**Report 12/12**

1. **Biểu đồ Histogram:**

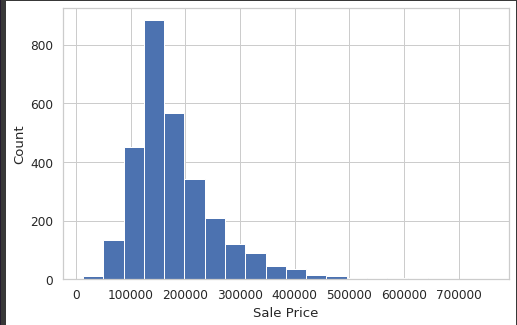
Histogram là biểu đồ áp dụng trên một biến liên tục nhằm tìm ra phân phối tần suất trong những khoảng giá trị được xác định trước của một biến.

Có 2 cách tạo biểu đồ histogram theo các khoảng giá trị đó là:

* Phân chia các khoảng giá trị có độ dài bằng nhau và độ dài được tính toán từ số lượng **bins** khai báo.
* Tự định nghĩa các khoảng giá trị dựa trên **bins\_edge** là các đầu mút của khoảng.

## Plt.hist() dùng hàm này để vẽ biễu đồ histogram trong python

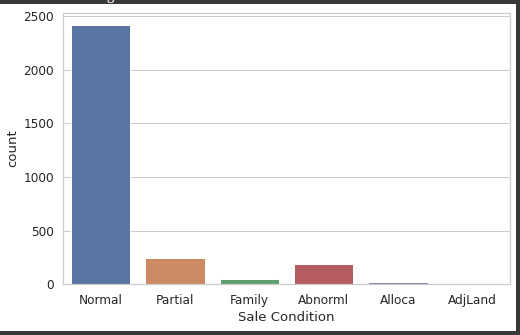
* **Value distribution (histogram):**Cho biết cách các giá trị trong cột của bạn được phân phối. Thanh càng cao, càng nhiều giá trị rơi vào một phạm vi.
* **Min & Max:** cho ra các giá trị min,max.
* **Mean:**thể hiện giá trị trung bình.
* **Median:**thể hiện các giá trị trung vị.



1. **Biểu đồ column hoặc biểu đồ bar để thể hiện các biến phân loại:**

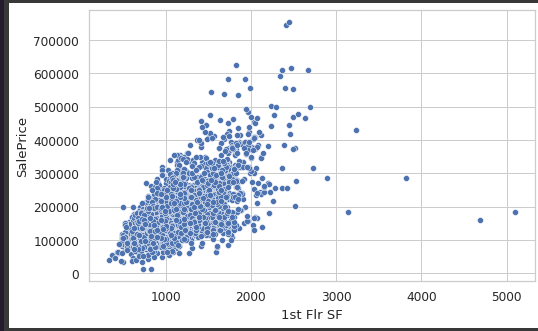
Sử dụng **Countplot()** dùng để biểu diễn các biến phân loại theo 2 trục là trục tung và trục hoành

Sử dụng hàm  **sns.countplot()** để vẽ biểu diễn biến phân loại



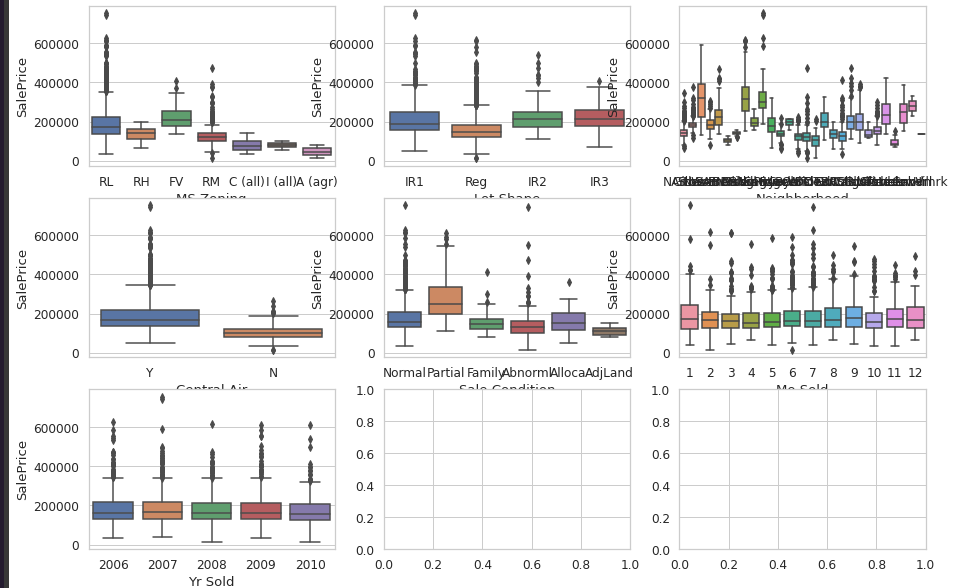
1. **Sự tương quan giữa 2 biến định lượng:**

Sử dụng **scatterplot()** để biển diễn sự quan hệ giữa 2 biến định lượng



1. **Sự tương quan giữa 2 biến định lượng và biến phân loại:**

Sử dụng **boxplot**() để biểu diễn sự tương quan giữa 2 biến định lượng và phân loại

****

Ở **boxplot()** biểu đồ diễn tả 5 vị trí phân bố của dữ liệu, đó là: giá trị nhỏ nhất (min), tứ phân vị thứ nhất (Q1), trung vị (median), tứ phân vị thứ 3 (Q3) và giá trị lớn nhất (max).

1. **Đọc file ở local host, ở 1 driver khác**

**Upload file lên colab.**

from google.colab import files

uploaded = files.upload()

**Sử dụng data ở một driver khác**

train\_df = pd.read\_csv(‘URL’)

test\_df = pd.read\_csv(‘URL’)

1. **Xử lý tiền dữ liệu:**

Dữ liệu null, dữ liệu bị mất: dùng hàm train\_df.isnull().sum()

Dùng hàm này để tìm giá trị bị thiếu tính phần trăm dữ liệu dữ liệu bị miss hoặc giá trị null nếu nhưng tỉ lệ bị mất dữ liệu nếu tỉ lệ null hoặc mất dữ liệu

quá nhiều thì tiến hành drop dữ liệu dòng đó. Nếu có 1 vài dữ liệu bị mất hoặc

null thì tiến hành set giá trị trùng bình cho giá trị đó.

Dùng (train\_df['Age'].isnull().sum()/train\_df.shape[0])\*100

Tính giá trị tỉ lệ dữ liệu bị mất:

Tính giá trị trung bình: train\_df["Age"].mean(skipna=True).

Tính giá trị trung vị: train\_df["Age"].median(skipna=True).

Tìm dữ liệu bị trùng duplicateRowsDF = dfObj[dfObj.duplicated()]

Xóa dữ liệu bị trùng: data.drop\_duplicates(subset ="First Name", keep = False, inplace = True)

1. **Summary**: dữ liệu dùng các đại lượng thống kê mô tả chúng ta sẽ sử dụng lệnh Describe () để xem các só liệu thống kê Count: là dữ liệu mẫu hoặc dữ liệu tổng Mean là trung bình Std là độ lệnh chuẩn là độ tương đối của điểm trên đồ thị Min: điểm nhỏ nhất 25% và 75%: Là tứ vị ví dụ vị lớp 100 người bạn đứng 10 thì bạn hơn 89 người khác là bạn hơn 89% 50%:là trung vị của dữ liệu dữ liệu được chia làm 4 phần bằng nhau 25% để có thể dễ dàng tập trung phân tích vào các khoản dữ liệu mà mình muốn phân tích Max là điểm cao nhất các điểm người Min và Max là outlier.
2. **Trực quan dữ liệu định lượng thì ta dùng biểu đồ histogram Histogram** là biểu đồ áp dụng trên một biến liên tục nhằm tìm ra phân phối tần suất trong những khoảng giá trị được xác định trước của một biến. Có 2 cách tạo biểu đồ histogram theo các khoảng giá trị đó là:

• Phân chia các khoảng giá trị có độ dài bằng nhau và độ dài được tính toán từ số lượng bins khai báo.

• Tự định nghĩa các khoảng giá trị dựa trên bins\_edge là các đầu mút của khoảng. Plt.hist() dùng hàm này để vẽ biễu đồ histogram trong python

• Value distribution (histogram): Cho biết cách các giá trị trong cột của bạn được phân phối. Thanh càng cao, càng nhiều giá trị rơi vào một phạm vi.

• Min & Max: cho ra các giá trị min,max.

• Mean: thể hiện giá trị trung bình.

• Median: thể hiện các giá trị trung vị.