



中山大學

室内定位
《生产实习报告》

小组成员：郑炳坤、齐瑞敏、文豪、詹俊辉、周文韬

日期：2020年10月6号

一、 实习背景

随着近些年来信息技术的发展及移动硬件设备的功能越来越完善，定位方面的应用在人们生活扮演的角色更加重要，但在享受着这些应用带来方便的同时，人们对这些应用的定位性能要求也不断提高，各种新的需求也相应出现。目前人们使用较多的定位导航软件有：百度地图、高德地图、Google 地图等。这些地图软件绝大部分采用的定位技术是 GPS，在户外定位中能取得较好效果。但在室内环境中，这些地图软件会出现较大的定位误差，原因是室内环境中，由于建筑物的遮挡，室内环境的复杂多变，对卫星信号产生了极大的衰减和干扰，导致 GPS 在室内定位中定位精确度低，达不到室内定位的定位性能需求。随着手机等移动设备的普及，人们对于室内能实现高精度定位的呼声越来越高，因此众多研究者们开始从各个方面寻求合适的方法来实现室内环境中的精确定位。

红外线定位、RFID 定位等主流室内定位技术在这些年在国内外得到了一定程度的研究，取得了不俗的成果。这些定位技术进行应用时往往需要部署专口的网络，建设成本较高，使得它们各自的扩展性与实用性受到了不同程度的限制。相比之下，Wifi 具有覆盖面积广、廉价等优点，基于 Wifi 的室内定位技术具有较大的优势，基于 Wifi 的室内定位系统逐渐成为了室内定位系统的主流。

二、数据库的构建

下面介绍怎么得到在位置指纹法中进行仿真所需的数据库

- **RSS 仿真环境数据集：**设置好房间尺寸和各个 AP 的位置等各种参数，使用射线跟踪计算每个位置的 RSS，位置点的间隔设得小一点（这里设为 0.01m），计算一次射线跟踪后把数据保存起来，以后所有的 RSS 数据都从这个“RSS 仿真环境数据集”中获取。
- **一个典型的离线指纹库：**模拟数据采集的过程，比如每个 1m 采集一次 RSS 数据，数据都从上面的“RSS 仿真环境中”获取。也可以考虑其他的一些采集方式，或者为采集过程加入噪声，设置采集样本的个数等。
- **在线定位测试数据：**模拟一个目标在房间中运动，获得一条运动轨迹，以及每个轨迹点上的 RSS，用来作为定位算法的测试数据。

这个系列的位置指纹法都使用生成的这些数据，用测试数据验证算法的性能。

具体代码的功能作用如下：

- **main.m：**主程序，在仿真环境中，得到离线指纹库，以及在线阶段的测试数据，用于以后的定位测试。
- **get_rss_by_ray_tracing.m：**简化场景下（空旷房间）的射线跟踪。
- **generate_radio_map.m：**生成“RSS 仿真环境数据集”。
- **get_random_trace.m：**生成一条随机轨迹。
- **get_offline_data_random.m：**模拟随机数据采集，生成位置指纹库。
- **get_offline_data_uniform.m：**模拟均匀数据采集，生成位置指纹库。
- **get_online_data.m：**模拟在线阶段，生成测试数据。
- **radio_map_20_15.mat：**生成的“RSS 仿真环境数据集”，199914996 的数组，比如 `fingerprint(1000, 1000, 2)` 代表的是仿真环境中位置（100, 100）上接收到的第 2 个 AP 的 RSS。
- **offline_data_rss.mat：**离线数据 RSS，每行为一个 RSS 向量。
- **offline_data_location.mat：**离线数据位置点，每行为一个位置点 x, y。
- **online_data_trace.mat：**生成测试数据的运动轨迹，10000*2 的数组，比如 `trace(10, :)` 代表的是第 10 个时刻目标的位置 x 和 y。
- **online_data_rss.mat：**生成测试数据中与运行轨迹对应的 RSS，10000*6 的数组，比如 `trace(10, :)` 代表的是第 10 个时刻时目标测得的各个 RSS。

三、 Python 实现 WiFi 定位

4.1 KNN 算法

KNN 算法是通过测量不同特征值之间的距离进行分类。它的思路是：如果一个样本在特征空间中的 k 个最相似(即特征空间中最邻近)的样本中的大多数属于某一个类别,则该样本也属于这个类别,其中 K 通常是不大于 20 的整数。KNN 算法中,所选择的邻居都是已经正确分类的对象。该方法在定类决策上只依据最邻近的一个或者几个样本的类别来决定待分样本所属的类别。

运行 `run_KNN.py`, 得到 KNN 算法的精度和计算时间、超参数 k 与 `score` 的关系曲线(图 1)以及训练量与 `accuracy` 的关系曲线(图 2)。

accuracy: 2.253 m Running time: 0.838 s

图 1 超参数 k 与 `score` 的关系曲线

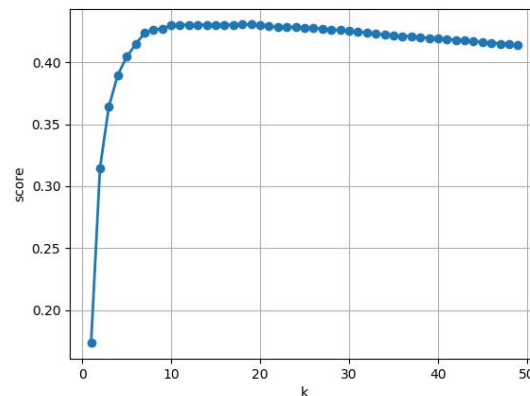
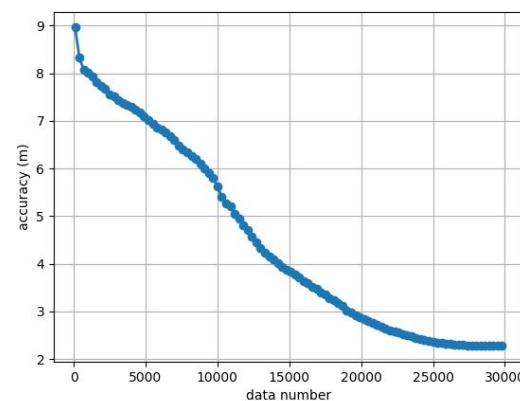


图 2 训练量与 `accuracy` 的关系曲线



4.2 梯度提升法

当损失函数是平方误差损失函数和指数损失函数时, 每一步优化是很简单的。但对一般损失函数而言, 往往每一步优化并不那么容易。针对这一问题, Freidman 提出了梯度提升 (gradient boosting) 算法。Gradient Boosting 是 Boosting 中的一大类算法, 它的思想借鉴于梯度下降法, 其基本原理是根据当前模型损失函数的负梯度信息来训练新加入的弱分类器, 然后将训练好的弱分类器以累加的

形式结合到现有模型中。

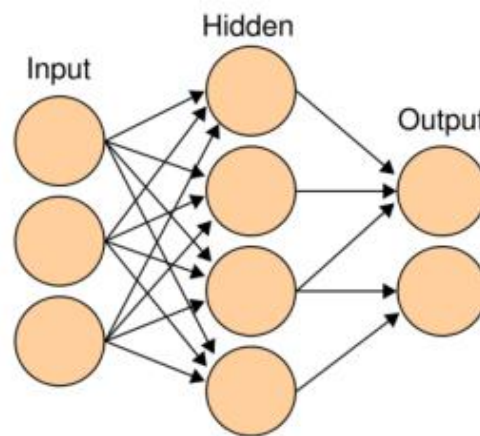
运行 `run_GB.py`，得到梯度提升法的精度和计算时间如下：

accuracy: 2.284 m Running time: 30.234 s

4.3 神经网络多层感知器法

多层感知器（MP，Multilayer Perceptron）是一种前馈人工神经网络模型，其将输入的多个数据集映射到单一的输出的数据集上，如图 3 所示。

图 3 多层感知器的概述图



运行 `run_MP.py`，得到神经网络多层感知器法的精度和计算时间如下：

accuracy: 2.399 m Running time: 130.669 s

4.4 随机森林法

随机森林顾名思义，是用随机的方式建立一个森林，森林里面有很多的决策树组成，随机森林的每一棵决策树之间是没有关联的。在得到森林之后，当有一个新的输入样本进入的时候，就让森林中的每一棵决策树分别进行一下判断，看看这个样本应该属于哪一类（对于分类算法），然后看看哪一类被选择最多，就预测这个样本为那一类。

运行 `run_RF.py`，得到随机森林法的精度和计算时间如下：

accuracy: 2.640 m Running time: 35.894 s

4.5 SVM 法

支持向量机（Support Vector Machine, SVM）是一类按监督学习（supervised learning）方式对数据进行二元分类的广义线性分类器（generalized linear classifier），其决策边界是对学习样本求解的最大边距超平面（maximum-margin hyperplane）。

运行 `run_SVM.py`，得到 SVM 法的精度和计算时间如下：

accuracy: 2.460 m Running time: 72.939 s

4.6 二分法

将平面区域分成相等大小的四部分，如图 4，计算出每部分中所有测量点的平均值，根据欧氏距离最近的原则预测数据点所在的位置。然后再将所在部分分成相等大小的四部分，预测数据点所在的位置，如图 5。依次类推，知道分成最小部分，即预测的数据点的具体位置。

图 4 第一次划分

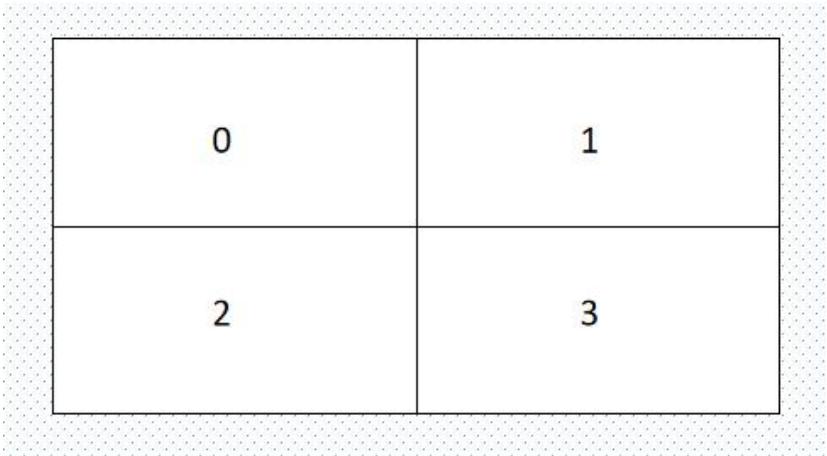
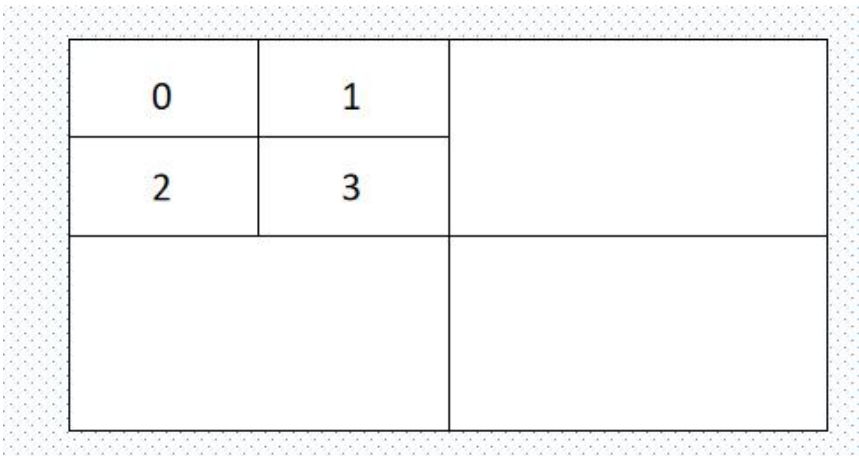


图 5 第二次划分

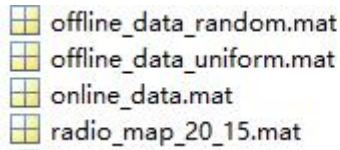


运行 run_Division.py，得到二分法的精度和计算时间为：

accuracy: 2.850 m Running time: 8.230 s

四、 Matlab 实现 WiFi 定位

构建数据库：运行 main_raytracing.m 文件，得到我们模拟的数据库



offline_data_random.mat
offline_data_uniform.mat
online_data.mat
radio_map_20_15.mat

构建离线数据库，即就是我们的训练数据集 offline_data_random.mat 对数据进行随机采样，默认 30000 个数据

```
function [data, labels] = get_offline_data_random(fingerprint, data_num)
%模拟离线数据采集，随机采样
if nargin == 1
    data_num = 30000; %默认30000个数据
end
```

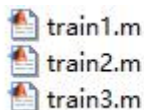
利用 MATLAB 自带的神经网络模型对得到的数据进行训练：（分别采用不同的训练函数，得到不同的网络训练结果，并保存在对应的.mat 文件中）

```
load('offline_data_random.mat');
x = offline_rss';
t = offline_location';

trainFcn = 'trainscg'; %共轭梯度下降算法
% Create a Fitting Network
hiddenLayerSize = 10;
net = fitnet(hiddenLayerSize, trainFcn);

% Setup Division of Data for Training, Validation, Testing
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
```

我们使用了三个不同的训练函数，最终训练得到三个不同的结果.



train1.m
train2.m
train3.m

接着就是构建测试数据集，运行 get_online_data.m 文件，得到我们的测试数据集。

```
function [ trace, rss ] = get_online_data( fingerprint, gridSize, roomLength, roomWidth, t )
%得到仿真位置指纹法的数据
%输入：rss仿真环境数据集，这个数据集的gridSize，房间尺寸，生成数据的个数
%输出：位置点轨迹，以及轨迹上每个点上的rss
```

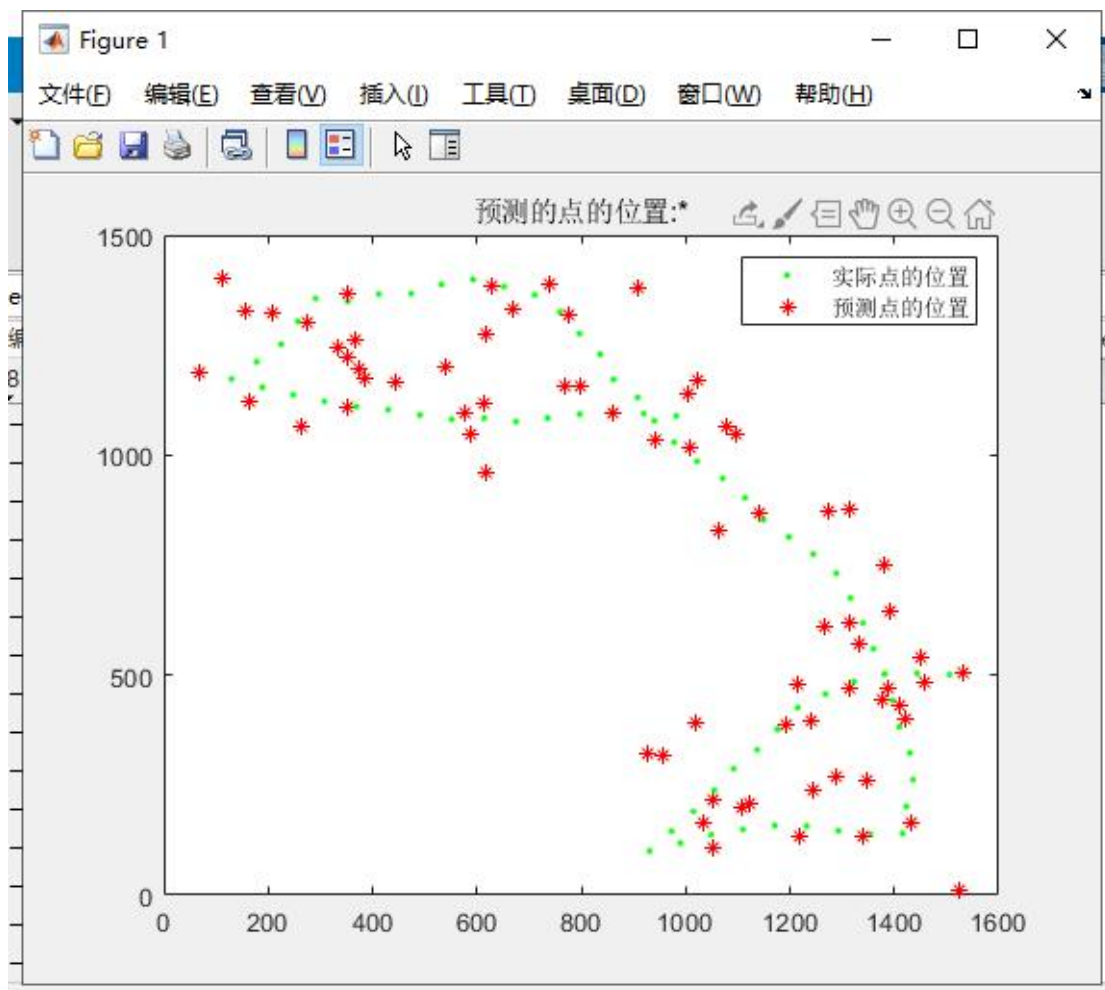
最后就是利用之前训练的到的神经网络对我们的测试数据集进行测试。运行 test.m 文件，并在文件中修改你所需要的神经网络模型（训练步骤得到的网络模型）

```

load net_br.mat;
load('online_data.mat')
rss=rss(150:250,:);
trace=trace(150:250,:);
a=rss';
result=sim(net,a);
result=result';
figure;
l=length(rss);
for i=1:l
    h1=plot(trace(i, 1), trace(i, 2), 'g.'):
    hold on;
    h2=plot(result(i, 1), result(i, 2), 'r*');
    legend([h1(1),h2(1)], '实际点的位置', '预测点的位置');
    title('预测的点的位置:');
end

```

结果如图所示：



五、 卡尔曼滤波和粒子滤波

卡尔曼滤波可以根据一些已知的量来预测未知的量, 这些量受到的干扰必须得近似高斯噪声。卡尔曼滤波对近期未来的数据的预测还是很精确的, 只要我们的数据只受到高斯噪声的干扰。

粒子滤波算法源于蒙特卡洛思想, 即以某事件出现的频率来指代该事件的概率。通俗的讲, 粒子滤波也是能用已知的一些数据预测未来的数据, 但是粒子滤波可以不局限于高斯噪声, 原理上粒子滤波可以驾驭所有的非线性、非高斯系统。

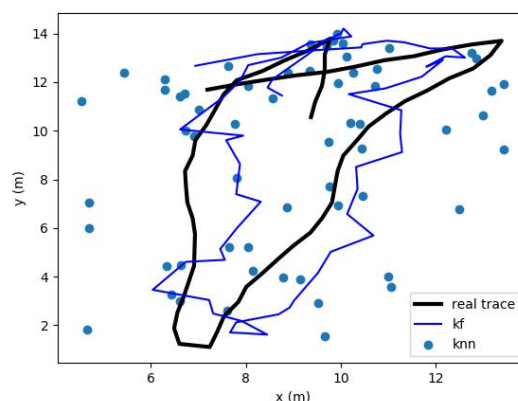
分别用卡尔曼滤波和粒子滤波对 KNN 算法得到的预测结果进行处理:

运行 `Kalman_Filter.py`, 得到卡尔曼滤波前后精度的变化和卡尔曼滤波的可视化效果图 (图 6)。

卡尔曼滤波前的精度: **2.254 m**

卡尔曼滤波后的精度: **1.848 m**

图 6 卡尔曼滤波的可视化效果图

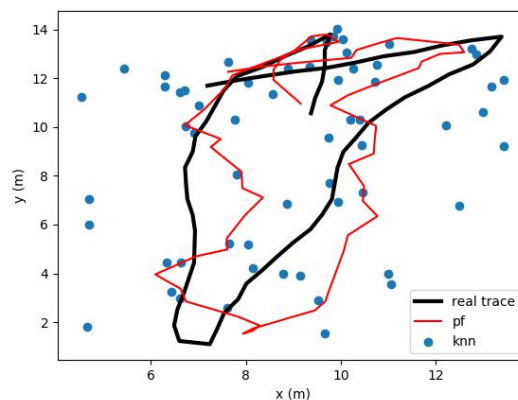


运行 `Particle_Filter.py`, 得到粒子滤波前后精度的变化和粒子滤波的可视化效果图 (图 7)。

粒子滤波前的精度: **2.254 m**

粒子滤波后的精度: **1.791 m**

图 7 粒子滤波的可视化效果图



六、 实习总结

这次实习我们主要研究了室内定位数据集的构建、机器学习算法和卡尔曼滤波或粒子滤波在室内定位的应用。

在现实中构建室内定位数据集需要的硬件设施多，付出的人力资源大，基于这种条件的限制，我们决定使用仿真数据集。室内定位仿真数据集的构建主要应用了射线跟踪法，通过计算出发射点与接收点之间的多条传播路径，主要包括了直接反射路径，前、后、左、右、上、下等六个平面的反射路径，分别对各个路径的信号进行分析，然后叠加，再乘以一个误差系数，模拟真实环境中的障碍物等阻碍信息得到接收点上的信号。

构建完数据集后我们开始利用各种机器学习算法对数据集进行预测，并计算出各种机器学习算法的预测精度和计算时间。综合比较之后不难发现，在所有的机器学习算法之中，KNN 算法表现最理想。在实验中，我们还对超参数 k 以及训练量进行了研究，得到了超参数 k 与 $score$ 的关系曲线以及训练量与 $accuracy$ 的关系曲线。可以发现，随着超参数和训练量的增加，KNN 算法的预测精度趋于稳定。

在利用 KNN 算法得到预测结果之后，我们还可以通过卡尔曼滤波或粒子滤波进一步提高预测的精度，其效果明显而且计算速度快。

综上所述，室内定位技术是处于发展中的研究领域，虽然有着各种各样的方法，但是定位精度普遍不高。再加上室内环境不可控的温度，物体遮挡等因素都会对定位结果产生影响，这就导致室内定位技术无法广泛应用到生活领域。我相信，在未来，室内定位技术会得到进一步的发展，提出更多的算法，达到可以具体应用的精度要求。

注：此报告参考了网上的很多资料，出于方便，没有在报告中标注出来，如有侵权，请联系改正。