data : DataFrame---Tidy (long-form) dataframe where each column is a variable and each row is an observation.

LoadData

1. getTXTpv(filepath,filename):

txt文本每行一个pv名。

Filepath：文件绝对路径

filename：文件名，

返回：list

1. getChanArch(ipaddr, key, pvnames, start, end, how=0):

获取原始的ChannelArchiver 数据。

Ipaddr：server端ip

key：Engine对应的Engine key序号

pvnames：pv名list

start,end：起始时间，结束时间，格式：“%m/%d/%Y %H:%M:%S”

how：ChannelArchiver中API的参数0-raw,1-spreadsheet,2-avg,3-plot-binning,4-linear

返回：dict

1. getChanArchEngineKey(ipaddr,enginename):

根据engine名获取key

Ipaddr：server端ip

Enginename：engine名

返回：key数字

1. getKey(ipaddr,pvnames)：

打印pv所在的Engine 名称和key序号，不存在给出提示

Ipaddr：server端ip

pvnames：pv名list

无返回

1. getFormatChanArch(ipaddr, key, pvnames, start, end, merge\_type,interpolate\_type='linear',fillna\_type=None,how=0,dropna=True):

获取同engine的多个pv历史数据，可选择对齐时间戳的类型等参数,对于

Ipaddr：server端ip

key：Engine对应的Engine key序号

pvnames：pv名list

start,end：起始时间，结束时间，格式：“%m/%d/%Y %H:%M:%S”

merge\_type：对齐时间戳的类型，包括outer-插值补全null，保持最小粒度period；inner-删除额外的，保持最大粒度period；number-指定period对齐数据

interpolate\_type：插值的类型，包括{'linear','time','index','values', 'nearest','zero','slinear', 'quadratic','cubic','barycentric','krogh','polynomial', 'spline', 'piecewise\_polynomial', 'from\_derivatives', 'pchip', 'akima'}，默认linear

fillna\_type：对于空值得填充类型，包括{'backfill', 'bfill', 'pad', 'ffill', None}, 默认 None

how：ChannelArchiver中API的参数，包括0-raw,1-spreadsheet,2-avg,3-plot-binning,4-linear

dropna：是否删除空值

返回：DataFrame

1. 待进一步测试：getArchAppl(data\_retrieval\_url,pvnames,start,end,merge\_type):

获取ArchiverAppliance的历史数据

data\_retrieval\_url：server设置里面的url，类似'192.168.44.168:17665/retrieval'

pvnames：pv名list

start,end：起始时间，结束时间，格式：“%m/%d/%Y %H:%M:%S”

merge\_type：对齐时间戳的类型，包括outer-插值补全null，保持最小粒度period；inner-删除额外的，保持最大粒度period；number-指定period对齐数据

1. getLocalFile(filepath,filename,skiprows=0):

获取本地文件，列为参数

filepath：文件所在路径

filename：文件名

skiprows：跳过多少行

1. generate\_live\_data (duration, pvnames):

产生实时数据

duration：产生的时间区间长度，单位s

pvnames：pv名list

1. dataset2df (dataset):

sklearn中的dataset转pandas DataFrame

dataset：示例数据

返回：DataFrame

1. np2df(data,col=""):

将numpy中的list、dict、ndarray转换为DataFrame

data：需要转换的数据

col：选填，list和ndarray需要column名

返回：DataFrame

1. df2np(data,datatype):

将DataFrame转换为指定类型，包括list、dict、ndarray

data：dataframe格式数据

datatype：指定类型，list、dict、ndarray，若不指定，转为dict

返回：指定数据类型

1. datetime2utc(datestr ,dtformat='%m/%d/%Y %H:%M:%S'):

把指定格式的string类型时间戳转换为unix time

datestr：字符串时间

dtformat：时间类型

返回：float类型时间

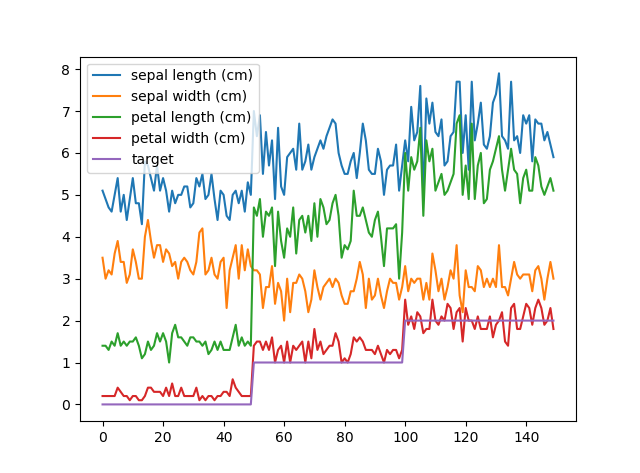
DisplayData

1. showPlot(data):

将所有列数据绘制到一张图像中

data：DataFrame格式

示例：

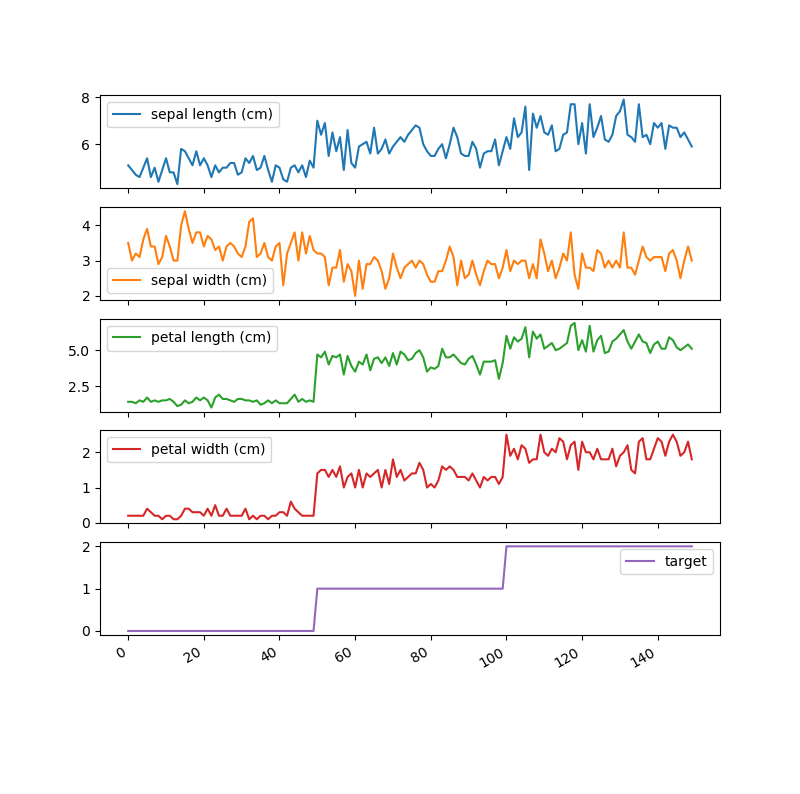


1. showSubPlot(data):

将每一列数据分别绘制到一张图像中

data：DataFrame格式

示例：



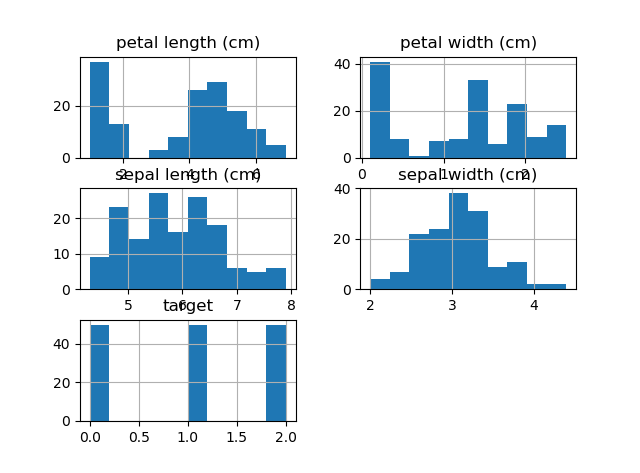
1. showHist(data，column=None):

绘制每一列数据的直方图

data：DataFrame格式

column：需要展示的列名，list，默认全部列

示例：



1. showStatistic(data):

打印所有数据的统计信息，含NaN；包括count，mean，std，min，25%，50%，75%，max

data：DataFrame格式

示例：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | sepal length (cm) | sepal width (cm) | petal length (cm) | petal width (cm) | target |
| count | 150.000000 | 150.000000 | 150.000000 | 150.000000 | 150.000000 |
| mean | 5.843333 | 3.054000 | 3.758667 | 1.198667 | 1.000000 |
| std | 0.828066 | 0.433594 | 1.764420 | 0.763161 | 0.819232 |
| min | 4.300000 | 2.000000 | 1.000000 | 0.100000 | 0.000000 |
| 25% | 5.100000 | 2.800000 | 1.600000 | 0.300000 | 1.000000 |
| 50% | 5.800000 | 3.000000 | 4.350000 | 1.300000 | 1.000000 |
| 75% | 6.400000 | 3.300000 | 5.100000 | 1.800000 | 2.000000 |
| max | 7.900000 | 4.400000 | 6.900000 | 2.500000 | 2.000000 |

1. showCorr (data,method='pearson'):

打印列之间的相关性，含NaN

data：DataFrame格式

method：pearson : standard correlation coefficient

kendall : Kendall Tau correlation coefficient

spearman : Spearman rank correlation

示例：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | sepal length (cm) | sepal width (cm) | petal length (cm) | petal width (cm) | target |
| sepal length (cm) | 1.000000 | -0.109369 | 0.871754 | 0.817954 | 0.782561 |
| sepal width (cm) | -0.109369 | 1.000000 | -0.420516 | -0.356544 | -0.419446 |
| petal length (cm) | 0.871754 | -0.420516 | 1.000000 | 0.962757 | 0.949043 |
| petal width (cm) | 0.817954 | -0.356544 | 0.962757 | 1.000000 | 0.956464 |
| target | 0.782561 | -0.419446 | 0.949043 | 0.956464 | 1.000000 |

1. showCorrMap (data,method='pearson'):

利用变量之间的相关系数矩阵生成热力图

data：DataFrame格式

method：pearson : standard correlation coefficient

kendall : Kendall Tau correlation coefficient

spearman : Spearman rank correlation

示例：



1. showSPLOM(data, hue=None, hue\_order=None, palette=None, vars=None, x\_vars=None, y\_vars=None, kind='scatter', diag\_kind='hist', markers=None, size=2.5, aspect=1, dropna=True, plot\_kws=None, diag\_kws=None, grid\_kws=None):

利用seaborn绘制多变量图, Plot pairwise relationships in a dataset

By default, this function will create a grid of Axes such that each variable in ``data`` will by shared in the y-axis across a single row and in the x-axis across a single column. The diagonal Axes are treated differently, drawing a plot to show the univariate distribution of the data for the variable in that column.

data：DataFrame格式

hue : string (variable name), optional

Variable in ``data`` to map plot aspects to different colors.

hue\_order : list of strings

Order for the levels of the hue variable in the palette

palette : dict or seaborn color palette

Set of colors for mapping the ``hue`` variable. If a dict, keys

should be values in the ``hue`` variable.

vars : list of variable names, optional

Variables within ``data`` to use, otherwise use every column with

a numeric datatype.

{x, y}\_vars : lists of variable names, optional

Variables within ``data`` to use separately for the rows and

columns of the figure; i.e. to make a non-square plot.

kind : {'scatter', 'reg'}, optional

Kind of plot for the non-identity relationships.

diag\_kind : {'hist', 'kde'}, optional

Kind of plot for the diagonal subplots.

markers : single matplotlib marker code or list, optional

Either the marker to use for all datapoints or a list of markers with

a length the same as the number of levels in the hue variable so that

differently colored points will also have different scatterplot

markers.

size : scalar, optional

Height (in inches) of each facet.

aspect : scalar, optional

Aspect \* size gives the width (in inches) of each facet.

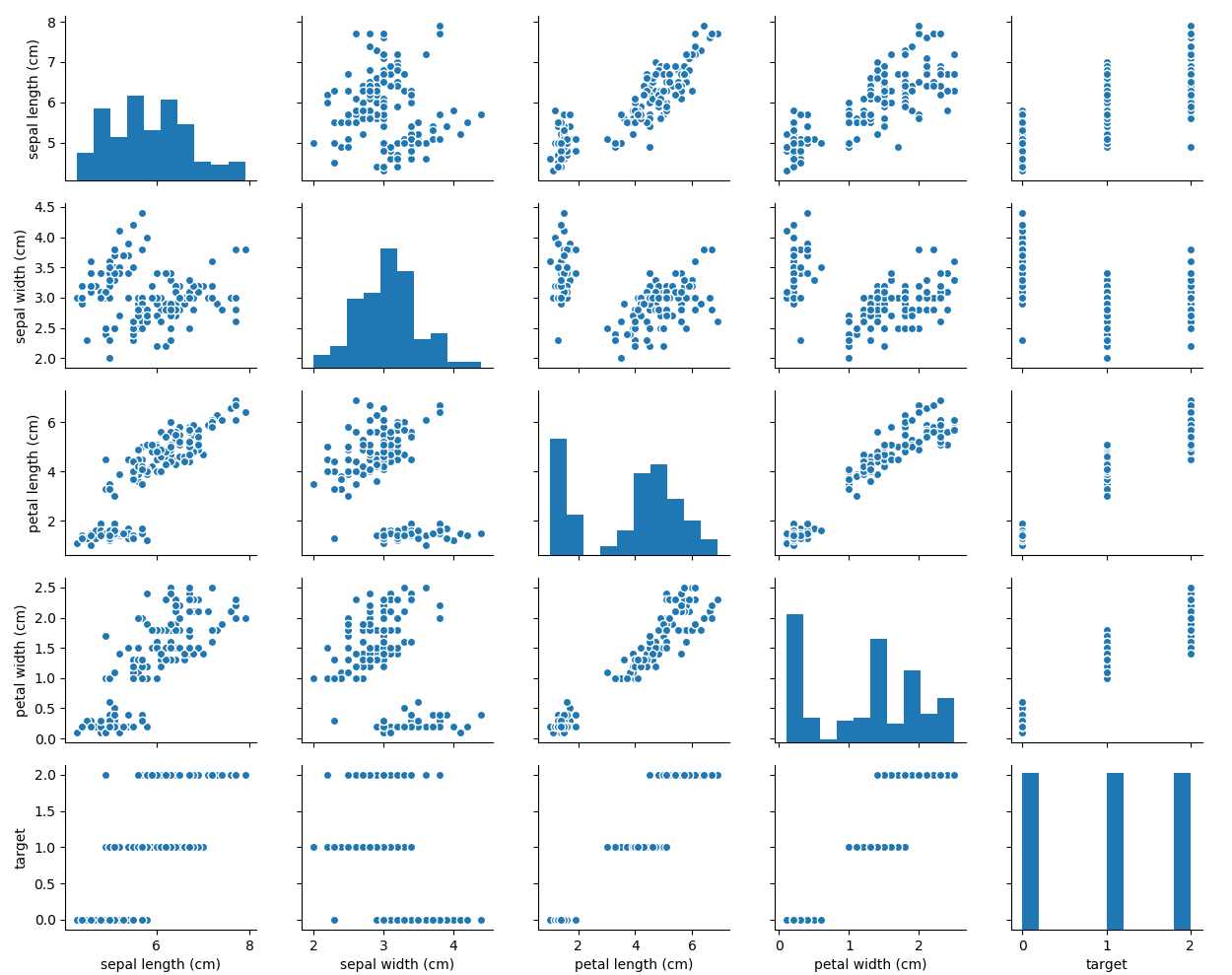
dropna : boolean, optional

Drop missing values from the data before plotting.

{plot, diag, grid}\_kws : dicts, optional

Dictionaries of keyword arguments.

示例：

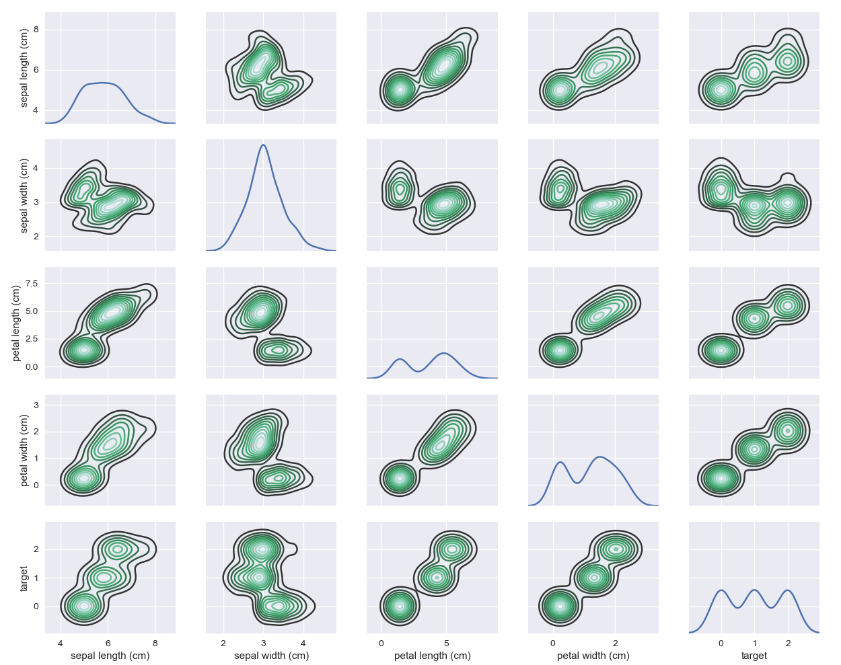


1. showSPLOM\_G(data, hue=None, hue\_order=None, palette=None, hue\_kws=None, vars=None, x\_vars=None, y\_vars=None, diag\_sharey=True, height=2.5, aspect=1, despine=True, dropna=True, size=None):

Subplot grid for plotting pairwise relationships in a dataset

data：DataFrame格式

参数类似showSPLOM



1. showStd(data,ddof=1):

打印变量的标准差

data：DataFrame格式

ddof：默认为1，为样本标准偏差，分母为n-1。ddof=0为总体标准偏差，分母为n

示例：

sepal length (cm) 0.828066

sepal width (cm) 0.433594

petal length (cm) 1.764420

petal width (cm) 0.763161

target 0.819232

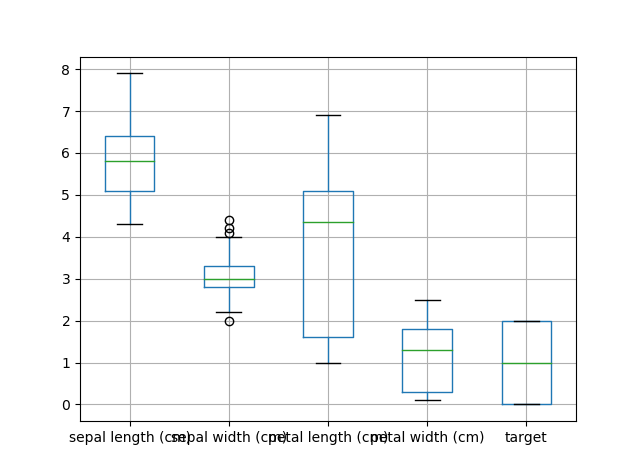
dtype: float64

1. showBins(data)

绘制箱体图，由五个分位数组成

data：DataFrame格式

示例：



1. plot\_predict(y\_test,y\_pred,title='Figure'):

展示预测值和实际值的折线-散点图

y\_test：测试值list

y\_pred：预测值list

title='Figure'：图题，默认Figure

1. plot\_predict(y\_test,y\_pred,title='Figure'):

拆分y\_test和y\_pred，每1000个点做一幅图，展示预测值和实际值的折线-散点图

y\_test：测试值list

y\_pred：预测值list

title='Figure'：图题，默认Figure

CleanData

1. NormData(data,norm\_type,ddof=0):

变量归一化

data：DataFrame格式

norm\_type：

min-max：离差标准化，是对原始数据的线性变换，使结果值映射到[0 - 1]之间

Z-score：给予原始数据的均值（mean）和标准差（standard deviation）进行数据的标准化。经过处理的数据符合标准正态分布，即均值为0，标准差为1

返回：归一化后的DataFrame

1. EnumData(data,pvname,cut\_ranges, right=True,enumnames=None):

将DataFrame中的某列数据转换为Enum类型

data：DataFrame格式

pvname：要转换的列名

cut\_ranges：划分区间范围，例如分为两类，cut\_range=[1,2,3]，即1-2范围为一类，2-3范围为一类，不能包含未在范围里面的数据，报错

right：默认Ture,代表是否包含右边界。 例如right == True，范围 [1,2,3,4] 代表 (1,2], (2,3], (3,4]

enumnames：array or boolean, default None. Used as labels for the resulting bins. Must be of the same length as the resulting bins. If False, return only integer indicators of the bins.

1. fillempty (data,method='pad'):

data：DataFrame格式

fillna\_type:{'backfill', 'bfill', 'pad', 'ffill', None}, default pad 向后补全

返回：补全后的DataFrame

1. DataFrame数据过滤

use of boolean vectors to filter the data. The operators are: | for or, & for and, and ~ for not. These must be grouped by using parentheses.

例如：

data[(data['pvname']>1000)&data['pvname']<20]

Train Data

1. split\_data(data,t\_pvname,test\_size = 0.3):

将数据集分成训练集和测试集

data：DataFrame格式

t\_pvname:作为目标变量的列名

test\_size：测试集所占比例

返回：X\_train, X\_test, y\_train, y\_test

1. MLLinearRegression(X\_train, X\_test, y\_train, y\_test):
2. MLGaussianNB\_testmodel(X\_train, X\_test, y\_train, y\_test):
3. MLGaussianNB(data,target\_pv,predict\_data):

用于预测指定数据

1. MLDesionTrees\_testmodel(X\_train, X\_test, y\_train, y\_test):
2. MLDecisionTrees(data,target\_pv,predict\_data):

用于预测指定数据

1. MLPolynomialRegression(X\_train, X\_test, y\_train, y\_test,degree=2):
2. MLKNN(X\_train, X\_test, y\_train, y\_test):
3. MLLogisticRegression\_testmodel(X\_train, X\_test, y\_train, y\_test,multinominal=False,c=1.0):
4. MLLogisticRegression(X,y,predict\_data,multinominal=False,c=1.0):

用于预测指定数据

1. test\_LogisticRegression\_C(X\_train, X\_test, y\_train, y\_test,a=-2,b=4,n=100):
2. MLDBSCAN(data,target\_pv,eps=0.5,min\_samples=10):
3. MLKMeans(data,feature\_pv1,feature\_pv2,cluster=2):
4. MLMLPClassifier(X\_train, X\_test, y\_train, y\_test):
5. 以下函数用来计算真实值与预测值之间的预测误差，r2, absolute\_error, quare\_error, explained\_viarance\_score：
6. performance\_metric\_r2(y\_true, y\_predict):
7. performance\_metric\_abstErr(y\_true, y\_predict):
8. performance\_metric\_sqrErr(y\_true, y\_predict):
9. performance\_metric\_expVia(y\_true, y\_predict):
10. fit\_model\_k\_fold(X, y):
11. fit\_model\_shuffle(X, y):