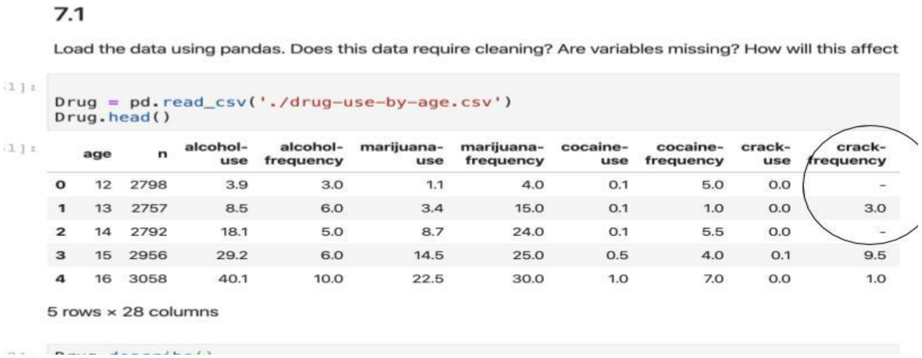
Project 2 SAT and DRUG DATA

Question 7.1

Missing Data in Drug

I noticed there was some missing data in the dataframe and that the missing values were marked with ‘-’. I wanted to find out how much missing data there was



7.1

Load the data using pandas. Does this data require cleaning? Are variables missing? How will this affect

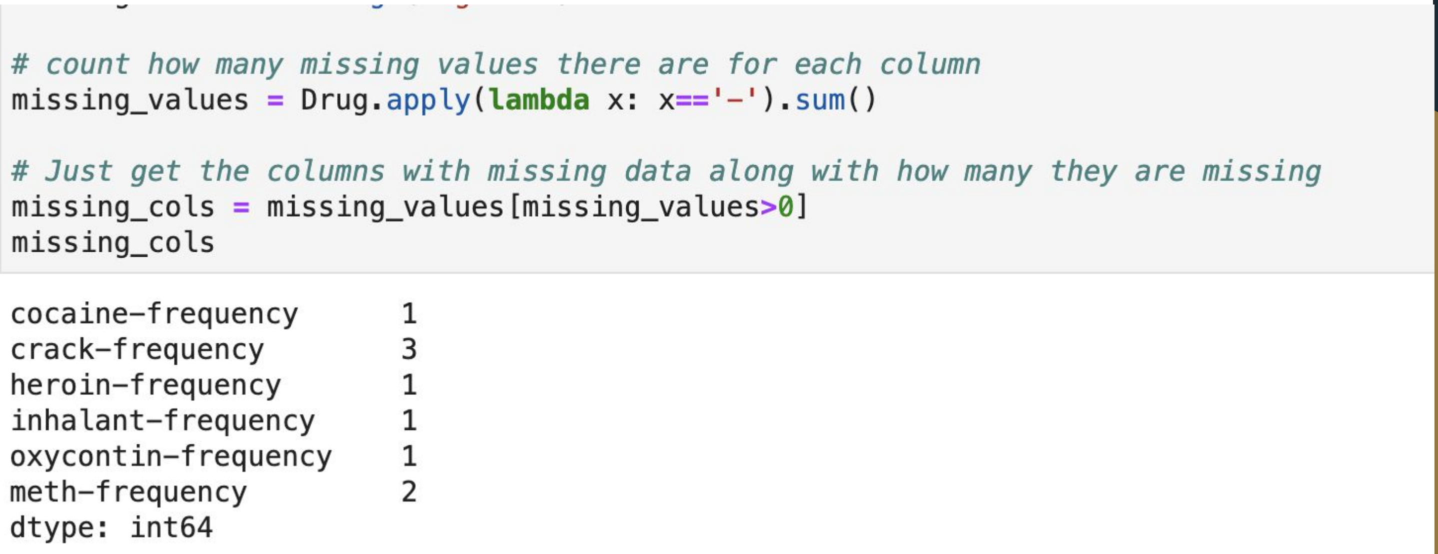
```
Drug = pd.read_csv('./drug-use-by-age.csv')
Drug.head()
```

	age	n	alcohol-use	alcohol-frequency	marijuana-use	marijuana-frequency	cocaine-use	cocaine-frequency	crack-use	crack-frequency
0	12	2798	3.9	3.0	1.1	4.0	0.1	5.0	0.0	-
1	13	2757	8.5	6.0	3.4	15.0	0.1	1.0	0.0	3.0
2	14	2792	18.1	5.0	8.7	24.0	0.1	5.5	0.0	-
3	15	2956	29.2	6.0	14.5	25.0	0.5	4.0	0.1	9.5
4	16	3058	40.1	10.0	22.5	30.0	1.0	7.0	0.0	1.0

5 rows × 28 columns

I used this code to find out

There were missing values in several columns, but not a large total number



```
# count how many missing values there are for each column
missing_values = Drug.apply(lambda x: x=='-').sum()

# Just get the columns with missing data along with how many they are missing
missing_cols = missing_values[missing_values>0]
missing_cols
```

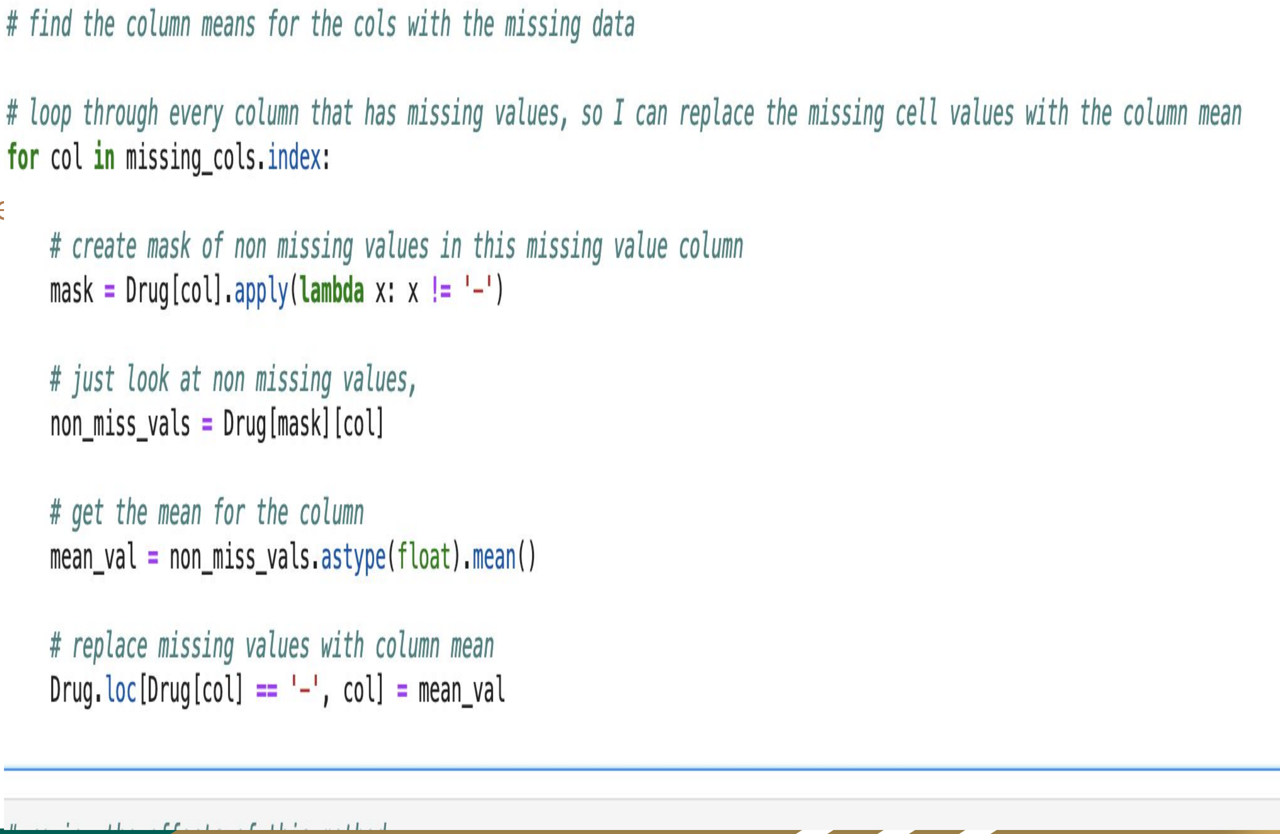
```
cocaine-frequency      1
crack-frequency        3
heroin-frequency       1
inhalant-frequency     1
oxycontin-frequency    1
meth-frequency         2
dtype: int64
```

Replace Missing Values- Code Notes

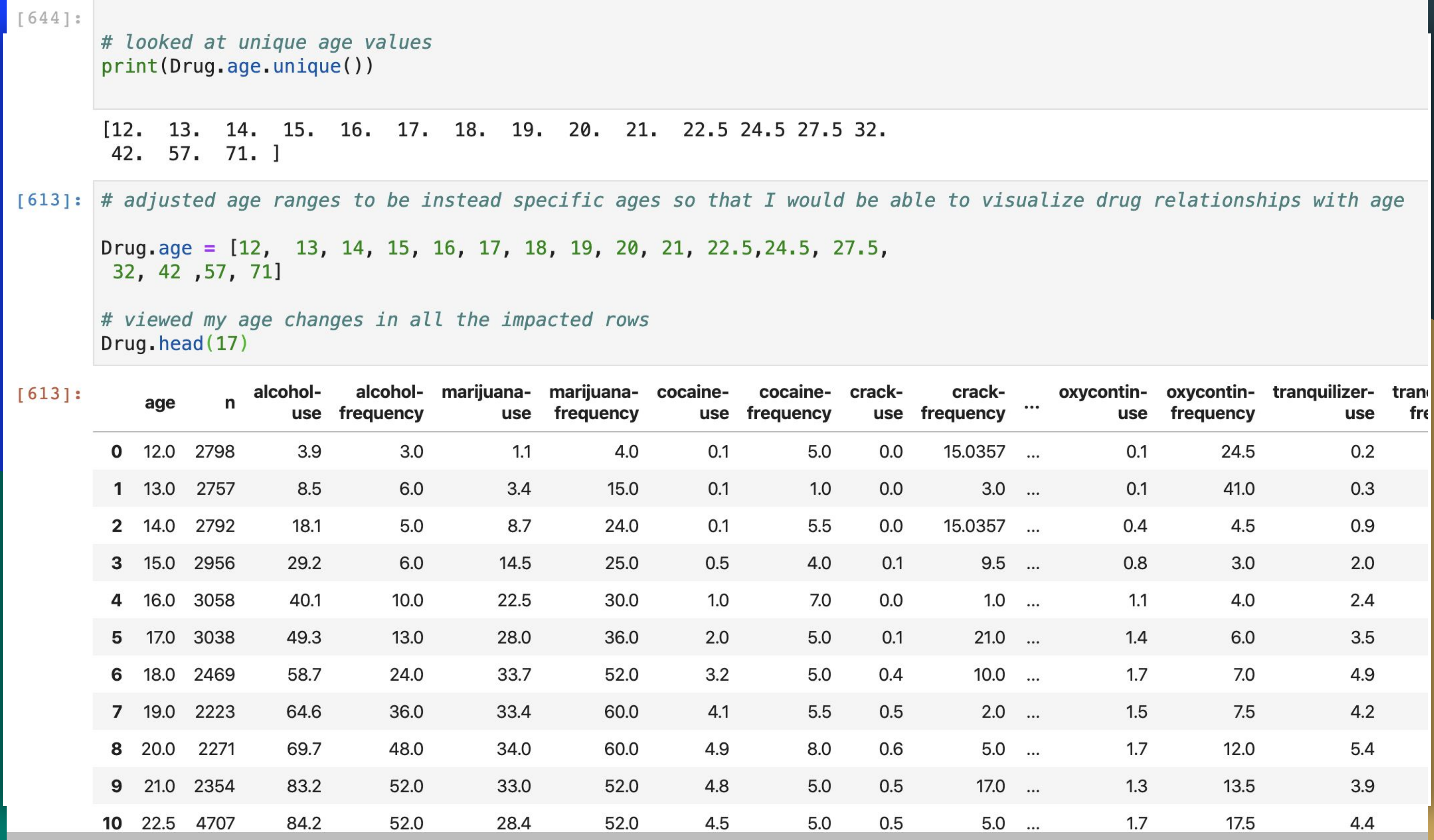
At worst 17.6% of the data per column was. In most cases only 5.6% was. I decided not to remove the missing data columns them from the dataframe. I considered options for replacing the missing values:

1. Use the column means.
2. Use neighboring data to assign approximate value to missing fields.

I decided to use column means.



```
# find the column means for the cols with the missing data

# loop through every column that has missing values, so I can replace the missing cell values with the column mean
for col in missing_cols.index:

    # create mask of non missing values in this missing value column
    mask = Drug[col].apply(lambda x: x != '-')

    # just look at non missing values,
    non_miss_vals = Drug[mask][col]

    # get the mean for the column
    mean_val = non_miss_vals.astype(float).mean()

    # replace missing values with column mean
    Drug.loc[Drug[col] == '-', col] = mean_val
```



[644]:

```
# looked at unique age values
print(Drug.age.unique())
```

```
[12.   13.   14.   15.   16.   17.   18.   19.   20.   21.   22.5 24.5 27.5 32.
 42.   57.   71. ]
```

[613]:

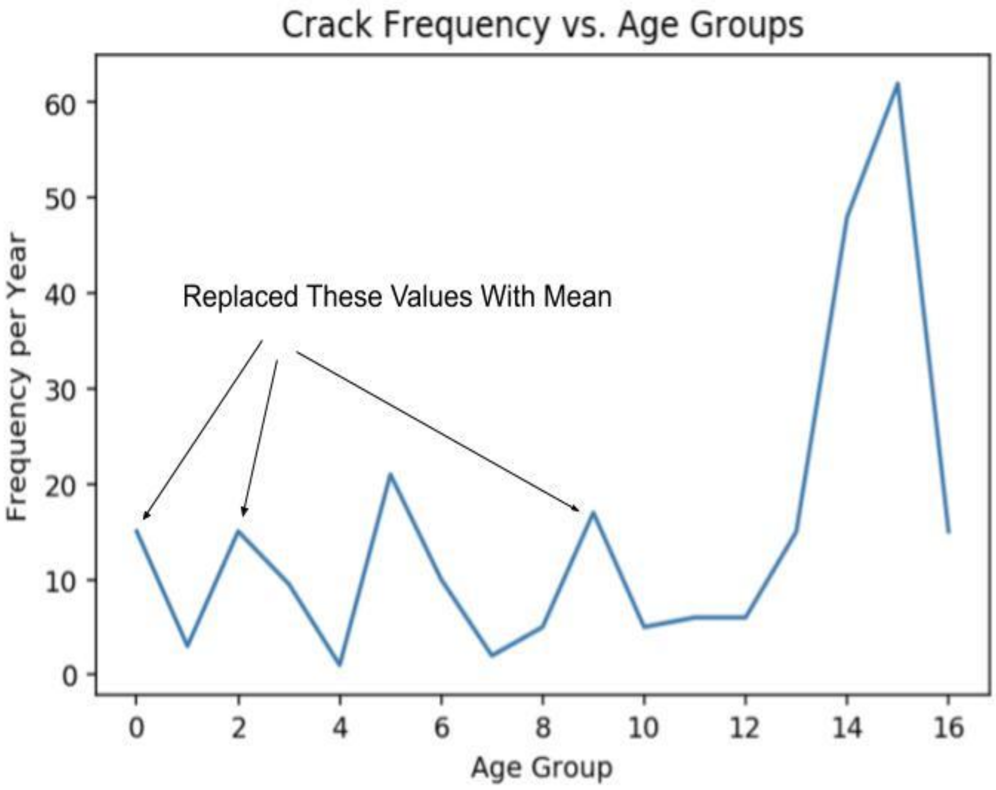
```
# adjusted age ranges to be instead specific ages so that I would be able to visualize drug relationships with age

Drug.age = [12,  13, 14, 15, 16, 17, 18, 19, 20, 21, 22.5,24.5, 27.5,
 32, 42 ,57, 71]

# viewed my age changes in all the impacted rows
Drug.head(17)
```

[613]:

	age	n	alcohol-use	alcohol-frequency	marijuana-use	marijuana-frequency	cocaine-use	cocaine-frequency	crack-use	crack-frequency	...	oxycontin-use	oxycontin-frequency	tranquilizer-use	tran fre
0	12.0	2798	3.9	3.0	1.1	4.0	0.1	5.0	0.0	15.0357	...	0.1	24.5	0.2	
1	13.0	2757	8.5	6.0	3.4	15.0	0.1	1.0	0.0	3.0	...	0.1	41.0	0.3	
2	14.0	2792	18.1	5.0	8.7	24.0	0.1	5.5	0.0	15.0357	...	0.4	4.5	0.9	
3	15.0	2956	29.2	6.0	14.5	25.0	0.5	4.0	0.1	9.5	...	0.8	3.0	2.0	
4	16.0	3058	40.1	10.0	22.5	30.0	1.0	7.0	0.0	1.0	...	1.1	4.0	2.4	
5	17.0	3038	49.3	13.0	28.0	36.0	2.0	5.0	0.1	21.0	...	1.4	6.0	3.5	
6	18.0	2469	58.7	24.0	33.7	52.0	3.2	5.0	0.4	10.0	...	1.7	7.0	4.9	
7	19.0	2223	64.6	36.0	33.4	60.0	4.1	5.5	0.5	2.0	...	1.5	7.5	4.2	
8	20.0	2271	69.7	48.0	34.0	60.0	4.9	8.0	0.6	5.0	...	1.7	12.0	5.4	
9	21.0	2354	83.2	52.0	33.0	52.0	4.8	5.0	0.5	17.0	...	1.3	13.5	3.9	
10	22.5	4707	84.2	52.0	28.4	52.0	4.5	5.0	0.5	5.0	...	1.7	17.5	4.4	

Considered the Impact of Replacement

The crack-frequency column had the most missing data (3). I plotted crack-frequency to see the impact of replacement. The points that were replaced show spikes in the plot, where you would expect to find a more gradual and increase.


Crack Frequency vs. Age Groups
Frequency per Year
Age Group
Replaced These Values With Mean