

# KASUS PENGGUNAAN ENERGI

Disusun oleh :

- Adawia Ananda (2106724883)
- Fernaldy (2106706464)
- Myra Azzahra Putri Syah Indra (2106726844)
- Nadia Sukesi Sianipar (2106700776)
- Najwa Salsabila (2106726812)
- Whitney (2106700946)

# PENDAHULUAN

## Masalah yang akan dibahas

“-----  
Permasalahan prediksi penggunaan energi gedung dari PT Ashrae - American Society of Heating, Refrigerating and Air-Conditioning Engineers  
-----”

## Jumlah pengukuran

“**16**”

## Sumber data

“-----  
<https://www.ashrae.org/about>  
-----”

“-----  
<https://www.kaggle.com/c/ashrae-energy-prediction/data>  
-----”

# Arti atau maksud dari pengukuran

## Tipe data

“ - - - - -

1. building\_id=int
2. meter=int
3. timestamp=object
4. meter\_reading=float64
5. site\_id=int
6. primary\_use=object
7. square\_feet=int
8. year\_built=float
9. floor\_count=float
10. air\_temperature=float
11. cloud\_coverage=float
12. dew\_temperature=float
13. precip\_depth\_1\_hr=float
14. sea\_level\_pressure=float
15. wind\_direction=float
16. wind\_speed=float

- - - - - ”

“ - - - - -

- 1. site\_id & building\_id: id lokasi dan gedung
- 2. primary\_use: Peruntukan Gedung
- 3. square\_feet: Luas bangunan gedung
- 4. year\_built: Tahun pembuatan gedung
- 5. floor\_count: Banyaknya lantai yang ada di gedung.
- 6. meter : Jenis meter reading penggunaan energy gedung.
- 7. timestamp : Waktu saat pengukuran (per-jam)
- 8. meter\_reading: Penggunaan energi.
- 9. air\_temperature: suhu udara
- 10. cloud\_coverage: ukuran berawan
- 11. dew\_temperature: suhu embun
- 12. precip\_depth\_1\_hr: precipitation (banyaknya air dari langit, karena sebab apapun)
- 13. sea\_level\_pressure: Tekanan permukaan laut.
- 14. wind\_direction & wind\_speed: arah dan kecepatan angin

- - - - - ”

# PRE-PROCESSING

```
import numpy as np
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
import gc

building_meta = pd.read_csv('/content/ashrae-energy-
prediction/building_metadata.csv')
train = pd.read_csv('/content/ashrae-energy-prediction/train.csv')
weather_train = pd.read_csv('/content/ashrae-energy-prediction/weather_train.csv')
```

building\_meta

train

weather\_train

# PRE-PROCESSING

**Selanjutnya akan dilakukan MEMORY REDUCTION**

Berdasarkan referensi kode dari

<https://www.kaggle.com/kernels/scriptcontent/3684066/download>

**akan dilakukan pengurangan memori agar tidak memakan memory banyak saat melakukan pre-processing.**

```
[ ] building_meta = reduce_mem_usage(building_meta, use_float16=True)
train = reduce_mem_usage(train, use_float16=True)
weather_train = reduce_mem_usage(weather_train, use_float16=True)

Memory usage of dataframe is 0.07 MB
Memory usage after optimization is: 0.02 MB
Decreased by 73.9%
Memory usage of dataframe is 616.95 MB
Memory usage after optimization is: 173.84 MB
Decreased by 71.8%
Memory usage of dataframe is 9.60 MB
Memory usage after optimization is: 2.59 MB
Decreased by 73.1%
```

# PRE-PROCESSING

#1

Data Integration

#2

Cleaning Data

#3

Saving (preprocessed) Data

# A. DATA INTEGRATION

Menggabungkan database building\_metadata, train, weather\_train menjadi 1 data besar.

#Melakukan join pada train

```
join1=pd.merge(building_meta,train,on='building_id')
join2=pd.merge(weather_train,building_meta,on='site_id')
join=pd.concat([join1,join2])
join
```

	site_id	building_id	primary_use	square_feet	year_built	floor_count	meter	timestamp	meter_reading	air_temperature	cloud_coverage	dew_temperature	precip_depth_1_hr	sea_level_pressure	wind_direction	wind_speed
0	0	0	Education	7432	2008.0	NaN	0.0	2016-01-01 00:00:00	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	0	0	Education	7432	2008.0	NaN	0.0	2016-01-01 01:00:00	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	0	0	Education	7432	2008.0	NaN	0.0	2016-01-01 02:00:00	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	0	0	Education	7432	2008.0	NaN	0.0	2016-01-01 03:00:00	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	0	0	Education	7432	2008.0	NaN	0.0	2016-01-01 04:00:00	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
12676161	15	1444	Entertainment/public assembly	19619	1914.0	NaN	NaN	2016-12-31 23:00:00	NaN	1.700195	NaN	-5.601562	-1.0	1008.5	180.0	8.796875
12676162	15	1445	Education	4298	NaN	NaN	NaN	2016-12-31 23:00:00	NaN	1.700195	NaN	-5.601562	-1.0	1008.5	180.0	8.796875
12676163	15	1446	Entertainment/public assembly	11265	1997.0	NaN	NaN	2016-12-31 23:00:00	NaN	1.700195	NaN	-5.601562	-1.0	1008.5	180.0	8.796875
12676164	15	1447	Lodging/residential	29775	2001.0	NaN	NaN	2016-12-31 23:00:00	NaN	1.700195	NaN	-5.601562	-1.0	1008.5	180.0	8.796875
12676165	15	1448	Office	92271	2001.0	NaN	NaN	2016-12-31 23:00:00	NaN	1.700195	NaN	-5.601562	-1.0	1008.5	180.0	8.796875

Akan dihasilkan tabel 32.892.266 rows × 16 columns

Data yang telah di satukan ini akan di gunakan untuk pre-processing data selanjutnya

## **2. CLEANING DATA**

**Akan melalui beberapa tahap yaitu:**

- 1. Mengoreksi Tipe Variabel**
  - 2. Menelusuri Statistika Deskriptif**
  - 3. Fixing Variabel Type => yaitu memangkas kolom menjadi kolom yang kita butuhkan saja untuk dilakukan analisis lebih lanjut**
  - 4. Removing duplicate data**
  - 5. Menentukan outlier**
  - 6. Remove Outlier ke variabel baru**
  - 7. Melihat missing value dan membuat keputusan apakah missing value akan di isi dengan mean, modus, median atau justru membiarkan missing value tersebut.**
-

## 1. Mengoreksi Tipe Variabel

#Melakukan join pada train

```
pd.DataFrame.info(join)
```

#Tipe data di setiap kolom

#Wajib di periksa apakah tipe datanya sudah tepat?

```
join.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 13022714 entries, 0 to 12676165
Data columns (total 16 columns):
 #   Column           Dtype  
 --- 
 0   site_id          int64  
 1   building_id      int64  
 2   primary_use       object  
 3   square_feet       int64  
 4   year_built        float64 
 5   floor_count       float64 
 6   meter             float64 
 7   timestamp         object  
 8   meter_reading     float64 
 9   air_temperature   float64 
 10  cloud_coverage   float64 
 11  dew_temperature  float64 
 12  precip_depth_1_hr float64 
 13  sea_level_pressure float64 
 14  wind_direction   float64 
 15  wind_speed        float64 
dtypes: float64(11), int64(3), object(2)
memory usage: 1.6+ GB
```

## 2. Statistika Deskriptif

```
join.describe(include='all')
```

	site_id	building_id	primary_use	square_feet	year_built	floor_count	meter	timestamp	meter_reading	air_temperature	cloud_coverage	dew_temperature	precip_depth_1_hr	sea_level_pressure	wind_direction	wind_speed
count	3.289227e+07	3.289227e+07	32892266	3.289227e+07	13973882.0	6618639.0	2.021610e+07	32892266	2.021610e+07	1.267214e+07	7055022.0	1.267024e+07	9994336.0	11687011.0	12014216.0	1.265129e+07
unique	NaN	NaN	16	NaN	NaN	NaN	NaN	8784	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
top	NaN	NaN	Education	NaN	NaN	NaN	NaN	2016-12-27 22:00:00	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
freq	NaN	NaN	NaN	12965865	NaN	NaN	NaN	3819	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
mean	7.581502e+00	7.694347e+02	NaN	1.017398e+05	NaN	NaN	6.624412e-01	NaN	2.042377e+03	NaN	NaN	NaN	NaN	NaN	NaN	NaN
std	5.083742e+00	4.249266e+02	NaN	1.149868e+05	NaN	0.0	9.309921e-01	NaN	1.532176e+05	0.000000e+00	0.0	0.000000e+00	NaN	NaN	NaN	0.000000e+00
min	0.000000e+00	0.000000e+00	NaN	2.830000e+02	1900.0	1.0	0.000000e+00	NaN	0.000000e+00	-2.890625e+01	0.0	-3.500000e+01	-1.0	968.0	0.0	0.000000e+00
25%	3.000000e+00	3.770000e+02	NaN	2.799500e+04	1951.0	1.0	0.000000e+00	NaN	1.830000e+01	8.898438e+00	0.0	1.099609e+00	0.0	1012.0	80.0	2.099609e+00
50%	8.000000e+00	8.210000e+02	NaN	6.640400e+04	1970.0	3.0	0.000000e+00	NaN	7.877500e+01	1.609375e+01	2.0	9.398438e+00	0.0	1016.5	180.0	3.099609e+00
75%	1.300000e+01	1.150000e+03	NaN	1.305770e+05	1994.0	6.0	1.000000e+00	NaN	2.679840e+02	2.390625e+01	4.0	1.609375e+01	0.0	1021.0	280.0	4.601562e+00
max	1.500000e+01	1.448000e+03	NaN	8.750000e+05	2017.0	26.0	3.000000e+00	NaN	2.190470e+07	4.718750e+01	9.0	2.609375e+01	343.0	1046.0	360.0	1.900000e+01

### 3. Fixing Variable Type(s)

#Memangkas kolom yang tidak akan dilakukan analisis

```
join.drop(join.columns[[7,10,11,12,13,14,15]],axis=1,inplace=True)  
join.head()
```

	site_id	building_id	primary_use	square_feet	year_built	floor_count	meter	meter_reading	air_temperature
0	0	0	Education	7432	2008.0	NaN	0.0	0.0	NaN
1	0	0	Education	7432	2008.0	NaN	0.0	0.0	NaN
2	0	0	Education	7432	2008.0	NaN	0.0	0.0	NaN
3	0	0	Education	7432	2008.0	NaN	0.0	0.0	NaN
4	0	0	Education	7432	2008.0	NaN	0.0	0.0	NaN

### 4. Removing Duplicate Data

#Mencari tahu berapa banyak data yang duplikat

```
[ ] #Mencari tahu berapa banyak data yang duplikat  
join.duplicated().sum()  
#Terlihat terdapat data yang duplikat. Kita akan membuang data ini
```

24589426

```
[ ] join.drop_duplicates(keep="first", inplace=True)  
print(join.shape)  
print("jumlah data yang duplikat", join.duplicated().sum())  
#Data berhasil di bersihkan dari data duplikat
```

(8302840, 9)  
jumlah data yang duplikat 0

# 5. Menentukan Outliers

#Mencari nilai MAX

```
join.max()
```

```
site_id          15  
building_id      1448  
primary_use     Warehouse/storage  
square_feet      875000  
year_built       2017.0  
floor_count      26.0  
meter            3.0  
meter_reading    7644020.0  
air_temperature   47.2  
dtype: object
```

#Mencari nilai MIN

```
join.min()
```

```
site_id           0  
building_id       0  
primary_use      Education  
square_feet      283  
year_built       1900.0  
floor_count      1.0  
meter             0.0  
meter_reading    0.0  
air_temperature   -28.9  
dtype: object
```

#Mencari nilai MEAN

```
join.mean()
```

```
site_id           8.835581  
building_id      863.827343  
square_feet      123135.530473  
year_built        NaN  
floor_count       NaN  
meter             0.667881  
meter_reading    4587.591797  
air_temperature   NaN  
dtype: float64
```

Terlihat terdapat outlier pada square\_feet dan meter\_reading karena nilai min dan max nya sangat jauh dari nilai rata-rata

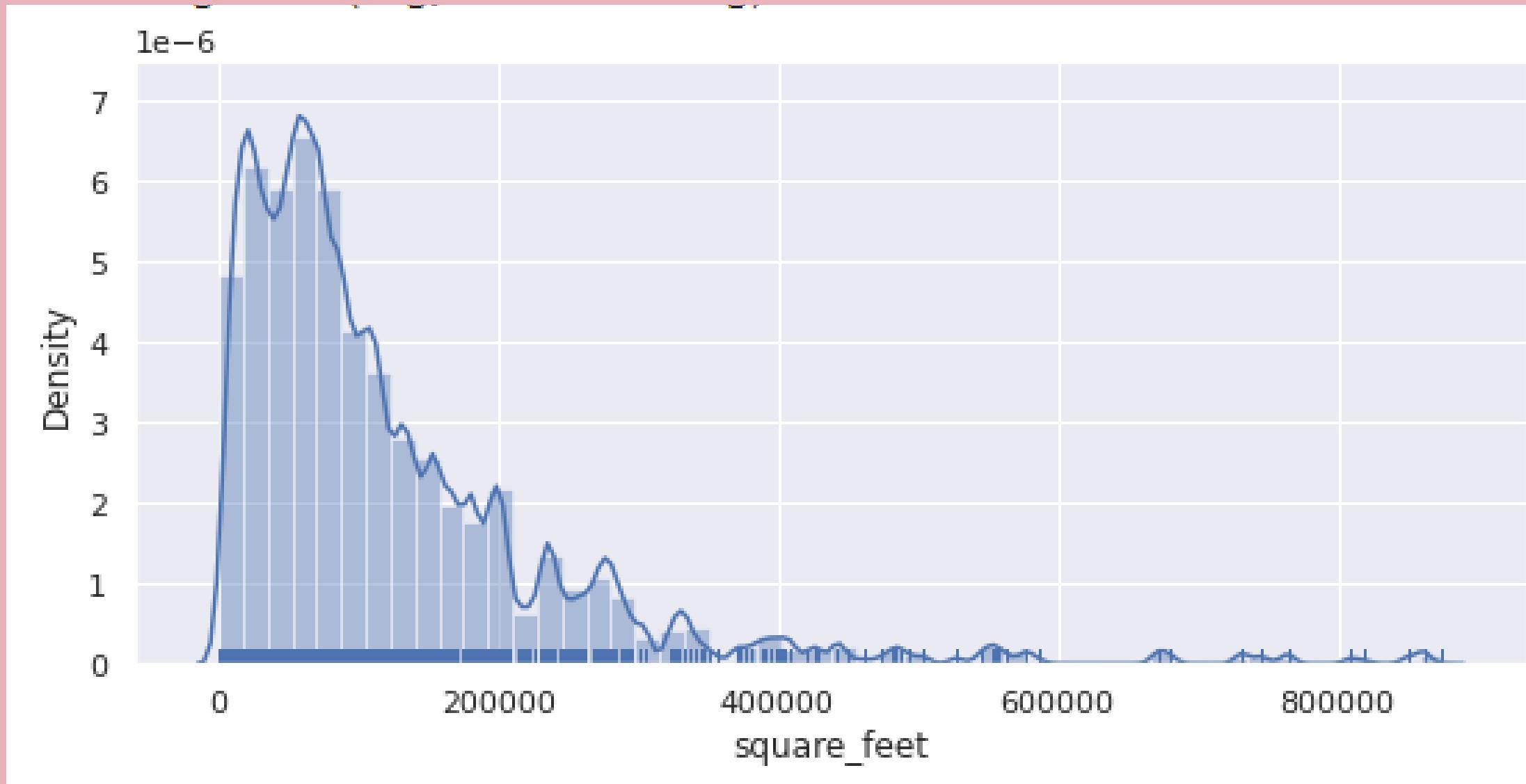
## 6.Remove Outliers ke variabel baru

Akan ditampilkan grafik outlier sebelum dan sesudah outlier diremove ke variabel baru.

**#GRAFIK OUTLIER SEBELUM OUTLIER DIREMOVE KE VARIABEL BARU (SQUARE\_FEET)**

**#Melihat data outlier yang terdapat pada variabel "square\_feet"**

```
plt.style.use('bmh'); sns.set() #style visualisasi  
fig, axes = plt.subplots(1, 1, figsize=(9, 4))  
p = sns.distplot(join['square_feet'], kde=True, rug=True)
```



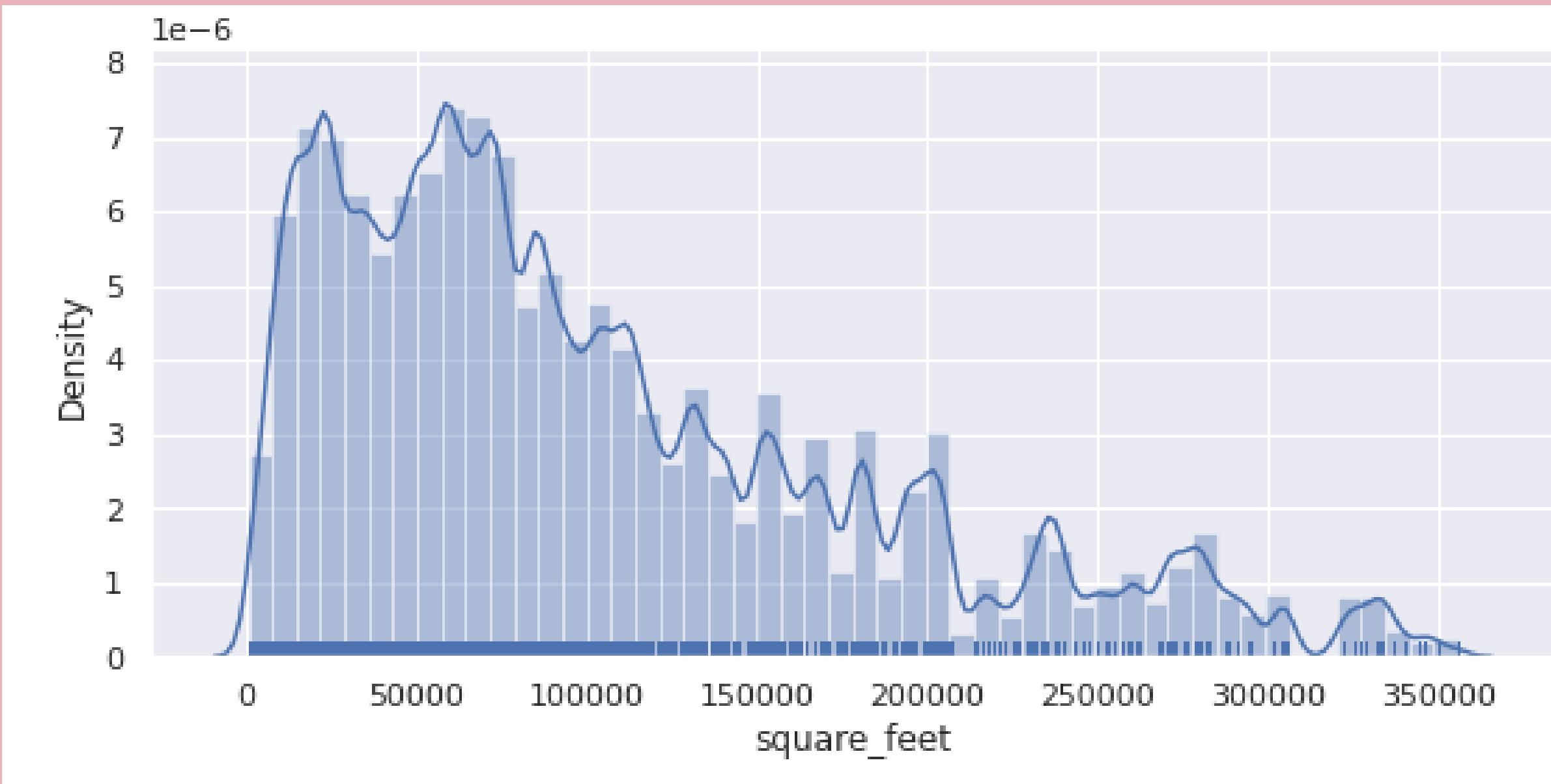
## #GRAFIK OUTLIER SESUDAH OUTLIER DIREMOVE KE VARIABEL BARU(SQUARE\_FEET)

#Data square\_feet tanpa outlier (remove outlier) tetapi di simpan di variabel baru

# Misal dengan asumsi data berdistribusi normal & menggunakan 95% confidence interval di sekitar variabel "square\_feet"

```
normal_data = abs(join.square_feet - join.square_feet.mean())<=(2*join.square_feet.std())
join2 = join[normal_data]
fig, axes = plt.subplots(1, 1, figsize=(9, 4))
p = sns.distplot(join2['square_feet'], kde=True, rug=True)
```

# Perhatikan disini sengaja data yang telah di remove outliernya disimpan dalam variabel baru "join2"

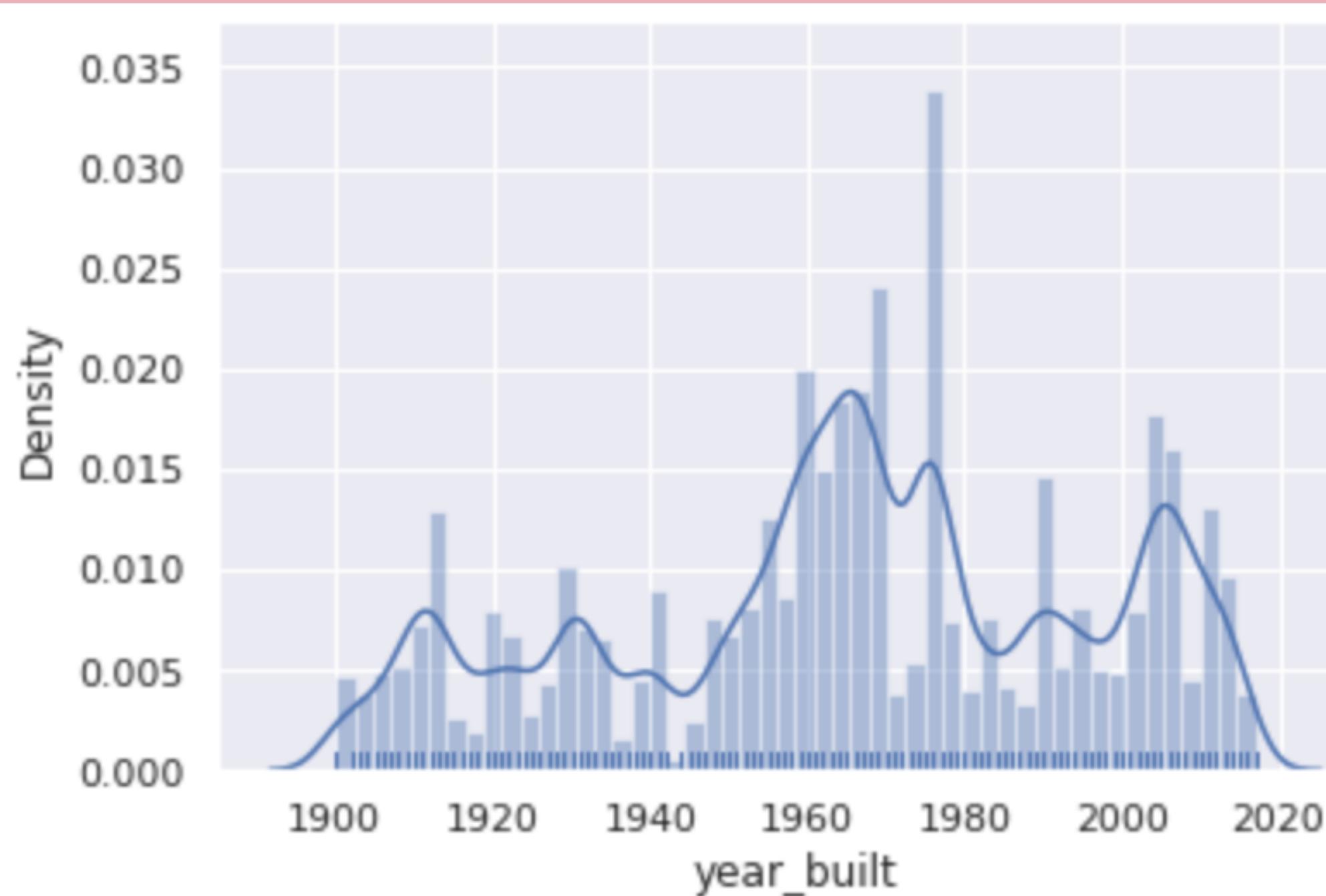


## #GRAFIK OUTLIER SEBELUM OUTLIER DIREMOVE KE VARIABEL BARU(YEAR\_BUILT)

#Melihat data outlier yang terdapat pada variabel "year\_built"

```
plt.style.use('bmh'); sns.set() #style visualisasi
```

```
p = sns.distplot(join['year_built'], kde=True, rug=True)
```



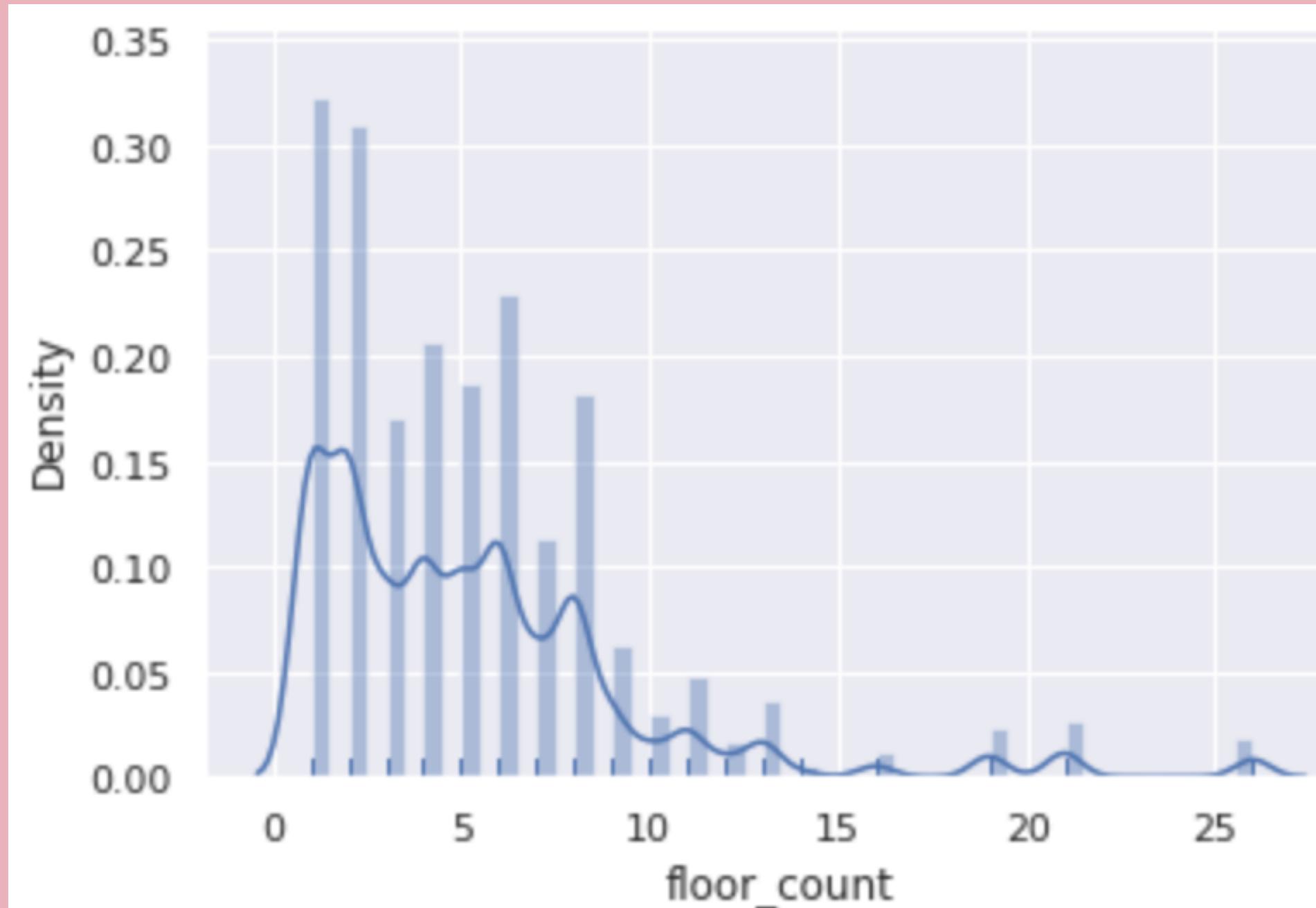
Terlihat dari grafik bahwa 'year\_built' tidak memiliki outlier.

## #GRAFIK OUTLIER SEBELUM OUTLIER DIREMOVE KE VARIABEL BARU(FLOOR\_COUNT)

#Melihat data outlier yang terdapat pada variabel "floor\_count"

```
plt.style.use('bmh'); sns.set() #style visualisasi
```

```
p = sns.distplot(join['floor_count'], kde=True, rug=True)
```

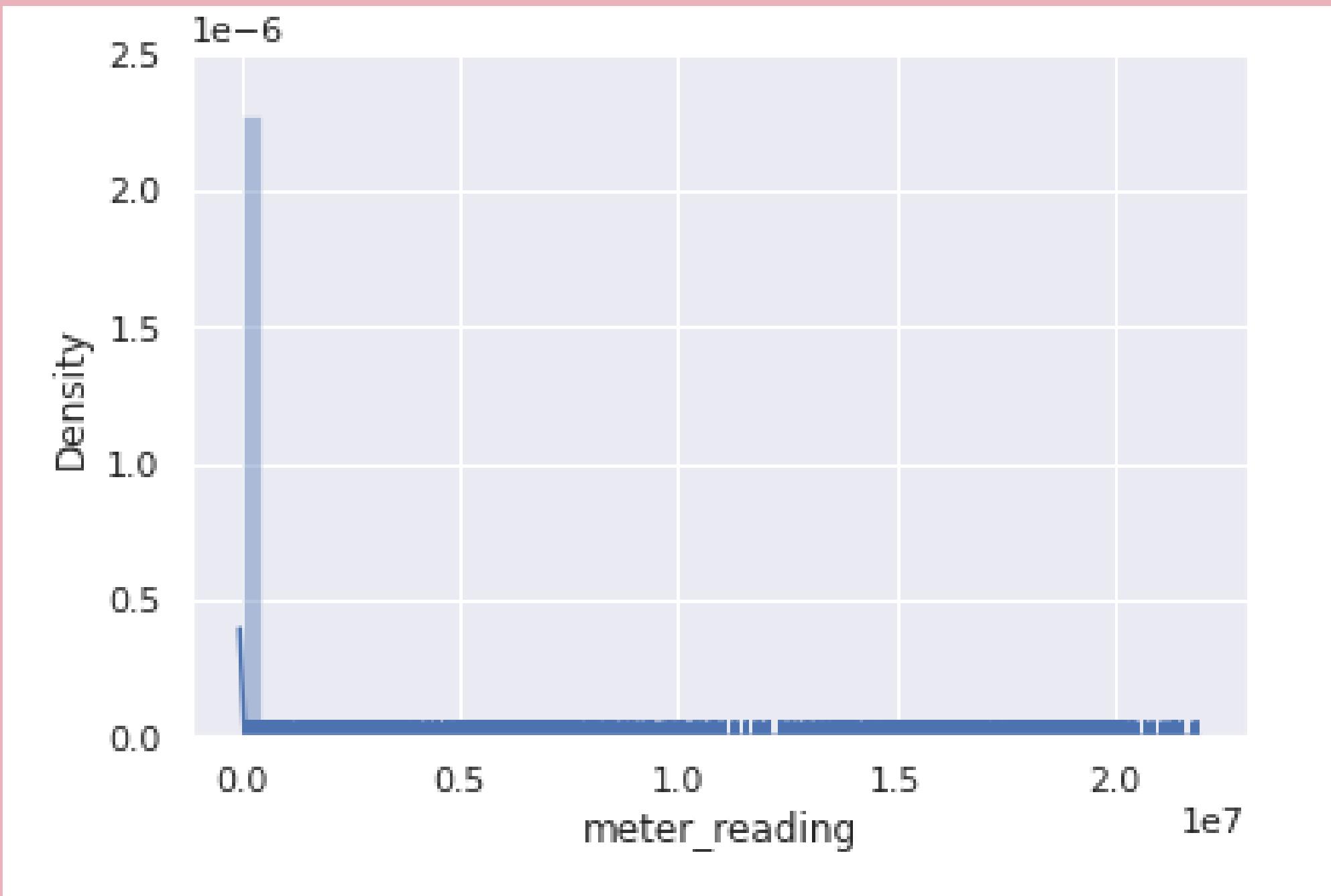


Terlihat dari grafik bahwa 'floor\_count' tidak memiliki outlier.

## #GRAFIK OUTLIER SEBELUM OUTLIER DIREMOVE KE VARIABEL BARU(METER\_READING)

#Melihat data outlier yang terdapat pada variabel "meter\_reading"

```
plt.style.use('bmh'); sns.set() #style visualisasi  
  
p = sns.distplot(join['meter_reading'], kde=True, rug=True)
```



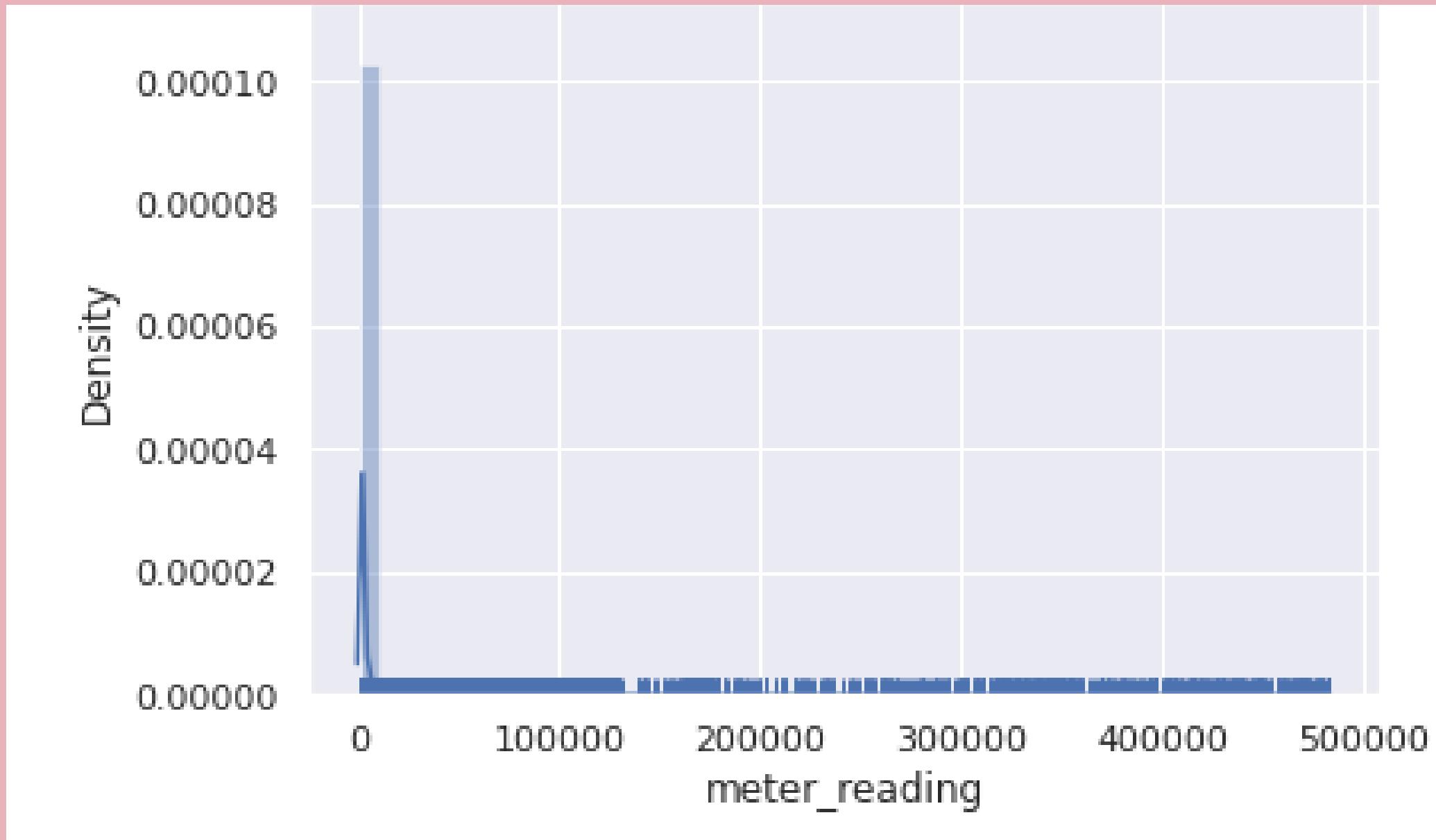
## #GRAFIK OUTLIER SESUDAH OUTLIER DIREMOVE KE VARIABEL BARU(METER\_READING)

#Data meter\_reading tanpa outlier (remove) tetapi di simpan di variabel baru

# Misal dengan asumsi data berdistribusi normal & menggunakan 95% confidence interval di sekitar variabel "meter\_reading"

```
normal_data4 = abs(join.meter_reading - join.meter_reading.mean())<=(2*join.meter_reading.std()) # mu-2s<x<mu+2s  
join5 = join[normal_data4]  
p = sns.distplot(join5['meter_reading'], kde=True, rug=True)
```

# Perhatikan disini sengaja data yang telah di remove outliernya disimpan dalam variabel baru "join5"



# 7. Menemukan Missing Value

#Mencari tau jumlah missing value tiap variabel

```
join.isna().sum()
```

```
site_id          0  
building_id      0  
primary_use      0  
square_feet       0  
year_built    4810785  
floor_count     7413705  
meter           160287  
meter_reading    160287  
air_temperature  8143731  
dtype: int64
```

#Mencari persentase missing value

```
(join.isnull().sum()/len(join)).to_frame('persentase missing')
```

	persentase missing
site_id	0.000000
building_id	0.000000
primary_use	0.000000
square_feet	0.000000
year_built	0.579414
floor_count	0.892912
meter	0.019305
meter_reading	0.019305
air_temperature	0.980837

#Menentukan apa yang harus dilakukan kepada missing values

Meskipun missing values yang terdapat dalam variable-variable tersebut lumayan banyak, kami memutuskan untuk tidak drop variable tersebut karena penting untuk analisis kami.

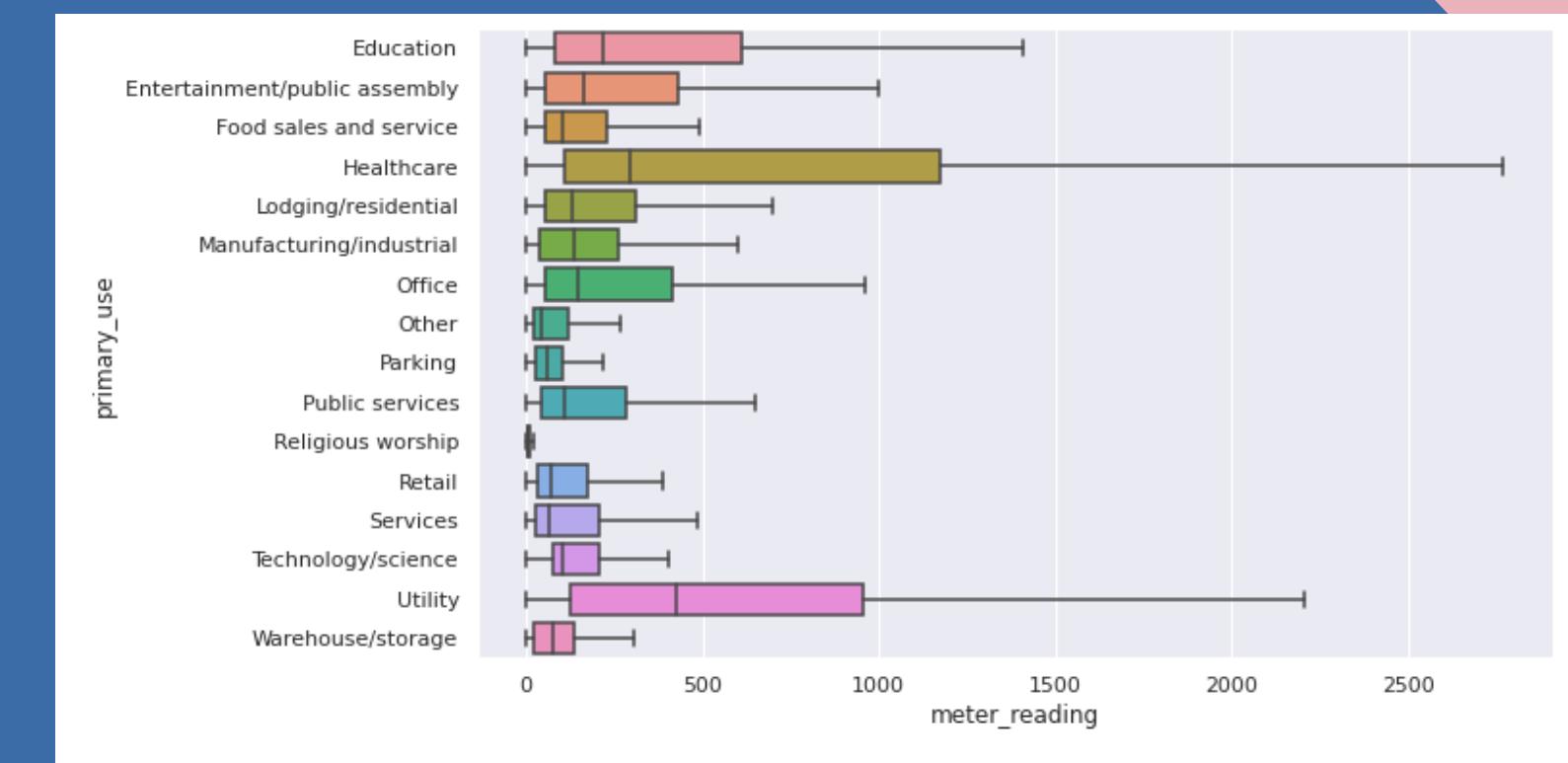
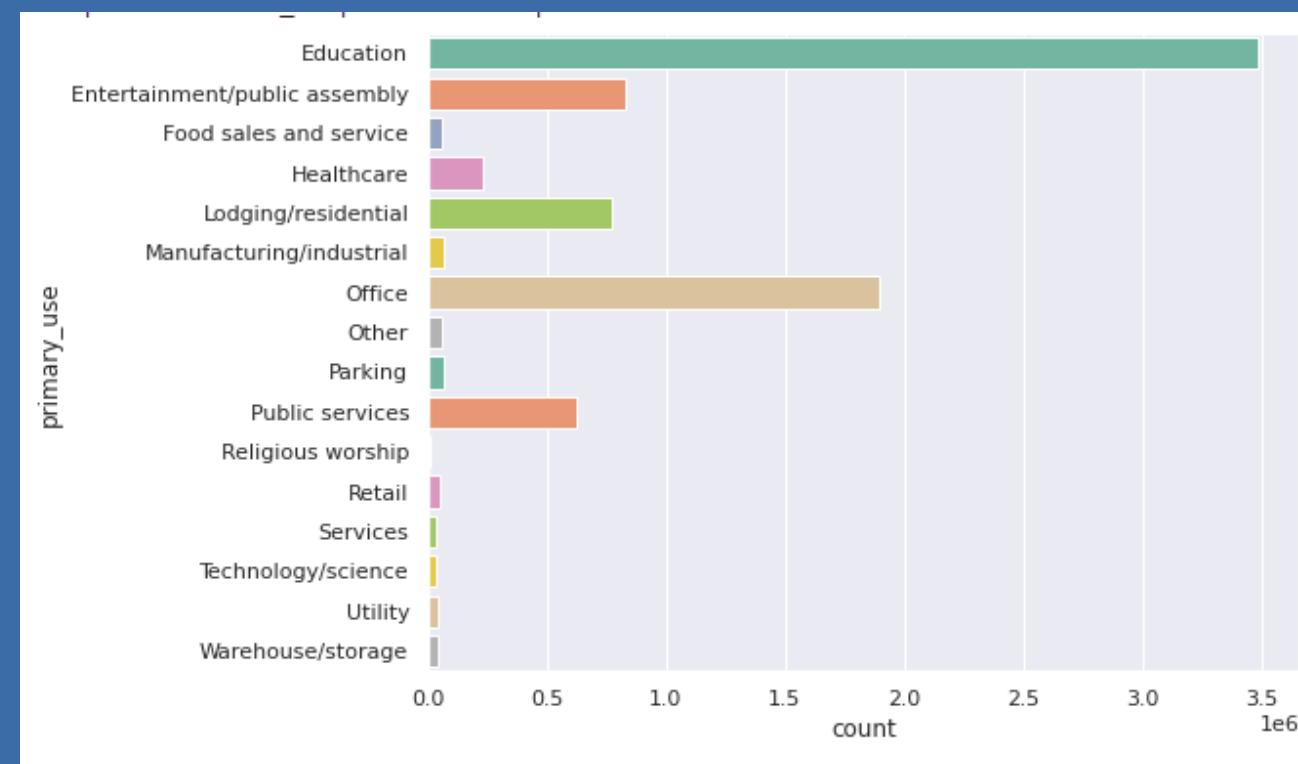
Selain itu, kami juga memutuskan untuk tidak mengimputasi karena persentase missing values yang cukup besar (di atas 10%). Dikhawatirkan apabila dilakukan imputasi, data yang dihasilkan tidak akurat

## C. SAVING (PREPROCESSED) DATA

```
# Saving the preprocessed Data for future use/analysis  
join.to_csv("join.csv", encoding='utf8', index=False)
```

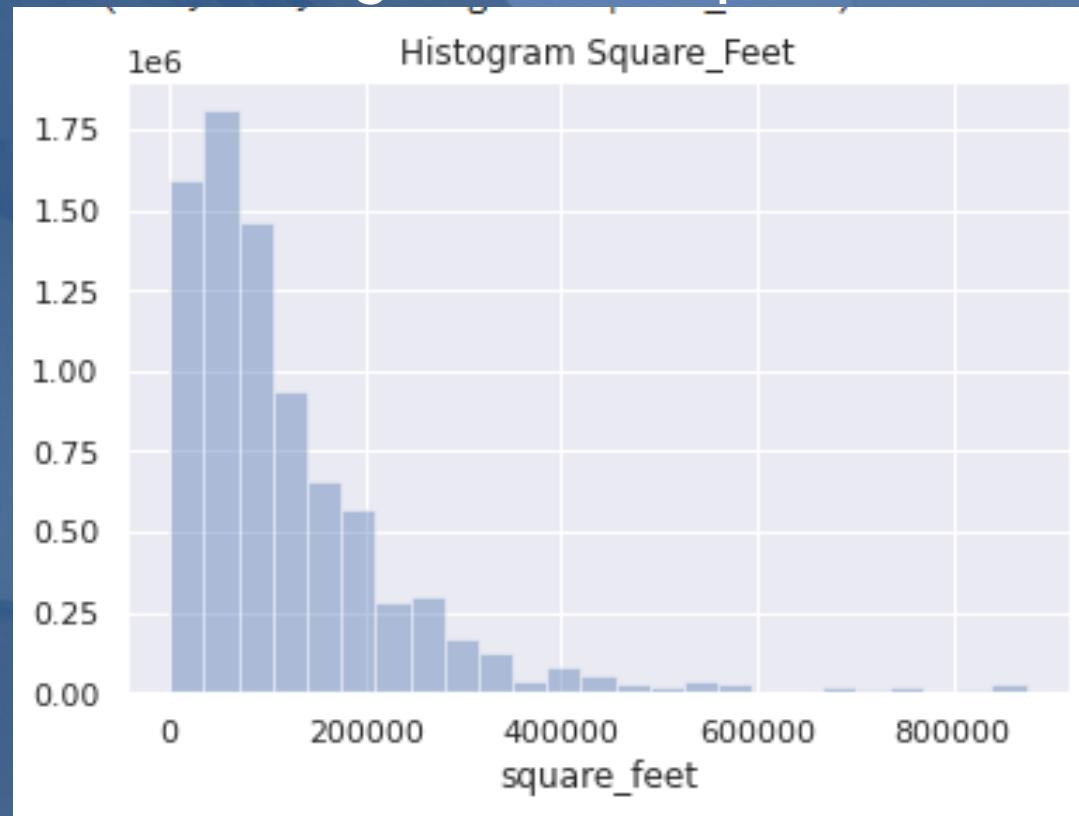
# VISUALISASI DATA

- Visualisasi kegunaan gedung di lingkungan tersebut
- Hubungan meter\_reading dengan primary\_use

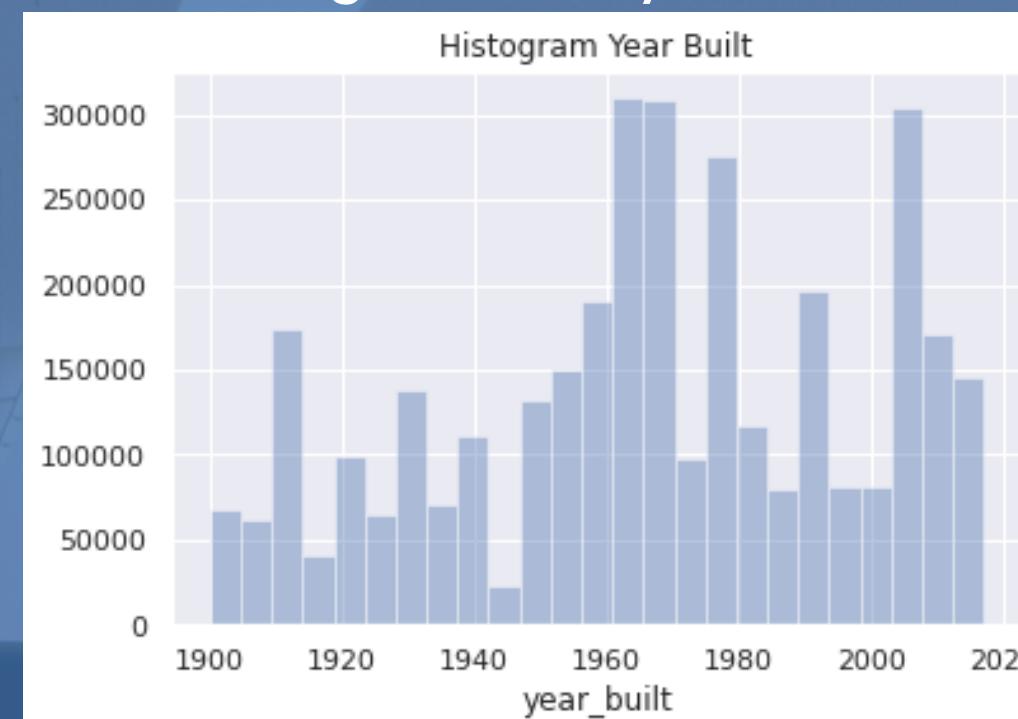


Dari kedua visualisasi tersebut, dapat dilihat bahwa sektor **Healthcare** dan **Utility** menggunakan **energi terbanyak**, sementara sektor **Religious Worship** menggunakan **energi paling sedikit**

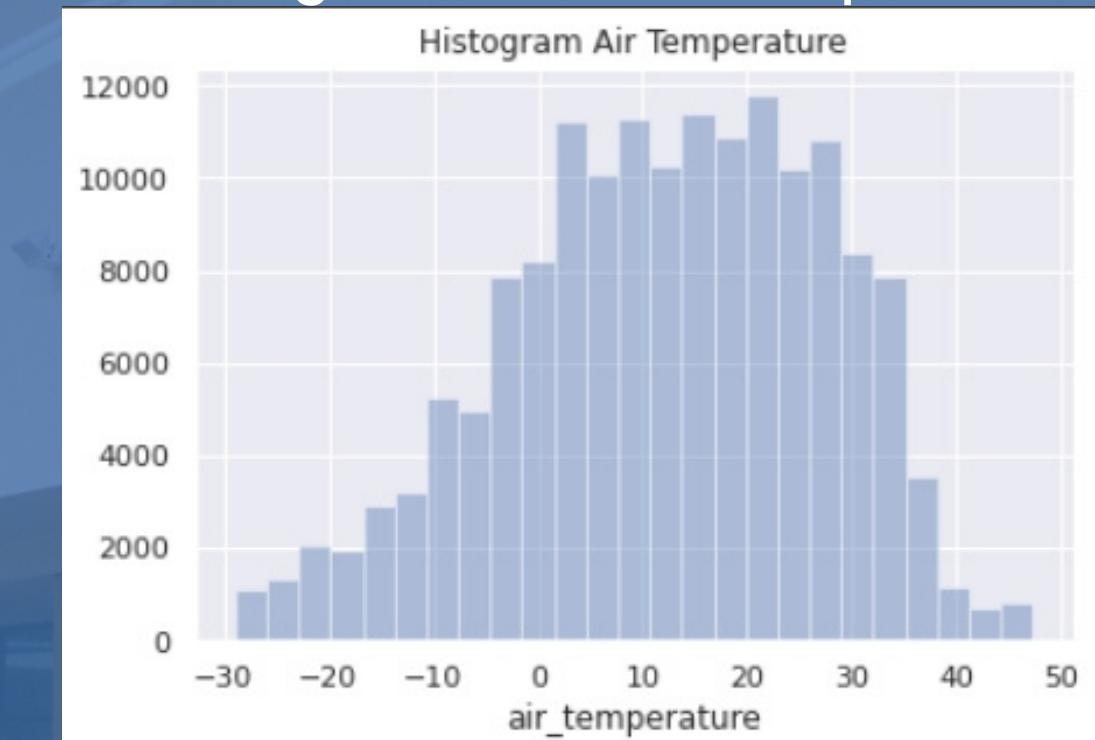
- Histogram dari square\_feet



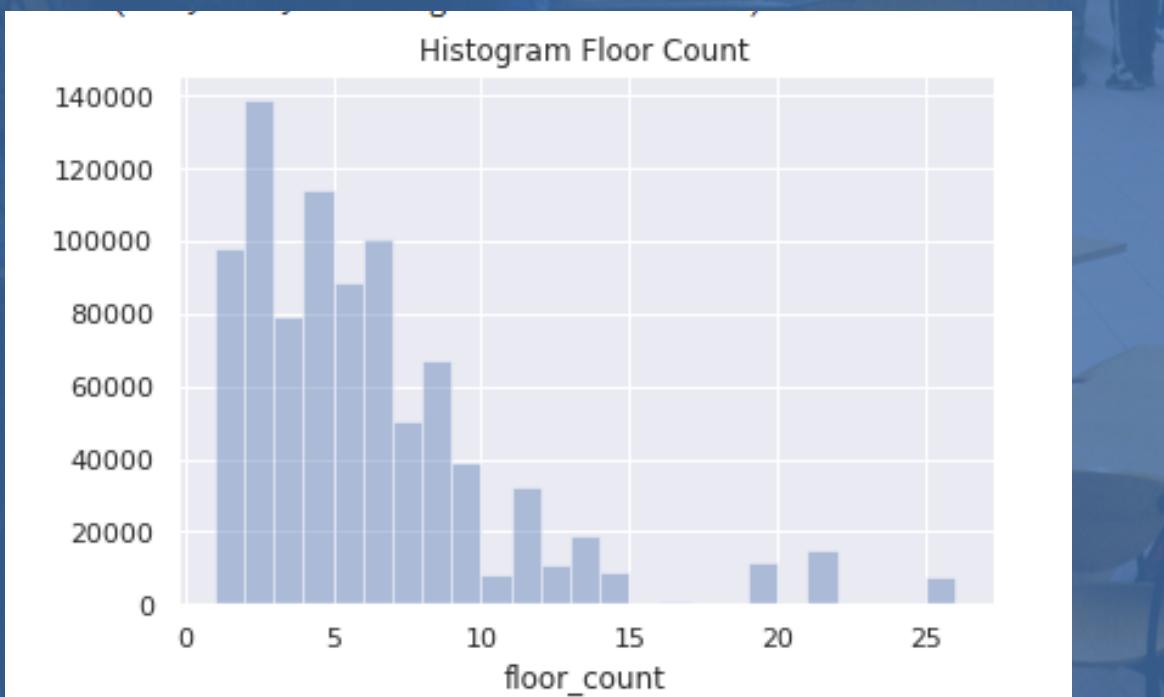
- Histogram dari year\_built



- Histogram dari air\_temperature



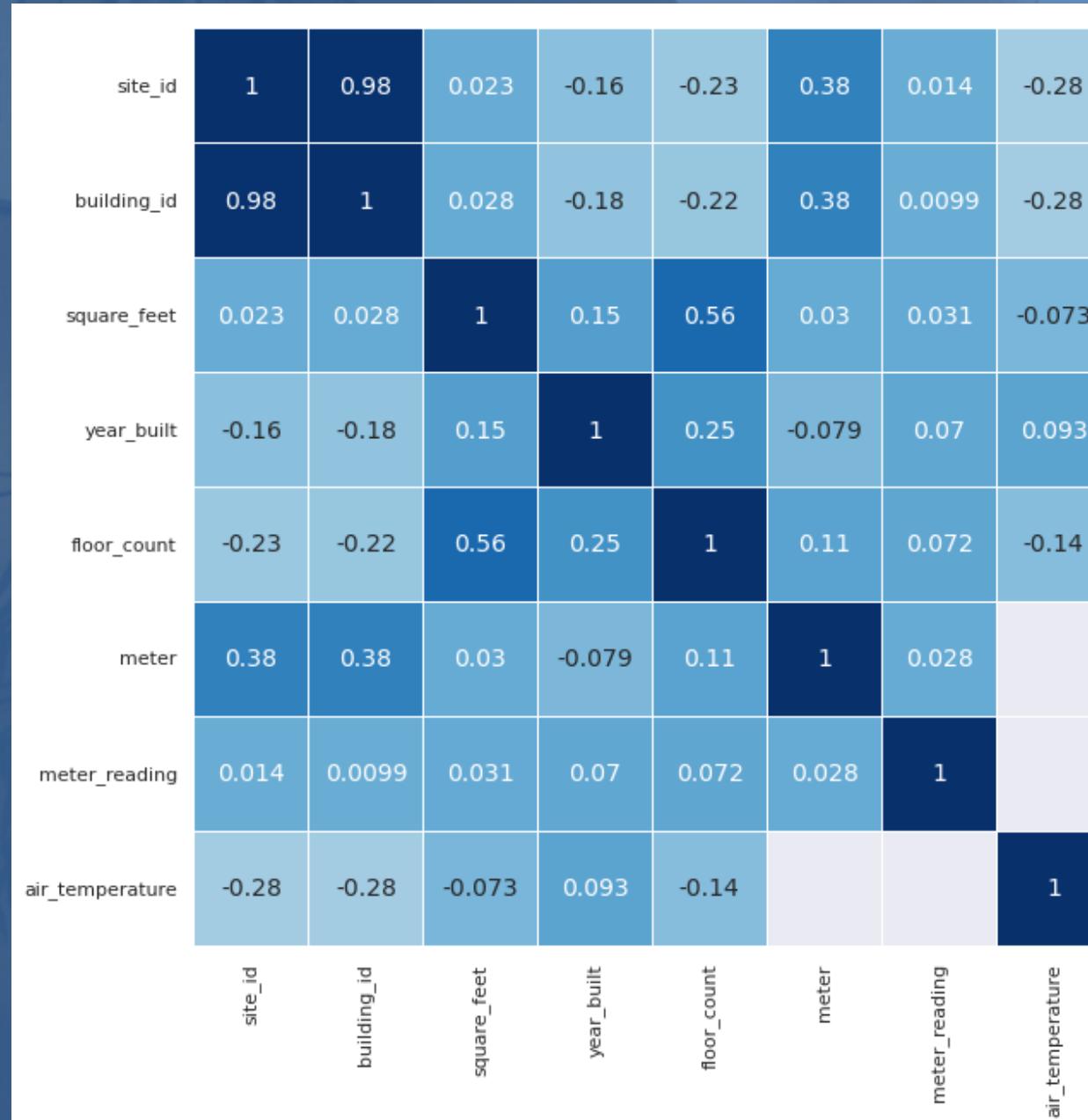
- Histogram dari floor\_count



Dari visualisasi tersebut, dapat dilihat bahwa mayoritas gedung :

- memiliki luas kurang dari 200000 square feet
- dibangun pada tahun 1960–1980
- memiliki suhu udara 10–20 derajat celcius
- memiliki jumlah lantai 1 sampai 5

- Heat map korelasi antar energi dengan variabel-variabel lainnya



- Tabel korelasi antar energi dengan variabel-variabel lainnya

	site_id	building_id	square_feet	year_built	floor_count	meter	meter_reading	air_temperature
site_id	1.000000	0.983414	0.023151	-0.155731	-0.228668	0.379930	0.014075	-0.277513
building_id	0.983414	1.000000	0.028087	-0.182125	-0.217098	0.380541	0.009930	-0.275911
square_feet	0.023151	0.028087	1.000000	0.146858	0.564485	0.029967	0.031280	-0.073478
year_built	-0.155731	-0.182125	0.146858	1.000000	0.246295	-0.078951	0.069819	0.092763
floor_count	-0.228668	-0.217098	0.564485	0.246295	1.000000	0.105812	0.071609	-0.139093
meter	0.379930	0.380541	0.029967	-0.078951	0.105812	1.000000	0.027716	NaN
meter_reading	0.014075	0.009930	0.031280	0.069819	0.071609	0.027716	1.000000	NaN
air_temperature	-0.277513	-0.275911	-0.073478	0.092763	-0.139093	NaN	NaN	1.000000

dari visualisasi tersebut, dapat dilihat bahwa korelasi antara energi yang digunakan dengan variabel lainnya cenderung rendah yaitu  $-0.2 < r < 0.2$

# **ANALISIS DASAR STATISTIKA**

**Analisis yang akan dilakukan adalah :**



Analisis untuk mengetahui korelasi waktu dengan besar penggunaan energi.



Analisis untuk mengetahui korelasi luas bangunan dengan besar penggunaan energi.



3

Analisis untuk mengetahui korelasi jumlah lantai dengan besar penggunaan energi.



4

Analisis untuk mengetahui korelasi suhu udara dengan besar penggunaan energi.



5

Mencari nilai maksimum meter\_reading untuk mengetahui bangunan dengan building\_id mana dan site\_id mana yang sebenarnya(pada fakta nya) menghabiskan energi paling banyak.



6

Membandingkan semua hasil pada poin 1,2,3,4 terhadap poin 5. Lalu, tarik kesimpulan

# 1. ANALISIS YEAR\_BUILT

```
join.min()
```

```
site_id          0  
building_id      0  
primary_use     Education  
square_feet      283  
year_built      1900.0  
floor_count      1.0  
meter            0.0  
meter_reading    0.0  
air_temperature -28.9  
dtype: object
```

Terlihat bahwa nilai minimum dari year\_built yaitu 1900. Artinya bangunan terlama di bangun sejak pada tahun 1900.

```
join[join['year_built']<1901].head(10)
```

	site_id	building_id	primary_use	square_feet	year_built	floor_count	meter	meter_reading	air_temperature
19620	1	124	Education	38319	1900.0	6.0	0.0	9.3	NaN
19621	1	124	Education	38319	1900.0	6.0	0.0	17.9	NaN
19622	1	124	Education	38319	1900.0	6.0	0.0	18.3	NaN
19624	1	124	Education	38319	1900.0	6.0	0.0	16.6	NaN
19625	1	124	Education	38319	1900.0	6.0	0.0	15.4	NaN
19626	1	124	Education	38319	1900.0	6.0	0.0	17.0	NaN
19627	1	124	Education	38319	1900.0	6.0	0.0	16.1	NaN
19628	1	124	Education	38319	1900.0	6.0	0.0	38.2	NaN
19629	1	124	Education	38319	1900.0	6.0	0.0	41.8	NaN
19631	1	124	Education	38319	1900.0	6.0	0.0	43.7	NaN

Dari tabel 10 nilai teratas di atas,dapat dilihat bahwa building dengan building\_id 124 dan site\_id 1 memiliki tahun berdiri paling lama yaitu sejak 1900. Dan semuanya digunakan untuk keperluan education.

## 2. ANALISIS SQUARE\_FEET

```
join.max()
```

```
site_id          15
building_id      1448
primary_use      Warehouse/storage
square_feet      875000
year_built       2017.0
floor_count      26.0
meter            3.0
meter_reading    7644020.0
air_temperature   47.2
dtype: object
```

```
join[join['square_feet']>874999].head(10)
```

	site_id	building_id	primary_use	square_feet	year_built	floor_count	meter	meter_reading	air_temperature
160700	8	869	Entertainment/public assembly	875000	NaN	1.0	0.0	1517.04	NaN
160724	8	869	Entertainment/public assembly	875000	NaN	1.0	0.0	1400.38	NaN
160748	8	869	Entertainment/public assembly	875000	NaN	1.0	0.0	1417.04	NaN
160772	8	869	Entertainment/public assembly	875000	NaN	1.0	0.0	1800.38	NaN
160796	8	869	Entertainment/public assembly	875000	NaN	1.0	0.0	1017.04	NaN
160820	8	869	Entertainment/public assembly	875000	NaN	1.0	0.0	1350.38	NaN
160844	8	869	Entertainment/public assembly	875000	NaN	1.0	0.0	1833.67	NaN
7054789	8	869	Entertainment/public assembly	875000	NaN	1.0	NaN	NaN	25.0
7054859	8	869	Entertainment/public assembly	875000	NaN	1.0	NaN	NaN	24.4
7054929	8	869	Entertainment/public assembly	875000	NaN	1.0	NaN	NaN	22.8

Terlihat bahwa luas bangunan paling maksimal adalah 875000

Dari tabel 10 nilai teratas di atas, dapat dilihat bahwa building dengan building\_id 869 dan site\_id 8 memiliki luas paling besar dan semuanya digunakan untuk keperluan entertainment/public assembly.

### 3. ANALISIS FLOOR\_COUNT

join.max()	
site_id	15
building_id	1448
primary_use	Warehouse/storage
square_feet	875000
year_built	2017.0
floor_count	26.0
meter	3.0
meter_reading	7644020.0
air_temperature	47.2
dtype: object	

Terlihat bahwa jumlah lantai paling maksimal adalah 26

join[join['floor_count']>25].head(10)										
	site_id	building_id	primary_use	square_feet	year_built	floor_count	meter	meter_reading	air_temperature	
148878	7	799	Education	527431	1976.0	26.0	0.0	0.00	NaN	
148879	7	799	Education	527431	1976.0	26.0	1.0	0.00	NaN	
148880	7	799	Education	527431	1976.0	26.0	2.0	1562.77	NaN	
148883	7	799	Education	527431	1976.0	26.0	2.0	1709.02	NaN	
148886	7	799	Education	527431	1976.0	26.0	2.0	1620.38	NaN	
148889	7	799	Education	527431	1976.0	26.0	2.0	1667.05	NaN	
148892	7	799	Education	527431	1976.0	26.0	2.0	1501.94	NaN	
148895	7	799	Education	527431	1976.0	26.0	2.0	2741.72	NaN	
148898	7	799	Education	527431	1976.0	26.0	2.0	3758.89	NaN	
148901	7	799	Education	527431	1976.0	26.0	2.0	3658.25	NaN	

Dari tabel 10 nilai teratas di atas, dapat dilihat bahwa building dengan building\_id 799 dan site\_id 7 memiliki jumlah lantai paling besar dan semuanya digunakan untuk keperluan education.

# 4. ANALISIS AIR\_TEMPERATURE

```
join.max()
```

```
site_id          15
building_id      1448
primary_use      Warehouse/storage
square_feet      875000
year_built       2017.0
floor_count      26.0
meter            3.0
meter_reading    7644020.0
air_temperature   47.2
dtype: object
```

Terlihat bahwa suhu udara paling maksimal adalah 47.2 derajat celcius.

```
join[join['air_temperature']>47].head(1)
```

```
site_id  building_id  primary_use  square_feet  year_built  floor_count  meter  meter_reading  air_temperature
1923003        2  Public services     43681    2007.0        NaN      NaN        NaN           47.2
```

Dari tabel nilai teratas pertama di atas,dapat dilihat bahwa building dengan building\_id 156 dan site\_id 2 memiliki suhu udara paling tinggi dan digunakan untuk keperluan public services.

# 4. ANALISIS AIR\_TEMPERATURE

```
join.min()

site_id          0
building_id      0
primary_use      Education
square_feet      283
year_built      1900.0
floor_count      1.0
meter            0.0
meter_reading    0.0
air_temperature -28.9
dtype: object
```

```
join[join['air_temperature']<-28.8].head(1)

  site_id  building_id  primary_use  square_feet  year_built  floor_count  meter  meter_reading  air_temperature
1 10683182        13       Office      1069      20864        NaN      NaN      NaN      NaN      -28.9
```

Terlihat bahwa suhu paling rendah adalah -28.9 derajat celcius

Dari tabel nilai teratas pertama di atas, dapat dilihat bahwa building dengan building\_id 1069 dan site\_id 13 memiliki suhu udara paling rendah dan digunakan untuk keperluan office.

# 5. BANGUNAN PENGGUNA ENERGI TEBESAR

join.max()

```
site_id      1.500000e+01
building_id   1.448000e+03
square_feet   8.750000e+05
year_built    2.017000e+03
floor_count   2.600000e+01
meter         3.000000e+00
meter_reading 2.190470e+07
air_temperature 4.718750e+01
dtype: float64
```

join[join['meter\_reading']>2190469].head(10)

	site_id	building_id	primary_use	square_feet	year_built	floor_count	meter	meter_reading	air_temperature
13749500	13	1099	Education	332884	NaN	NaN	2.0	3241630.0	NaN
13749518	13	1099	Education	332884	NaN	NaN	2.0	3554740.0	NaN
13749520	13	1099	Education	332884	NaN	NaN	2.0	3341210.0	NaN
13749522	13	1099	Education	332884	NaN	NaN	2.0	2695780.0	NaN
13749524	13	1099	Education	332884	NaN	NaN	2.0	2608420.0	NaN
13749526	13	1099	Education	332884	NaN	NaN	2.0	2583650.0	NaN
13749528	13	1099	Education	332884	NaN	NaN	2.0	2670430.0	NaN
13749530	13	1099	Education	332884	NaN	NaN	2.0	2769430.0	NaN
13749532	13	1099	Education	332884	NaN	NaN	2.0	3009500.0	NaN
13749534	13	1099	Education	332884	NaN	NaN	2.0	2379820.0	NaN

Dari hasil di atas dapat bahwa penggunaan energi paling maksimum(meter\_reading) dihasilkan sebesar 2190470

Dari hasil tabel penggunaan energi maksimum yang sebenarnya di atas terlihat bahwa pada faktanya gedung yang menghabiskan energi paling banyak sebesar 2190470 yaitu gedung dengan building\_id 1099 dan site\_id 13 dan digunakan untuk Education.

# KESIMPULAN

**Dari hasil analisis diperoleh 5 kesimpulan, yaitu:**

- Gedung yang menggunakan energi terbanyak berdasarkan year\_built (building\_id 124) berbeda dengan gedung yang sebenarnya menggunakan energi terbesar (building\_id 1099)
- Bangunan paling lama tidak menjamin bangunan itu akan menggunakan energi paling besar. Maka, tidak ada korelasi penggunaan energi terhadap waktu secara signifikan.
- Namun dapat dilihat, primary\_use kedua hasil itu sama-sama untuk keperluan pendidikan.

- Gedung yang menggunakan energi terbanyak berdasarkan square\_feet (building\_id 869) berbeda dengan gedung yang sebenarnya menggunakan energi terbesar (building\_id 1099)
- Bangunan dengan luas paling besar tidak menjamin bangunan itu akan menggunakan energi paling besar. Tidak ada korelasi penggunaan energi terhadap luas bangunan secara signifikan
- Selain itu dapat dilihat, primary\_use kedua hasil itu berbeda yaitu satu digunakan untuk keperluan pendidikan dan satu nya lagi untuk keperluan entertainment/public assembly.

- Gedung yang menggunakan energi terbanyak berdasarkan floor\_count (building\_id 799) berbeda dengan gedung yang sebenarnya menggunakan energi terbesar (building\_id 1099)
- Bangunan dengan jumlah lantai paling banyak tidak menjamin bangunan itu akan menggunakan energi paling besar. Maka, tidak ada korelasi penggunaan energi terhadap jumlah lantai secara signifikan.
- Namun dapat dilihat, primary\_use kedua hasil itu sama yaitu sama-sama digunakan untuk keperluan pendidikan.

- Gedung yang menggunakan energi terbanyak berdasarkan suhu udara maksimum (building\_id 156) dan suhu udara minimum (building\_id 1069) berbeda dengan gedung yang sebenarnya menggunakan energi terbesar (building\_id 1099)
- Hal ini menunjukkan suhu udara paling tertinggi maupun terendah tidak menjamin suatu bangunan akan menggunakan energi paling besar. Maka, tidak ada korelasi antara penggunaan energi terhadap suhu udara secara signifikan.
- Selain itu dapat dilihat, primary\_use kedua hasil itu berbeda yaitu satu digunakan untuk keperluan pendidikan dan selebihnya untuk keperluan public services dan office.

Tidak ada korelasi yang cukup kuat antara meter reading dengan variabel lainnya. Dari dua visualisasi terakhir di atas, dapat dilihat bahwa korelasi antara energi yang digunakan (meter\_reading) dengan variabel-variabel lainnya cenderung rendah, yaitu  $-0.2 < r < 0.2$



**SEKIAN DAN  
TERIMA KASIH**