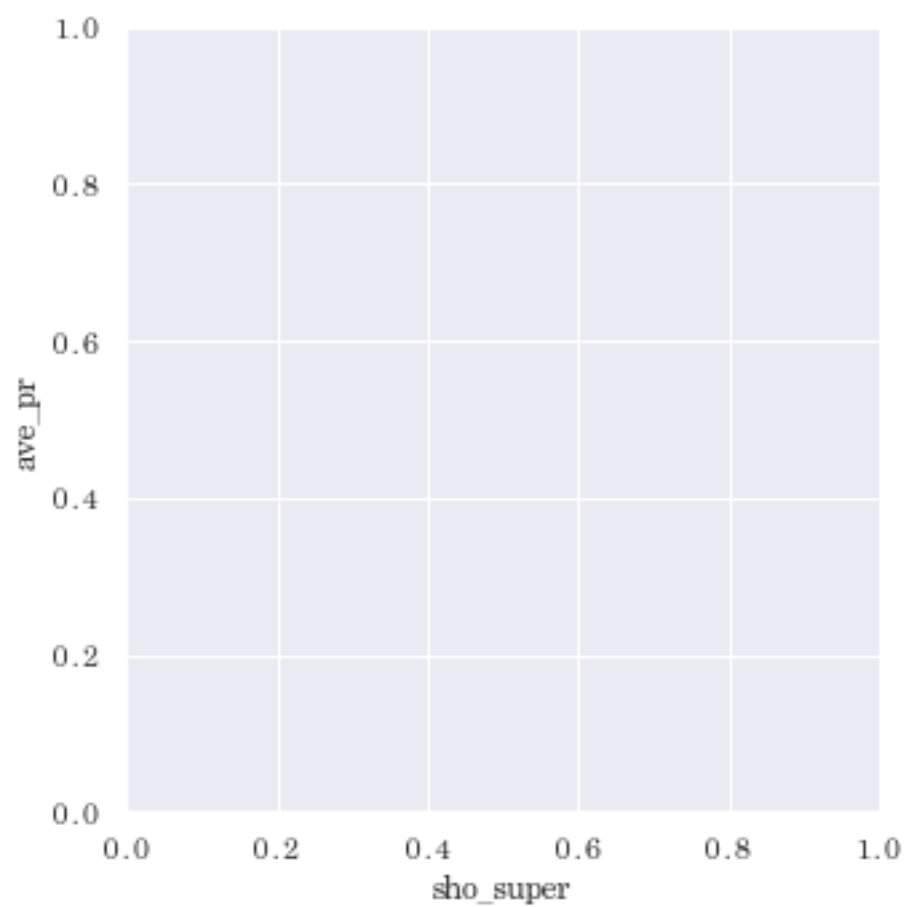


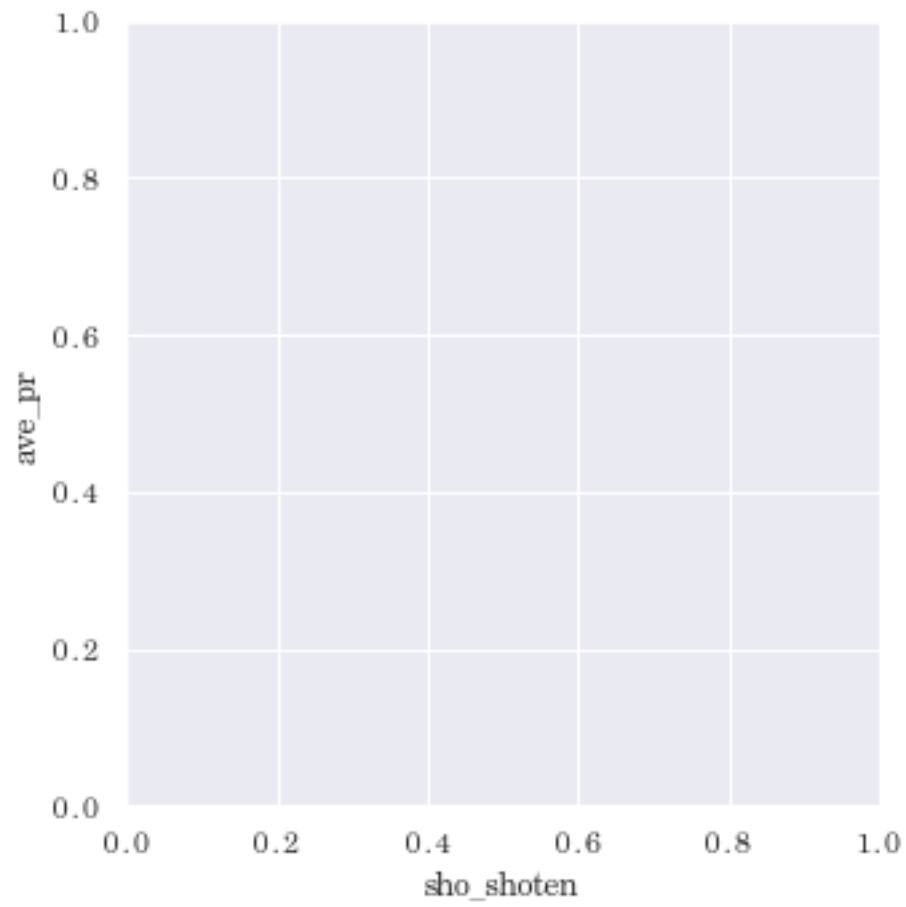


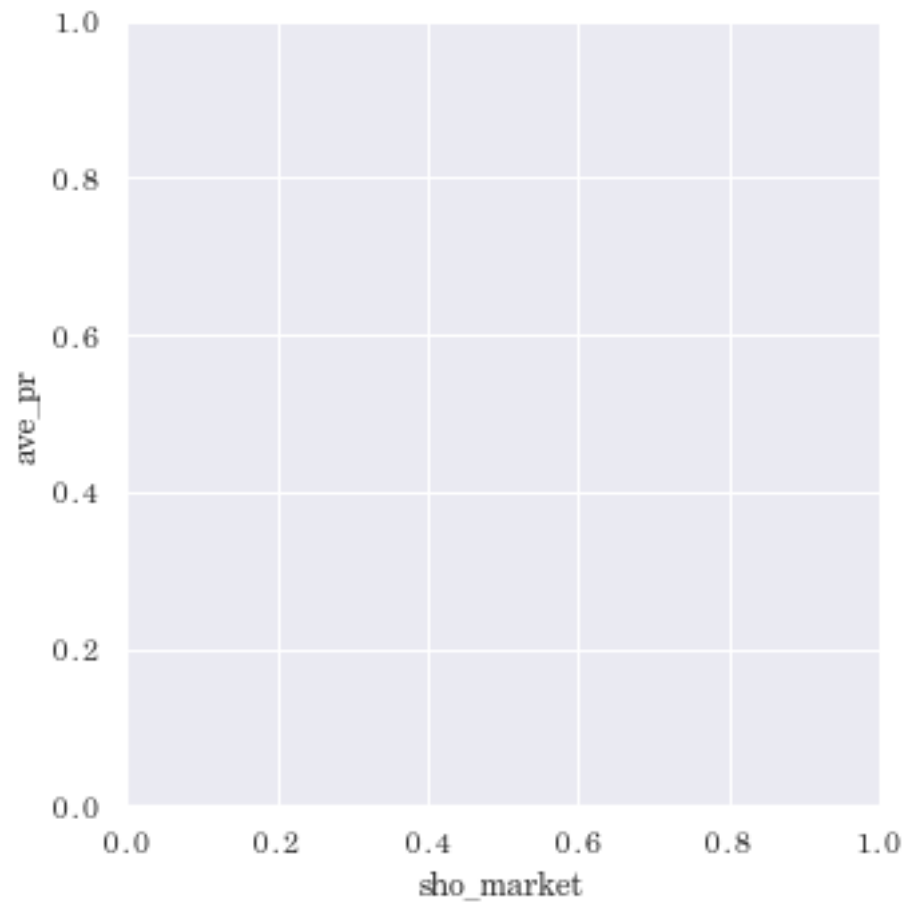


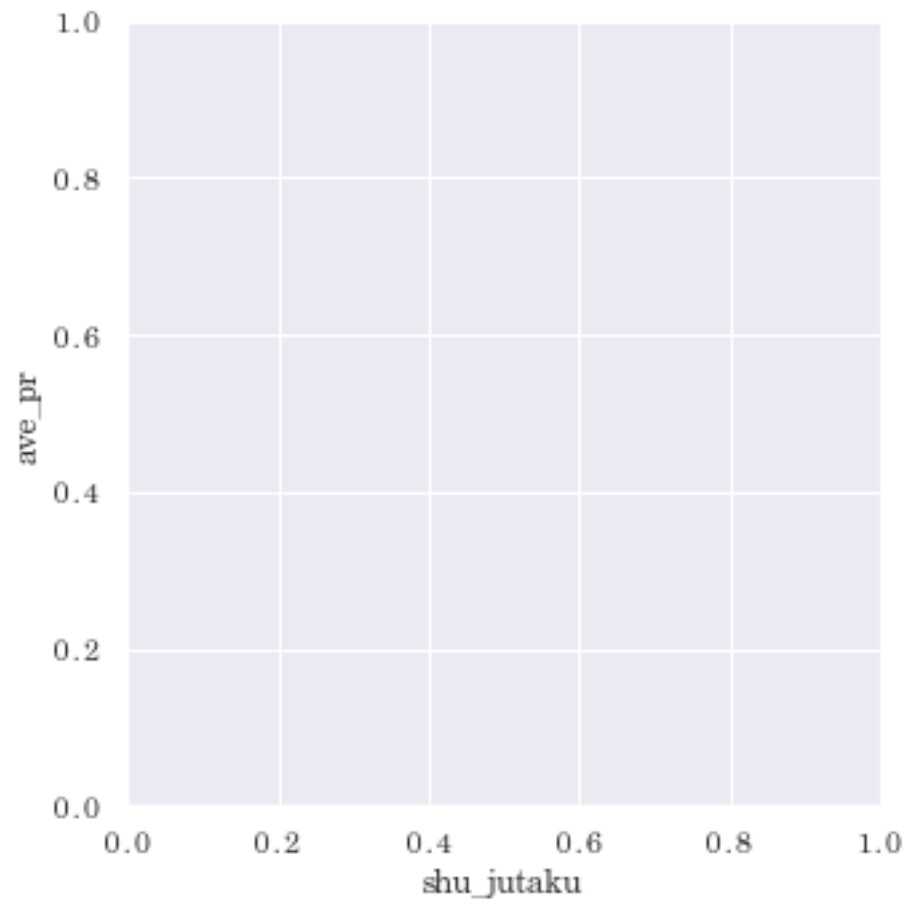


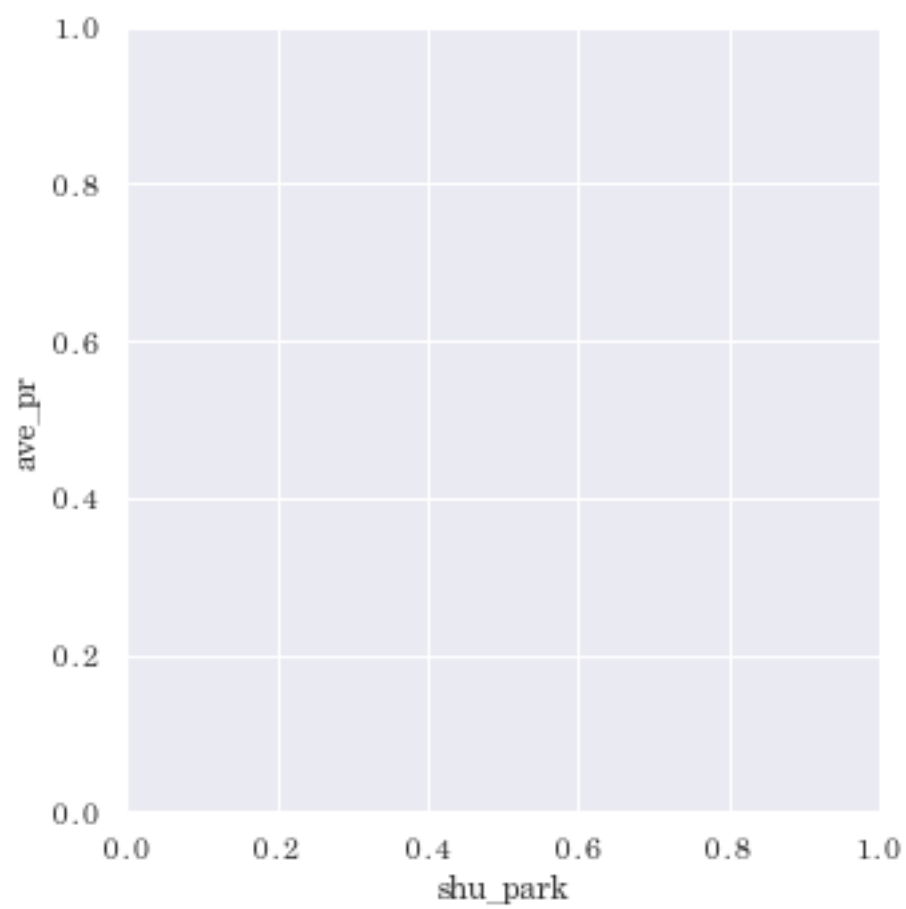




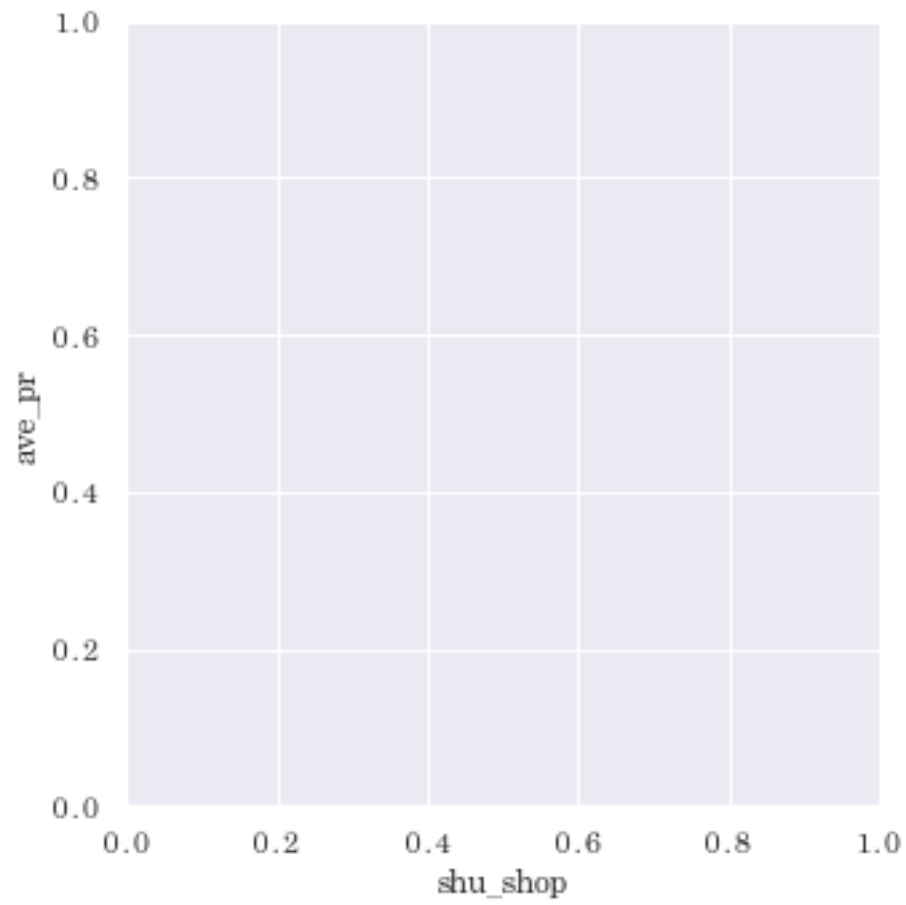


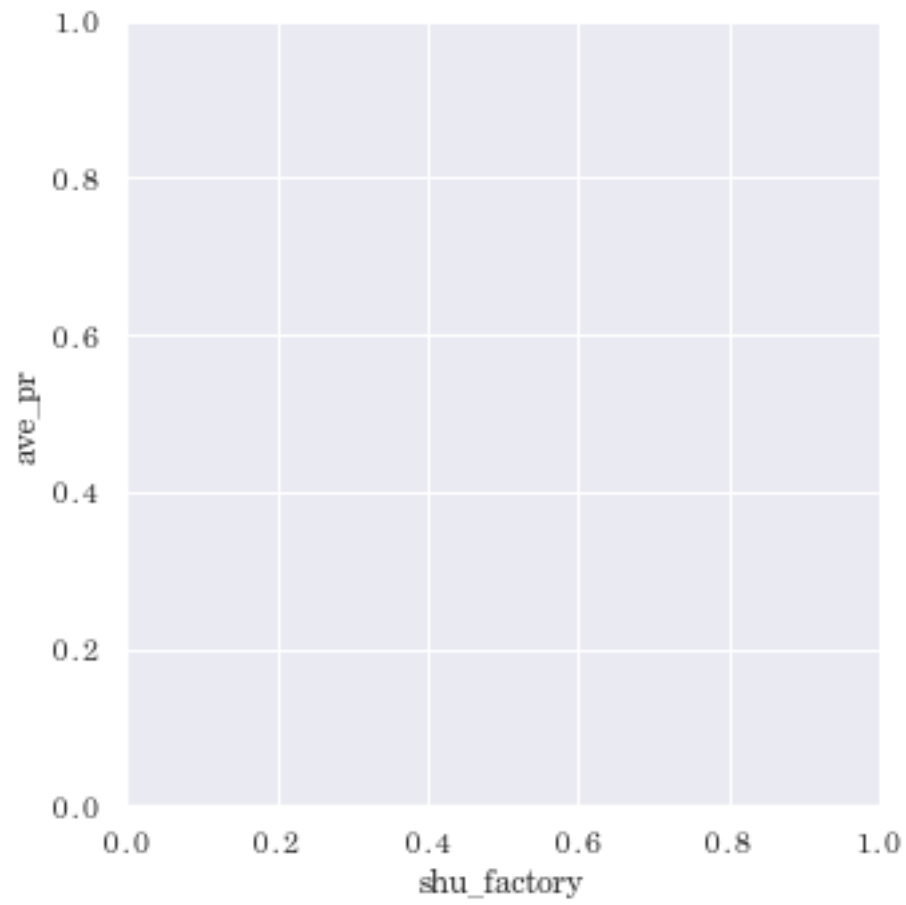


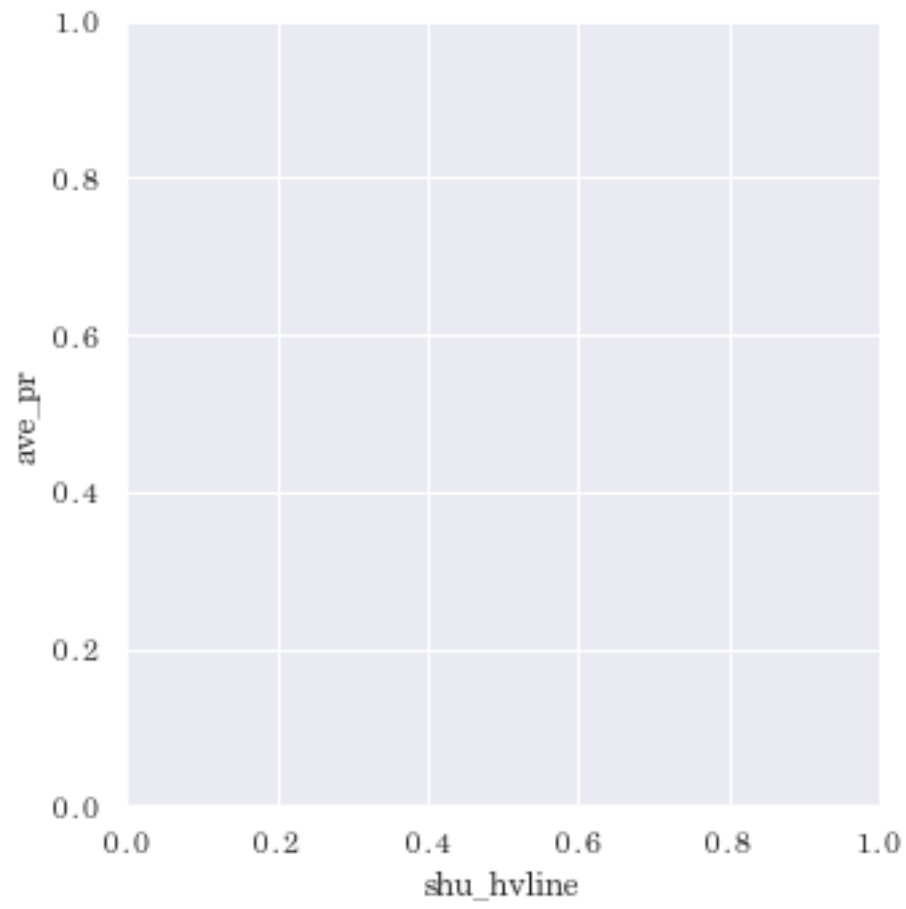


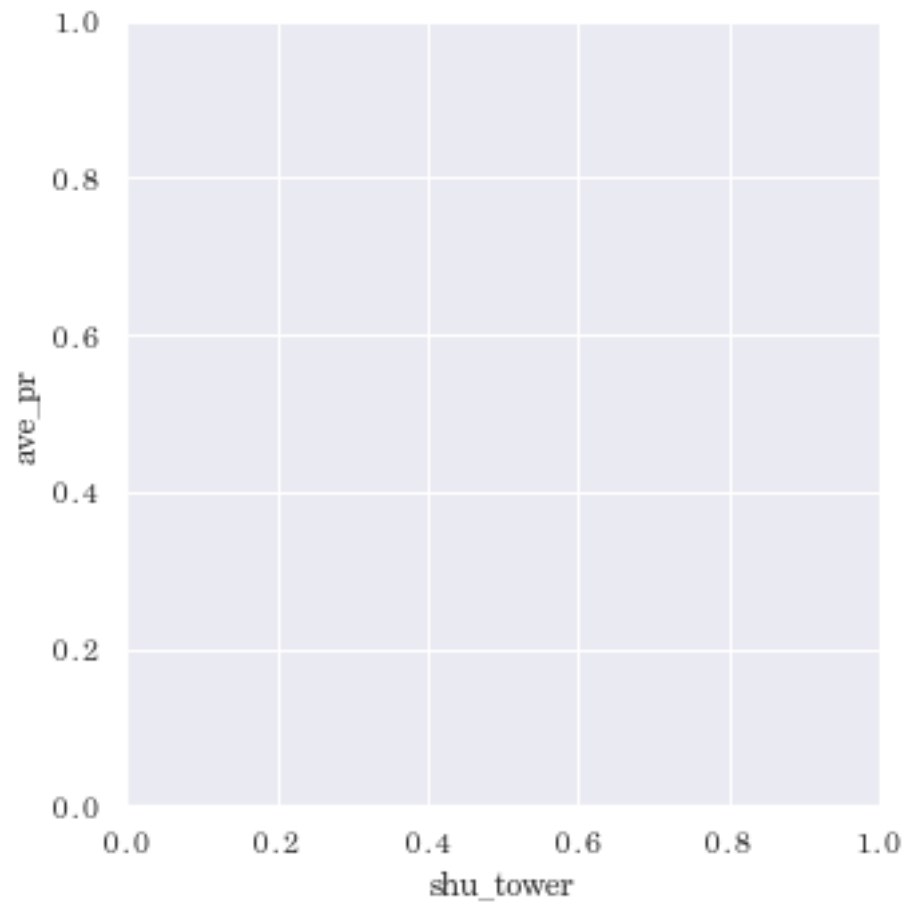


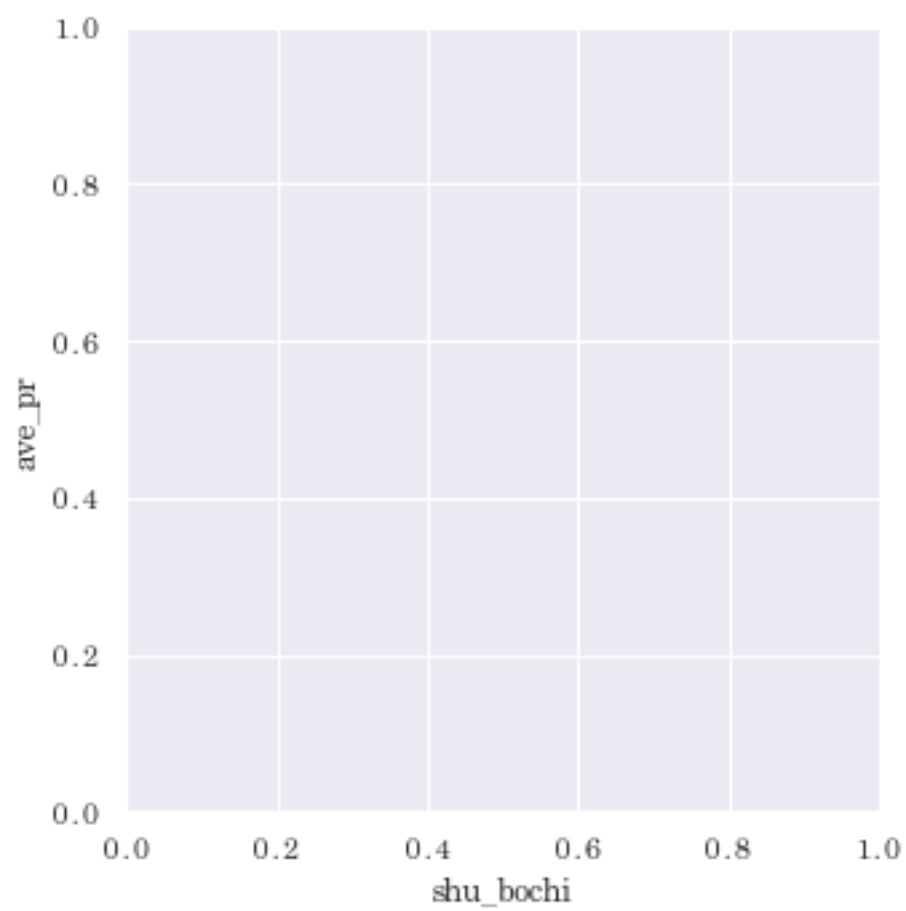


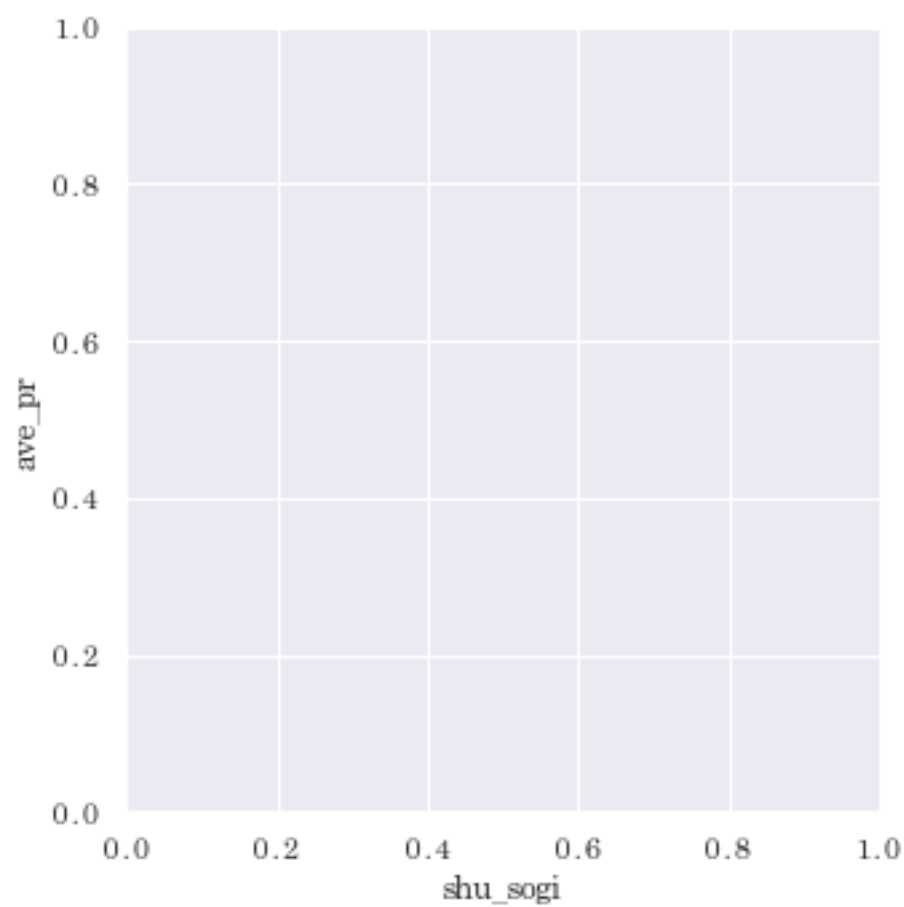


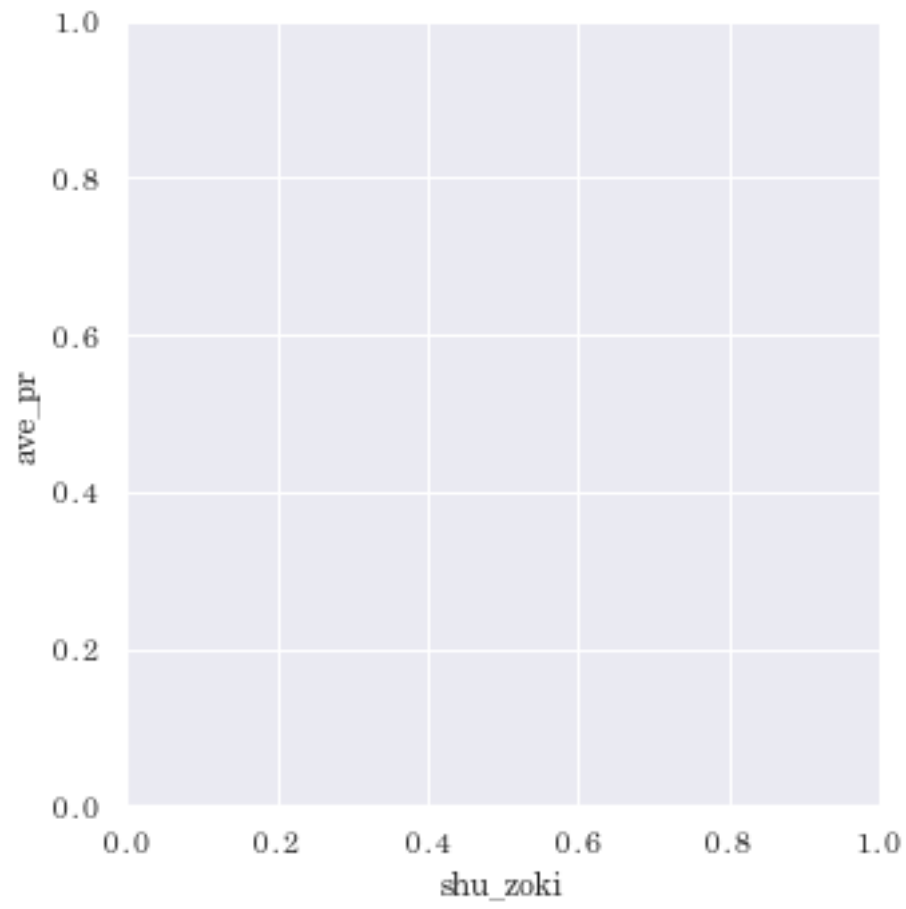


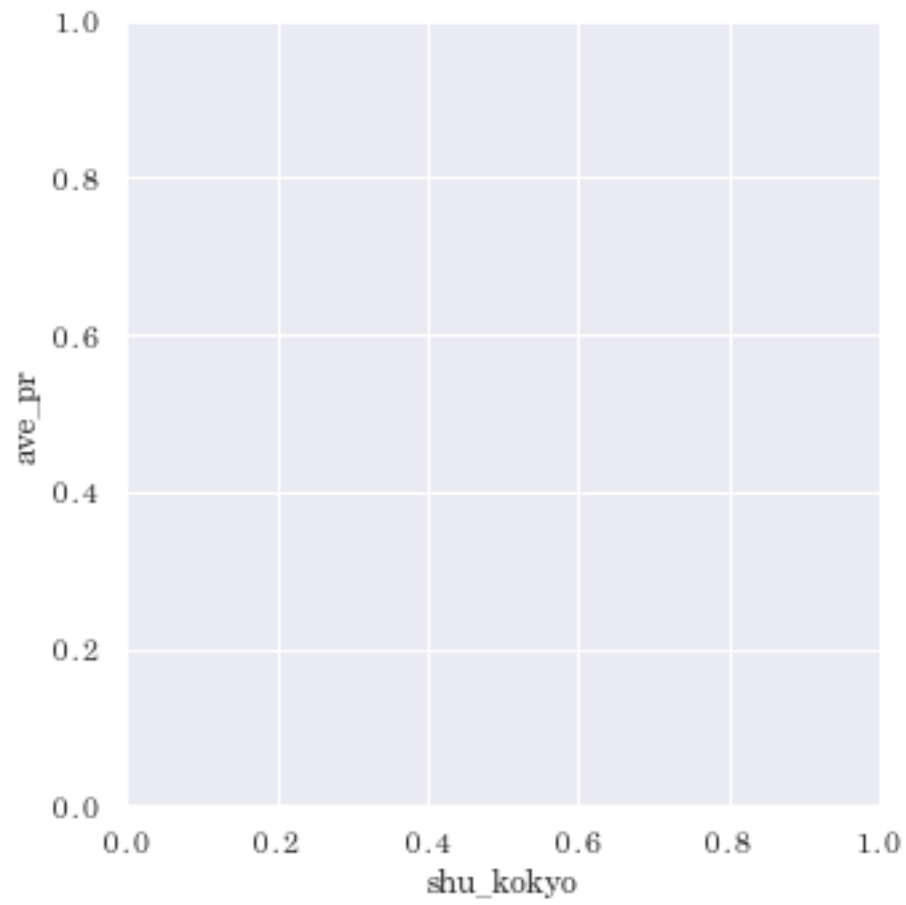




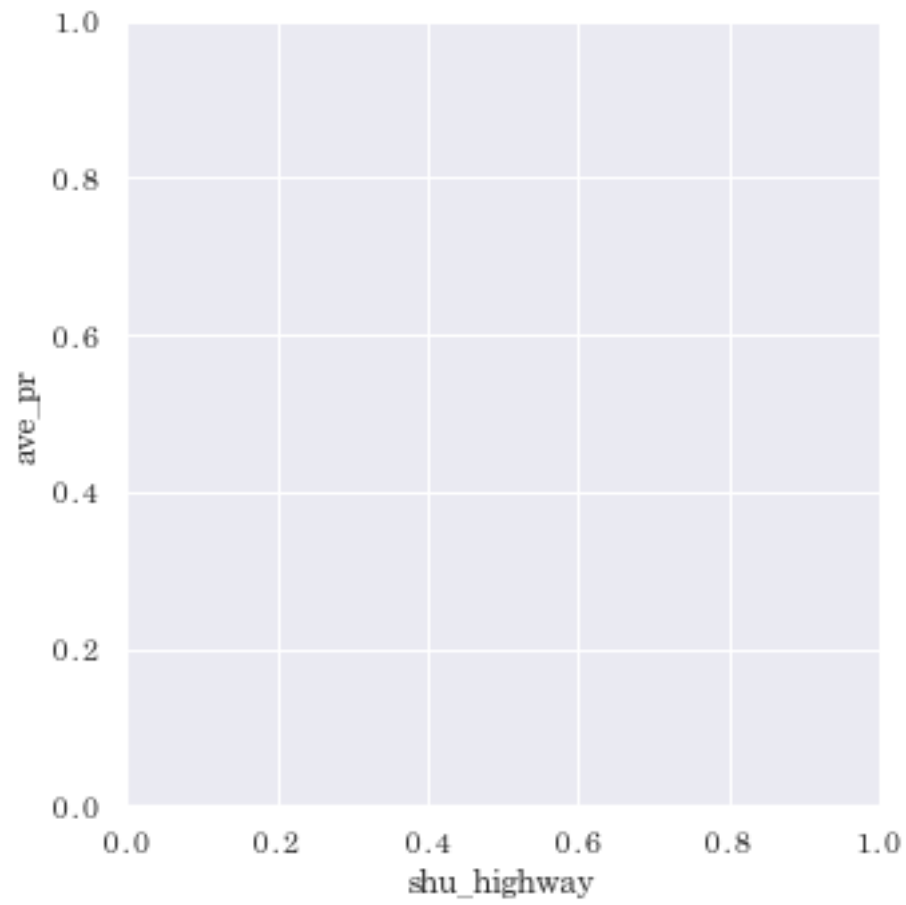


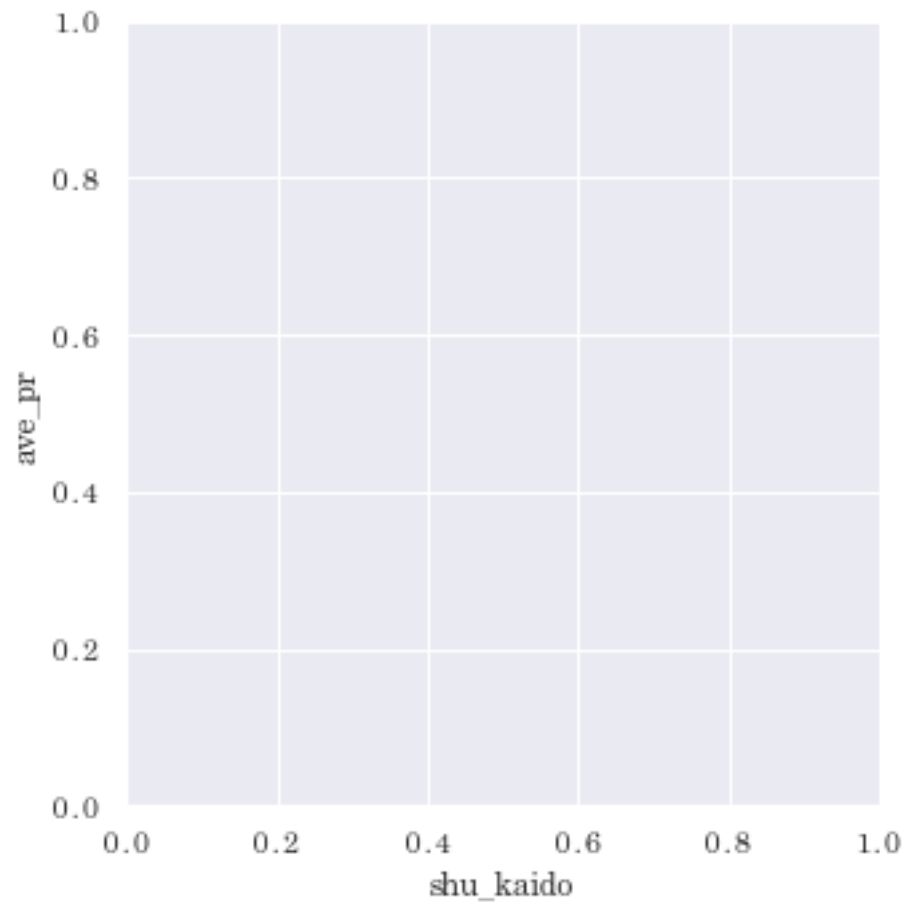


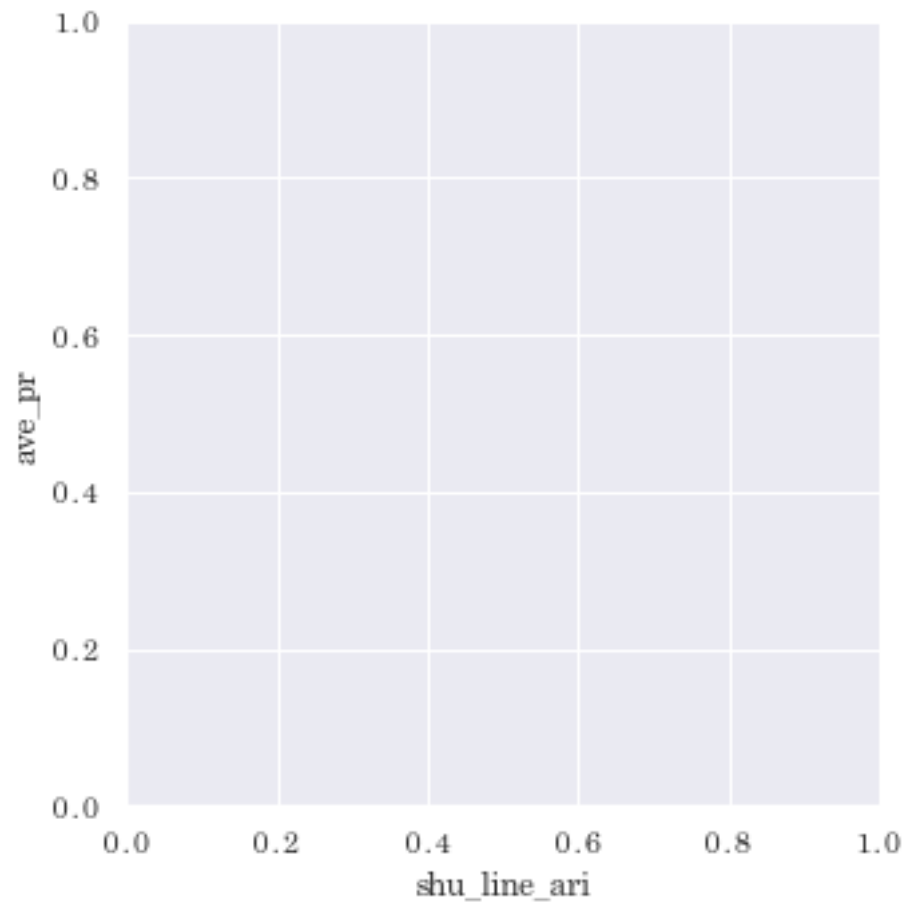


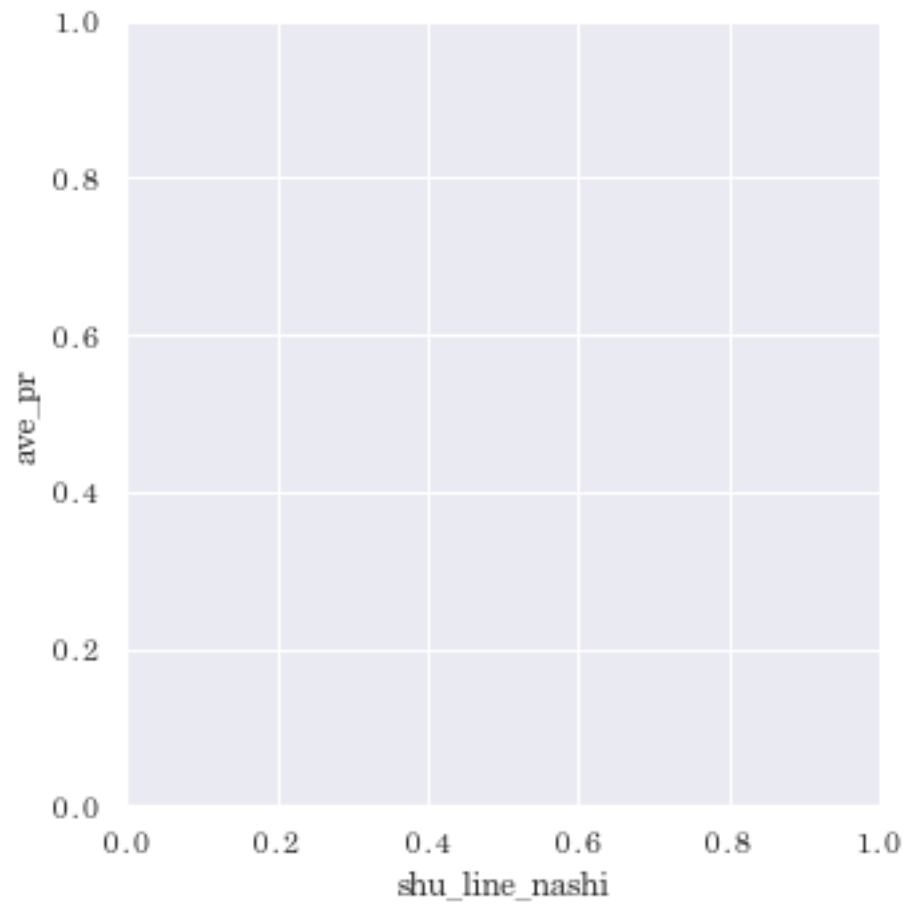


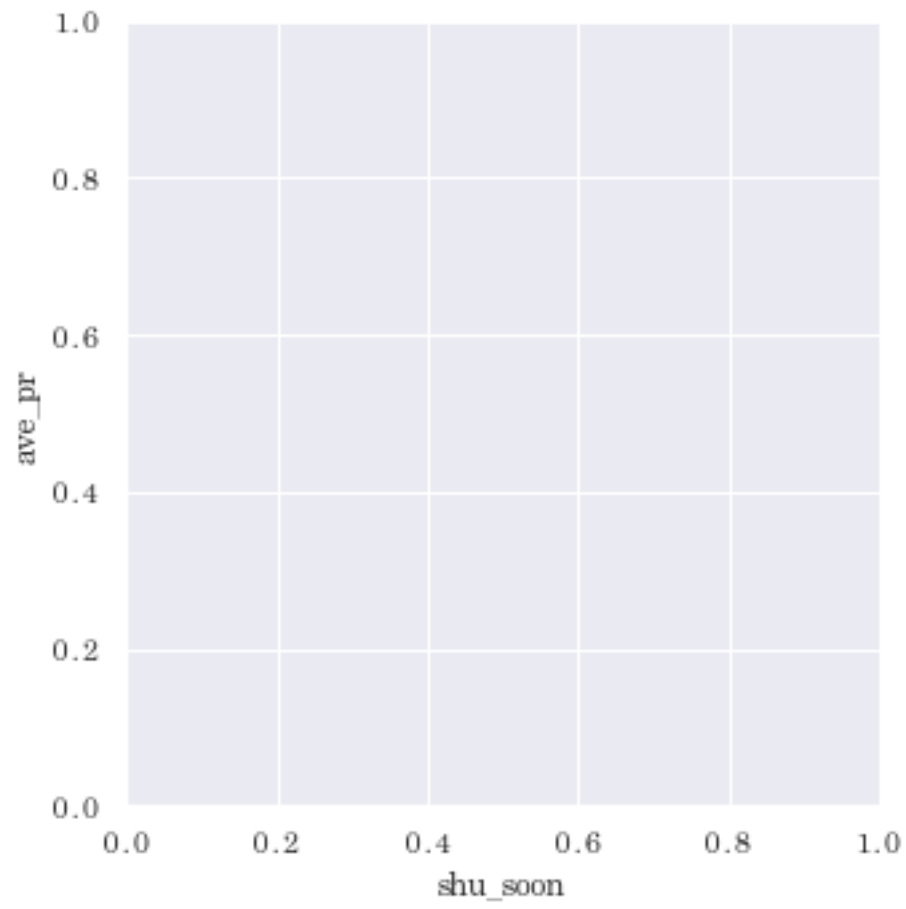


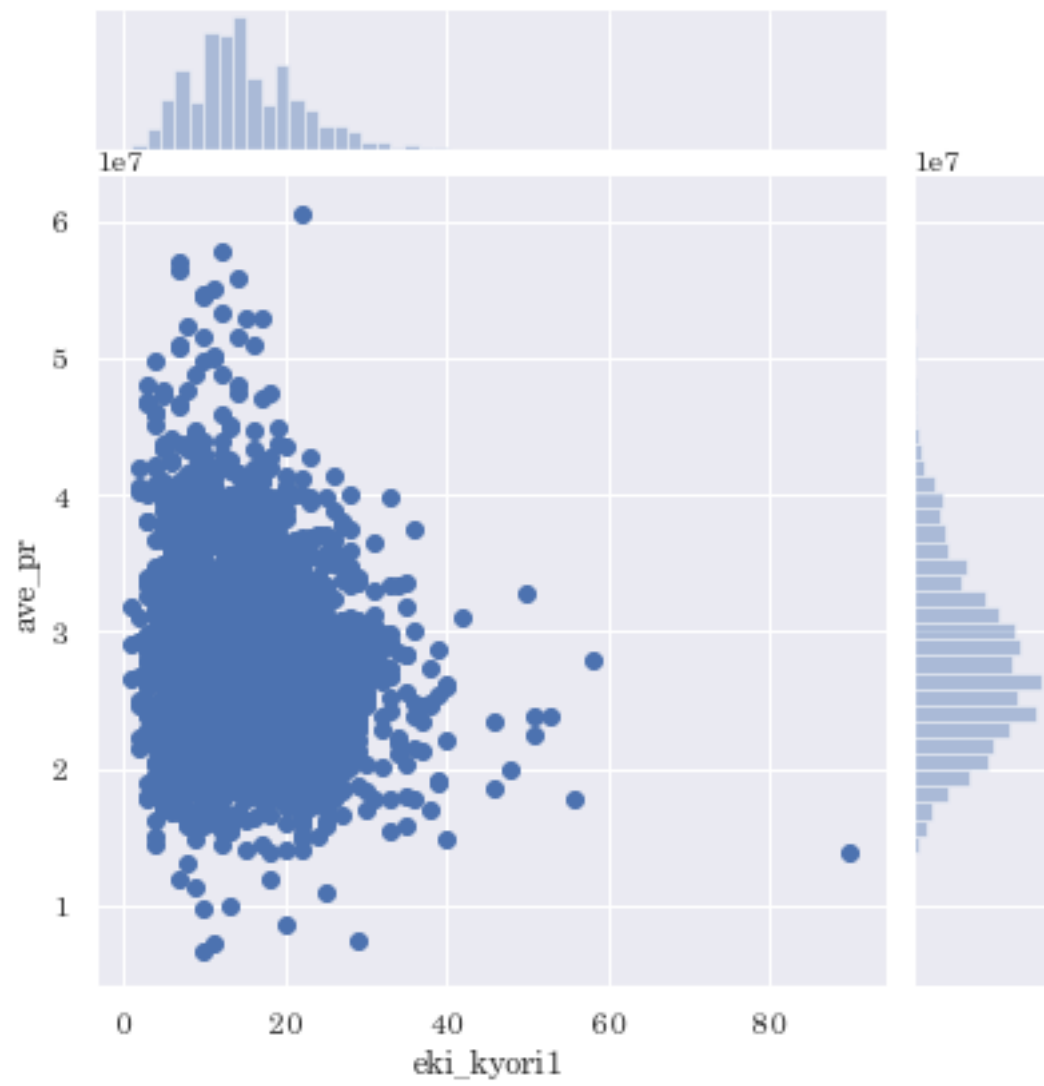


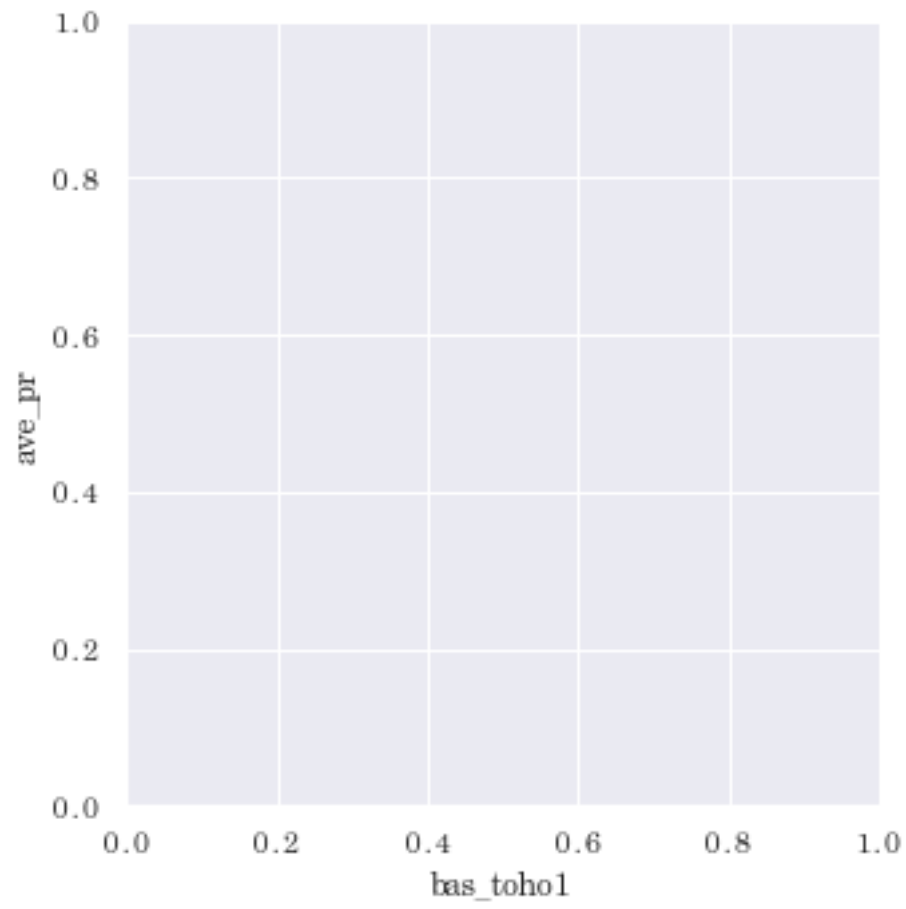


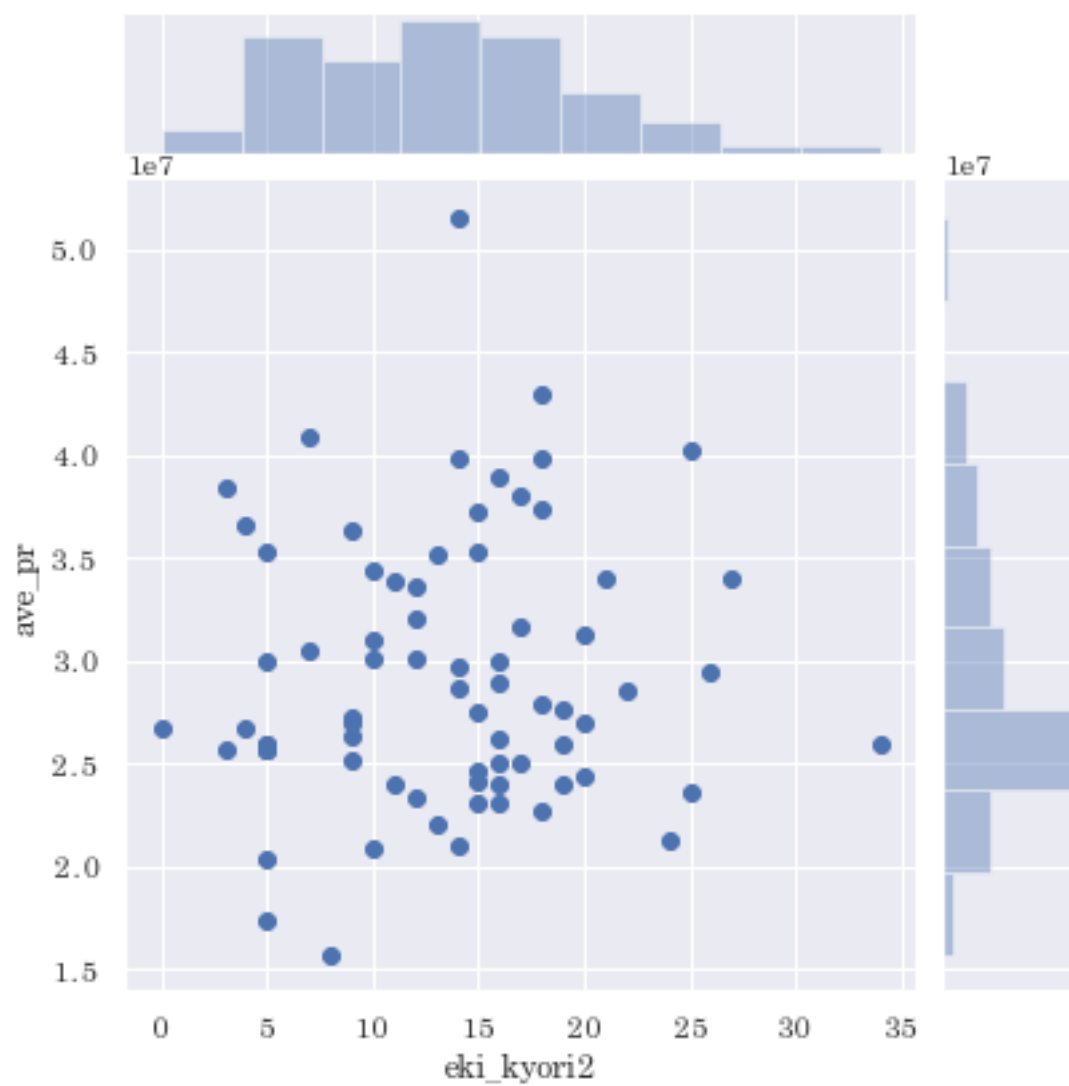




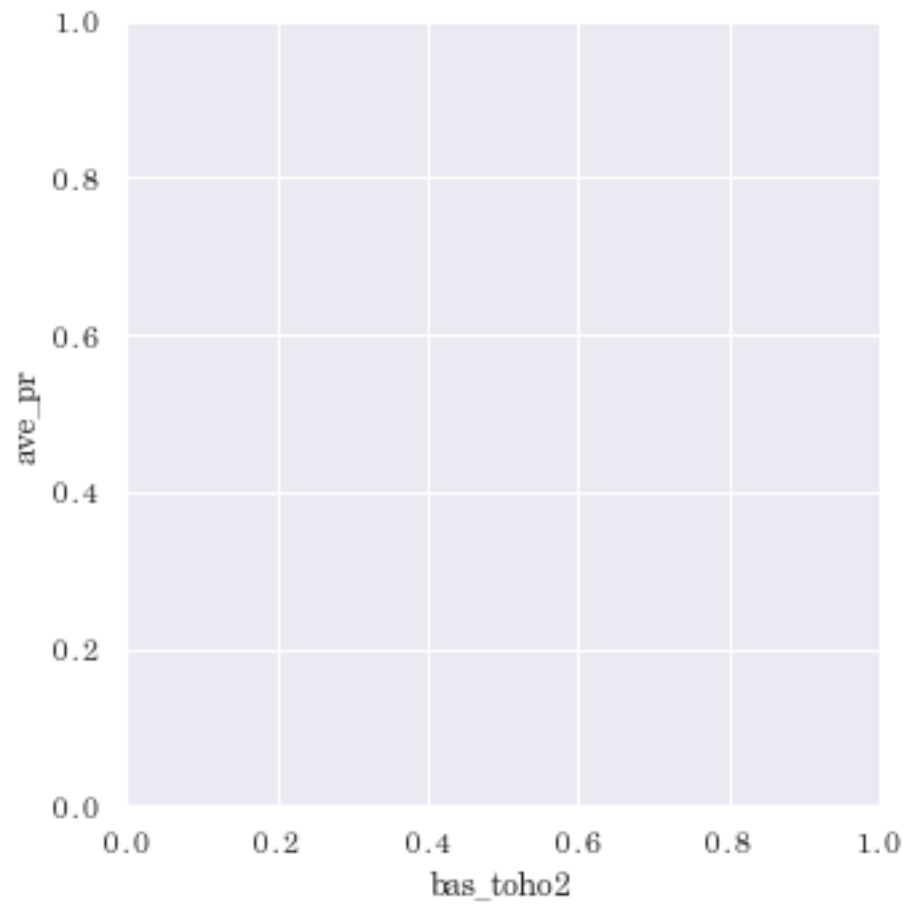












In [ ]: