**Московский государственный технический**

**университет им. Н.Э. Баумана.**

Факультет «Информатика и управление»

Кафедра ИУ5.

Курс «Технологии машинного обучения»

Отчет по лабораторной работе №1

«Разведочный анализ данных. Исследование и визуализация данных»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-62 |  | преподаватель каф. ИУ5 |
| Король Константин |  | Гапанюк Ю.Е. |
| Подпись и дата: |  | Подпись и дата: |

Москва, 2019 г.

Задание

* Выбрать набор данных (датасет). Вы можете найти список свободно распространяемых датасетов [здесь.](https://github.com/ugapanyuk/ml_course/wiki/DSLIST)

Для лабораторных работ не рекомендуется выбирать датасеты большого размера.

* Создать ноутбук, который содержит следующие разделы:

1. Текстовое описание выбранного Вами набора данных.
2. Основные характеристики датасета.
3. Визуальное исследование датасета.
4. Информация о корреляции признаков.

* Сформировать отчет и разместить его в своем репозитории на github.

**Код программы**

Текстовое описание набора данных

В качестве набора данных мы будем использовать набор данных по определению качества красного вина: <https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009> Датасет состоит из файла winequality-red.csv

Файл содержит следующие колонки:

fixed acidity - содержание неиспаряющейсяя кислотности

volatile acidity - содержание уксусной кислотности

citric acid - содержание лимонной кислоты

residual sugar - содержание сахара

chlorides - содержание соли

free sulfur dioxide - содержание свободного диоксида серы (в виде растворенного газа)

total sulfur dioxide - содержание общего диоксида серы

density - плотность

pH - степень кислотности вина от 0 до 14

sulphates - содержание сульфатов

alcohol - содержание алкоголя

quality - итоговая оценка качества вина от 0 до 10

**import** **numpy** **as** **np**

**import** **pandas** **as** **pd**

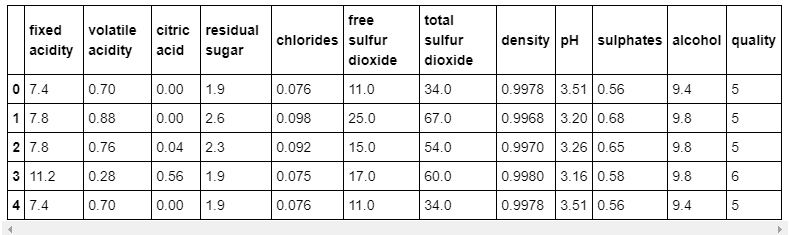
**import** **seaborn** **as** **sns**

**import** **matplotlib.pyplot** **as** **plt**

%matplotlib inline

data = pd.read\_csv('data/winequality-red.csv', sep=",")

data.head()



data.shape

OUT:(1599, 12)

total\_count = data.shape[0]

print('Всего строк: **{}**'.format(total\_count))

OUT: Всего строк: 1599

data.columns

OUT: Index(['fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar',

'chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'density',

'pH', 'sulphates', 'alcohol', 'quality'],

dtype='object')

data.dtypes

OUT: fixed acidity float64

volatile acidity float64

citric acid float64

residual sugar float64

chlorides float64

free sulfur dioxide float64

total sulfur dioxide float64

density float64

pH float64

sulphates float64

alcohol float64

quality int64

dtype: object

**for** col **in** data.columns:

temp\_null\_count = data[data[col].isnull()].shape[0]

print('**{}** - **{}**'.format(col, temp\_null\_count))

OUT: fixed acidity - 0

volatile acidity - 0

citric acid - 0

residual sugar - 0

chlorides - 0

free sulfur dioxide - 0

total sulfur dioxide - 0

density - 0

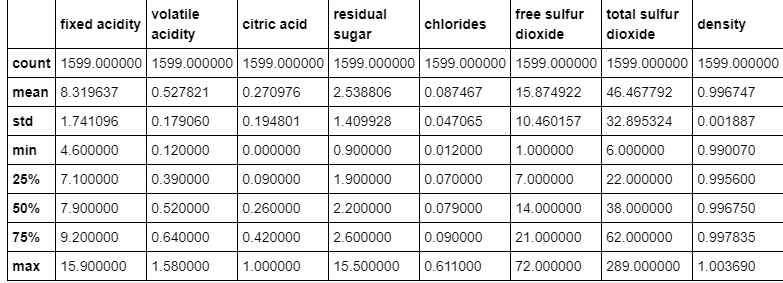
pH - 0

sulphates - 0

alcohol - 0

quality – 0

data.describe()

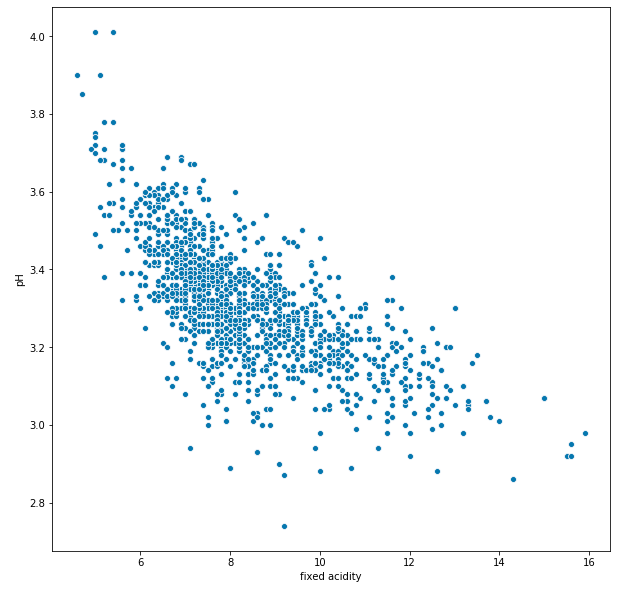


data['quality'].unique()

OUT: array([5, 6, 7, 4, 8, 3], dtype=int64)

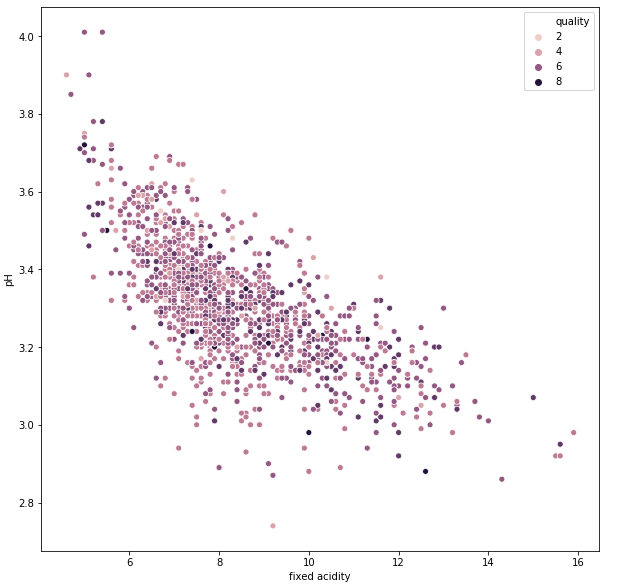
fig, ax = plt.subplots(figsize=(10,10))

sns.scatterplot(ax=ax, x='fixed acidity', y='pH', data=data)



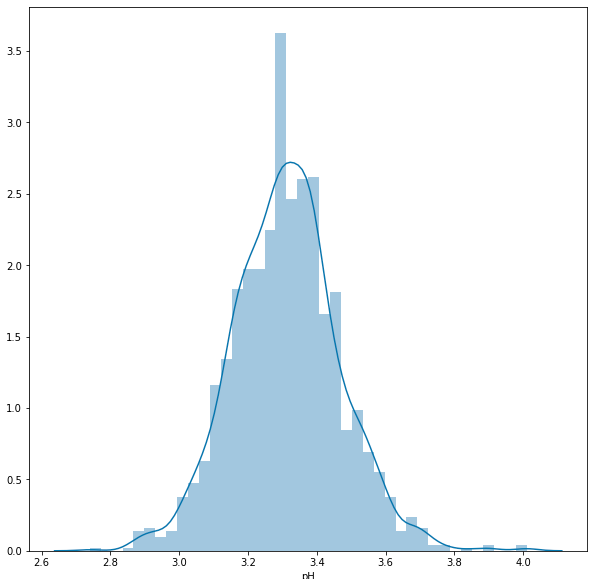
fig, ax = plt.subplots(figsize=(10,10))

sns.scatterplot(ax=ax, x='fixed acidity', y='pH', data=data, hue='quality')

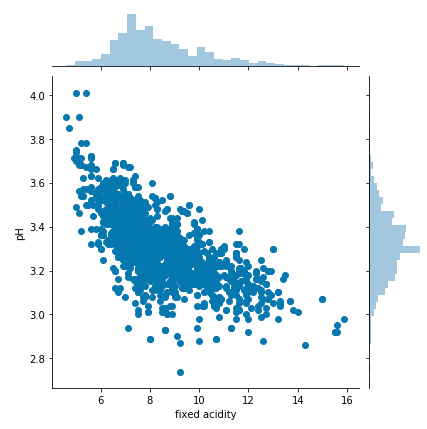


fig, ax = plt.subplots(figsize=(10,10))

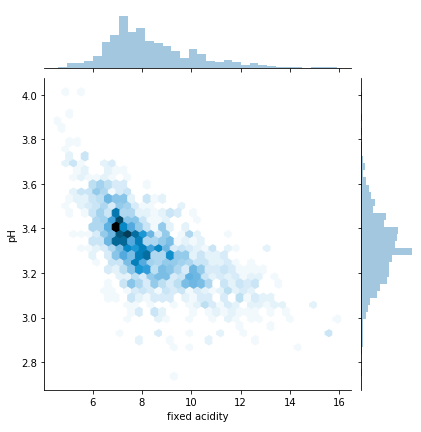
sns.distplot(data['pH'])



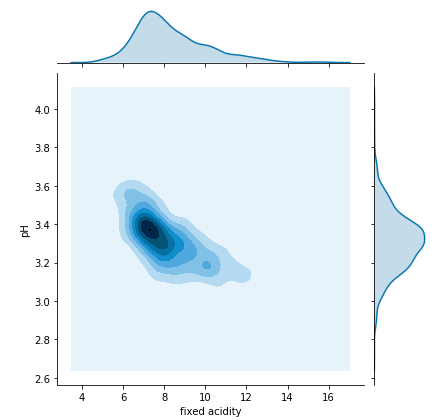
sns.jointplot(x='fixed acidity', y='pH', data=data)



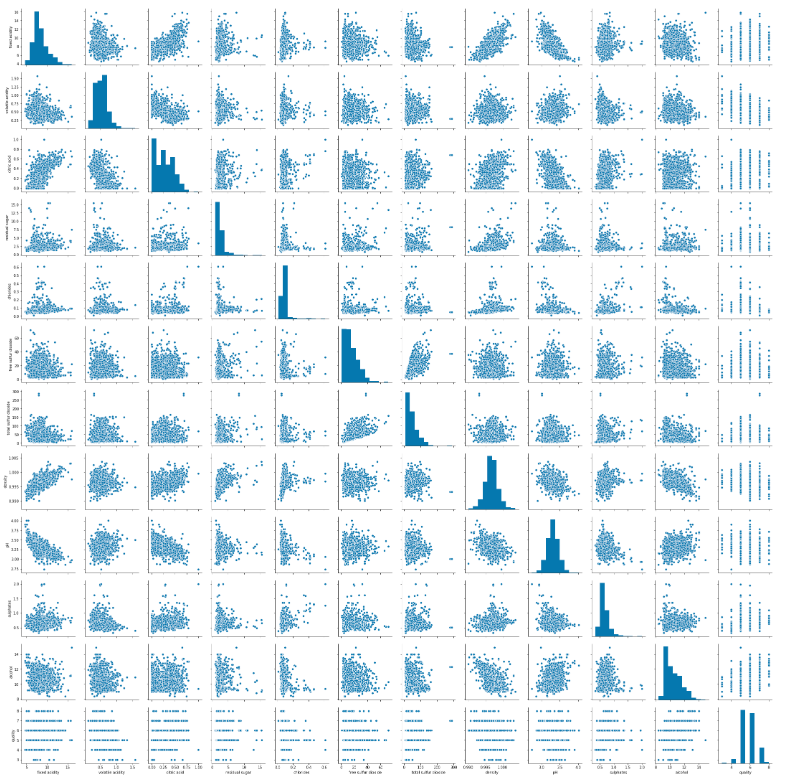
sns.jointplot(x='fixed acidity', y='pH', data=data, kind='hex')



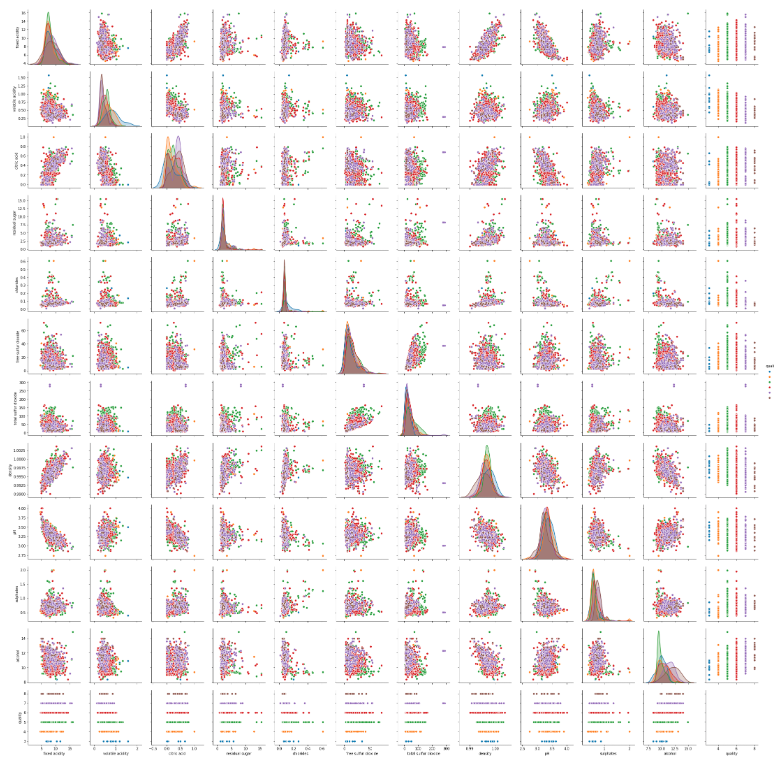
sns.jointplot(x='fixed acidity', y='pH', data=data, kind='kde')



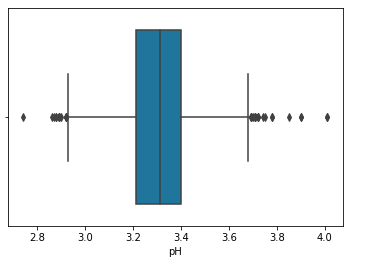
sns.pairplot(data)



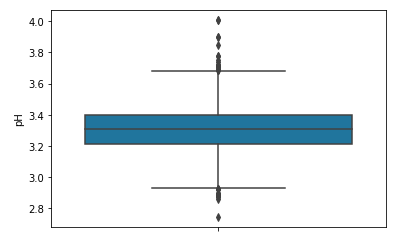
sns.pairplot(data, hue='quality')



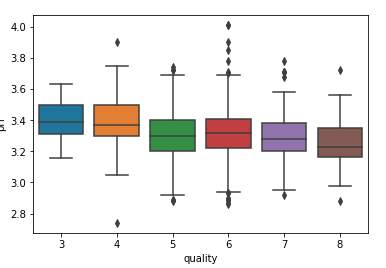
sns.boxplot(x=data['pH'])



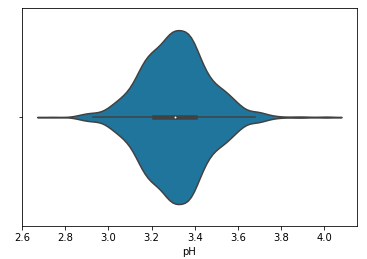
sns.boxplot(y=data['pH'])



sns.boxplot(x='quality', y='pH', data=data)



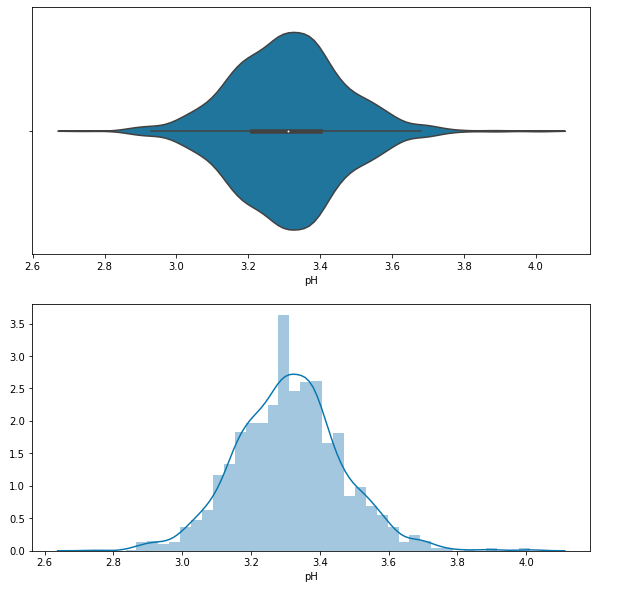
sns.violinplot(x=data['pH'])



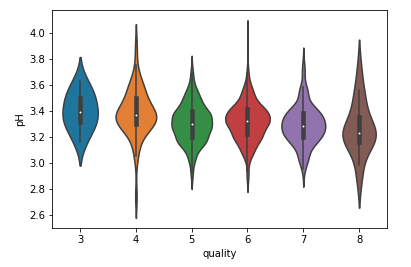
fig, ax = plt.subplots(2, 1, figsize=(10,10))

sns.violinplot(ax=ax[0], x=data['pH'])

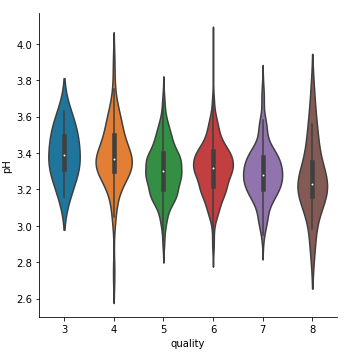
sns.distplot(data['pH'], ax=ax[1])



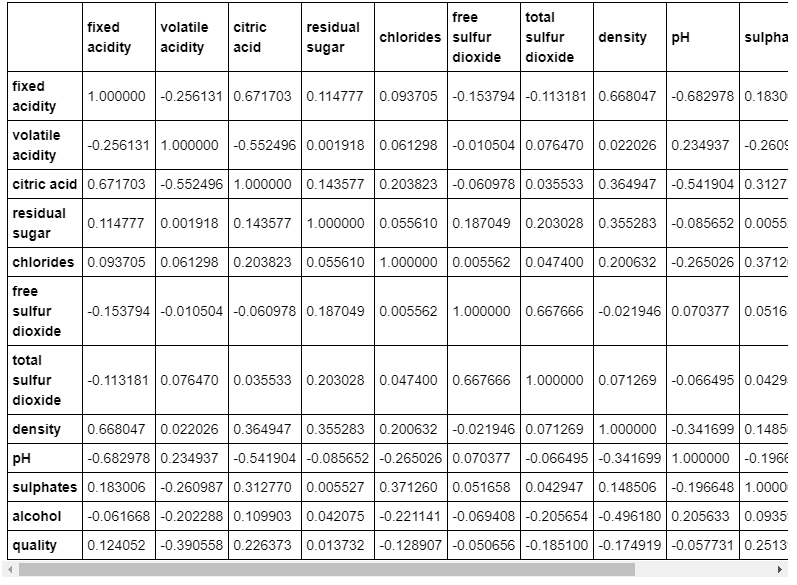
sns.violinplot(x='quality', y='pH', data=data)



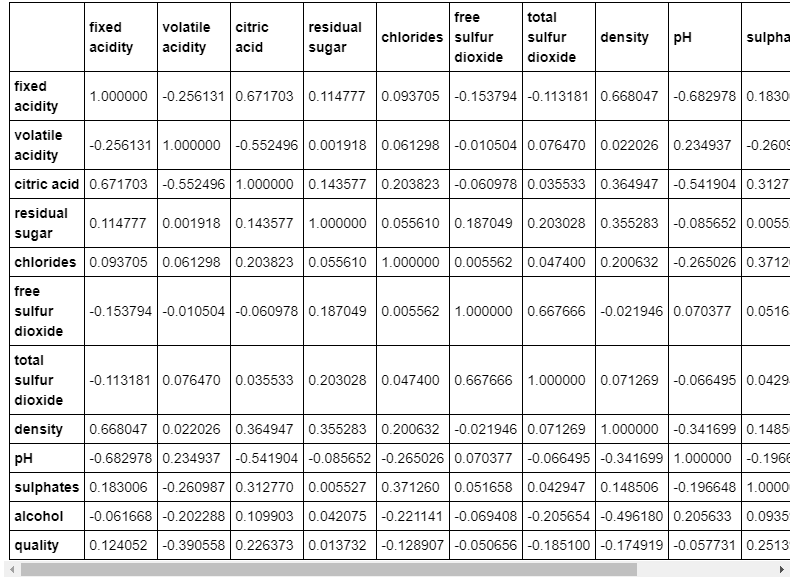
sns.catplot(x='quality', y='pH', data=data, kind="violin", split=**True**)



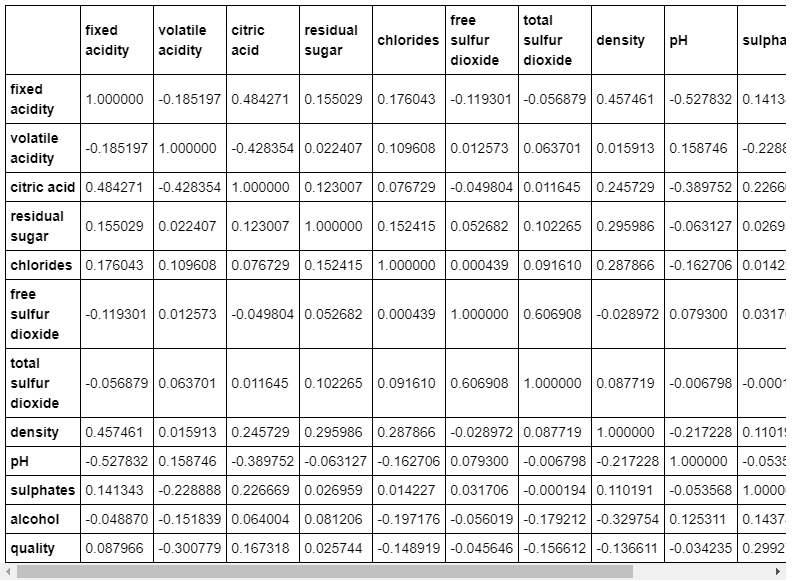
data.corr()



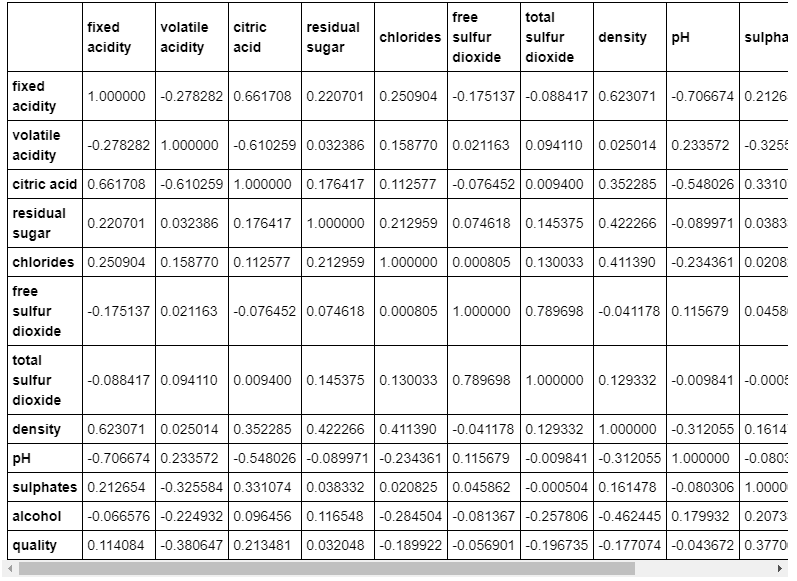
data.corr(method='pearson')



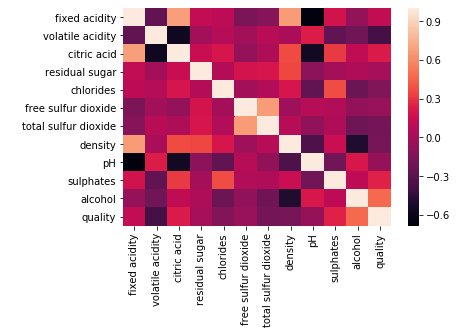
data.corr(method='kendall')



data.corr(method='spearman')

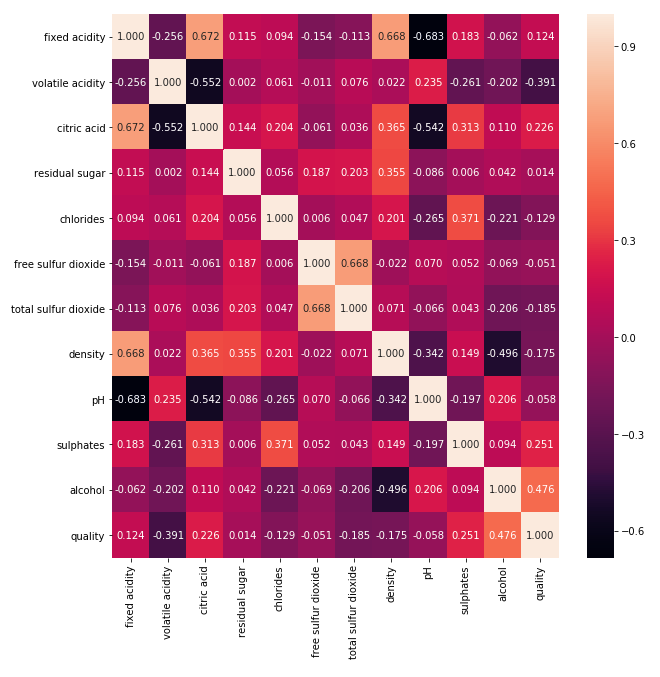


sns.heatmap(data.corr())



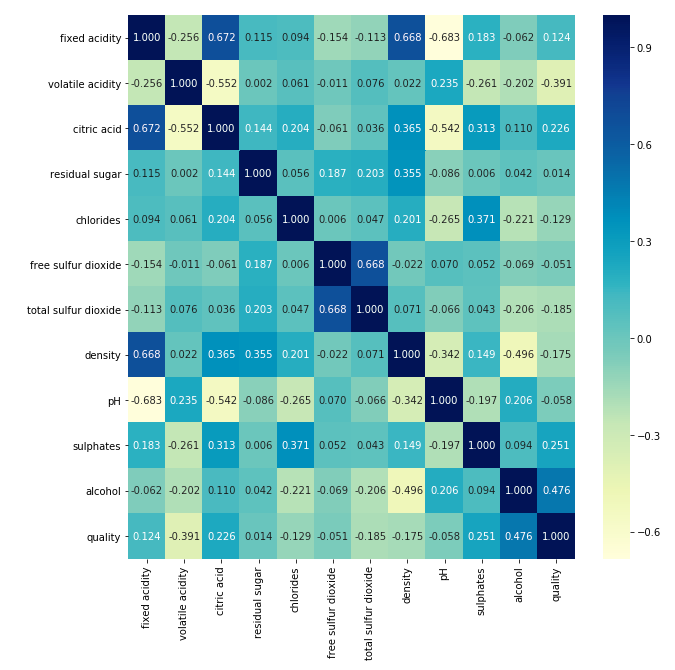
fig, ax = plt.subplots(figsize=(10,10))

sns.heatmap(data.corr(), annot=**True**, fmt='.3f')



fig, ax = plt.subplots(figsize=(10,10))

sns.heatmap(data.corr(), cmap='YlGnBu', annot=**True**, fmt='.3f')



fig, ax = plt.subplots(figsize=(10,10))

*# Треугольный вариант матрицы*

mask = np.zeros\_like(data.corr(), dtype=np.bool)

*# чтобы оставить нижнюю часть матрицы*

*# mask[np.triu\_indices\_from(mask)] = True*

*# чтобы оставить верхнюю часть матрицы*

mask[np.tril\_indices\_from(mask)] = **True**

sns.heatmap(data.corr(), mask=mask, annot=**True**, fmt='.3f')

