Distribution_Analysis

October 13, 2022

0.0.1 Sarthak Arora

[3]:

```
import pandas as pd
import numpy as np
from sqlalchemy import create_engine
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm
import pylab
from scipy.stats import norm
engine = create_engine('sqlite://', echo=False)
```

0.0.2 loading the dataset and removing cases with priv_counts 0 or NaN's

0.0.3 also filtering out for cases where both mcare_pay_median and priv_pay_median are present

[4]:

```
df_main = pd.read_csv("/home/lennon_mccartney/Downloads/priv_mcare_f_pay.csv")
# df_hsp = pd.read_csv("/home/lennon_mccartney/Downloads/Hospital_Master_Sheet.
 ↪csv")
```

[5]:

```
df_main.head()
```

[5]:

```
     msa  year       site                  group  priv_count  priv_pay_mean  \
0  10180  2018  Inpatient  breast reconstruction         NaN            NaN
1  10420  2018  Inpatient  breast reconstruction         8.0    19937.08375
2  10500  2018  Inpatient  breast reconstruction         NaN            NaN
3  10540  2018  Inpatient  breast reconstruction         NaN            NaN
4  10580  2018  Inpatient  breast reconstruction         4.0    14837.26000

   priv_pay_median  priv_pay_iqr  mcare_los  mcare_pay_mean  mcare_pay_median  \
0              NaN           NaN        NaN             NaN               NaN
1        16147.330       5692.86   2.000000       8313.8475           8298.49
2              NaN           NaN   2.000000       9155.9400           9155.94
3              NaN           NaN        NaN             NaN               NaN
4        10420.675       4474.06   2.888889       9230.5000           8003.40

   mcare_pay_sd                    CBSA_NAME     State  FIPS State Code  \
0          NaN                  Abilene, TX     Texas                48
1  1575.325296                    Akron, OH      Ohio                39
2          NaN                   Albany, GA   Georgia                13
3          NaN           Albany-Lebanon, OR    Oregon                41
4  6267.381132  Albany-Schenectady-Troy, NY  New York                36

          lon        lat
0  -99.733144  32.448736
1  -81.519005  41.081445
2  -84.155741  31.578507
3 -122.907034  44.536512
4  -73.653621  42.763648
```

```
[6]: ## removing cases where priv_count is 0 or NaN as they belong to the prediction␣
      ↪set
     df_train = df_main[(df_main['priv_count'] != 0) & (df_main['priv_count'].
      ↪notnull()) & (df_main['mcare_pay_median'].notnull()) &␣
      ↪(df_main['priv_pay_median'].notnull()) ]
```

```
[7]: df_train.reset_index(inplace=True)
```

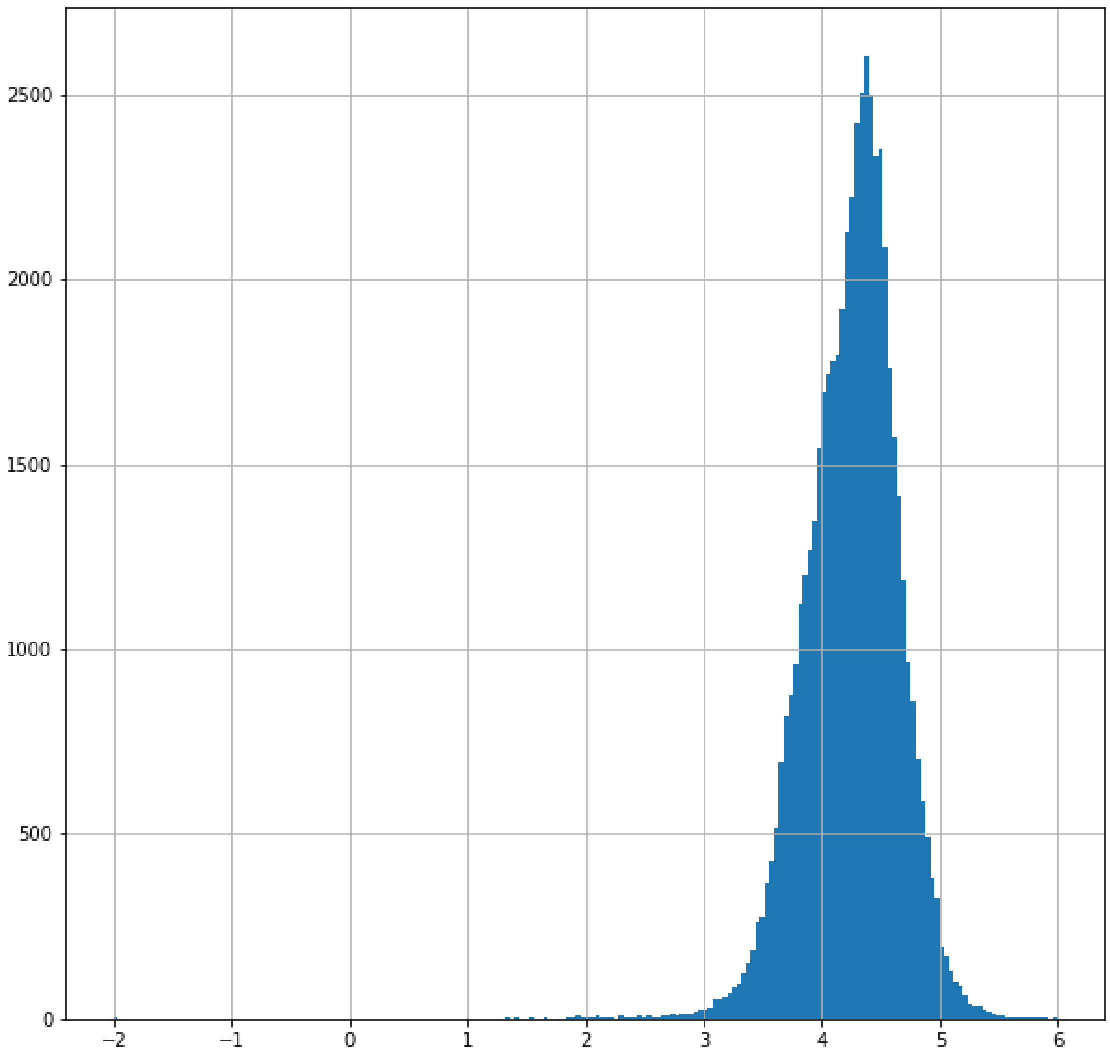
```
[8]: # df_train.to_sql('train_data', con=engine)
```

```
[9]: # engine.execute("SELECT msa,[group],count(*) as ct FROM train_data group by␣
      ↪1,2 order by 3 desc").fetchall()
```

0.0.4 looking at how the distribution of log(priv__pay__median) looks like

```
[10]: fig = plt.figure(figsize = (10,10))
      ax = fig.gca()
      np.log10(df_main[df_main["priv_pay_median"] > 0]["priv_pay_median"]).hist(bins␣
       ↪= 200, ax = ax)
```

```
[10]: <AxesSubplot:>
```



0.0.5 looking at the distribution of the difference between the median values for private and medicare

[11]:
```
(df_train["priv_pay_median"]-df_train["mcare_pay_median"]).describe()
```

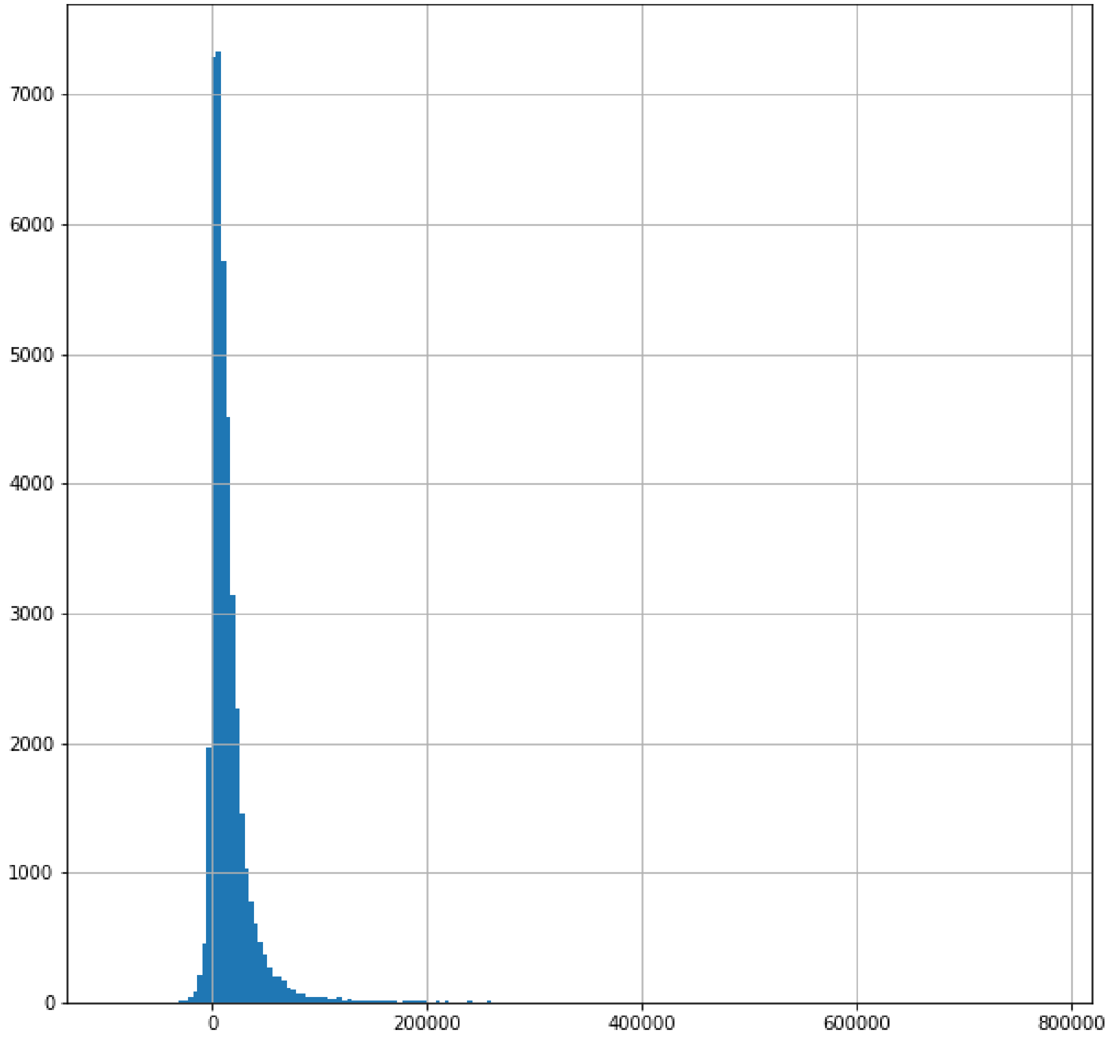
[11]:
```
count     39459.000000
mean      15816.831965
std       25474.567066
min      -91088.045000
25%        4129.882500
50%       10215.610000
75%       19718.665000
max      775469.400000
dtype: float64
```

[12]:

```
fig = plt.figure(figsize = (10,10))
ax = fig.gca()
(df_train["priv_pay_median"]-df_train["mcare_pay_median"]).hist(bins = 200, ax␣
 ↪= ax)
# plt.savefig("hist.png")
```

[12]:

```
<AxesSubplot:>
```



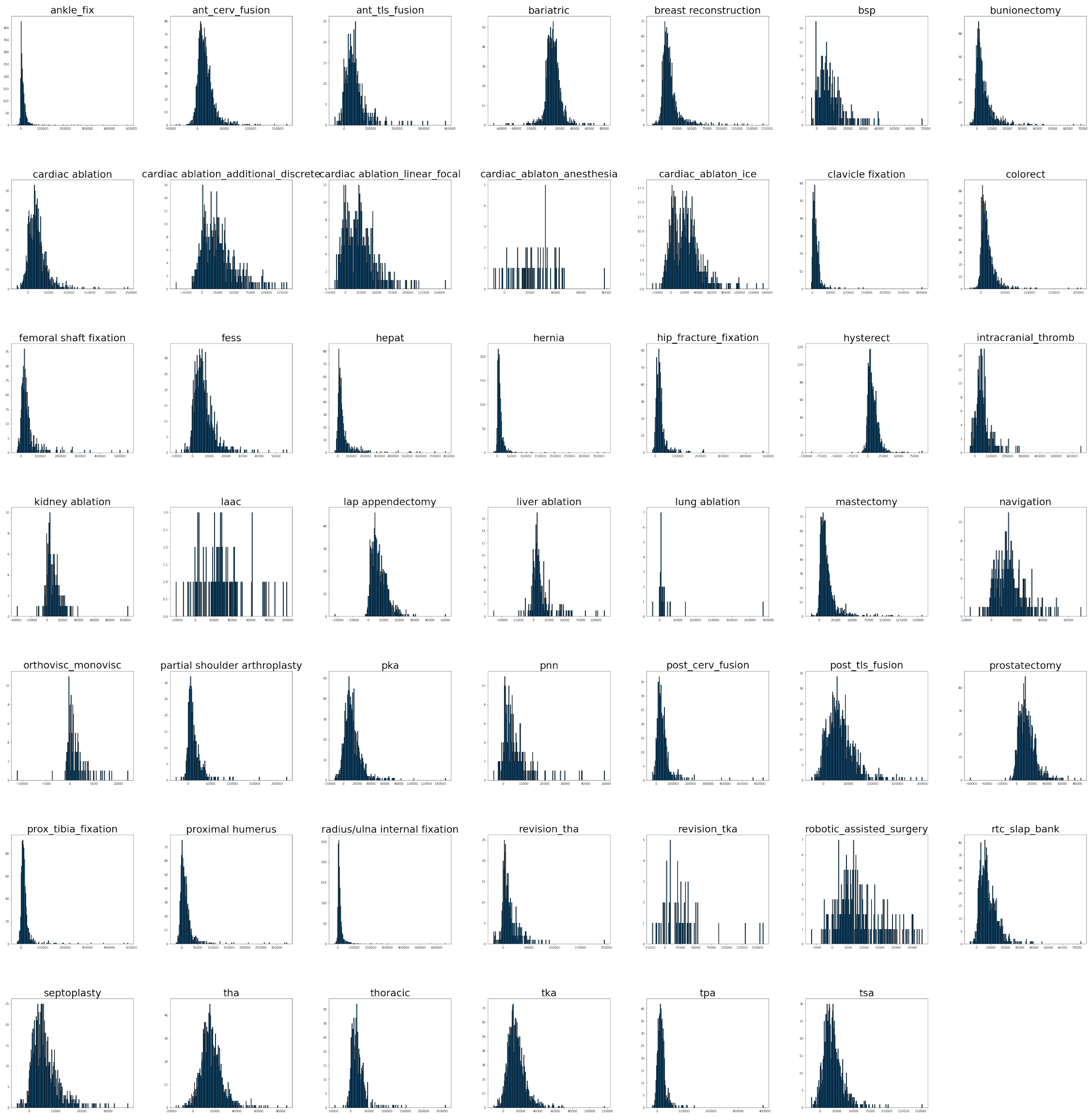
0.0.6 looking at distribution of the above difference at a group level

[14]:

```
fig = plt.figure(figsize = (60,65))
ax = fig.gca()
plt.tight_layout()
```

```python
fig = (df_train["priv_pay_median"]-df_train["mcare_pay_median"]).hist(by = df_train["group"] ,bins = 200, ax = ax,ec="k",rot = 0)
[x.title.set_size(32) for x in fig.ravel()]
plt.savefig("diff_gp_hist.png",dpi = 100)
```

```
/home/lennon_mccartney/.local/lib/python3.8/site-packages/pandas/plotting/_matplotlib/hist.py:370: UserWarning: To output multiple subplots, the figure containing the passed axes is being cleared
  axes = _grouped_hist(
```



0.0.7 checking if tka group histogram above can pass as a normal distribution

[15]:

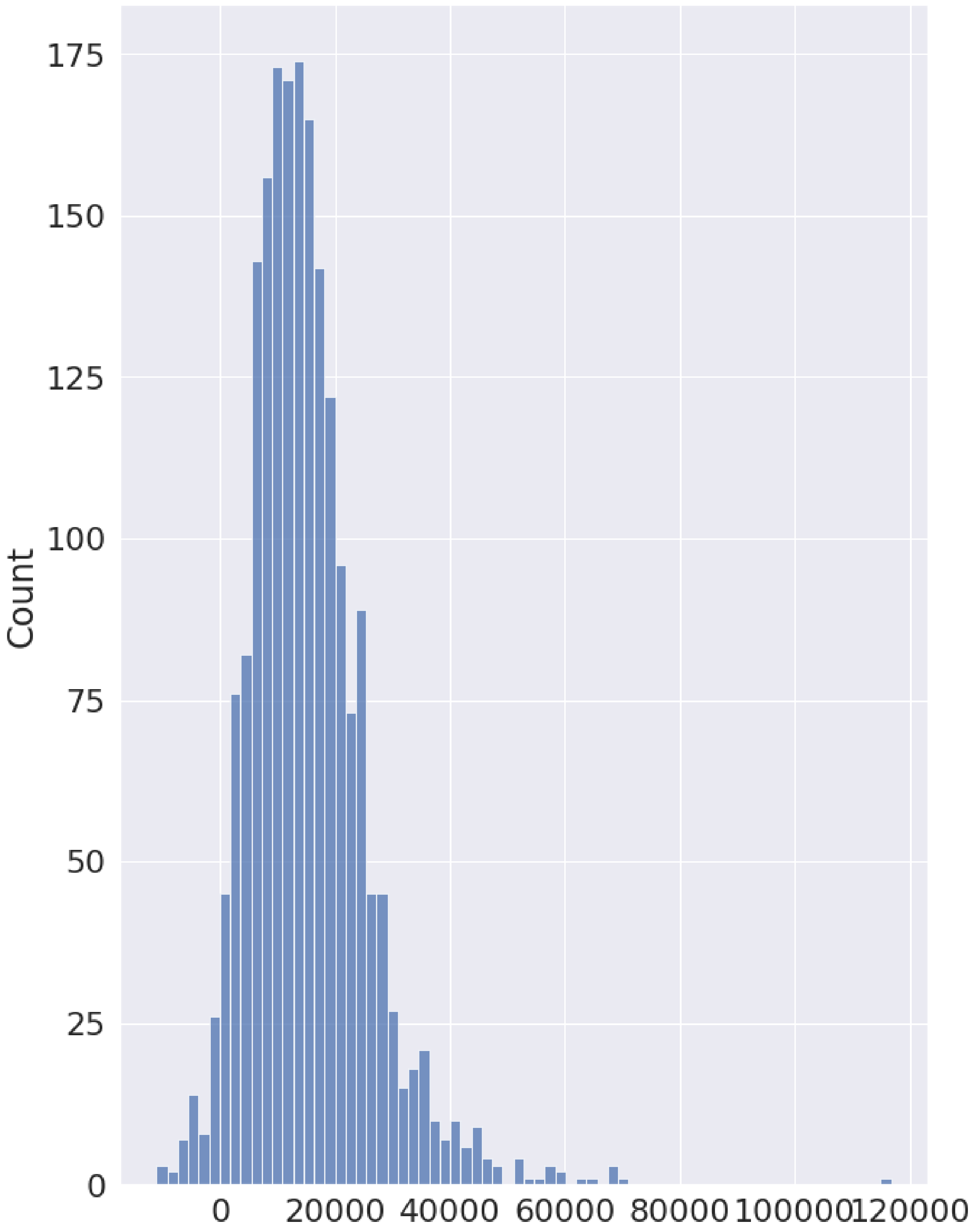
```
dt_pts = df_train[df_train["group"] == "tka"]["priv_pay_median"] -␣
 ↪df_train[df_train["group"] == "tka"]["mcare_pay_median"]
gp_stats = dt_pts.describe()
mu = gp_stats[1]
std = gp_stats[2]
```

[16]:

```
plt.figure(figsize=(10,15))
sns.set(font_scale=2)
sns.histplot(dt_pts)
```

[16]:

```
<AxesSubplot:ylabel='Count'>
```

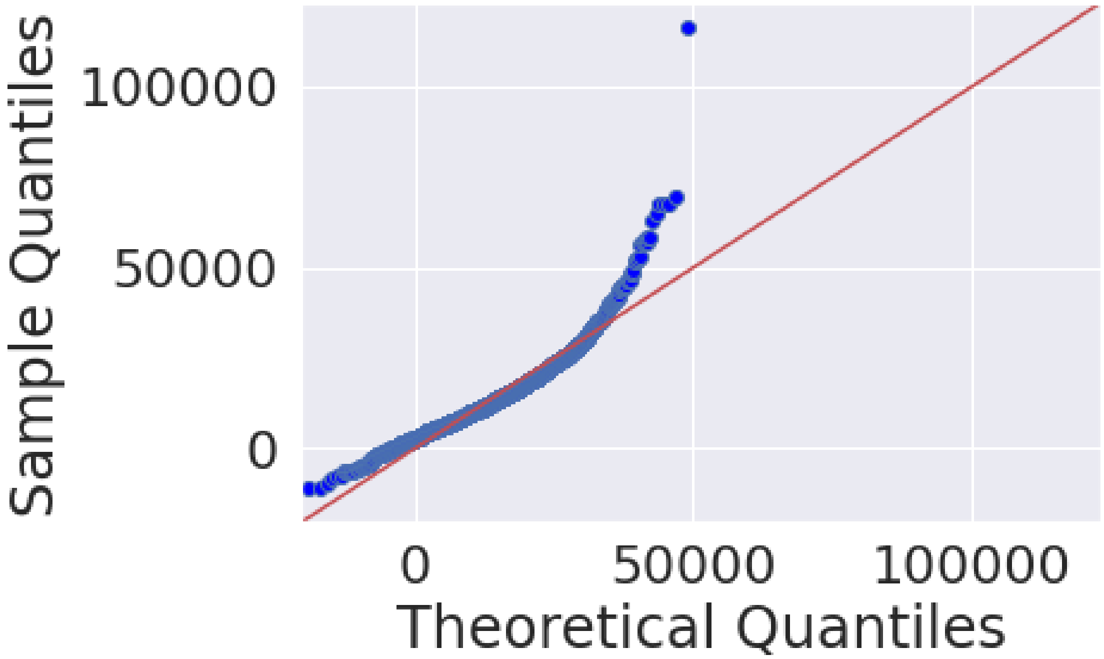


[17]:

```
sm.qqplot(dt_pts,dist=norm(mu,std), line ='45')
pylab.show()
```

```
/home/lennon_mccartney/anaconda3/lib/python3.8/site-
packages/statsmodels/graphics/gofplots.py:993: UserWarning: marker is
```

```
redundantly defined by the 'marker' keyword argument and the fmt string "bo" (->
marker='o'). The keyword argument will take precedence.
  ax.plot(x, y, fmt, **plot_style)
```



0.0.8 the QQ plot above shows a huge deviation from normal behaviour, hence we try to remove outliers and make a QQ plot again, we also look at what kind of rows are we removing as outliers

[19]:

```
rmv_outliers = dt_pts[(dt_pts > mu - 3*std) & (dt_pts < mu + 3*std)].values
outliers = list(dt_pts[(dt_pts <= mu - 3*std) | (dt_pts >= mu + 3*std)].index)
dt_out = df_train.loc[df_train.apply(lambda x: x['index'] in outliers, axis=1)]
dt_out
```

[19]:

```
      index    msa  year        site             group  priv_count  \
5609  14628  47894  2020  Outpatient  post_cerv_fusion         1.0
5814  14938  38340  2018   Inpatient   post_tls_fusion         2.0
5819  14945  39340  2018   Inpatient   post_tls_fusion        16.0
5879  15016  45820  2018   Inpatient   post_tls_fusion         3.0
5881  15018  46060  2018   Inpatient   post_tls_fusion        18.0
5951  15113  15764  2019   Inpatient   post_tls_fusion        47.0
5959  15121  16300  2019   Inpatient   post_tls_fusion         9.0
6058  15265  29820  2019   Inpatient   post_tls_fusion        63.0
6059  15267  30020  2019   Inpatient   post_tls_fusion         3.0
6081  15301  33700  2019   Inpatient   post_tls_fusion         2.0
6091  15314  34980  2019   Inpatient   post_tls_fusion        62.0
```

```
6108  15336  37340  2019   Inpatient   post_tls_fusion        17.0
6155  15390  42200  2019   Inpatient   post_tls_fusion         2.0
6253  15518  15540  2020   Inpatient   post_tls_fusion         4.0
6505  15866  49620  2020   Inpatient   post_tls_fusion        12.0
6525  15955  17980  2018  Outpatient   post_tls_fusion         1.0
6527  15962  19124  2018  Outpatient   post_tls_fusion        21.0
6551  16088  30780  2018  Outpatient   post_tls_fusion         1.0
6557  16107  33124  2018  Outpatient   post_tls_fusion         1.0

      priv_pay_mean  priv_pay_median  priv_pay_iqr  mcare_los  mcare_pay_mean  \
5609    12235.62000        12235.620        0.0000   0.000000     6985.335000
5814    19587.46000        19587.460      681.6100  10.000000    32043.380000
5819    50772.16250        45869.805    30407.6800   3.082677    27785.656060
5879   105737.26000        90744.500    56941.3100   3.457627    26749.052710
5881    53965.38778        55533.395    30599.4875   3.413551    30583.553060
5951    60486.61723        52193.060    28620.9650   3.594444    37652.319940
5959    52371.88333        57260.130    28023.5300   3.727273    26221.624550
6058    70990.90889        63192.900    28453.1400   3.432967    32706.230440
6059    73439.13667        64040.000    30049.3950   3.217391    26939.818700
6081    98507.12000        98507.120    69509.7000   5.523810    51870.478570
6091    79848.38355        70663.230    38198.3925   3.521082    29219.691670
6108    72042.71941        66402.860    42681.7300   4.355263    28141.249670
6155   112852.68500       112852.685    31537.6850   4.190476    43442.541140
6253    79903.40000        72053.155    33163.3800   4.500000    25369.431250
6505    99551.83000        87720.575    65813.4650   3.213793    30365.414830
6525    36267.77000        36267.770        0.0000   0.000000     3112.276000
6527    31047.67238        18710.810    30821.0100   0.000000     7233.178155
6551     8303.00000         8303.000        0.0000   0.000000     4554.950870
6557    71906.39000        71906.390        0.0000   0.000000     6264.940000

      mcare_pay_median  mcare_pay_sd  \
5609             71.85  13875.035360
5814          32043.38           NaN
5819          24986.60   8950.401953
5879          22177.32  11400.822400
5881          29827.73  11552.783620
5951          34805.26  12922.453230
5959          24042.03   5321.346872
6058          32790.17  17845.910310
6059          23182.96   5893.185582
6081          40709.44  25302.974220
6091          26331.29  17963.519410
6108          23007.59  11301.341810
6155          42985.24  11924.036560
6253          30461.27  19461.509320
6505          27819.72  15979.296770
6525              0.00   5072.834190
```

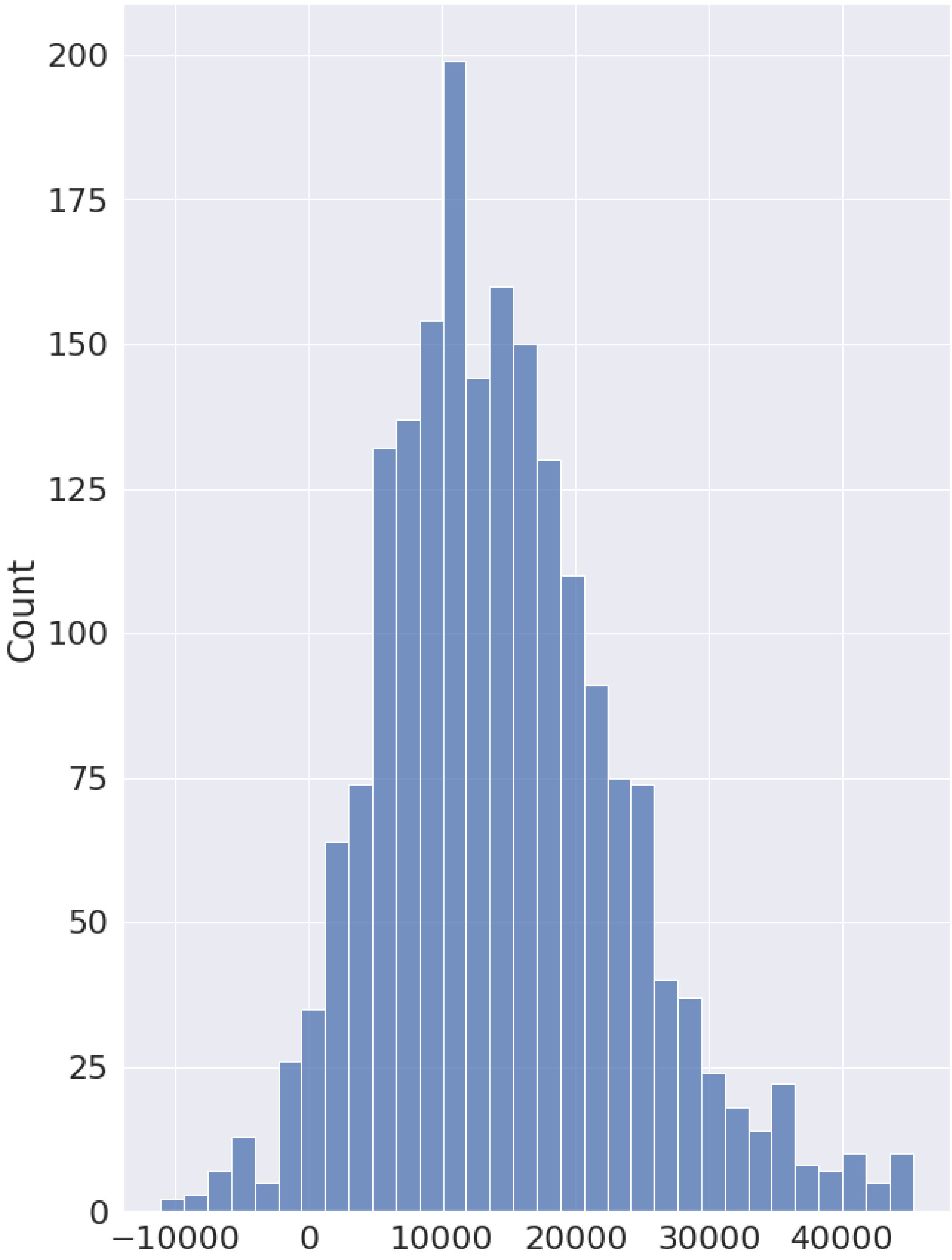
```
6527          8462.03   3501.119206
6551          7592.99   4084.511503
6557          8218.57   4346.051307

                                             CBSA_NAME          State  \
5609    Washington-Arlington-Alexandria, DC-VA-MD-WV       Virginia
5814                                  Pittsfield, MA  Massachusetts
5819                                  Provo-Orem, UT           Utah
5879                                      Topeka, KS         Kansas
5881                                      Tucson, AZ        Arizona
5951                  Boston-Cambridge-Newton, MA-NH  Massachusetts
5959                                Cedar Rapids, IA           Iowa
6058                Las Vegas-Henderson-Paradise, NV         Nevada
6059                                      Lawton, OK       Oklahoma
6081                                     Modesto, CA     California
6091  Nashville-Davidson--Murfreesboro--Franklin, TN      Tennessee
6108               Palm Bay-Melbourne-Titusville, FL        Florida
6155                   Santa Maria-Santa Barbara, CA     California
6253                 Burlington-South Burlington, VT        Vermont
6505                                York-Hanover, PA   Pennsylvania
6525                                 Columbus, GA-AL        Alabama
6527                 Dallas-Fort Worth-Arlington, TX          Texas
6551        Little Rock-North Little Rock-Conway, AR       Arkansas
6557                   Miami-Miami Beach-Kendall, FL        Florida

      FIPS State Code         lon        lat
5609               51  -77.368316  39.134974
5814               25  -73.245382  42.450085
5819               49 -111.694648  40.296898
5879               20  -95.675158  39.047345
5881                4 -110.974711  32.222607
5951               25  -71.058830  42.360071
5959               19  -91.665623  41.977880
6058               32 -115.146665  36.097195
6059               40  -98.395929  34.603567
6081                6 -120.997001  37.639260
6091               47  -86.580447  36.214401
6108               12  -80.721442  28.263933
6155                6 -120.435719  34.953034
6253               50  -73.212072  44.475882
6505               42  -76.983036  39.800655
6525                1  -84.987709  32.460976
6527               48  -96.920913  32.707875
6551                5  -92.322162  34.729938
6557               12  -80.133611  25.806053
```

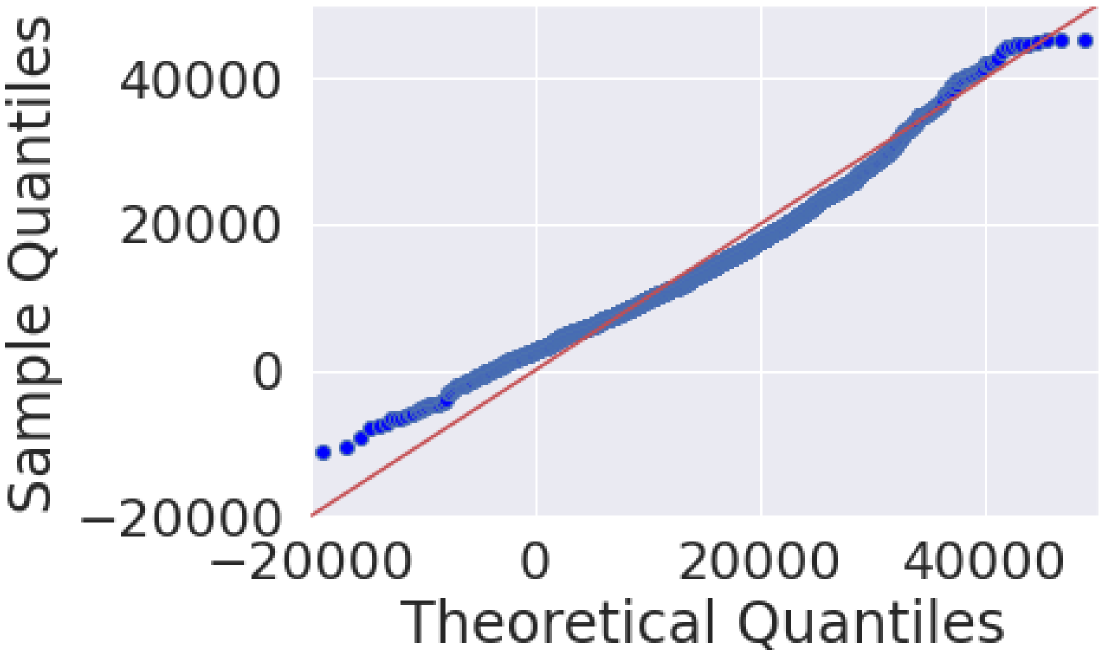
[20]:
```
plt.figure(figsize=(10,15))
sns.set(font_scale=2)
sns.histplot(rmv_outliers)
```

[20]:
```
<AxesSubplot:ylabel='Count'>
```



[21]:
```
sm.qqplot(rmv_outliers,dist=norm(mu,std), line ='45')
pylab.show()
```

```
/home/lennon_mccartney/anaconda3/lib/python3.8/site-
packages/statsmodels/graphics/gofplots.py:993: UserWarning: marker is
redundantly defined by the 'marker' keyword argument and the fmt string "bo" (->
marker='o'). The keyword argument will take precedence.
  ax.plot(x, y, fmt, **plot_style)
```



0.0.9 Repeating the above steps to create QQ plots for all the groups after removing outliers from every group

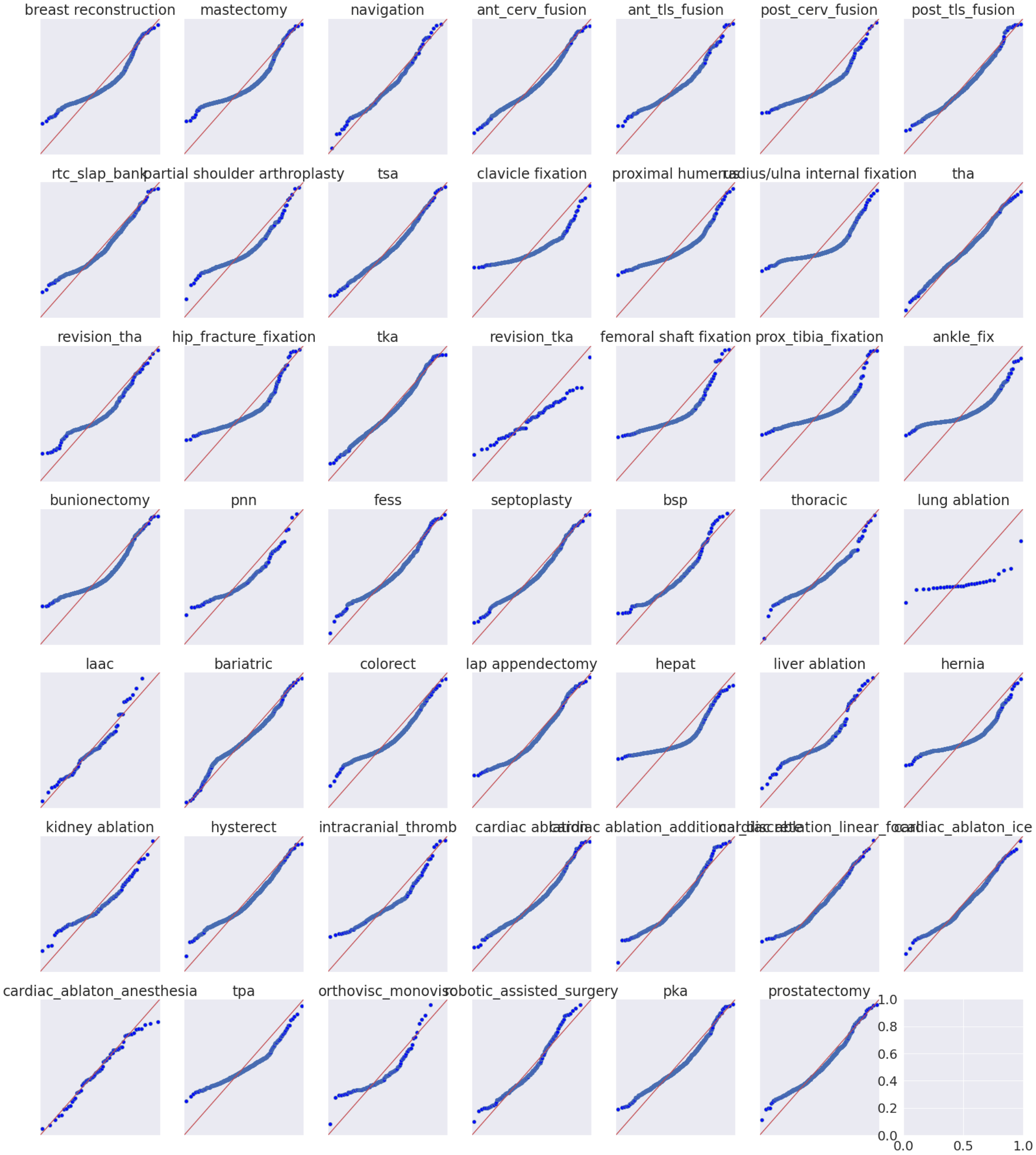
[22]:
```
fig, axs = plt.subplots(7, 7, figsize=(30,35))
gp_list = df_train["group"].unique()
outlier_data = []
figs = []
for i,x in enumerate(gp_list):
    dt_pts = df_train[df_train["group"] == x]["priv_pay_median"] -␣
 ↪df_train[df_train["group"] == x]["mcare_pay_median"]
    gp_stats = dt_pts.describe()
    mu = gp_stats[1]
    std = gp_stats[2]
    rmv_outliers = dt_pts[(dt_pts > mu - 3*std) & (dt_pts < mu + 3*std)].values
    outliers = list(dt_pts[(dt_pts <= mu - 3*std) | (dt_pts >= mu + 3*std)].
 ↪index)
    dt_out = df_train.loc[df_train.apply(lambda x: x['index'] in outliers,␣
 ↪axis=1)]
#     ax.set(xlabel=None)
#     ax.set(ylabel=None)

#     plt.gca().set_title(x)

    sm.qqplot(rmv_outliers,dist=norm(mu,std), line ='45',ax=axs[int(i/7),i%7])
    axs[int(i/7),i%7].get_yaxis().set_visible(False)
    axs[int(i/7),i%7].get_xaxis().set_visible(False)
    axs[int(i/7),i%7].set_title(x)
    outlier_data.append(dt_out)

plt.savefig("qqplots.png")
```

```
/home/lennon_mccartney/anaconda3/lib/python3.8/site-
packages/statsmodels/graphics/gofplots.py:993: UserWarning: marker is
redundantly defined by the 'marker' keyword argument and the fmt string "bo" (->
marker='o'). The keyword argument will take precedence.
  ax.plot(x, y, fmt, **plot_style)
/home/lennon_mccartney/anaconda3/lib/python3.8/site-
packages/statsmodels/graphics/gofplots.py:993: UserWarning: marker is
redundantly defined by the 'marker' keyword argument and the fmt string "bo" (->
marker='o'). The keyword argument will take precedence.
  ax.plot(x, y, fmt, **plot_style)
/home/lennon_mccartney/anaconda3/lib/python3.8/site-
packages/statsmodels/graphics/gofplots.py:993: UserWarning: marker is
redundantly defined by the 'marker' keyword argument and the fmt string "bo" (->
marker='o'). The keyword argument will take precedence.
  ax.plot(x, y, fmt, **plot_style)
/home/lennon_mccartney/anaconda3/lib/python3.8/site-
packages/statsmodels/graphics/gofplots.py:993: UserWarning: marker is
redundantly defined by the 'marker' keyword argument and the fmt string "bo" (->
marker='o'). The keyword argument will take precedence.
  ax.plot(x, y, fmt, **plot_style)
/home/lennon_mccartney/anaconda3/lib/python3.8/site-
packages/statsmodels/graphics/gofplots.py:993: UserWarning: marker is
redundantly defined by the 'marker' keyword argument and the fmt string "bo" (->
marker='o'). The keyword argument will take precedence.
  ax.plot(x, y, fmt, **plot_style)
/home/lennon_mccartney/anaconda3/lib/python3.8/site-
packages/statsmodels/graphics/gofplots.py:993: UserWarning: marker is
redundantly defined by the 'marker' keyword argument and the fmt string "bo" (->
marker='o'). The keyword argument will take precedence.
```


breast reconstruction
mastectomy
navigation
ant_cerv_fusion
ant_tls_fusion
post_cerv_fusion
post_tls_fusion
rtc_slap_bank
partial shoulder arthroplasty
tsa
clavicle fixation
proximal humerus
radius/ulna internal fixation
tha
revision_tha
hip_fracture_fixation
tka
revision_tka
femoral shaft fixation
prox_tibia_fixation
ankle_fix
bunionectomy
pnn
fess
septoplasty
bsp
thoracic
lung ablation
laac
bariatric
colorect
lap appendectomy
hepat
liver ablation
hernia
kidney ablation
hysterect
intracranial_thromb
cardiac ablation
cardiac ablation_additional_discrete
cardiac ablation_linear_focal
cardiac_ablaton_ice
cardiac_ablaton_anesthesia
tpa
orthovisc_monovisc
robotic_assisted_surgery
pka
prostatectomy
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.5
1.0