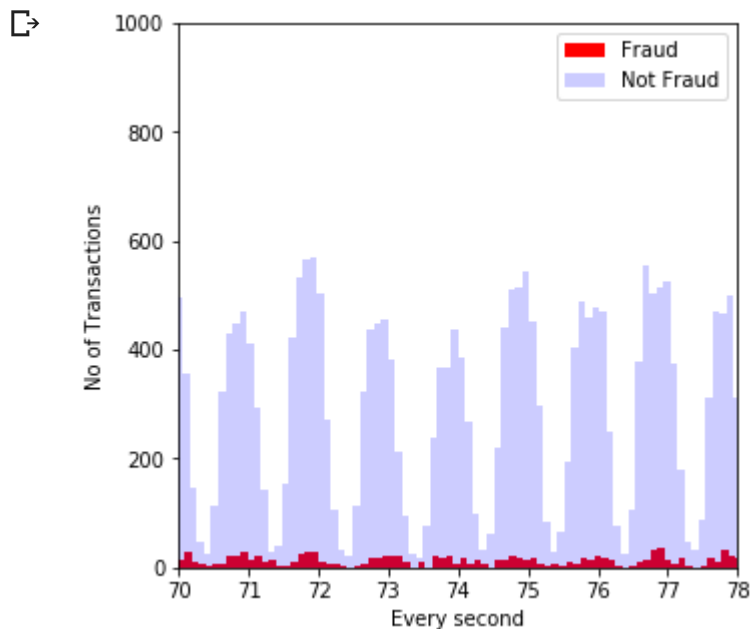


```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
```

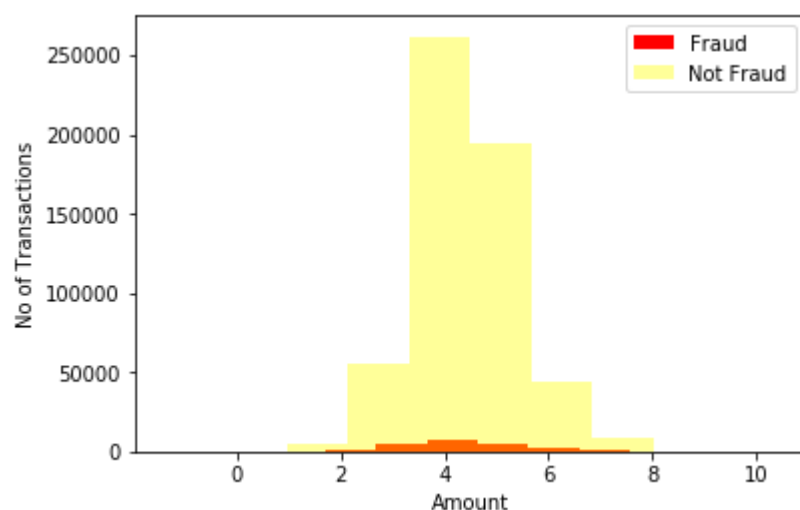
```
train_transaction = pd.read_csv("/content/drive/My Drive/train_transaction.csv", index_col= 'TransactionID')
train_identity = pd.read_csv("/content/drive/My Drive/train_identity.csv", index_col= 'TransactionID')
test_transaction = pd.read_csv("/content/drive/My Drive/test_transaction.csv", index_col= 'TransactionID')
test_identity = pd.read_csv("/content/drive/My Drive/test_identity.csv", index_col= 'TransactionID')
```

```
train_transaction_fraud = train_transaction[train_transaction['isFraud'] == 1]
train_transaction_notFraud = train_transaction[train_transaction['isFraud'] == 0]
```

```
figure = plt.figure(figsize=(5,5))
plt.xlim(70,78)
plt.ylim(0,1000)
plt.hist(train_transaction_fraud['TransactionDT']/86400, bins = 1800, alpha = 1, label = 'Fraud', color = 'red')
plt.hist(train_transaction_notFraud['TransactionDT']/86400, bins = 1800, alpha = 0.2, label = 'Not Fraud', color = 'blue')
plt.legend(loc='upper right')
plt.xlabel("Every second")
plt.ylabel("No of Transactions")
plt.show()
```



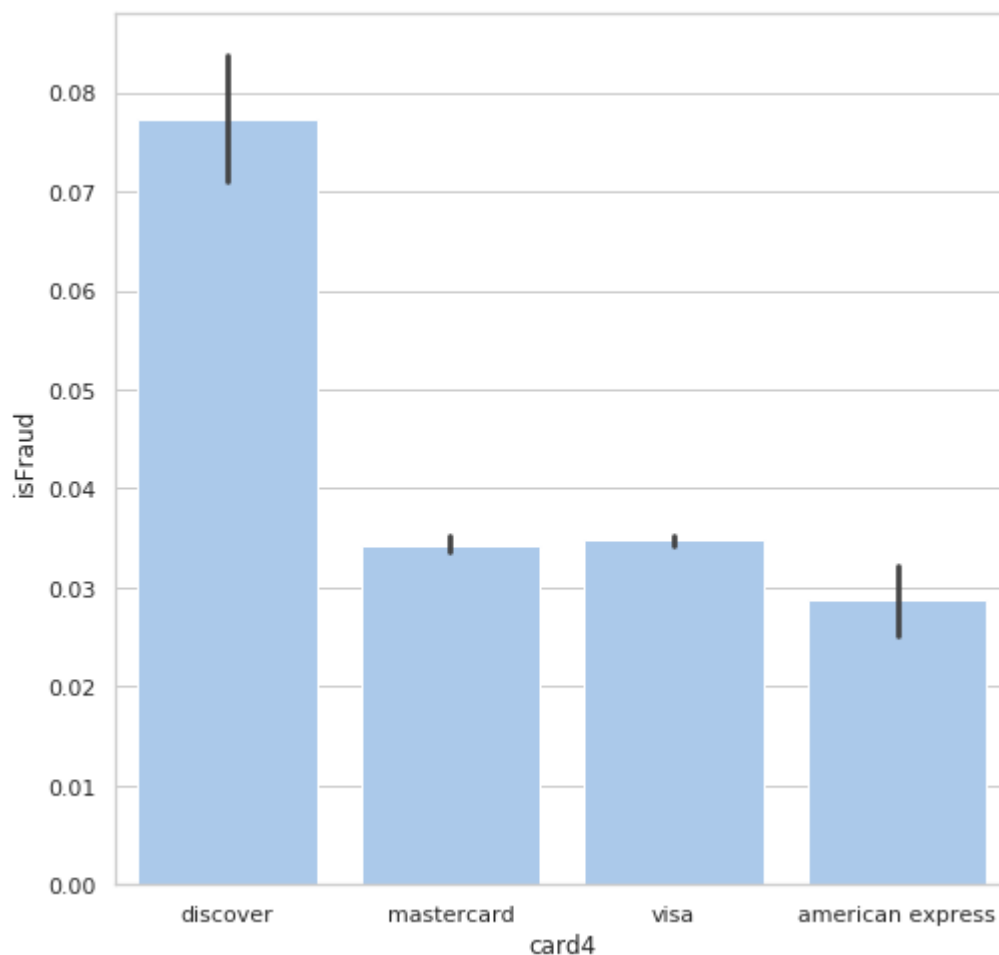
```
import numpy as np
plt.hist(np.log(train_transaction_fraud['TransactionAmt']), alpha = 1, label = 'Fraud', color = 'red')
plt.hist(np.log(train_transaction_notFraud['TransactionAmt']), alpha = 0.4, label = 'Not Fraud', color = 'blue')
plt.legend(loc='upper right')
plt.xlabel("Amount")
plt.ylabel("No of Transactions")
plt.show()
```



```
sns.set(style="whitegrid")
figure = plt.figure(figsize=(8,8))
sns.set_color_codes("pastel")
sns.barplot(x=train_transaction['card4'],y=train_transaction['isFraud'],data=train_transaction,lab=
```



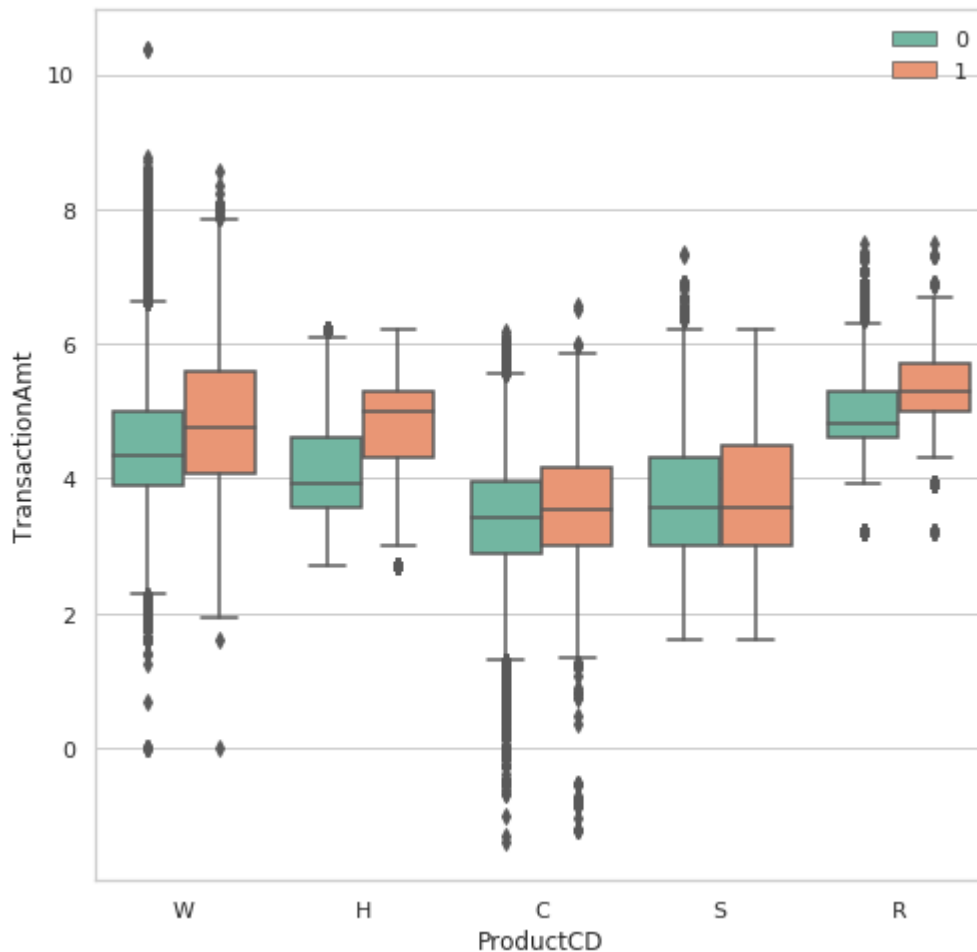
<matplotlib.axes.\_subplots.AxesSubplot at 0x7f9b5acf1748>



```
#sns.set(style="darkgrid")
figure = plt.figure(figsize=(8,8))
```

```
ax = sns.boxplot(x=train_transaction['ProductCD'], y=np.log(train_transaction['TransactionAmt']), f
ax.legend(frameon=False, loc='upper right', ncol=1)
```

↳ <matplotlib.legend.Legend at 0x7f9b58427908>



```
new_train_transaction = pd.read_csv("/content/drive/My Drive/train_transaction.csv")
sns.pairplot(new_train_transaction)
```

```
fig, ax = plt.subplots(2, 2, figsize = (10,10))
```

```
time = train_transaction_fraud['TransactionDT']
time2 = train_transaction_notFraud['TransactionDT']
```

```
log_time = np.log(time.values/86400)
log_time2 = np.log(time2.values/86400)
```

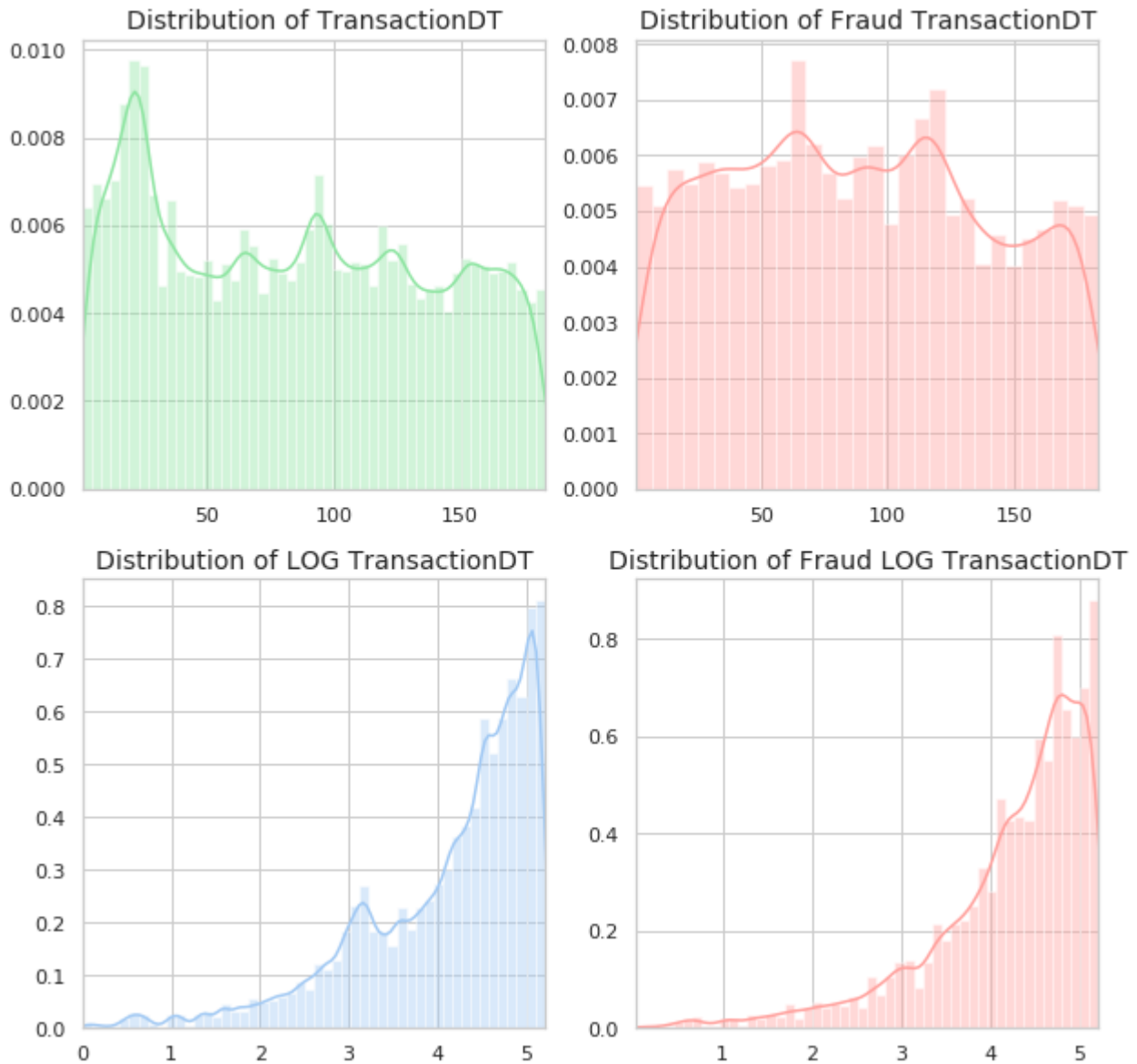
```
sns.distplot(time2.values/86400, ax = ax[0,0], color = 'g')
ax[0,0].set_title('Distribution of TransactionDT', fontsize=14)
ax[0,0].set_xlim([min(time2.values/86400), max(time2.values/86400)])
```

```
sns.distplot(time.values/86400, ax=ax[0,1], color='r')
ax[0,1].set_title('Distribution of Fraud TransactionDT', fontsize=14)
ax[0,1].set_xlim([min(time.values/86400), max(time.values/86400)])
```

```
sns.distplot(log_time2, ax=ax[1,0], color='b')
ax[1,0].set_title('Distribution of LOG TransactionDT', fontsize=14)
ax[1,0].set_xlim([min(log_time2), max(log_time2)])
```

```
sns.distplot(log_time, ax=ax[1,1], color='r')
ax[1,1].set_title('Distribution of Fraud LOG TransactionDT', fontsize=14)
ax[1,1].set_xlim([min(log_time), max(log_time)])

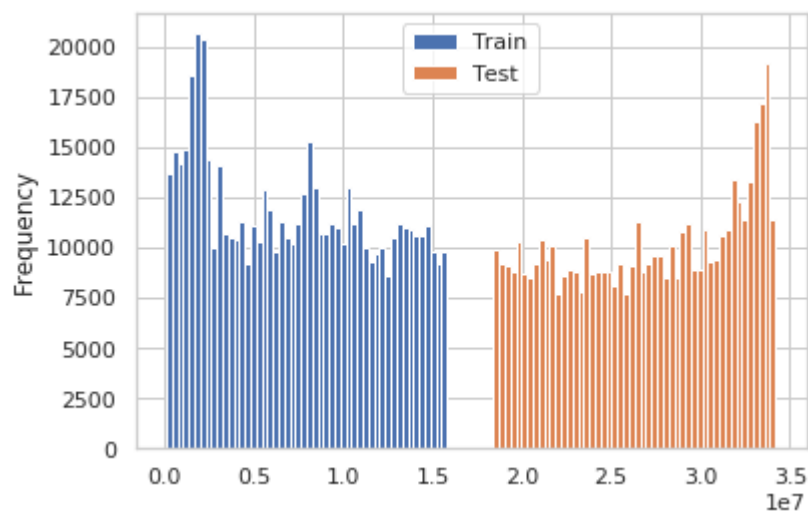
plt.show()
```



```
#No overlap between train and test data dates.
train_transaction['TransactionDT'].plot(kind = "hist", label = "Train", bins = 50)
test_transaction['TransactionDT'].plot(kind = "hist", label = "Test", bins = 50)
plt.legend()
```



<matplotlib.legend.Legend at 0x7f9b19c442b0>



```
sns.countplot(x = "DeviceType",data = train_identity, palette = "Set3")
```

☞ <matplotlib.axes.\_subplots.AxesSubplot at 0x7f9b19c29cc0>

