In [33]:
```
#1 program
import pandas as pd
import numpy as np
from scipy.io import arff
from sklearn.preprocessing import LabelEncoder
data = arff.loadarff('speeddating.arff')
train= pd.DataFrame(data[0])
train.head()
catCols = [col for col in train.columns if train[col].dtype=="O"]
catCols[:5]
train[catCols]=train[catCols].apply(lambda x: x.str.decode('utf8'))
train
```

Out[33]:

	has_null	wave	gender	age	age_o	d_age	d_d_age	race	race
0	0	1.0	female	21.0	27.0	6.0	[4-6]	Asian/Pacific Islander/Asian-American	European/Caucasi Ameri
1	0	1.0	female	21.0	22.0	1.0	[0-1]	Asian/Pacific Islander/Asian-American	European/Caucasi Ameri
2	1	1.0	female	21.0	22.0	1.0	[0-1]	Asian/Pacific Islander/Asian-American	Asian/Pac Islander/Asi Ameri
3	0	1.0	female	21.0	23.0	2.0	[2-3]	Asian/Pacific Islander/Asian-American	European/Caucasi Ameri
4	0	1.0	female	21.0	24.0	3.0	[2-3]	Asian/Pacific Islander/Asian-American	Latino/Hispa Ameri
...	...	...	...	...	...	...	...	...	
8373	1	21.0	male	25.0	26.0	1.0	[0-1]	European/Caucasian-American	Latino/Hispa Ameri
8374	1	21.0	male	25.0	24.0	1.0	[0-1]	European/Caucasian-American	Ot
8375	1	21.0	male	25.0	29.0	4.0	[4-6]	European/Caucasian-American	Latino/Hispa Ameri
8376	1	21.0	male	25.0	22.0	3.0	[2-3]	European/Caucasian-American	Asian/Pac Islander/Asi Ameri
8377	1	21.0	male	25.0	22.0	3.0	[2-3]	European/Caucasian-American	Asian/Pac Islander/Asi Ameri

8378 rows × 123 columns

In [34]:
```
#2nd program
from sklearn.preprocessing import LabelEncoder
l = LabelEncoder()
label = l.fit_transform(train['gender'])
train.drop("gender", axis=1, inplace=False)
train["gender"] = label
train
```

Out[34]:

	has_null	wave	gender	age	age_o	d_age	d_d_age	race	race_o
0	0	1.0	0	21.0	27.0	6.0	[4-6]	Asian/Pacific Islander/Asian-American	European/Caucasian-American
1	0	1.0	0	21.0	22.0	1.0	[0-1]	Asian/Pacific Islander/Asian-American	European/Caucasian-American
2	1	1.0	0	21.0	22.0	1.0	[0-1]	Asian/Pacific Islander/Asian-American	Asian/Pacific Islander/Asian-American
3	0	1.0	0	21.0	23.0	2.0	[2-3]	Asian/Pacific Islander/Asian-American	European/Caucasian-American
4	0	1.0	0	21.0	24.0	3.0	[2-3]	Asian/Pacific Islander/Asian-American	Latino/Hispanic American
...	...	...	...	...	...	...	...	...	...
373	1	21.0	1	25.0	26.0	1.0	[0-1]	European/Caucasian-American	Latino/Hispanic American
374	1	21.0	1	25.0	24.0	1.0	[0-1]	European/Caucasian-American	Other
375	1	21.0	1	25.0	29.0	4.0	[4-6]	European/Caucasian-American	Latino/Hispanic American
376	1	21.0	1	25.0	22.0	3.0	[2-3]	European/Caucasian-American	Asian/Pacific Islander/Asian-American
377	1	21.0	1	25.0	22.0	3.0	[2-3]	European/Caucasian-American	Asian/Pacific Islander/Asian-American

78 rows × 123 columns

In [25]:
```
train['age'].mean()
```

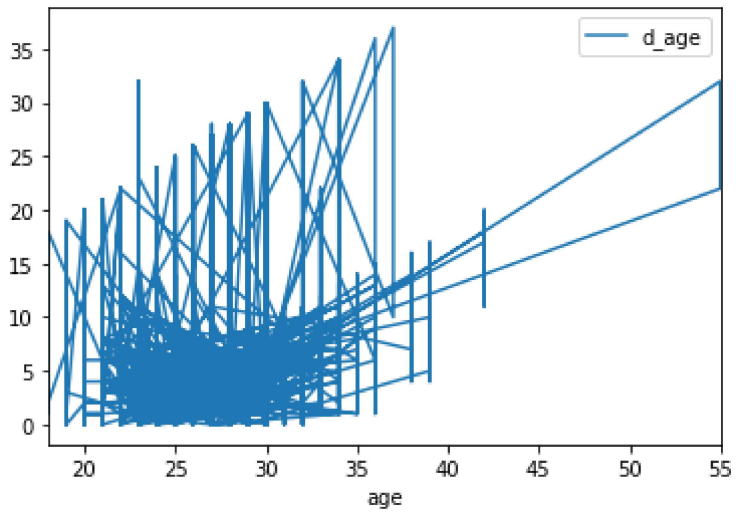
Out[25]:
```
26.358927924664975
```

In [24]:
```
print(train.dtypes)
```

```
has_null                object
wave                   float64
gender                   int32
age                    float64
age_o                  float64
                        ...
d_guess_prob_liked      object
met                    float64
decision                object
decision_o              object
match                   object
Length: 123, dtype: object
```

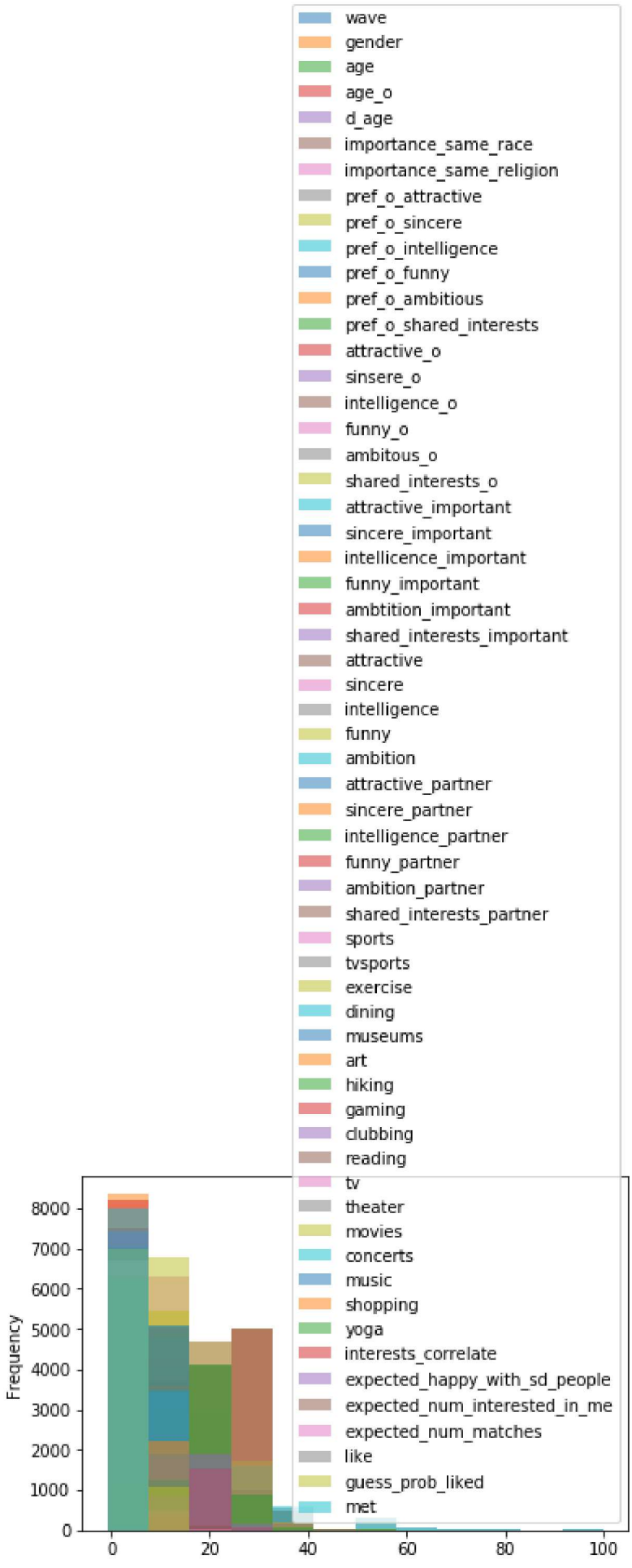
In [21]:
```
#3rd program
import matplotlib.pyplot as plt
train.plot(x="age",y="d_age")
```

Out[21]: <matplotlib.axes._subplots.AxesSubplot at 0x2350bc84788>



In [38]:
```
train.plot.hist(bins=12,alpha=0.5)
```

Out[38]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2ea8128ae08>
```


wave
gender
age
age_o
d_age
importance_same_race
importance_same_religion
pref_o_attractive
pref_o_sincere
pref_o_intelligence
pref_o_funny
pref_o_ambitious
pref_o_shared_interests
attractive_o
sinsere_o
intelligence_o
funny_o
ambitous_o
shared_interests_o
attractive_important
sincere_important
intellicence_important
funny_important
ambtition_important
shared_interests_important
attractive
sincere
intelligence
funny
ambition
attractive_partner
sincere_partner
intelligence_partner
funny_partner
ambition_partner
shared_interests_partner
sports
tvsports
exercise
dining
museums
art
hiking
gaming
clubbing
reading
tv
theater
movies
concerts
music
shopping
yoga
interests_correlate
expected_happy_with_sd_people
expected_num_interested_in_me
expected_num_matches
like
guess_prob_liked
met
Frequency
0
1000
2000
3000
4000
5000
6000
7000
8000
0
20
40
60
80
100

In [ ]:

Out[7]: <matplotlib.axes._subplots.AxesSubplot at 0x2ca86890848>

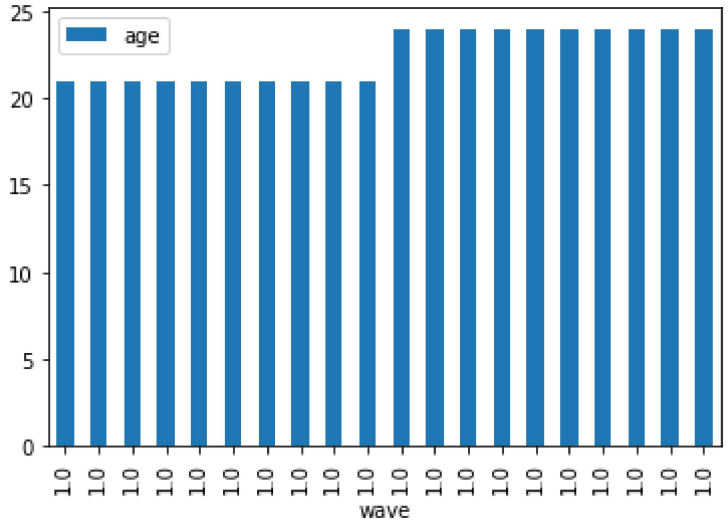
In [23]:
```
import pandas as pd
import numpy as np
from scipy.io import arff
from sklearn.preprocessing import LabelEncoder
data = arff.loadarff('speeddating.arff')
train= pd.DataFrame(data[0])
train.plot.line()
```

Out[23]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x23510a28688>
```


wave
age
age_o
d_age
importance_same_race
importance_same_religion
pref_o_attractive
pref_o_sincere
pref_o_intelligence
pref_o_funny
pref_o_ambitious
pref_o_shared_interests
attractive_o
sinsere_o
intelligence_o
funny_o
ambitous_o
shared_interests_o
attractive_important
sincere_important
intellicence_important
funny_important
ambtition_important
shared_interests_important
attractive
sincere
intelligence
funny
ambition
attractive_partner
sincere_partner
intelligence_partner
funny_partner
ambition_partner
shared_interests_partner
sports
tvsports
exercise
dining
museums
art
hiking
gaming
clubbing
reading
tv
theater
movies
concerts
music
shopping
yoga
interests_correlate
expected_happy_with_sd_people
expected_num_interested_in_me
expected_num_matches
like
guess_prob_liked
met
100
80
60
40
20
0
0
1000
2000
3000
4000
5000
6000
7000
8000

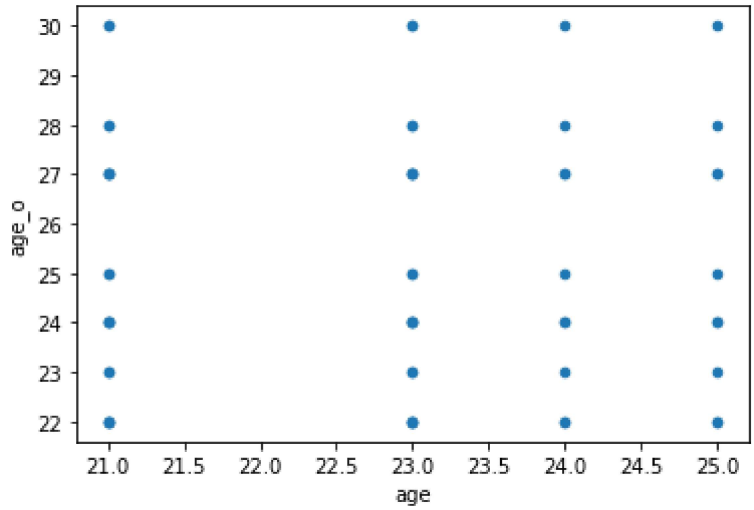
In [18]:
```
import pandas as pd
from scipy.io import arff
from sklearn.preprocessing import LabelEncoder
data = arff.loadarff('speeddating.arff')
train= pd.DataFrame(data[0])
train.head(20).plot.bar(x='wave',y='age')
```

Out[18]: <matplotlib.axes._subplots.AxesSubplot at 0x2350bc71e48>



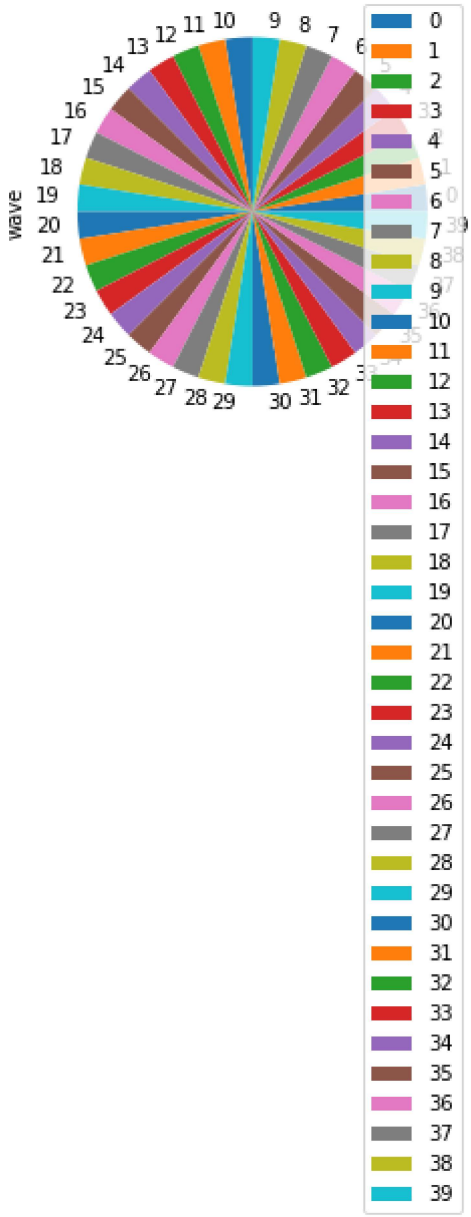
In [8]:
```
train.head(60).plot.scatter(x='age',y='age_o')
```

Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x23502b5ef48>



In [10]:
```
train.head(40).plot.pie(x='age',y='wave')
```

Out[10]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x23506834588>
```



In [11]:

0	b'0'	1.0	b'female'	21.0	27.0	0.162162	b'[4-6]'	b'Asian/Pacific Islander/Asian-American'	b'European/Caucas Americ
1	b'0'	1.0	b'female'	21.0	22.0	0.027027	b'[0-1]'	b'Asian/Pacific Islander/Asian-American'	b'European/Caucas Americ
2	b'1'	1.0	b'female'	21.0	22.0	0.027027	b'[0-1]'	b'Asian/Pacific Islander/Asian-American'	b'Asian/Pac Islander/As Americ
3	b'0'	1.0	b'female'	21.0	23.0	0.054054	b'[2-3]'	b'Asian/Pacific Islander/Asian-American'	b'European/Caucas Americ
4	b'0'	1.0	b'female'	21.0	24.0	0.081081	b'[2-3]'	b'Asian/Pacific Islander/Asian-American'	b'Latino/Hispa Americ

5 rows × 123 columns

In [16]:
```
import numpy as np
from scipy.io import arff
from sklearn.preprocessing import LabelEncoder
data = arff.loadarff('speeddating.arff')
train= pd.DataFrame(data[0])
catCols = [col for col in train.columns if train[col].dtype=="O"]
catCols[:5]
train[catCols]=train[catCols].apply(lambda x: x.str.decode('utf8'))
train=train.replace('?',np.nan)
train
```

Out[16]:

race	...	d_expected_num_interested_in_me	d_expected_num_matches	like	guess_prob_liked	d_like
0	...	[0-3]	[3-5]	7.0	6.0	[6-8]
0	...	[0-3]	[3-5]	7.0	5.0	[6-8]
1	...	[0-3]	[3-5]	7.0	NaN	[6-8]
0	...	[0-3]	[3-5]	7.0	6.0	[6-8]
0	...	[0-3]	[3-5]	6.0	6.0	[6-8]
...	...	...	...	...	...	...
0	...	[0-3]	[3-5]	2.0	5.0	[0-5]
0	...	[0-3]	[3-5]	4.0	4.0	[0-5]
0	...	[0-3]	[3-5]	6.0	5.0	[6-8]
0	...	[0-3]	[3-5]	5.0	5.0	[0-5]
0	...	[0-3]	[3-5]	4.0	5.0	[0-5]

In [25]:
```
#4th program
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
x=train.iloc[:,-1]
y=train.iloc[:,-2]
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=1/3,random_state=0)
```

In [ ]:
```

```