

```

In [1]: # 导入操作系统库
import os
# 更改工作目录
os.chdir(r"D:\softwares\applied statistics\pythoncodelearning\chap4\sourcecode")
# 导入计算库
import numpy as np
# 导入绘图库
import matplotlib.pyplot as plt
# 导入最近邻模型
from sklearn.neighbors import KNeighborsRegressor
# 导入绘图库中的字体管理包
from matplotlib import font_manager
# 实现中文字符正常显示
font = font_manager.FontProperties(fname=r"C:\Windows\Fonts\SimKai.ttf")
# 使用seaborn风格绘图
plt.style.use("seaborn-v0_8")
# 生成数据
np.random.seed(0)
X = np.sort(5 * np.random.rand(40, 1), axis=0)
# 用于预测的X
T = np.linspace(0, 5, 500)[: , np.newaxis]
y = np.sin(X).ravel()
# 添加噪声
y[:5] += 1 * (0.5 - np.random.rand(8))
n_neighbors = 5
fig, axs = plt.subplots(2, 1, figsize=(6,6), tight_layout=True)
# 拟合回归模型
for i, weights in enumerate(["uniform", "distance"]):
    # 构建模型
    knn = KNeighborsRegressor(n_neighbors, weights=weights)
    # 模型拟合
    knn.fit(X, y)
    # 预测
    y_ = knn.predict(T)
    # 图形
    ax = axs.flatten()[i]
    ax.scatter(X, y, color="darkorange", label="data")
    ax.plot(T, y_, color="navy", label="prediction")
    ax.legend()
    ax.set_title("KNeighborsRegressor (k = %i, weights = '%s')" % (n_neighbors,
plt.show()
fig.savefig("../codeimage/code3.pdf")

```

