1.绘制经过两个点直线方程。两个点坐标是（1,3）和（4,5），绘制穿过这两个点的直线，得到直线方程。采用线性回归模型sklearn.linear\_model实现

import numpy as np

import matplotlib. pyplot as plt

from sklearn. linear\_model import LinearRegression

X=[ [1],[4]]

y=[3,5]

lr=LinearRegression().fit(X,y)

z=np. linspace(0,5,20)

plt. scatter(X,y,s=80)

plt. plot(z,lr.predict(z.reshape(-1,1)),c='k')

plt. title ('Straight Line')

plt. show()

print("直线方程是:")

print('y={:.3f}'.format(lr.coef\_[0]),'x','+{:.3f}'.format(lr.intercept\_))

2.已知训练样本X = np.array([[-1, -1], [-2, -1], [1, 1], [2, 1]])；目标值数组 y = np.array([1, 1, 2,2])。采用线性核函数的支持向量机预测 [[-0.5,-0.8]])) 所属类别。

import numpy as np

X=np.array([[-1,-1],[-2,-1],[1,1],[2,1]])

y=np.array([1,1,2,2])

from sklearn. svm import SVC

SVCClf = SVC(kernel ='linear')

SVCClf. fit (X,y)

print(SVCClf.predict([[-0.5,-0.8]]))

3.使用 linspace 生成[100,200]区间内 80 个数据，在[5,20]之间随机增幅，使用polyfit进行抛 物线拟合

import numpy as np

from matplotlib import pyplot as plt

x=np. linspace(100,200,80)

y = x +np. random. random\_integers(5,20,80)

p1=np.polyfit(x,y,deg=2)

q1=np.polyval(p1,x)

plt. plot(x,y,'o')

plt. plot (x,q1,'k')

plt. show()

4.在同一子图中绘制学生各学期的语文、数学和英语成绩分析图。

使用rcParams参数设置画布大小[(10, 6)]、中文字体（黑体）、坐标轴刻度线显示方向（向内）；然后使用plot()函数绘制图形，并设置语文图形样式为上三角双画线、数学图形样式为实心圆点画线、英语图形样式为红色小菱形虚线；最后使用列表同时设置3个图形的图例。

import matplotlib.pyplot as plt

import pandas as pd

df = pd.read\_excel('学生各学期成绩表.xlsx')

plt.rcParams['figure.figsize'] = (10, 6)

plt.rcParams['font.sans-serif'] = 'SimHei'

plt.xticks(rotation=15)

plt.grid()

x = df['学期']

plt.plot(x, df['语文'], '--^')

plt.plot(x, df['数学'], '-.o')

plt.plot(x, df['英语'], 'r:d')

plt.legend(['语文', '数学', '英语'])

plt.show()

5.对data=[[-1,2],[-0.5,6],[0,10],[1,18]]进行归一化处理

import numpy as np

X=np.array([[-1,2],[-0.5,6],[0,10],[1,18]])

X\_nor=(X-X.min(axis=0))/(X.max(axis=0)-X.min(axis=0))

print (X\_nor)

6.采用Sklearn的StandardScaler对x=[[1.,-1.,2.],[2.,0.,0.],[0.,1.,-1.]]进行标准化处理,求数据的均值、方差以及标准化数据。

from sklearn. preprocessing import StandardScaler

import numpy as np

X=np.array([[1.,-1.,2.],[2.,0.,0.],[0.,1.,-1.]])

scaler=StandardScaler().fit(X)

print('\n均值:',scaler.mean\_)

print('方差:',scaler. var\_)

X\_scale=scaler. transform(X)

print('\n标准化数据:\n',X\_scale)

1.绘制经过两个点直线方程。两个点坐标是（1,3）和（4,5），绘制穿过这两个点的直线，得到直线方程。采用线性回归模型sklearn.linear\_model实现

import numpy as np

import matplotlib. pyplot as plt

from sklearn. linear\_model import LinearRegression

X=[ [1],[4]]

y=[3,5]

lr=LinearRegression().fit(X,y)

z=np. linspace(0,5,20)

plt. scatter(X,y,s=80)

plt. plot(z,lr.predict(z.reshape(-1,1)),c='k')

plt. title ('Straight Line')

plt. show()

print("直线方程是:")

print('y={:.3f}'.format(lr.coef\_[0]),'x','+{:.3f}'.format(lr.intercept\_))

2.已知训练样本X = np.array([[-1, -1], [-2, -1], [1, 1], [2, 1]])；目标值数组 y = np.array([1, 1, 2,2])。采用线性核函数的支持向量机预测 [[-0.5,-0.8]])) 所属类别。

import numpy as np

X=np.array([[-1,-1],[-2,-1],[1,1],[2,1]])

y=np.array([1,1,2,2])

from sklearn. svm import SVC

SVCClf = SVC(kernel ='linear')

SVCClf. fit (X,y)

print(SVCClf.predict([[-0.5,-0.8]]))

3.使用 linspace 生成[100,200]区间内 80 个数据，在[5,20]之间随机增幅，使用polyfit进行抛 物线拟合

import numpy as np

from matplotlib import pyplot as plt

x=np. linspace(100,200,80)

y = x +np. random. random\_integers(5,20,80)

p1=np.polyfit(x,y,deg=2)

q1=np.polyval(p1,x)

plt. plot(x,y,'o')

plt. plot (x,q1,'k')

plt. show()

4.在同一子图中绘制学生各学期的语文、数学和英语成绩分析图。

使用rcParams参数设置画布大小[(10, 6)]、中文字体（黑体）、坐标轴刻度线显示方向（向内）；然后使用plot()函数绘制图形，并设置语文图形样式为上三角双画线、数学图形样式为实心圆点画线、英语图形样式为红色小菱形虚线；最后使用列表同时设置3个图形的图例。

import matplotlib.pyplot as plt

import pandas as pd

df = pd.read\_excel('学生各学期成绩表.xlsx')

plt.rcParams['figure.figsize'] = (10, 6)

plt.rcParams['font.sans-serif'] = 'SimHei'

plt.xticks(rotation=15)

plt.grid()

x = df['学期']

plt.plot(x, df['语文'], '--^')

plt.plot(x, df['数学'], '-.o')

plt.plot(x, df['英语'], 'r:d')

plt.legend(['语文', '数学', '英语'])

plt.show()

5.对data=[[-1,2],[-0.5,6],[0,10],[1,18]]进行归一化处理

import numpy as np

X=np.array([[-1,2],[-0.5,6],[0,10],[1,18]])

X\_nor=(X-X.min(axis=0))/(X.max(axis=0)-X.min(axis=0))

print (X\_nor)

6.采用Sklearn的StandardScaler对x=[[1.,-1.,2.],[2.,0.,0.],[0.,1.,-1.]]进行标准化处理,求数据的均值、方差以及标准化数据。

from sklearn. preprocessing import StandardScaler

import numpy as np

X=np.array([[1.,-1.,2.],[2.,0.,0.],[0.,1.,-1.]])

scaler=StandardScaler().fit(X)

print('\n均值:',scaler.mean\_)

print('方差:',scaler. var\_)

X\_scale=scaler. transform(X)

print('\n标准化数据:\n',X\_scale)

1.绘制经过两个点直线方程。两个点坐标是（1,3）和（4,5），绘制穿过这两个点的直线，得到直线方程。采用线性回归模型sklearn.linear\_model实现

import numpy as np

import matplotlib. pyplot as plt

from sklearn. linear\_model import LinearRegression

X=[ [1],[4]]

y=[3,5]

lr=LinearRegression().fit(X,y)

z=np. linspace(0,5,20)

plt. scatter(X,y,s=80)

plt. plot(z,lr.predict(z.reshape(-1,1)),c='k')

plt. title ('Straight Line')

plt. show()

print("直线方程是:")

print('y={:.3f}'.format(lr.coef\_[0]),'x','+{:.3f}'.format(lr.intercept\_))

2.已知训练样本X = np.array([[-1, -1], [-2, -1], [1, 1], [2, 1]])；目标值数组 y = np.array([1, 1, 2,2])。采用线性核函数的支持向量机预测 [[-0.5,-0.8]])) 所属类别。

import numpy as np

X=np.array([[-1,-1],[-2,-1],[1,1],[2,1]])

y=np.array([1,1,2,2])

from sklearn. svm import SVC

SVCClf = SVC(kernel ='linear')

SVCClf. fit (X,y)

print(SVCClf.predict([[-0.5,-0.8]]))

3.使用 linspace 生成[100,200]区间内 80 个数据，在[5,20]之间随机增幅，使用polyfit进行抛 物线拟合

import numpy as np

from matplotlib import pyplot as plt

x=np. linspace(100,200,80)

y = x +np. random. random\_integers(5,20,80)

p1=np.polyfit(x,y,deg=2)

q1=np.polyval(p1,x)

plt. plot(x,y,'o')

plt. plot (x,q1,'k')

plt. show()

4.在同一子图中绘制学生各学期的语文、数学和英语成绩分析图。

使用rcParams参数设置画布大小[(10, 6)]、中文字体（黑体）、坐标轴刻度线显示方向（向内）；然后使用plot()函数绘制图形，并设置语文图形样式为上三角双画线、数学图形样式为实心圆点画线、英语图形样式为红色小菱形虚线；最后使用列表同时设置3个图形的图例。

import matplotlib.pyplot as plt

import pandas as pd

df = pd.read\_excel('学生各学期成绩表.xlsx')

plt.rcParams['figure.figsize'] = (10, 6)

plt.rcParams['font.sans-serif'] = 'SimHei'

plt.xticks(rotation=15)

plt.grid()

x = df['学期']

plt.plot(x, df['语文'], '--^')

plt.plot(x, df['数学'], '-.o')

plt.plot(x, df['英语'], 'r:d')

plt.legend(['语文', '数学', '英语'])

plt.show()

5.对data=[[-1,2],[-0.5,6],[0,10],[1,18]]进行归一化处理

import numpy as np

X=np.array([[-1,2],[-0.5,6],[0,10],[1,18]])

X\_nor=(X-X.min(axis=0))/(X.max(axis=0)-X.min(axis=0))

print (X\_nor)

6.采用Sklearn的StandardScaler对x=[[1.,-1.,2.],[2.,0.,0.],[0.,1.,-1.]]进行标准化处理,求数据的均值、方差以及标准化数据。

from sklearn. preprocessing import StandardScaler

import numpy as np

X=np.array([[1.,-1.,2.],[2.,0.,0.],[0.,1.,-1.]])

scaler=StandardScaler().fit(X)

print('\n均值:',scaler.mean\_)

print('方差:',scaler. var\_)

X\_scale=scaler. transform(X)

print('\n标准化数据:\n',X\_scale)