Import Libraries

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

Import SK learn

In [9]:
```
from sklearn.model_selection import train_test_split
from sklearn.impute import SimpleImputer
from sklearn.compose import ColumnTransformer
```

Import Dataset

In [10]:
```
df = pd.read_csv('titanic_toy.csv')
```

In [11]:
```
df
```

Out[11]:

	Age	Fare	Family	Survived
0	22.0	7.2500	1	0
1	38.0	71.2833	1	1
2	26.0	7.9250	0	1
3	35.0	53.1000	1	1
4	35.0	8.0500	0	0
...	...	...	...	...
886	27.0	13.0000	0	0
887	19.0	30.0000	0	1
888	NaN	23.4500	3	0
889	26.0	NaN	0	1
890	32.0	7.7500	0	0

891 rows × 4 columns

In [12]:
```
df.head()
```

Out[12]:

	Age	Fare	Family	Survived
0	22.0	7.2500	1	0
1	38.0	71.2833	1	1
2	26.0	7.9250	0	1
3	35.0	53.1000	1	1
4	35.0	8.0500	0	0

In [13]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 4 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   Age       714 non-null    float64
 1   Fare      846 non-null    float64
 2   Family    891 non-null    int64  
 3   Survived  891 non-null    int64  
dtypes: float64(2), int64(2)
memory usage: 28.0 KB
```

Perform Train Test Split

In [18]:
```
X = df.drop(columns=['Survived'])
y = df['Survived']
```

In [19]:
```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=2)
```

In [20]:
```
X_train['Age_99'] = X_train['Age'].fillna(99)
X_train['Age_minus1'] = X_train['Age'].fillna(-1)
X_train['Fare_999'] = X_train['Fare'].fillna(999)
X_train['Fare_minus1'] = X_train['Fare'].fillna(-
1)
```

Create New Column and Replace Value (Age-99 | Fare-999)

In [21]:
```
X_train['Age_99'] = X_train['Age'].fillna(99)
X_train['Age_minus1'] = X_train['Age'].fillna(-1)
X_train['Fare_999'] = X_train['Fare'].fillna(999)
X_train['Fare_minus1'] = X_train['Fare'].fillna(-1)
```

In [23]:
```
X_train.sample(10)
```

Out[23]:

	Age	Fare	Family	Age_99	Age_minus1	Fare_999	Fare_minus1
861	21.0	11.5000	1	21.0	21.0	11.5000	11.5000
428	NaN	7.7500	0	99.0	-1.0	7.7500	7.7500
97	23.0	63.3583	1	23.0	23.0	63.3583	63.3583
31	NaN	146.5208	1	99.0	-1.0	146.5208	146.5208
378	20.0	NaN	0	20.0	20.0	999.0000	-1.0000
358	NaN	7.8792	0	99.0	-1.0	7.8792	7.8792
323	22.0	29.0000	2	22.0	22.0	29.0000	29.0000
121	NaN	8.0500	0	99.0	-1.0	8.0500	8.0500
165	9.0	20.5250	2	9.0	9.0	20.5250	20.5250
494	21.0	8.0500	0	21.0	21.0	8.0500	8.0500

Review Variance

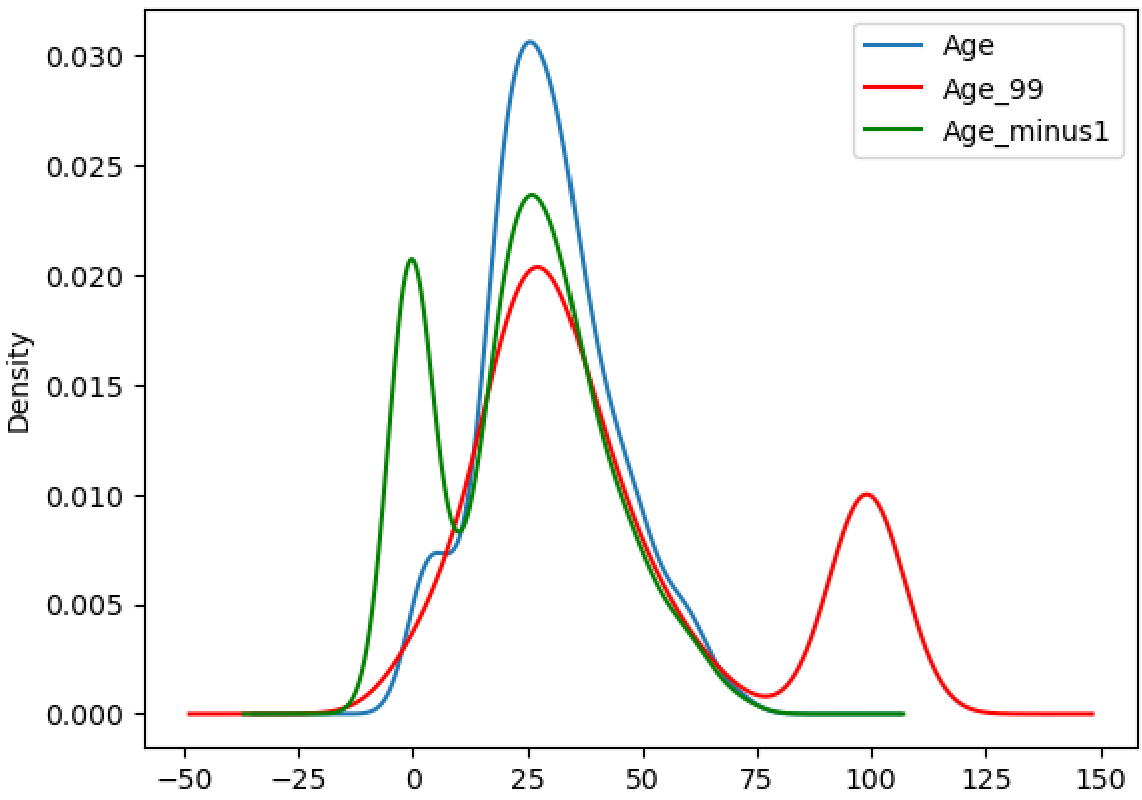
In [27]:
```
print('Original Age variable variance: ', X_train['Age'].var())
print('Age Variance after 99 use imputation: ', X_train['Age_99'].var())
print('Age Variance after -1 use imputation: ', X_train['Age_minus1'].var())
print('Original Fare variable variance: ', X_train['Fare'].var())
print('Fare Variance after 999 use imputation: ', X_train['Fare_999'].var())
print('Fare Variance after -1 use imputation: ', X_train['Fare_minus1'].var())
```

```
Original Age variable variance:  204.3495133904614
Age Variance after 99 use imputation:  951.7275570187172
Age Variance after -1 use imputation:  318.0896202624484
Original Fare variable variance:  2448.197913706318
Fare Variance after 999 use imputation:  47219.20265217623
Fare Variance after -1 use imputation:  2378.5676784883503
```

Review Distribution - Age:

In [29]:
```
fig = plt.figure()
ax = fig.add_subplot(111)
# original variable distribution
X_train['Age'].plot(kind='kde', ax=ax)
# variable imputed with the median
X_train['Age_99'].plot(kind='kde', ax=ax, color='red')
# variable imputed with the mean
X_train['Age_minus1'].plot(kind='kde', ax=ax, color='green')
# add legends
lines, labels = ax.get_legend_handles_labels()
ax.legend(lines, labels, loc='best')
```

Out[29]:
```
<matplotlib.legend.Legend at 0x200c9e32ad0>
```

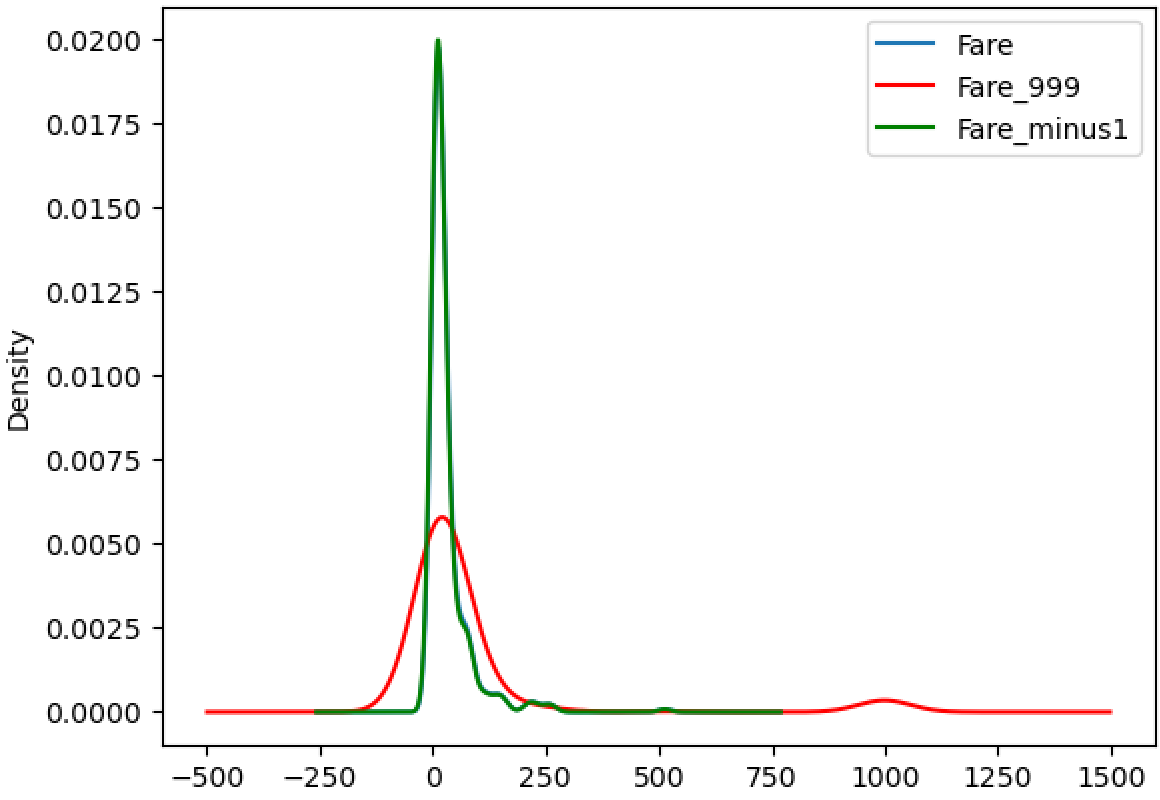


Review Distribution –Fare:

In [30]:
```python
fig = plt.figure()
ax = fig.add_subplot(111)
# original variable distribution
X_train['Fare'].plot(kind='kde', ax=ax)
# variable imputed with the median
X_train['Fare_999'].plot(kind='kde', ax=ax, color='red')
# variable imputed with the mean

X_train['Fare_minus1'].plot(kind='kde', ax=ax, color='green')
# add legends
lines, labels = ax.get_legend_handles_labels()
ax.legend(lines, labels, loc='best')
```

Out[30]: <matplotlib.legend.Legend at 0x200c9147090>



Check Covariance

In [31]:
```python
X_train.cov()
```

Out[31]:

	Age	Fare	Family	Age_99	Age_minus1	Fare_999	Fare_minus1
Age	204.349513	70.719262	-6.498901	204.349513	204.349513	162.793430	63.321188
Fare	70.719262	2448.197914	17.258917	-101.671097	125.558364	2448.197914	2448.197914
Family	-6.498901	17.258917	2.735252	-7.387287	-4.149246	11.528625	16.553989
Age_99	204.349513	-101.671097	-7.387287	951.727557	-189.535540	-159.931663	-94.317400
Age_minus1	204.349513	125.558364	-4.149246	-189.535540	318.089620	257.379887	114.394141
Fare_999	162.793430	2448.197914	11.528625	-159.931663	257.379887	47219.202652	762.474982
Fare_minus1	63.321188	2448.197914	16.553989	-94.317400	114.394141	762.474982	2378.567678

Check Correlation

In [32]:

```
X_train.corr()
```

Out[32]:

	Age	Fare	Family	Age_99	Age_minus1	Fare_999	Fare_minus1
Age	1.000000	0.092644	-0.299113	1.000000	1.000000	0.051179	0.084585
Fare	0.092644	1.000000	0.208268	-0.066273	0.142022	1.000000	1.000000
Family	-0.299113	0.208268	1.000000	-0.144787	-0.140668	0.032079	0.205233
Age_99	1.000000	-0.066273	-0.144787	1.000000	-0.344476	-0.023857	-0.062687
Age_minus1	1.000000	0.142022	-0.140668	-0.344476	1.000000	0.066411	0.131514
Fare_999	0.051179	1.000000	0.032079	-0.023857	0.066411	1.000000	0.071946
Fare_minus1	0.084585	1.000000	0.205233	-0.062687	0.131514	0.071946	1.000000

In [ ]: