Chapter 07

August 4, 2023

[1]:
```
import pandas as pd
import numpy as np
import os

import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
```

[6]:
```
titanic = pd.read_csv(os.path.join('data','Titanic.csv'))
```

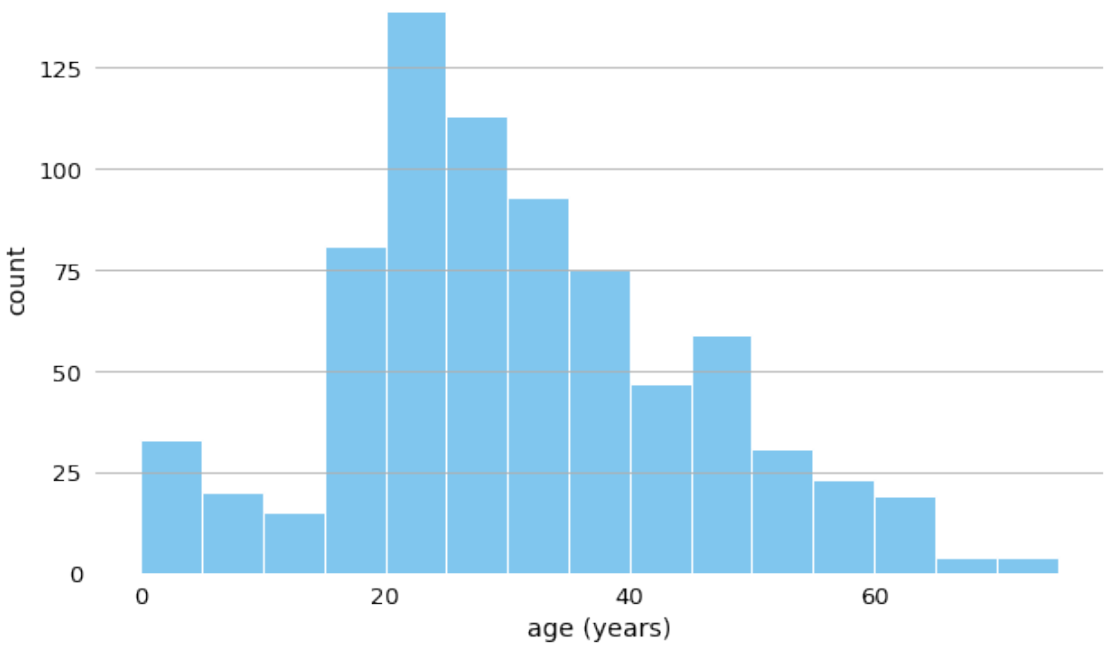
[8]:
```
titanic_age = titanic.dropna()
titanic_age['Age'] = np.round(titanic_age.Age) # change ages that less than 1␣
 ↪years old to 1
```

```
C:\Users\bpei\AppData\Local\Temp\ipykernel_1928\1395678662.py:2:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  titanic_age['Age'] = np.round(titanic_age.Age) # change ages that less than 1
years old to 1
```

[9]:
```
fig, ax = plt.subplots(1,1, figsize = (10,6))
sns.histplot(data=titanic_age.Age, binwidth=5, edgecolor='white', color =␣
 ↪'#56B4E9', ax=ax)
ax.spines[:].set_visible(False)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')

ax.xaxis.set_major_locator(ticker.MultipleLocator(20))
ax.yaxis.set_major_locator(ticker.MultipleLocator(25))
# ax.xaxis.set_minor_locator(ticker.MultipleLocator(1))

ax.yaxis.grid()
ax.tick_params(axis = 'both', which = 'major', labelsize = 13)
ax.set_xlabel('age (years)', fontsize = 14)
```

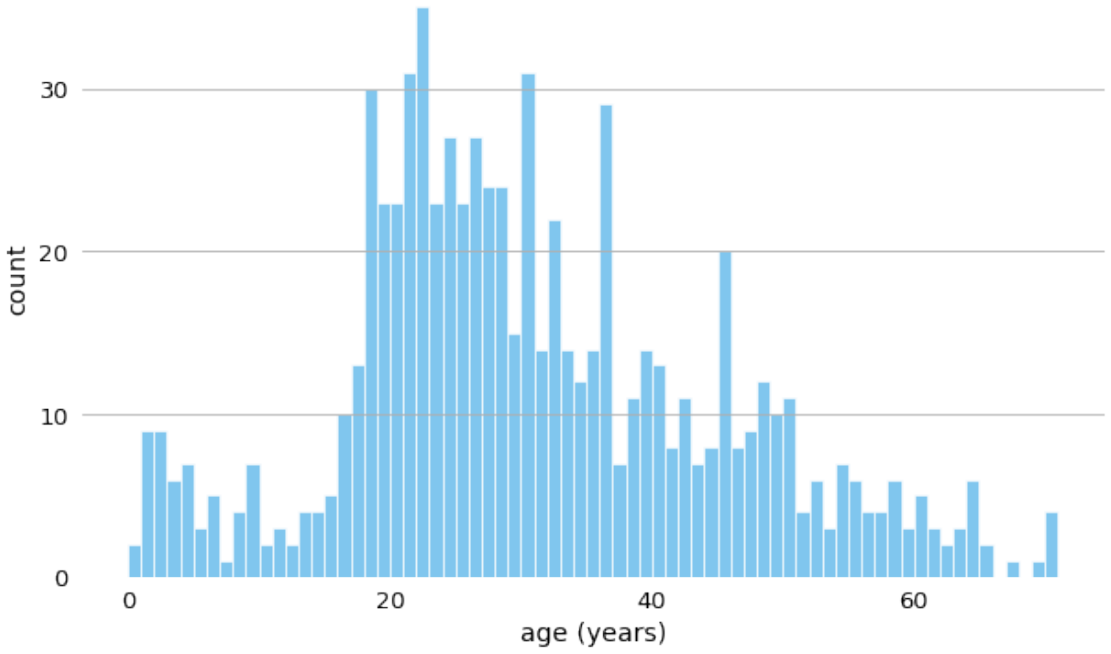
```
ax.set_ylabel('count', fontsize = 14)
ax.plot()
```

[9]: []



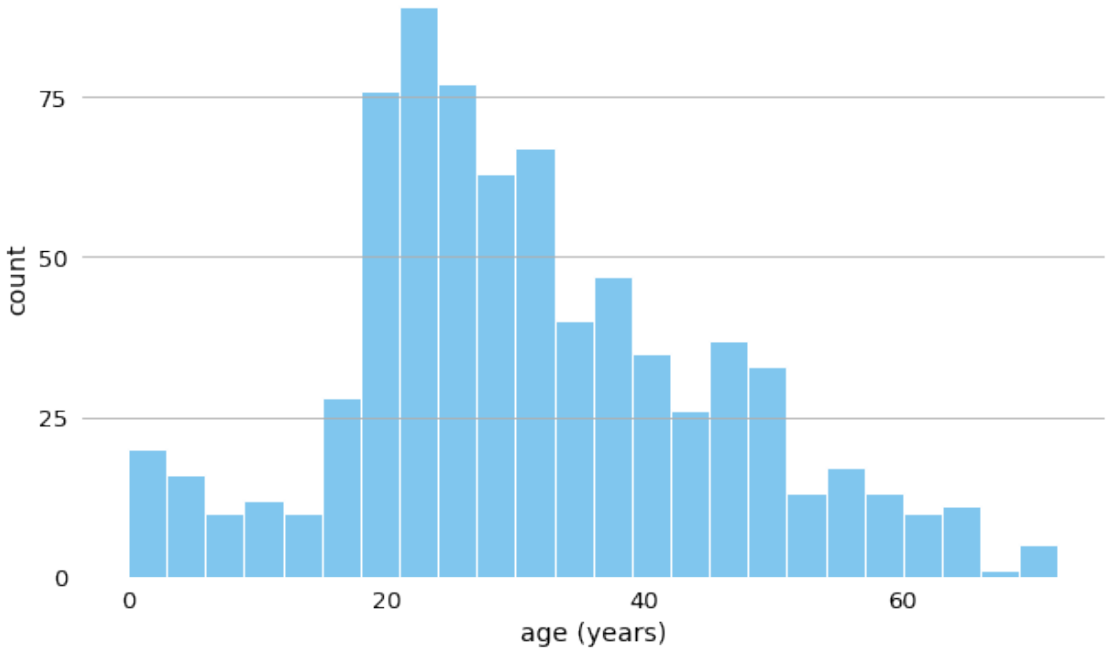
[6]:
```
fig, ax = plt.subplots(1,1, figsize = (10,6))
sns.histplot(data=titanic_age.Age, binwidth=1, edgecolor='white', color =␣
↪'#56B4E9', ax=ax)
ax.spines[:].set_visible(False)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')

ax.xaxis.set_major_locator(ticker.MultipleLocator(20))
ax.yaxis.set_major_locator(ticker.MultipleLocator(10))
# ax.xaxis.set_minor_locator(ticker.MultipleLocator(1))

ax.yaxis.grid()
ax.tick_params(axis = 'both', which = 'major', labelsize = 13)
ax.set_xlabel('age (years)', fontsize = 14)
ax.set_ylabel('count', fontsize = 14)
ax.plot()
```

[6]: []



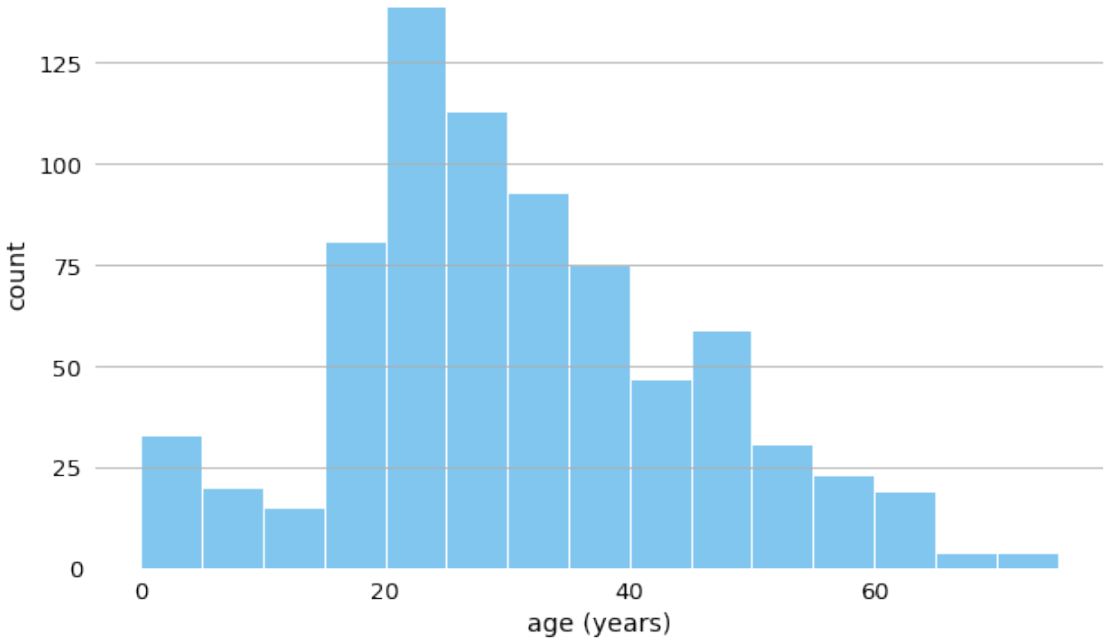
[7]:
```
fig, ax = plt.subplots(1,1, figsize = (10,6))
sns.histplot(data=titanic_age.Age, binwidth=3, edgecolor='white', color =␣
↪'#56B4E9', ax=ax)
ax.spines[:].set_visible(False)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')

ax.xaxis.set_major_locator(ticker.MultipleLocator(20))
ax.yaxis.set_major_locator(ticker.MultipleLocator(25))
# ax.xaxis.set_minor_locator(ticker.MultipleLocator(1))

ax.yaxis.grid()
ax.tick_params(axis = 'both', which = 'major', labelsize = 13)
ax.set_xlabel('age (years)', fontsize = 14)
ax.set_ylabel('count', fontsize = 14)
ax.plot()
```

[7]: []



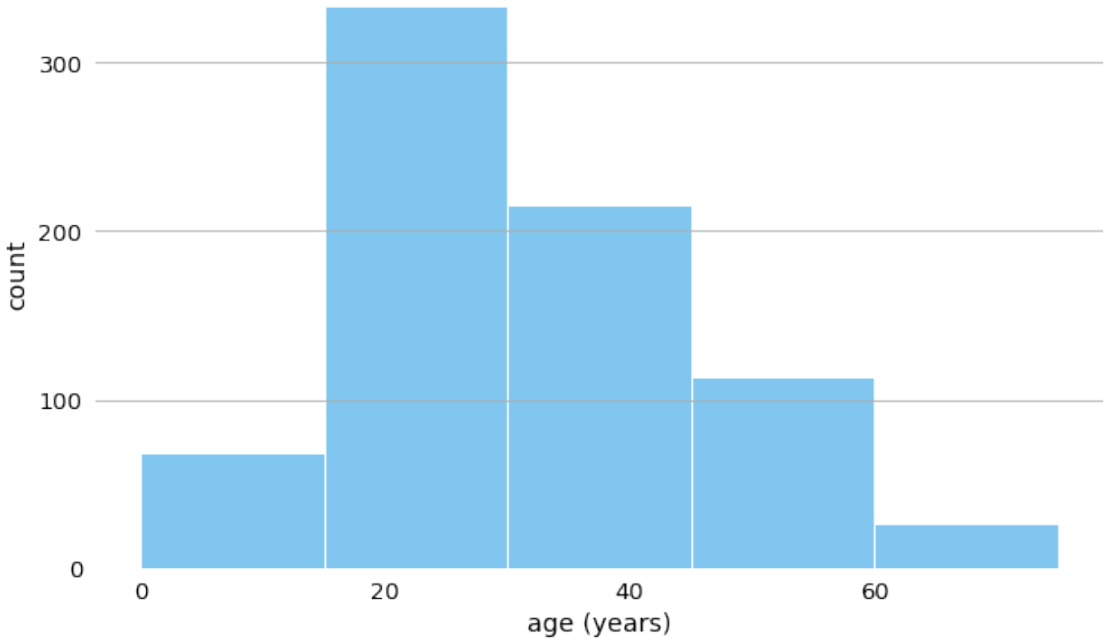
[8]:
```
fig, ax = plt.subplots(1,1, figsize = (10,6))
sns.histplot(data=titanic_age.Age, binwidth=5, edgecolor='white', color =␣
↪'#56B4E9', ax=ax)
ax.spines[:].set_visible(False)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')

ax.xaxis.set_major_locator(ticker.MultipleLocator(20))
ax.yaxis.set_major_locator(ticker.MultipleLocator(25))
# ax.xaxis.set_minor_locator(ticker.MultipleLocator(1))

ax.yaxis.grid()
ax.tick_params(axis = 'both', which = 'major', labelsize = 13)
ax.set_xlabel('age (years)', fontsize = 14)
ax.set_ylabel('count', fontsize = 14)
ax.plot()
```

[8]: []



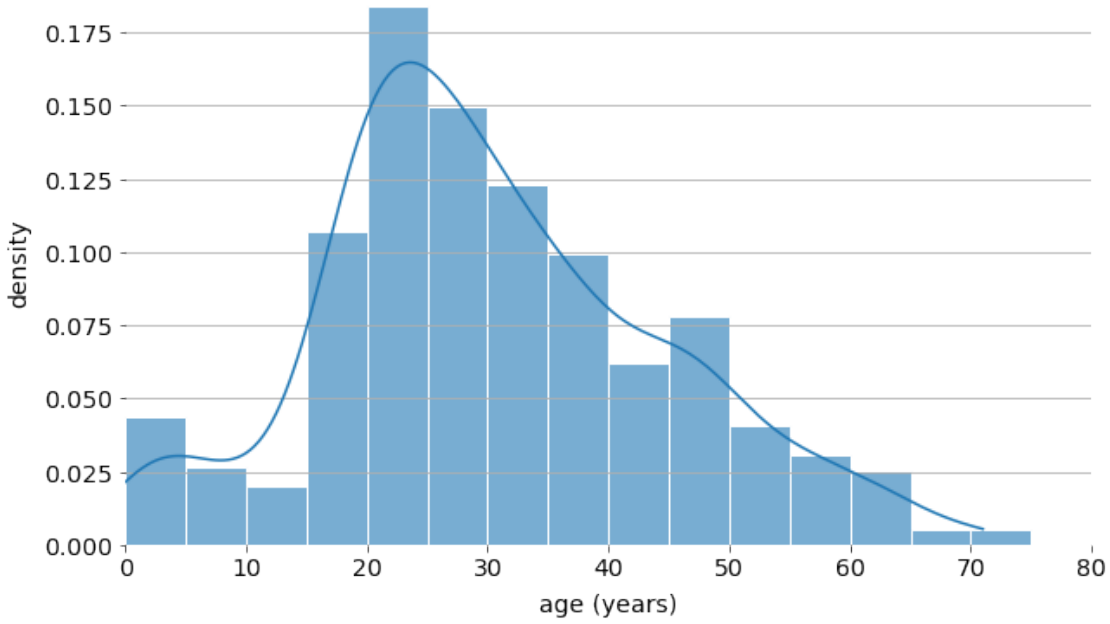
[9]:
```
fig, ax = plt.subplots(1,1, figsize = (10,6))
sns.histplot(data=titanic_age.Age, binwidth=15, edgecolor='white', color =␣
↪'#56B4E9', ax=ax)
ax.spines[:].set_visible(False)
ax.xaxis.set_ticks_position('none')
ax.yaxis.set_ticks_position('none')

ax.xaxis.set_major_locator(ticker.MultipleLocator(20))
ax.yaxis.set_major_locator(ticker.MultipleLocator(100))
# ax.xaxis.set_minor_locator(ticker.MultipleLocator(1))

ax.yaxis.grid()
ax.tick_params(axis = 'both', which = 'major', labelsize = 13)
ax.set_xlabel('age (years)', fontsize = 14)
ax.set_ylabel('count', fontsize = 14)
ax.plot()
```

[9]: []



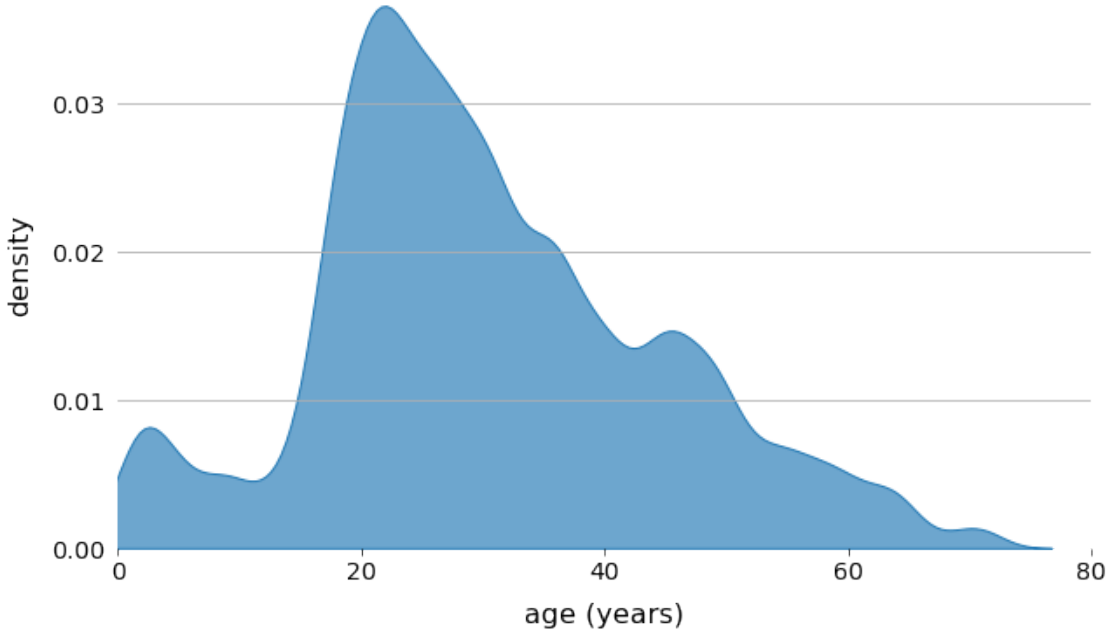
[39]:
```
from sklearn.neighbors import KernelDensity

fig, ax = plt.subplots(1,1,figsize = (10,6))
model = KernelDensity(bandwidth= 2, kernel= 'gaussian')
age_dist = titanic_age['Age'].values.reshape(-1,1)
model.fit(age_dist)

values = np.asarray([value for value in range(0,84)])
values = values.reshape(-1,1)
probabilities = model.score_samples(values)
probabilities = np.exp(probabilities)
# sns.histplot(age_dist, binwidth=5, kde=False, edgecolor='w', color='#56B4E9',␣
 ↪alpha = 0.7, stat='proportion') # proportion of each category in total
sns.histplot(age_dist, binwidth=5, kde=True, edgecolor='w', color='#56B4E9',␣
 ↪alpha = 0.6, stat='proportion') # proportion of each category in total

# plt.plot(values[:],probabilities*len(titanic_age), color='k')

ax.spines['left'].set_position(('data',0))
ax.spines['bottom'].set_position(('data',0))
ax.spines[:].set_visible(False)
ax.yaxis.grid()
ax.tick_params(axis = 'both', which = 'major', labelsize = 14)
ax.set_ylabel('density', fontsize=14, labelpad =7)
ax.set_xlabel('age (years)', fontsize=14, labelpad =7)
```

```
ax.set_xlim([0,80])
ax.get_legend().remove()
```

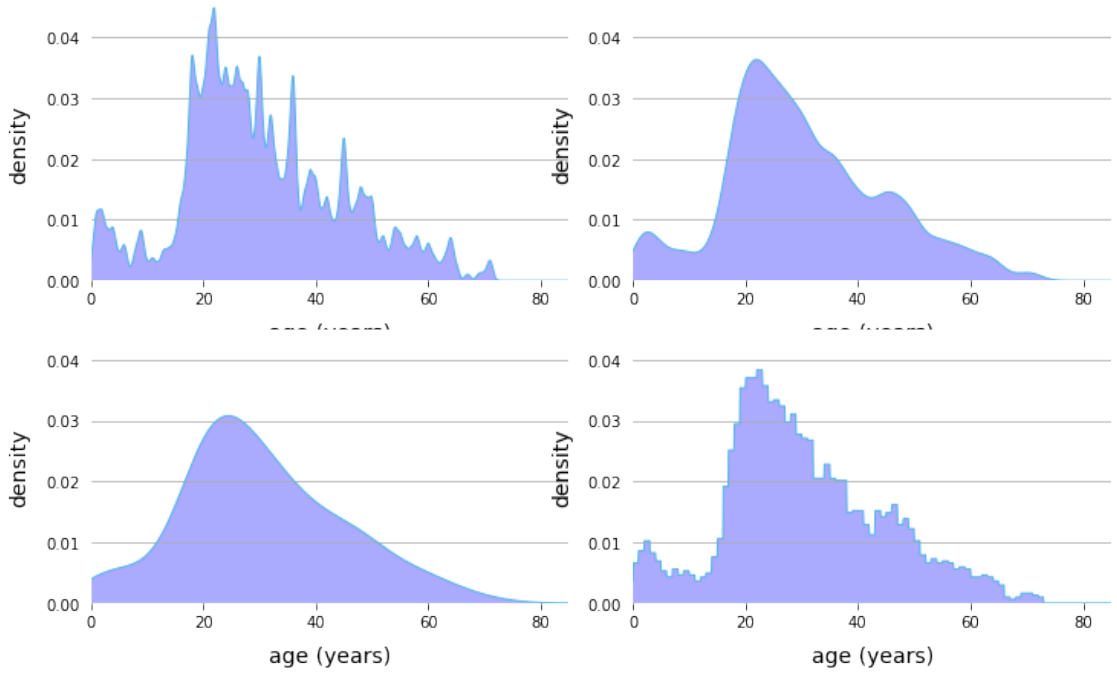


[11]:
```
fig, ax = plt.subplots(1,1, figsize = (10,6))
sns.kdeplot(titanic_age['Age'], fill='sky',alpha =.65, bw_adjust=.5)
ax.spines['left'].set_position(('data',0))
ax.spines[:].set_visible(False)
ax.xaxis.set_major_locator(ticker.MultipleLocator(20))
ax.yaxis.set_major_locator(ticker.MultipleLocator(0.01))
ax.tick_params(axis = 'both', which='major', labelsize = 14)
ax.yaxis.set_ticks_position('none')
ax.yaxis.grid()
ax.set_xlim([0,80])
ax.set_ylabel('density', fontsize =16, labelpad= 10)
ax.set_xlabel('age (years)', fontsize =16, labelpad= 10)
plt.plot()
```

[11]: []



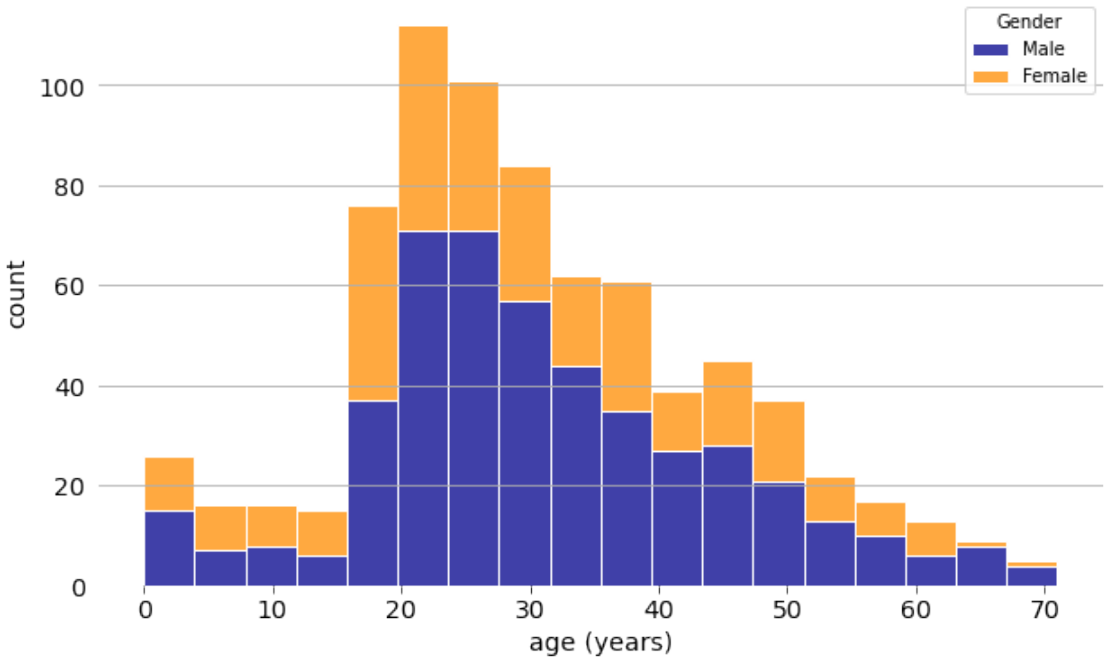
[12]:
```
fig, ax = plt.subplots(2,2,figsize=(10,6))
kernels = ['gaussian','gaussian','gaussian','tophat']
bandwidths = [0.5, 2, 5, 2]
x = titanic_age['Age'].values.reshape(-1,1)
x_plot = np.linspace(0, 85, 1000).reshape(-1,1)
fig.tight_layout()
for i, (kernel, bandwidth) in enumerate(zip(kernels, bandwidths)):
    kde = KernelDensity(kernel= kernel, bandwidth= bandwidth).fit(x)
    log_dens = kde.score_samples(x_plot)
    axi = ax.ravel()[i]
    axi.plot(x_plot[:,0], np.exp(log_dens), color='#56B4E9', linewidth=1)
    axi.fill_between(x_plot[:,0], np.exp(log_dens),fc='#AAAAFF')

    axi.spines[:].set_visible(False)
    axi.yaxis.grid()

    axi.yaxis.set_major_locator(ticker.MultipleLocator(0.01))
    axi.xaxis.set_major_locator(ticker.MultipleLocator(20))
    axi.yaxis.set_ticks_position('none')

    axi.set_xlim([0, 85])
    axi.set_ylim([0,0.045])
    axi.set_ylabel('density', fontsize = 14, labelpad= 10)
    axi.set_xlabel('age (years)', fontsize = 14, labelpad= 10)
```



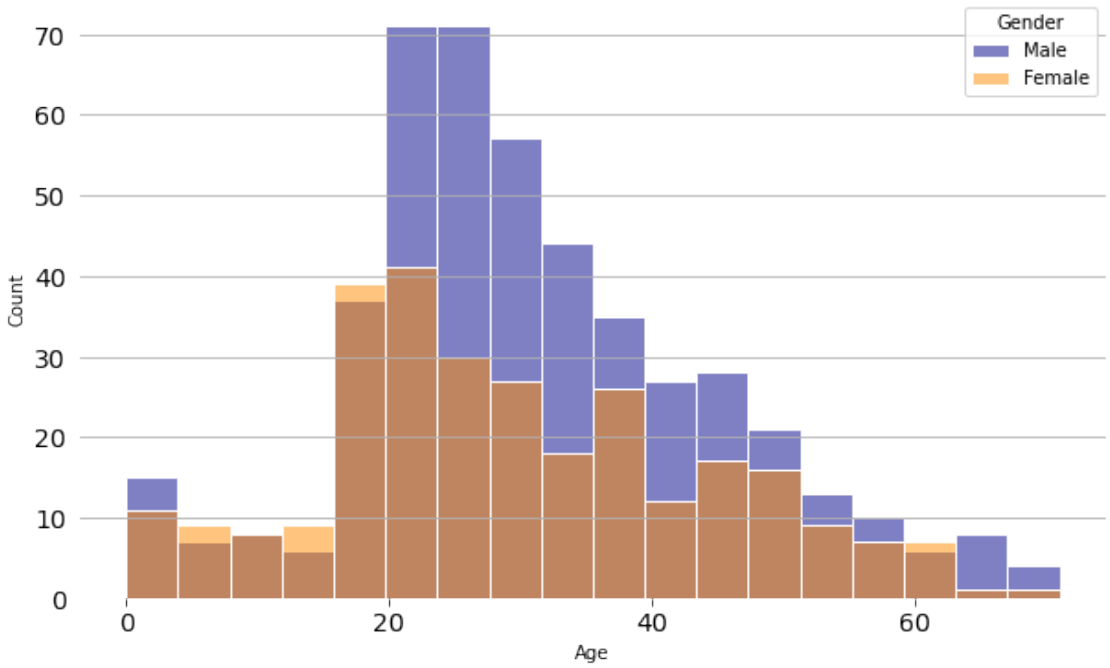
7.2 The drawbacks of stacked bar chart

[13]:
```
fig, ax = plt.subplots(1,1, figsize=(10,6))
axes = sns.histplot(titanic_age,
             x='Age',
             hue = 'SexCode',
             multiple='stack',
             hue_order=[1,0],
             color='red',
             palette=['darkorange','darkblue'],
             edgecolor='white'
             )

ax.legend(['Male','Female'], title='Gender')
ax.spines[:].set_visible(False)
ax.yaxis.set_ticks_position('none')
ax.yaxis.set_major_locator(ticker.MultipleLocator(20))
ax.tick_params(axis='both', which='major', labelsize=14)
ax.yaxis.grid()
ax.set_xlabel('age (years)', fontsize=14)
ax.set_ylabel('count',fontsize=14)
```

[13]:
```
Text(0, 0.5, 'count')
```



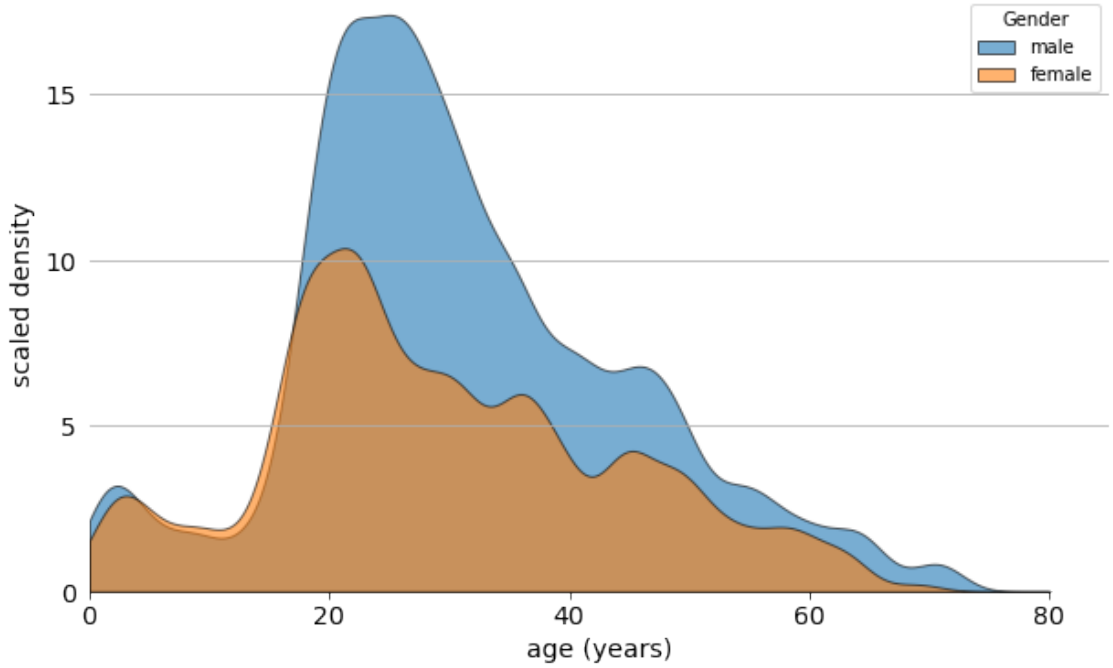
[14]:
```
fig, ax = plt.subplots(1,1, figsize=(10,6))
axes = sns.histplot(titanic_age,
             x='Age',
             hue = 'SexCode',
             hue_order=[1,0],
             color='red',
             palette=['darkorange','darkblue'],
             edgecolor='white',
             )

ax.legend(['Male','Female'],title='Gender')

ax.spines[:].set_visible(False)
ax.yaxis.set_ticks_position('none')
ax.xaxis.set_major_locator(ticker.MultipleLocator(20))
ax.tick_params(axis='both', which='major', labelsize = 14)
ax.yaxis.grid()
```



[15]:
```
fig, ax = plt.subplots(1,1, figsize=(10,6))
titanic_age_female = titanic_age[titanic_age['SexCode']==0]
titanic_age_male = titanic_age[titanic_age['SexCode']==1]
x_plot = np.linspace(0,80,1000).reshape(-1,1)
female_age = titanic_age_female['Age'].values.reshape(-1,1)
male_age = titanic_age_male['Age'].values.reshape(-1,1)

kde_female = KernelDensity(kernel='gaussian', bandwidth=2).fit(female_age)
kde_male = KernelDensity(kernel='gaussian', bandwidth=2).fit(male_age)

log_dens_female = kde_female.score_samples(x_plot)
log_dens_male = kde_male.score_samples(x_plot)
# ax.plot(x_plot[:,0], np.exp(log_dens_female))
ax.fill_between(x_plot[:,0], np.
 ↪exp(log_dens_female)*len(titanic_age_female),edgecolor='k', alpha=0.6)
# ax.plot(x_plot[:,0], np.exp(log_dens_female)*len(titanic_age_female), alpha=0.
 ↪6, color='k')
ax.fill_between(x_plot[:,0], np.
 ↪exp(log_dens_male)*len(titanic_age_male),edgecolor='k',alpha=0.6)
# ax.plot(x_plot[:,0], np.exp(log_dens_male)*len(titanic_age_male), alpha=0.6,␣
 ↪color='k')

ax.yaxis.set_major_locator(ticker.MultipleLocator(5))
ax.xaxis.set_major_locator(ticker.MultipleLocator(20))
```

```
ax.spines[:].set_visible(False)
ax.yaxis.set_ticks_position('none')
ax.yaxis.grid()
ax.tick_params(axis='both', which = 'major', labelsize=14)
ax.set_xlim([0,85])
ax.set_ylim([0,18])

ax.set_xlabel('age (years)', fontsize=14)
ax.set_ylabel('scaled density', fontsize=14)
ax.legend(['male','female'],title='Gender')
```

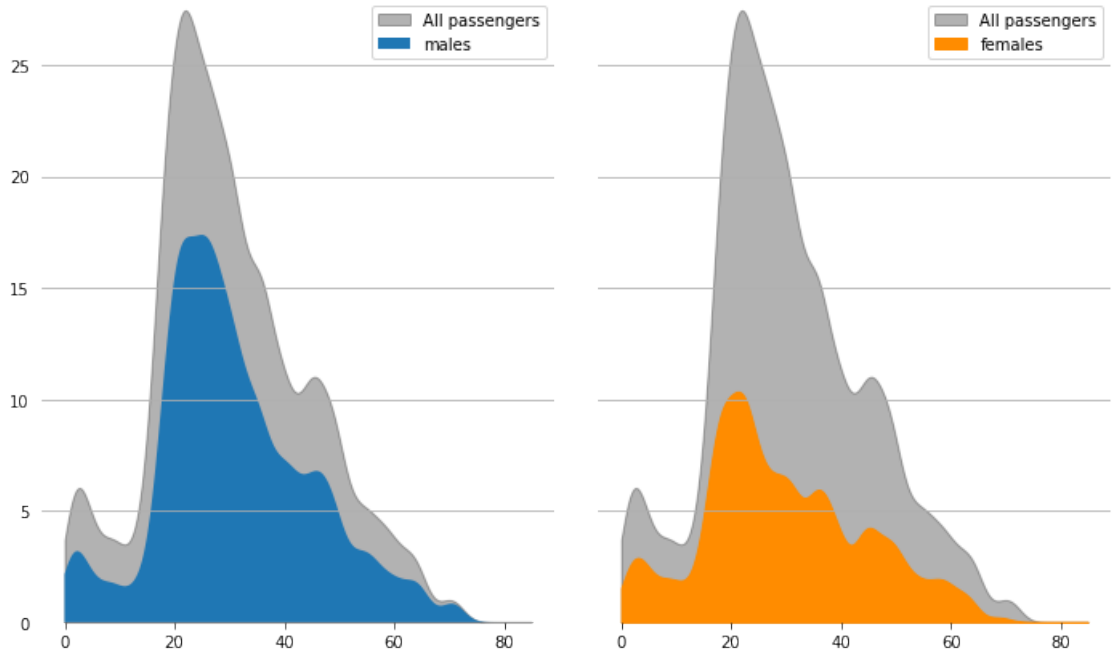
[15]: <matplotlib.legend.Legend at 0x256073f7be0>



[16]:
```
fig, (ax1, ax2) = plt.subplots(1,2,figsize =(10,6),sharey=True)
# fig.subplots_adjust(wspace=0)
x_plot = np.linspace(0,85,1000).reshape(-1,1)
total_age = titanic_age['Age'].values.reshape(-1,1)
titanic_age_male = titanic_age[titanic_age['SexCode']==0]['Age'].values.
 ↪reshape(-1,1)
titanic_age_female = titanic_age[titanic_age['SexCode']==1]['Age'].values.
 ↪reshape(-1,1)
fig.tight_layout()
kde_total = KernelDensity(kernel='gaussian', bandwidth=2).fit(total_age)
log_dens = kde_total.score_samples(x_plot)
```

```
ax1.fill_between(x_plot[:,0], np.
 ↪exp(log_dens)*len(titanic_age),color='black',alpha=0.3, label='All␣
 ↪passengers')
ax2.fill_between(x_plot[:,0], np.
 ↪exp(log_dens)*len(titanic_age),color='black',alpha=0.3, label='All␣
 ↪passengers')

kde_male = KernelDensity(kernel='gaussian',bandwidth=2).fit(titanic_age_male)
log_dens_male = kde_male.score_samples(x_plot)
ax1.fill_between(x_plot[:,0],np.exp(log_dens_male)*len(titanic_age_male),␣
 ↪label='males')
ax1.spines[:].set_visible(False)
ax1.yaxis.set_ticks_position('none')
ax1.yaxis.grid()
ax1.set_ylim([0,28])


kde_female = KernelDensity(kernel='gaussian',bandwidth=2).
 ↪fit(titanic_age_female)
log_dens_female = kde_female.score_samples(x_plot)
ax2.fill_between(x_plot[:,0],np.
 ↪exp(log_dens_female)*len(titanic_age_female),color='darkorange',label='females')
ax2.spines[:].set_visible(False)
ax2.yaxis.set_ticks_position('none')
ax2.yaxis.grid()
ax2.set_ylim([0,28])
ax1.legend()
ax2.legend()
```

[16]: <matplotlib.legend.Legend at 0x256073d98b0>

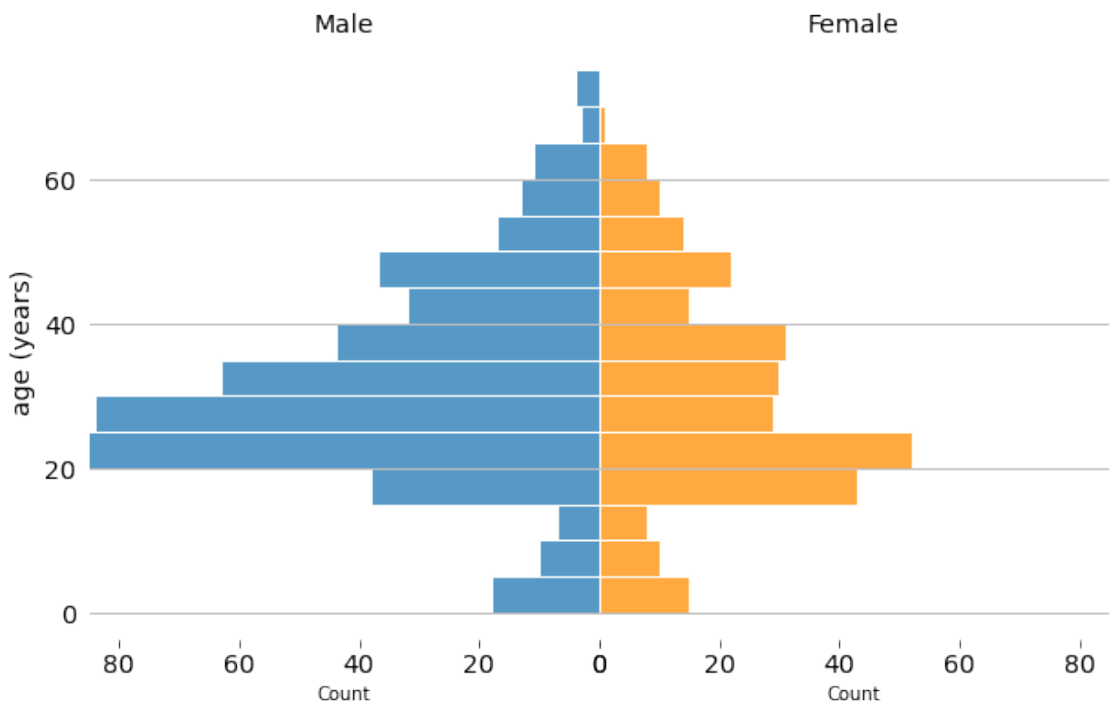


bidirectional bar chart

[17]:
```
titanic_age_male = titanic_age[titanic_age['SexCode']==0]
titanic_age_female = titanic_age[titanic_age['SexCode']==1]

fig, (ax1, ax2) = plt.subplots(1,2,figsize=(10,6), sharey=True)
fig.subplots_adjust(wspace=0)
sns.histplot(
    data=titanic_age_male,
    # y='SexCode',
    y='Age',
    binwidth=5,
    edgecolor='white',
    ax=ax1
    )

ax1.spines[:].set_visible(False)
ax1.yaxis.set_major_locator(ticker.MultipleLocator(20))
ax1.xaxis.set_major_locator(ticker.MultipleLocator(20))
ax1.yaxis.set_ticks_position('none')
ax1.tick_params(axis='both', which='major', labelsize=14)
ax1.yaxis.grid()
ax1.set_ylabel('age (years)', fontsize=14)
ax1.set_xlim([0,85])
ax1.set_title('Male',fontsize=14)
ax1.invert_xaxis()
```

```
sns.histplot(
    data=titanic_age_female,
    # y='SexCode',
    y='Age',
    binwidth=5,
    color='darkorange',
    edgecolor='white',
    ax=ax2
    )
ax2.spines[:].set_visible(False)
ax2.xaxis.set_major_locator(ticker.MultipleLocator(20))
ax2.yaxis.set_ticks_position('none')
ax2.tick_params(axis='both', which='major', labelsize=14)
ax2.yaxis.grid()
ax2.set_xlim([0,85])
ax2.set_title('Female',fontsize=14)
```

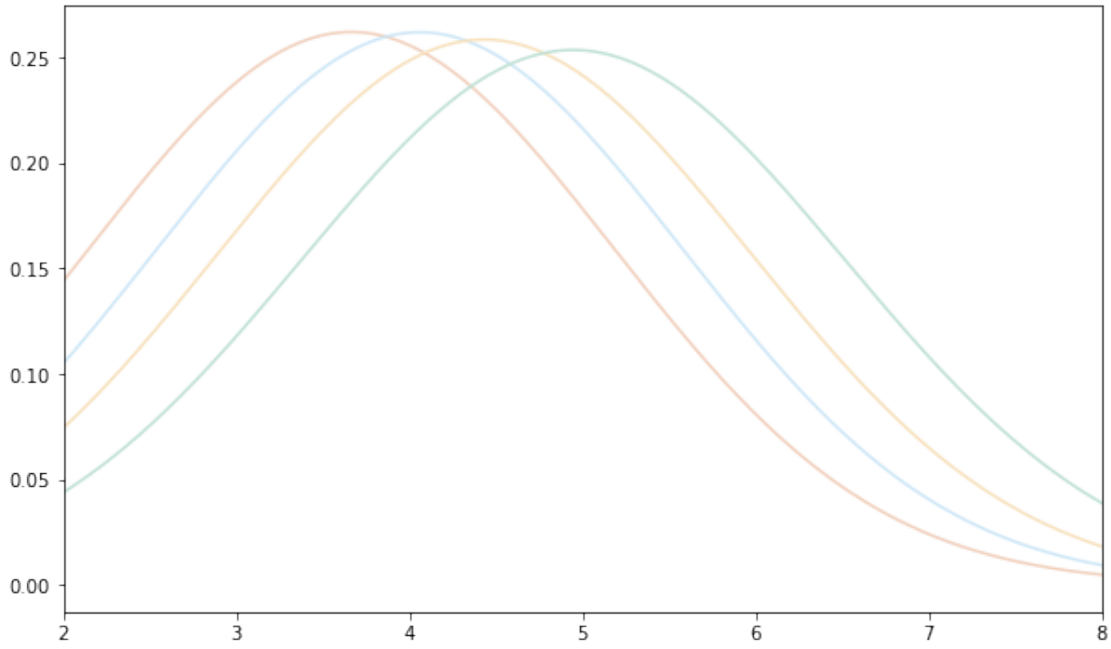
[17]: `Text(0.5, 1.0, 'Female')`



[43]:
```
cow_milk = pd.read_csv(os.path.join('.','data','cows_milk.csv'))
HolsteinF = cow_milk[cow_milk['breed']=='Holstein-Friesian']
Ayrshire = cow_milk[cow_milk['breed']=='Ayrshire']
canadian = cow_milk[cow_milk['breed']=='Canadian']
```

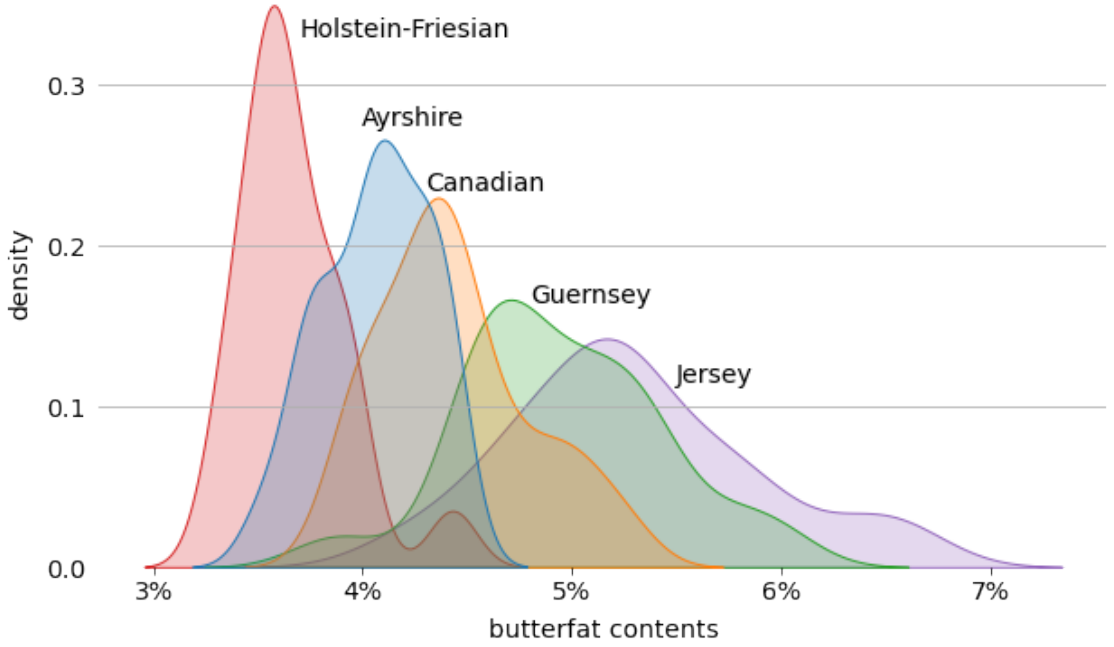
```
guernsey = cow_milk[cow_milk['breed']=='Guernsey']
jersey = cow_milk[cow_milk['breed']=='Jersey']
```

[113]:
```
fig, ax = plt.subplots(1,1,figsize=(10,6))
x_plot = np.linspace(0,10,1000).reshape(-1,1)
for cat, color in zip([HolsteinF, Ayrshire, canadian, guernsey,␣
 ↪jersey],['#f2d1be','#cfe8f8','#f8e2be','#c1e2d6']):
    kde = KernelDensity(kernel='gaussian', bandwidth=1.5).fit(cat['butterfat'].
 ↪values.reshape(-1,1))
    log_dens = kde.score_samples(x_plot)
    ax.plot(x_plot[:,0],np.exp(log_dens),color=color)
    ax.set_xlim([2,8])
```



[48]:
```
fig, ax = plt.subplots(figsize=(10,6))
axes = sns.kdeplot(data=cow_milk,
                   x='butterfat',
                   hue='breed',
                   bw_adjust=.8,
                   fill=True
                   )
ax.spines[:].set_visible(False)
ax.yaxis.set_ticks_position('none')
ax.yaxis.set_major_locator(ticker.MultipleLocator(0.1))
ax.tick_params(axis='both', which = 'major', labelsize=14)
ax.yaxis.grid()
```

```
ax.text(3.7, 0.33,'Holstein-Friesian',fontsize=14)
ax.text(4, 0.275,'Ayrshire',fontsize=14)
ax.text(4.3, 0.235,'Canadian',fontsize=14)
ax.text(4.8, 0.165,'Guernsey',fontsize=14)
ax.text(5.5, 0.115,'Jersey',fontsize=14)
ax.get_legend().set_visible(False)
ax.set_ylabel('density',fontsize=14,labelpad=7)
ax.set_xlabel('butterfat contents',fontsize=14,labelpad=7)
ax.xaxis.set_ticks([3,4,5,6,7],['3%','4%','5%','6%','7%'])
plt.show()
```



[ ]: