




```
import pandas as pd
import missingno as msno
import matplotlib.pyplot as plt
```

```
df=pd.read_csv("https://raw.githubusercontent.com/campusx-official/100-days-of-machine-learn
```

```
df.head()
```



	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

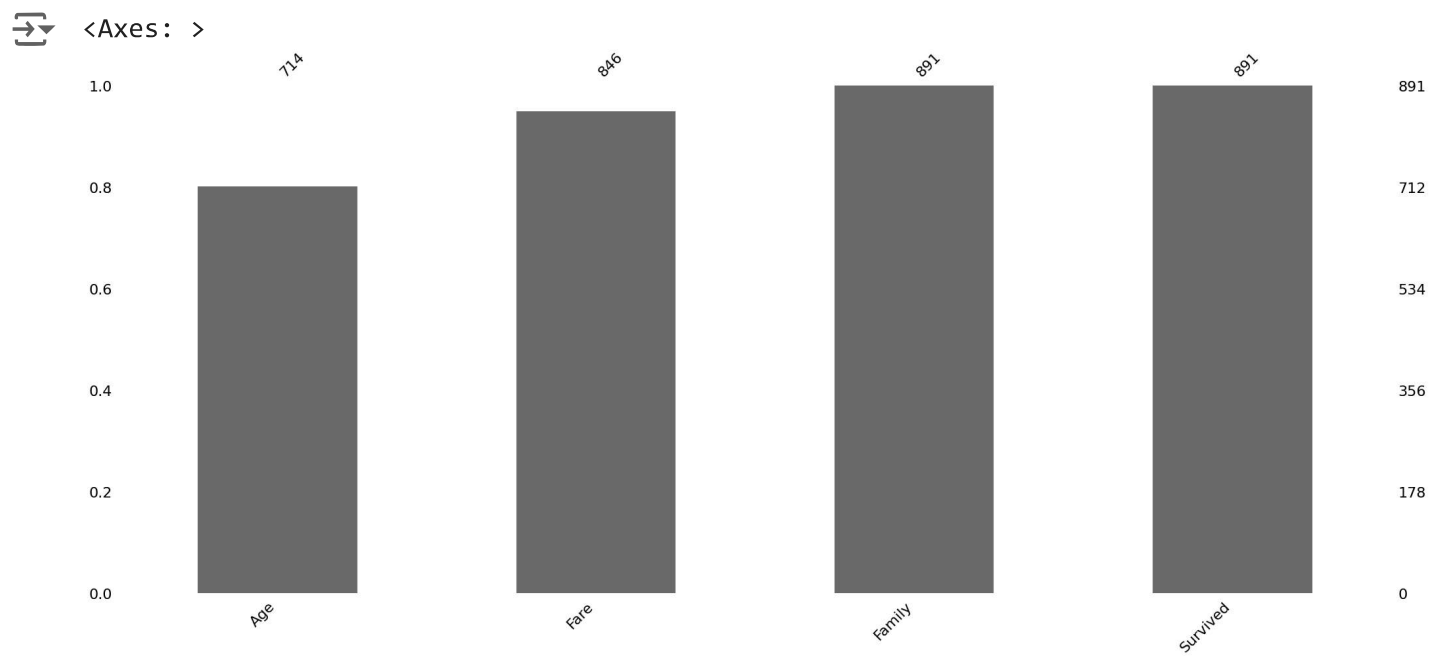


Next steps:

[Generate code with df](#)[View recommended plots](#)

Start coding or [generate](#) with AI.

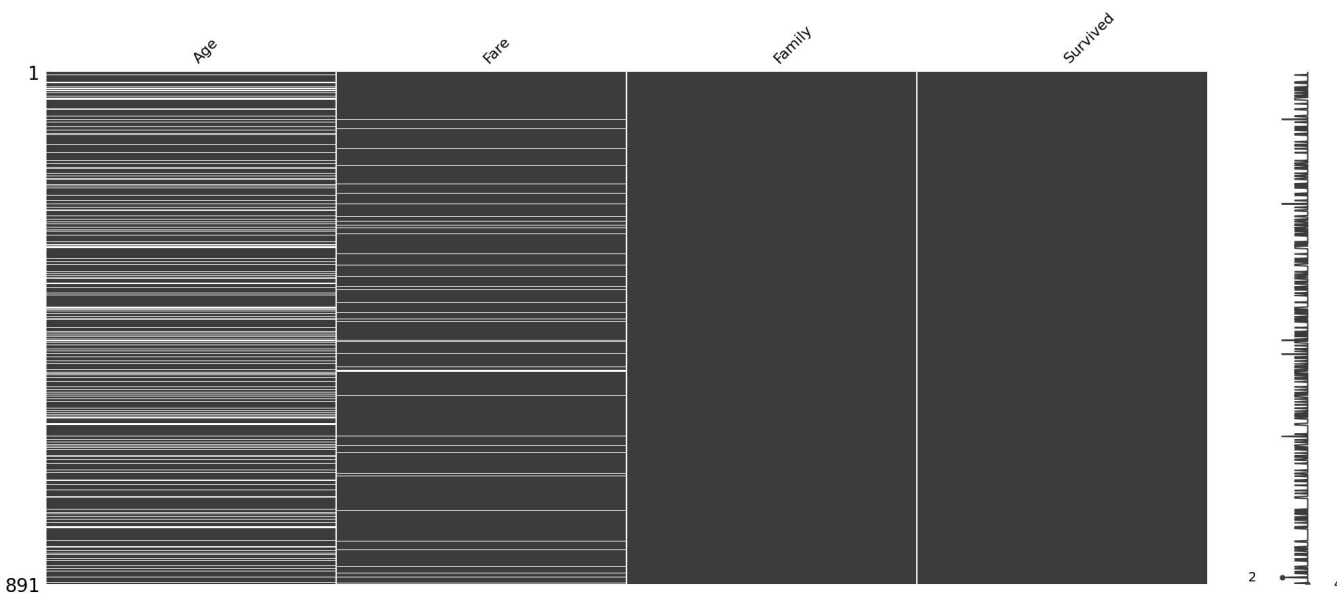
```
msno.bar(df)
```



```
msno.matrix(df)
```



<Axes: >



```
df.isnull().mean()
```

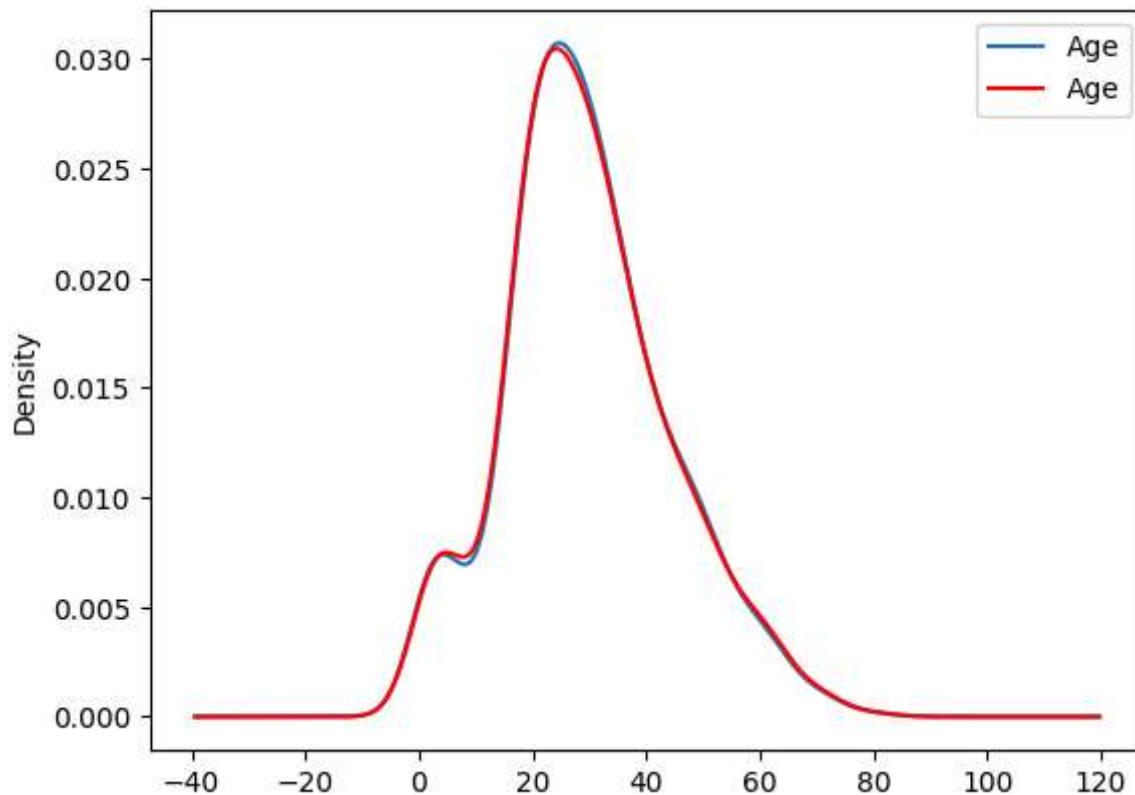


```
Age      0.198653
Fare      0.050505
Family    0.000000
Survived  0.000000
dtype: float64
```

```
#check the mcar mnar or mar
```

```
dfna=df.dropna()
```

```
fig=plt.figure()
ax=fig.add_subplot(111)
df["Age"].plot(kind="kde", ax=ax)
dfna["Age"].plot(kind="kde", ax=ax, color="red")
lines,labels=ax.get_legend_handles_labels()
ax.legend(lines,labels, loc="best")
plt.show()
```



If the distribution does not change, it suggests that the data is missing completely at random (MCAR).

```
from sklearn.model_selection import train_test_split
y=df["Survived"]
x=df.drop("Survived",axis=1)
xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=0.02,random_state=2)
```


```
for i in df.columns:
    print(i)
```




```
Age
Fare
Family
Survived
```

```
xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=0.02,random_state=2)
```


```
xtrain.head()
```



	Age	Fare	Family	
702	18.0	NaN	1	
280	65.0	7.7500	0	
505	18.0	108.9000	1	
128	NaN	22.3583	2	
333	16.0	18.0000	2	

Next steps:


[Generate code with xtrain](#)


 [View recommended plots](#)

```
import numpy as np
```

```
xtrain["Age_mean"]=xtrain["Age"].fillna(np.mean(xtrain["Age"]))
xtrain["Age_Median"]=xtrain["Age"].fillna(np.median(xtrain["Age"]))
xtrain["Fare_mean"]=xtrain["Fare"].fillna(np.mean(xtrain["Fare"]))
xtrain["Fare_Median"]=xtrain["Fare"].fillna(np.median(xtrain["Fare"]))
```


```
xtrain.head()
```



	Age	Fare	Family	Age_mean	Age_Median	Fare_mean	Fare_Median	
702	18.0	NaN	1	18.000000	18.0	32.212167	NaN	
280	65.0	7.7500	0	65.000000	65.0	7.750000	7.7500	
505	18.0	108.9000	1	18.000000	18.0	108.900000	108.9000	
128	NaN	22.3583	2	29.571059	NaN	22.358300	22.3583	
333	16.0	18.0000	2	16.000000	16.0	18.000000	18.0000	

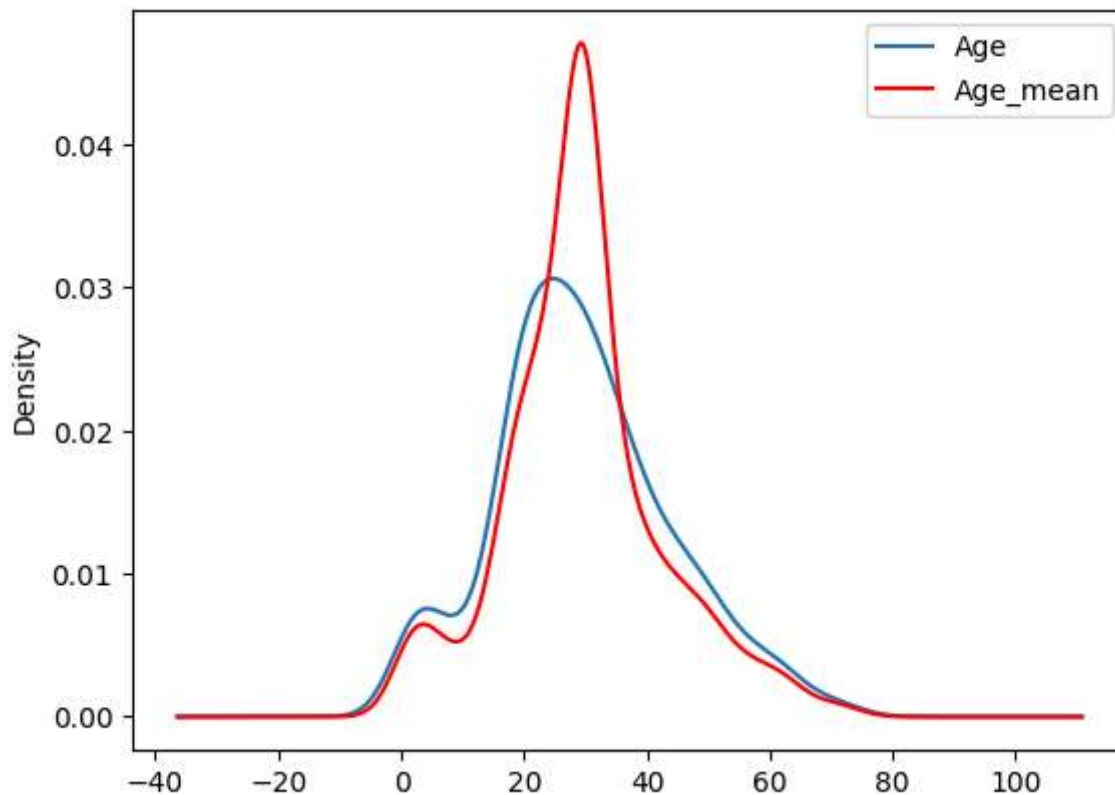
Next steps:

[Generate code with xtrain](#)

 [View recommended plots](#)

```
#Age vs Age_mean
#Age vs Age_median
#Fare vs Fare_mean
#Fare vs Fare_median
```

```
fig=plt.figure()
ax=fig.add_subplot(111)
xtrain["Age"].plot(kind="kde", ax=ax)
xtrain["Age_mean"].plot(kind="kde", ax=ax, color="red")
lines,labels=ax.get_legend_handles_labels()
ax.legend(lines,labels,loc="best")
plt.show()
```

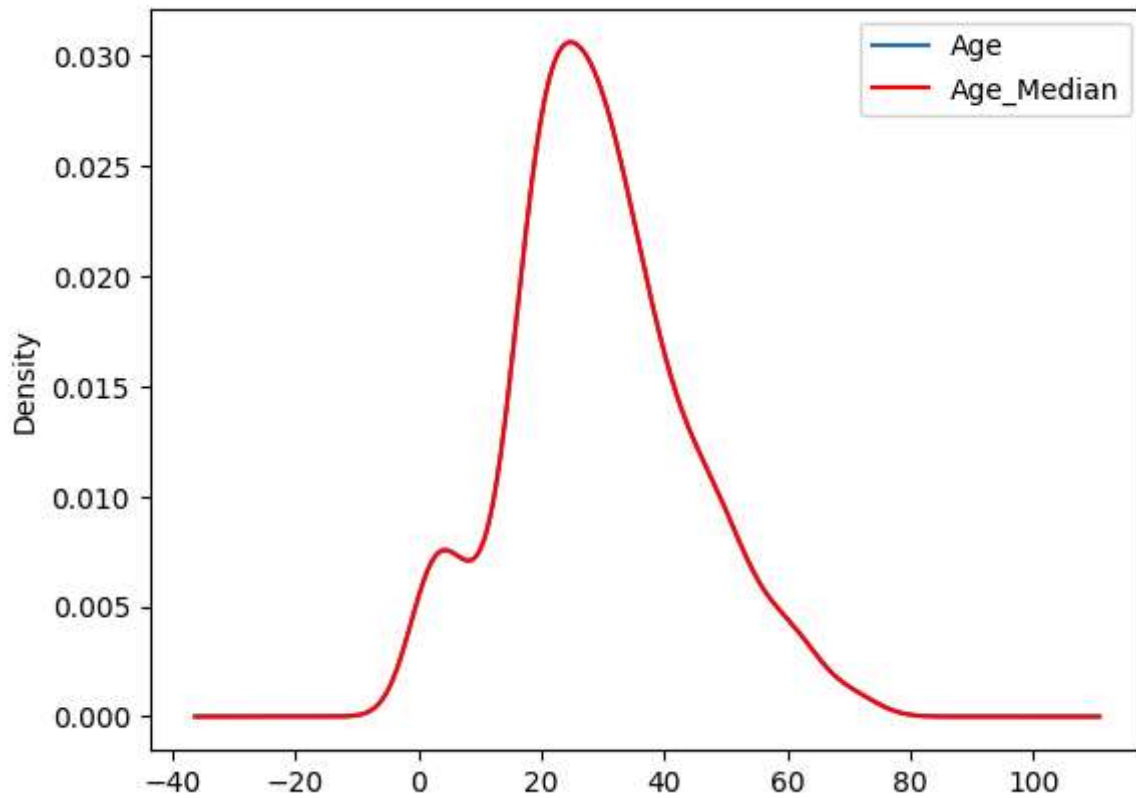


Age Distribution (Blue Line): This line represents the original distribution of the Age variable in the dataset.

Age_mean Distribution (Red Line): This line represents the distribution of the Age variable after mean imputation.

Distributions of Age and Age_mean show noticeable differences, indicating that the imputation process did significantly alter the distribution of the data.

```
fig=plt.figure()
ax=fig.add_subplot(111)
xtrain["Age"].plot(kind="kde", ax=ax)
xtrain["Age_Median"].plot(kind="kde", ax=ax, color="red")
lines,labels=ax.get_legend_handles_labels()
ax.legend(lines,labels,loc="best")
plt.show()
```

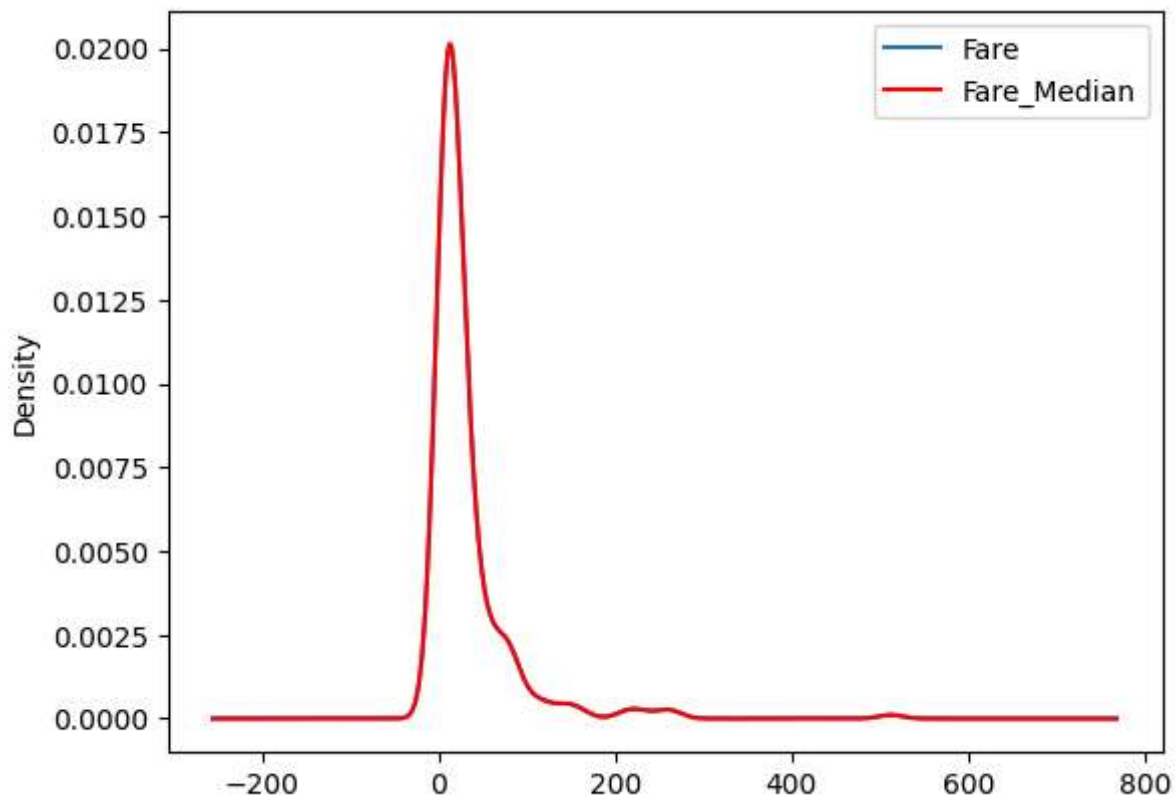


Age Distribution (Blue Line): This line represents the original distribution of the Age variable in the dataset.

Age_Median Distribution (Red Line): This line represents the distribution of the Age variable after median imputation.

Distributions of Age and Age_Median are very similar, indicating that the imputation process did not significantly alter the distribution of the data.

```
fig=plt.figure()
ax=fig.add_subplot(111)
xtrain["Fare"].plot(kind="kde",ax=ax)
xtrain["Fare_Median"].plot(kind="kde",ax=ax,color="red")
lines,labels=ax.get_legend_handles_labels()
ax.legend(lines,labels,loc="best")
plt.show()
```

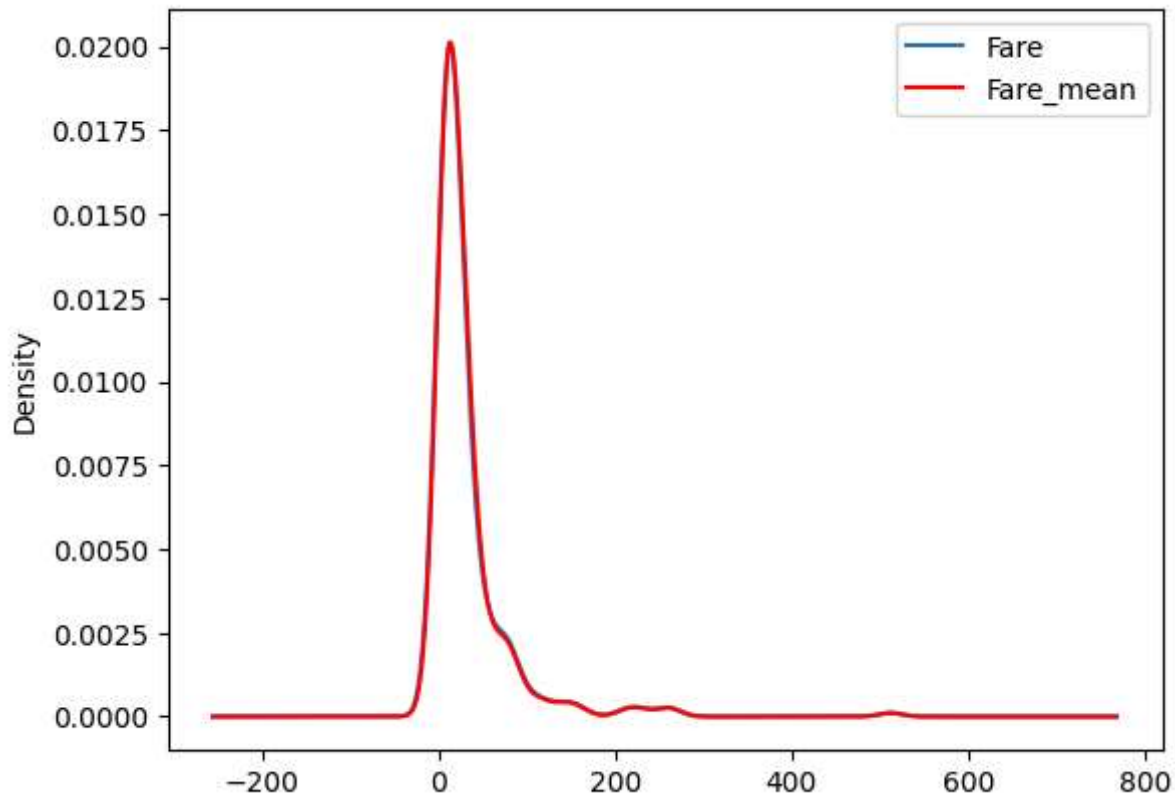


Fare Distribution (Blue Line): This line represents the original distribution of the Fare variable in the dataset.

Fare_Median Distribution (Red Line): This line represents the distribution of the Fare variable after median imputation.

Distributions of Fare and Fare_Median are very similar, indicating that the imputation process did not significantly alter the distribution of the data.

```
fig=plt.figure()
ax=fig.add_subplot(111)
xtrain["Fare"].plot(kind="kde",ax=ax)
xtrain["Fare_mean"].plot(kind="kde",ax=ax,color="red")
lines,labels=ax.get_legend_handles_labels()
ax.legend(lines,labels,loc="best")
plt.show()
```

Fare Distribution (Blue Line): This line represents the original distribution of the Fare variable in dataset.

Fare_mean Distribution (Red Line): This line represents the distribution of the Fare variable after imputation of mean.

Distributions of Fare and Fare_mean are very similar, this indicates that imputation process did not significantly alter the distribution of the data.

```
xtrain.columns
```



```
Index(['Age', 'Fare', 'Family', 'Age_mean', 'Age_Median', 'Fare_mean',
      'Fare_Median'],
      dtype='object')
```

```
variances=xtrain[['Age', 'Fare', 'Age_mean', 'Age_Median', 'Fare_mean', 'Fare_Median']].var()
variances
```



```
Age          208.956540
Fare         2560.172306
Age_mean     167.261084
Age_Median   208.956540
Fare_mean    2428.053322
Fare_Median  2560.172306
dtype: float64
```

✓ Variance Analysis Before and After Imputation

Let's analyze the variances of the Age and Fare variables before and after different imputation methods.

Age Variance (Original): The variance of the original Age values is 208.956540.

- This is the baseline variance of the Age variable in the dataset, reflecting the spread of the original Age values.

Fare Variance (Original): The variance of the original Fare values is 2560.172306.

- This is the baseline variance of the Fare variable in the dataset, reflecting the spread of the original Fare values.

Age_mean Variance (After Mean Imputation): The variance of the Age values after mean imputation is 167.261084.

- The variance of the Age values has decreased after mean imputation, suggesting that mean imputation has reduced the spread of Age values, likely because the mean imputation replaced missing values with the same mean value, reducing variability.

Age_Median Variance (After Median Imputation): The variance of the Age values after median imputation is 208.956540.

- The variance remains unchanged after median imputation, indicating that median imputation did not alter the spread of the Age values. This suggests that the median imputation maintained the original variability of the data.

Fare_mean Variance (After Mean Imputation): The variance of the Fare values after mean imputation is 2428.053322.

- The variance of the Fare values has decreased after mean imputation, suggesting that mean imputation has reduced the spread of Fare values, likely because the mean imputation replaced missing values with the same mean value, reducing variability.

Fare_Median Variance (After Median Imputation): The variance of the Fare values after median imputation is 2560.172306.

- The variance remains unchanged after median imputation, indicating that median imputation did not alter the spread of the Fare values. This suggests that the median imputation maintained the original variability of the data.

Interpretation of Changes:

- **Mean Imputation:** Both Age and Fare variances have decreased after mean imputation. This reduction in variance indicates that mean imputation reduced the spread of values by

replacing missing values with a constant mean, which reduces overall variability.

- **Median Imputation:** Both Age and Fare variances remained the same after median imputation. This indicates that median imputation preserved the original spread of the data, maintaining the original variability.

The unchanged variances after median imputation imply that this method preserved the original data distribution more effectively than mean imputation, which resulted in a decrease in the overall variances.

```
xtrain.cov()
```



	Age	Fare	Family	Age_mean	Age_Median	Fare_mean	Fare_Median
Age	208.956540	77.452762	-6.436331	208.956540	208.956540	73.029582	77.452762
Fare	77.452762	2560.172306	17.722764	61.638027	77.452762	2560.172306	2560.172306
Family	-6.436331	17.722764	2.609070	-5.152017	-6.436331	16.808172	17.722764
Age_mean	208.956540	61.638027	-5.152017	167.261084	208.956540	58.457166	61.638027
Age_Median	208.956540	77.452762	-6.436331	208.956540	208.956540	73.029582	77.452762
Fare_mean	73.029582	2560.172306	16.808172	58.457166	73.029582	2428.053322	2560.172306
Fare_Median	77.452762	2560.172306	17.722764	61.638027	77.452762	2560.172306	2560.172306

✓ Interpretation of Specific Pairs:

1. Age vs Fare_mean:

- Covariance is 73.029582.
- The relationship remains positive after mean imputation, indicating a good consistency with the original positive relationship.

2. Age vs Fare_Median:

- Covariance is 77.452762.
- The relationship remains positive after median imputation, maintaining the original positive relationship.

3. Fare vs Age_mean:

- Covariance is 61.638027.
- The relationship remains positive after mean imputation, reflecting a good consistency with the original positive relationship.

4. Fare vs Age_Median:

- Covariance is 77.452762.
- The relationship remains positive after median imputation, maintaining the original positive relationship.

Summary:

- **Age vs Fare_mean:** Shows a positive relationship. After imputation, it also shows positive (good).
- **Age vs Fare_Median:** Shows a positive relationship. After imputation, it also shows positive (good).
- **Fare vs Age_mean:** Shows a positive relationship. After imputation, it also shows positive (good).
- **Fare vs Age_Median:** Shows a positive relationship. After imputation, it also shows positive (good).

The positive covariance values indicate that the relationships between the pairs of variables remain consistent after both mean and median imputations, preserving the original positive relationships.

xtrain.corr()



	Age	Fare	Family	Age_mean	Age_Median	Fare_mean	Fare_Median
Age	1.000000	0.098578	-0.300660	1.000000	1.000000	0.096291	0.098578
Fare	0.098578	1.000000	0.214274	0.093781	0.098578	1.000000	1.000000
Family	-0.300660	0.214274	1.000000	-0.246625	-0.300660	0.211178	0.214274
Age_mean	1.000000	0.093781	-0.246625	1.000000	1.000000	0.091730	0.093781
Age_Median	1.000000	0.098578	-0.300660	1.000000	1.000000	0.096291	0.098578
Fare_mean	0.096291	1.000000	0.211178	0.091730	0.096291	1.000000	1.000000
Fare_Median	0.098578	1.000000	0.214274	0.093781	0.098578	1.000000	1.000000

✓ Interpretation of Strength Changes:

1. Age vs Fare_mean:

- Original correlation: 0.098578
- After mean imputation: 0.096291
- **Strength:** Decreased

2. Age vs Fare_Median:

- Original correlation: 0.098578

- After median imputation: 0.098578
- **Strength:** No change

3. Fare vs Age_mean:

- Original correlation: 0.098578
- After mean imputation: 0.093781
- **Strength:** Decreased

4. Fare vs Age_Median:

- Original correlation: 0.098578
- After median imputation: 0.098578
- **Strength:** No change

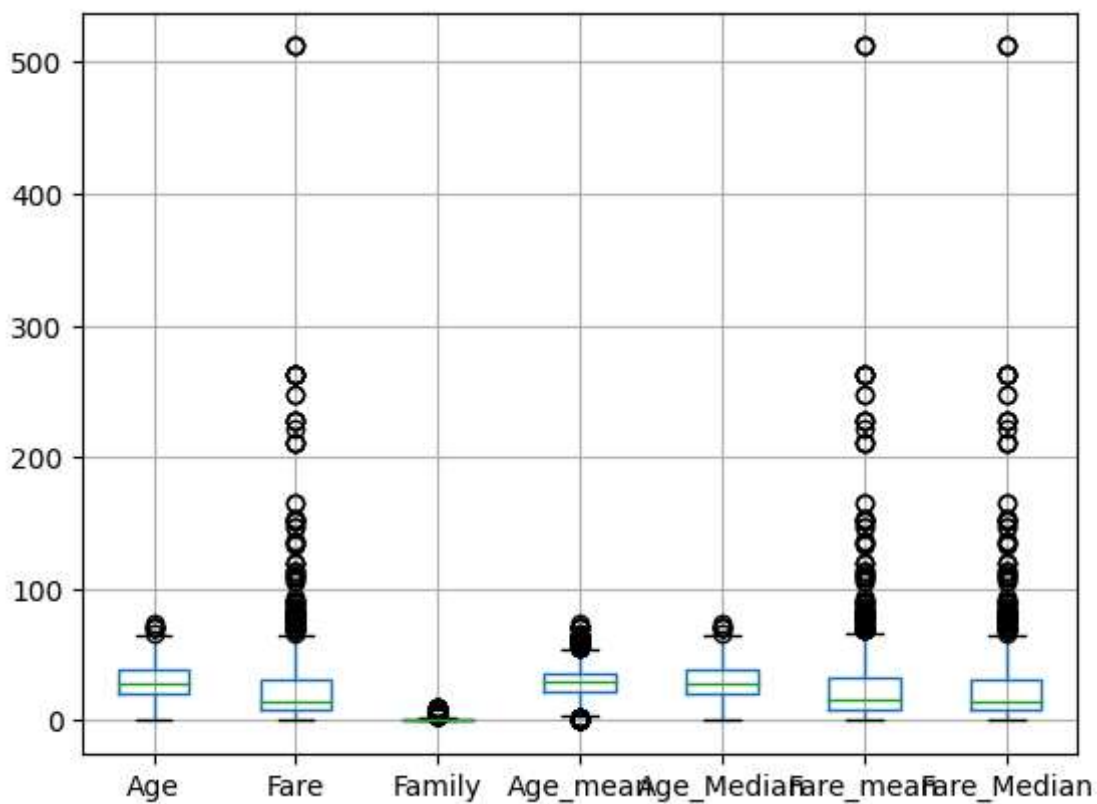
Summary:

- **Age vs Fare_mean:** Shows a positive relationship. After imputation, the strength decreased.
- **Age vs Fare_Median:** Shows a positive relationship. After imputation, the strength did not change.
- **Fare vs Age_mean:** Shows a positive relationship. After imputation, the strength decreased.
- **Fare vs Age_Median:** Shows a positive relationship. After imputation, the strength did not change.

The analysis indicates that the strength of the relationships remains mostly consistent after median imputation, whereas mean imputation results in a slight decrease in the strength of the relationships.

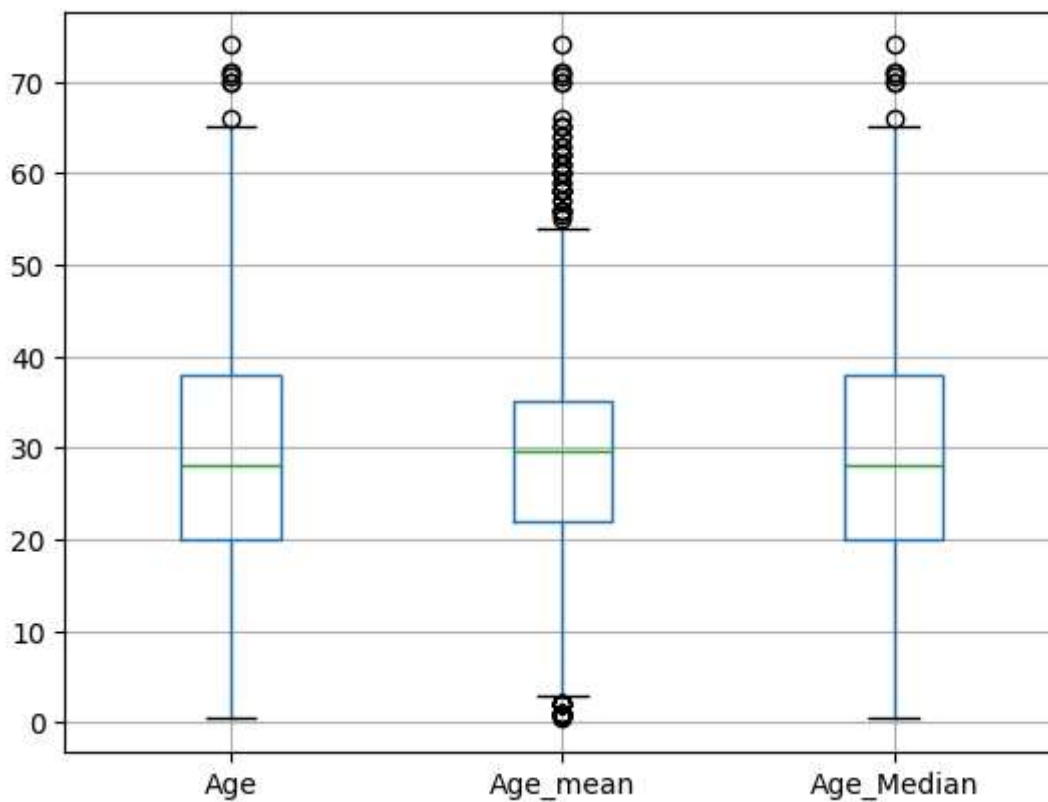
```
xtrain.boxplot()
```

↔ <Axes: >



```
xtrain[['Age', 'Age_mean', 'Age_Median']].boxplot()
plt.show()
```

↔



1. Age (Original Data):

- The data has a broad range of values, and there are some outliers present.
- The distribution seems fairly symmetric, with the median line approximately in the middle of the box.
- The interquartile range (IQR) is moderate, indicating a reasonable spread of the middle 50% of the data.

2. Age_mean (Mean Imputed Data):

- Imputing missing values with the mean introduces many outliers. This is evident from the numerous dots above the upper whisker.
- The first quartile (Q1) and third quartile (Q3) are closer together, which indicates a compression of the central values.
- The median remains approximately the same as the original data.

3. Age_Median (Median Imputed Data):

- Imputing with the median seems to maintain a similar spread to the original data, with fewer outliers compared to the mean imputation.
- The IQR remains similar to the original data, indicating that the spread of the central 50% is maintained.
- The median line is in the middle of the box, similar to the original data.

Summary:

- Imputing missing values with the mean can introduce a significant number of outliers and compress the IQR.
- Imputing with the median maintains the distribution's characteristics more closely, with fewer outliers and a similar spread of the central data.

```
xtrain[['Fare', 'Fare_mean',
        'Fare_Median']].boxplot()
plt.show()
```

