北京大学

硕士研究生学位论文

题目：基于大规模高斯过程回归的城市空气质量估计

姓　　名：陈晓东

学　　号：1301214327

院　　系：信息科学技术学院

专　　业：智能科学与技术

研究方向：复杂网络建模与智能分析

导　　师：宋国杰 副教授

二零一六年六月

版权声明

任何收存和保管本论文各种版本的单位和个人, 未经本论文作者同意, 不得将本论文转借他人, 亦不得随意复制、抄录、拍照或以任何方式传播。否则, 引起有碍作者著作权之问题, 将可能承担法律责任。

摘要

空气质量和人类日常生活息息相关，它反映了城市的空气污染状况。空气污染常常是由人类生产生活的排放所造成，而反过来空气污染又会阻碍人类的正常生活生产活动。全面实时地对城市空气质量进行监测成为了一个新挑战。但是无论采用何种方式对城市空气质量进行监测，都无法完整地覆盖所有区域和所有时间段。因此，我们需要研究如何根据采样的观察样本，来估计出其他未监测区域的空气质量状况。

传统的方式主要是通过监测站来采样空气质量的，但是由于监测站造价昂贵，在城区覆盖面非常稀疏，无法适用于细粒度的空气质量估计，实用性不足。在本文，我们关注一种新型的空气监测方式——移动传感器监测。通过安装传感器，使得我们可以在汽车行进过程中进行细粒度时空的空气质量状况采样。与此同时，这种移动采样方式带来的大量数据，需要我们提出一个更高效和准确的方法来处理。

本文利用了高斯过程模型的适用性、扩展性和理论性强等优点，结合了空气质量监测场景下的时空近似性的特性，提出了基于大规模高斯过程的城市空气质量估计算法，并在真实数据集上验证了各种场景算法的性能表现。

本文创新点如下：

- 结合移动监测数据和气候数据，提出了一种基于高斯过程回归的回归方法，通过引入新的核函数来描述采样数据在气候、时空等特征上的距离，用来估计没有监测到区域的当前空气污染状况。
- 针对移动车采样数据量大的情况，给出一种基于 k-d 树的空间划分和采样方法，用来提高空气质量估计的效率。
- 实验表明本文提出方法准确度优于传统的插值方法，并在时间效率上有了若干个量级的提升。

关键词：高斯过程回归, 空气质量估计, 可伸缩算法

Estimating the Urban Air using the Scalable Gaussian Process Regression

Chen Xiaodong (Machine Intelligence)
Directed by Associate Prof. Song Guojie

ABSTRACT

The air quality is closely related to people's daily life. It reflects the pollution status of urban air. The production, life of human activities usually produce the air pollution. In turn, air pollution is also a serious impediment to development of human beings. It becomes a challenge to monitor the urban air quality in anywhere and anytime. However, it is no way to cover all the area in all the time using any sampling methods. And because of that, we need to study how to estimate the other unmonitored areas with sampled observations.

Currently, major air pollutants are typically monitored by networks of static stations. However, these stations are also costly to ac- quire and maintain, which results in limited information being collected about the spatial distribution of air pollutants. In this paper, we focus on a new way to monitor the air quality – sensors located on the car, which differ from the traditional static station. By the way of the mobile sensors, we can sample the air quality in a small fined grained while the cars is moving along the road. In the same time, we need to propose a new framework to handle the numerous records generated by this mobile sampling methods

In this paper, taking advantage of both the Gauss process and the locality of the air pollution, we propose a urban air quality estimation method based on the scalable gaussian process for air pollution of the unmonitored areas.

we summary our contributions as follow:

- Combined with the mobile sampled data and the climate data, we propose a newly Gauss process based regression method to estimate the air quality of the unmonitored area, which incorporates a kernel function to measure the distance between the spatio-tempo, climate and pollutions features.

- we also proposed a method based on the k-d tree structure to divide the sample space into layers to enhance the efficiency of the process of estimation.
- Experiments show that our proposed method outperforms the traditional spatial regression method in both precision and running time.

KEYWORDS: Gauss Process Regression, Air Quality Estimation, Scalable Algorithm

目录

插图

表格

引言

0.1 研究背景

城市的空气质量与人类日常生产和生活休戚相关。空气质量的好坏反映了一个城市空气污染的程度，它通常依据的是空气中污染物的浓度高低来判断的。在不同的时间和地点，空气污染物的浓度都会有所不同。这其中受到很多因素的影响，来自固定和流动污染源的人为污染物排大小是其中主要的因素之一，其中包括交通工具、工业污染、居民生活等因素相关。另外，城市的建筑物密度和地形地貌和当时刻的气象条件也影响了空气质量的好坏。在空气污染物中，例如超细悬浮微粒 (ultrafine particles) 在空间分布有很大的差异性 (如图 1)。常见的代表性超细悬浮微粒, 如 $PM_{2.5}$ 和 PM_{10} 对于研究如何控制空气污染有很大帮助。

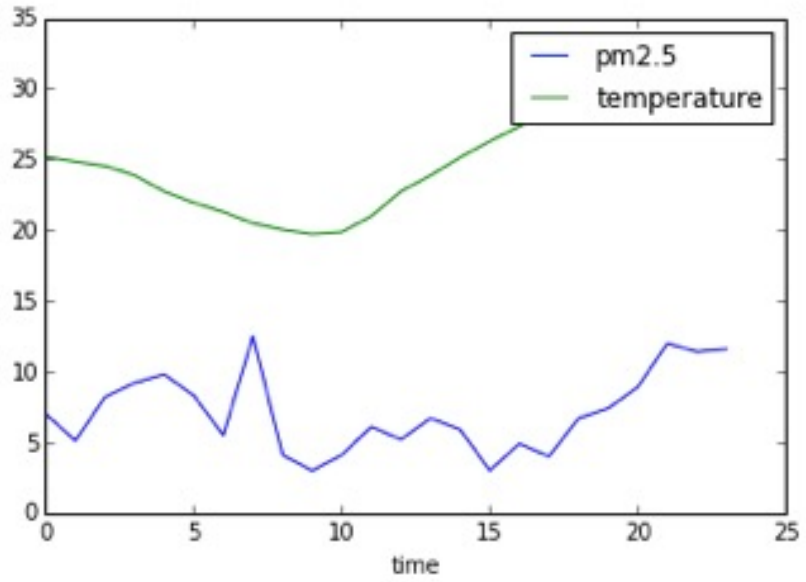

图 1　PM2.5 和气温随时间变化图

城市空气质量等级是据城市空气环境质量标准和各项污染物的生态环境效应及其对人体健康的影响, 所确定的污染指数分级以及相应的污染物浓度限值, 图 2是是美国常用的污染物分集标准 ①。该分级标准是城市空气质量预报的实施标准，也是进行城

① www.airnow.gov

市环境功能分区和空气质量评价的主要依据。

空气质量指数	空气质量指数级别（状况）及表示颜色	对健康影响情况	建议采取的措施
0 - 50	一级（优）	空气质量令人满意，基本无空气污染	各类人群可正常活动
51 -100	二级（良）	空气质量可接受，但某些污染物可能对极少数异常敏感人群健康有较弱影响	极少数异常敏感人群应减少户外活动
101-150	三级（轻度污染）	易感人群症状有轻度加剧，健康人群出现刺激症状	儿童、老年人及心脏病、呼吸系统疾病患者应减少长时间、高强度的户外锻炼
151-200	四级（中度污染）	进一步加剧易感人群症状，可能对健康人群心脏、呼吸系统有影响	儿童、老年人及心脏病、呼吸系统疾病患者避免长时间、高强度的户外锻炼，一般人群适量减少户外运动
201-300	五级（重度污染）	心脏病和肺病患者症状显著加剧，运动耐受力降低，健康人群普遍出现症状	儿童、老年人及心脏病、肺病患者应停留在室内，停止户外运动，一般人群减少户外运动
300+	六级（严重污染）	健康人群运动耐受力降低，有明显强烈症状，提前出现某些疾病	儿童、老年人和病人应停留在室内，避免体力消耗，一般人群避免户外活动

图 2　空气质量指数

在现实中，我们通常利用固定的空气质量监测站来监测空气质量。它的功能是对存在于大气、空气中的污染物质进行定点、连续或者定时的采样、测量和分析。为了对空气进行监测，一般在一个环保重点城市设立若干个空气站，站内安装多参数自动监测仪器作连续自动监测，将监测结果实时存储并加以分析后得到相关的数据。其中待监测因子包括：污染极细颗粒物（PM2.5，PM10）、臭氧、二氧化硫、一氧化碳、硫化氢、氮氧化物、挥发性有机污染物、总悬浮颗粒物、铅、苯、气象参数和能见度等。但是这一种方式金钱和人力成本都非常高。在一个城区里，往往只能建数量有限的基站，覆盖面非常稀疏。大量没有监测到的区域需要我们提出高效的方法进行估计。常见的机器学习做法，就是利用空间插值的方法，对未监测区域进行回归预测。这类空间插值方法的本质都是基于已有的观察记录加权，然后回归估计出其他区域的空气质量。

近年来，兴起一种新的监测方式。通过在汽车顶部安装传感器，那样就可以在汽车行驶过程中，随时随地地对空气质量进行采样。这一种新的监测方式，它的成本非常低廉，安装非常方变。通过在多辆车上安装传感器（如图 3），我们的采样可以覆盖几乎整个市区。但与此同时，因为传感器放在了车上，它们有时会过于接近污染源，导致测量带有偏差性，使得这一种测量方式的精度相对于传统基站来得没有这么高。

0.2　研究意义

传统的空气质量估计往往有两大类，一类是基于领域知识，利用气候学的知识，模拟污染物的扩散过程，建立传播的微分方程，从而推断污染物的浓度。这一类方法，计算

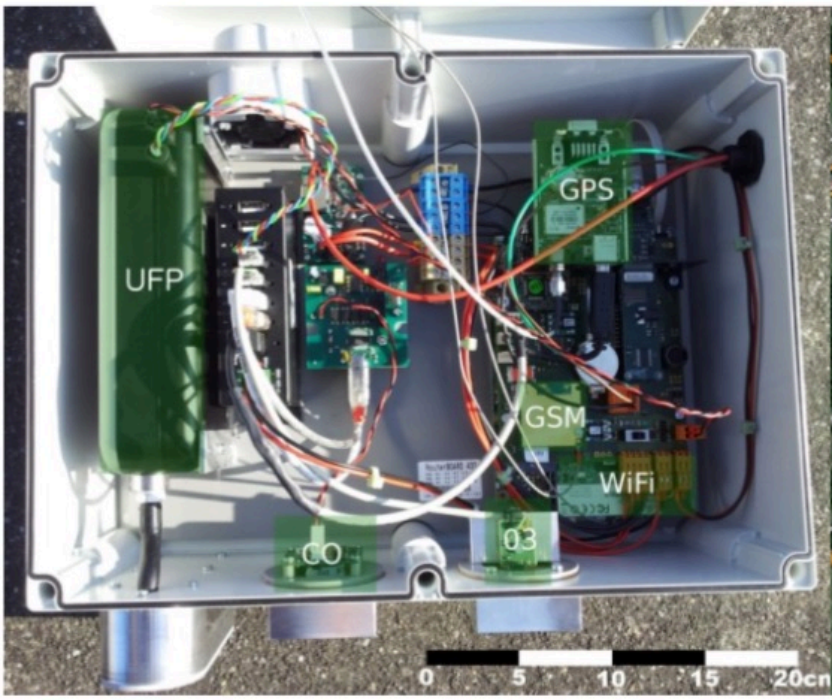

图 3　移动监测设备

成本极大，需要极强的领域知识。第二大类方法就是基于数据驱动的方法，建立统计学模型，利用数学上的回归预测，求出没有监测到区域的污染物浓度。过去基于第二类方法的工作，受限于采集回来的数据时空覆盖面非常有限（例如图 4所示），而且采样的粒度至少为一个小时。导致如果数据很难利用于实时预测当前的空气质量，只能用于预测长期（如一周，一月和一季度）的平均污染状况。这样一来，空气质量对人类出行活动的意义大大减少。

移动检测设备的引入，大大缓解了采样稀疏的问题。通过多辆承载设备的汽车在城区行驶，我们可以获得时空范围覆盖较广的采样数据。并且其采样频率远高于传统监测站，甚至达到几十秒一次。通过利用移动采样数据，我们就可以对城区各个区域进行细粒度的空气质量分析，从而对人们出行、政府决策提供便利。

移动车的引入，将会带来两方面的新问题：其一，它将会产生大量的记录，其数据规模往往是传统方法的几个量级。同时，我们估计空气质量的频率也远高于传统方法（接近实时，如 10 分钟一次）。一些经典的有效分析方法因为时间复杂度的原因无法在很好的处理这些数据；其二，传统数据采样方法，采样（时空）粒度大，分析处理的时间空间尺度也大（一周以上，几平方公里大小），估计出来的结果往往是一个较大大区域的平均水平，无法小尺度的情况。

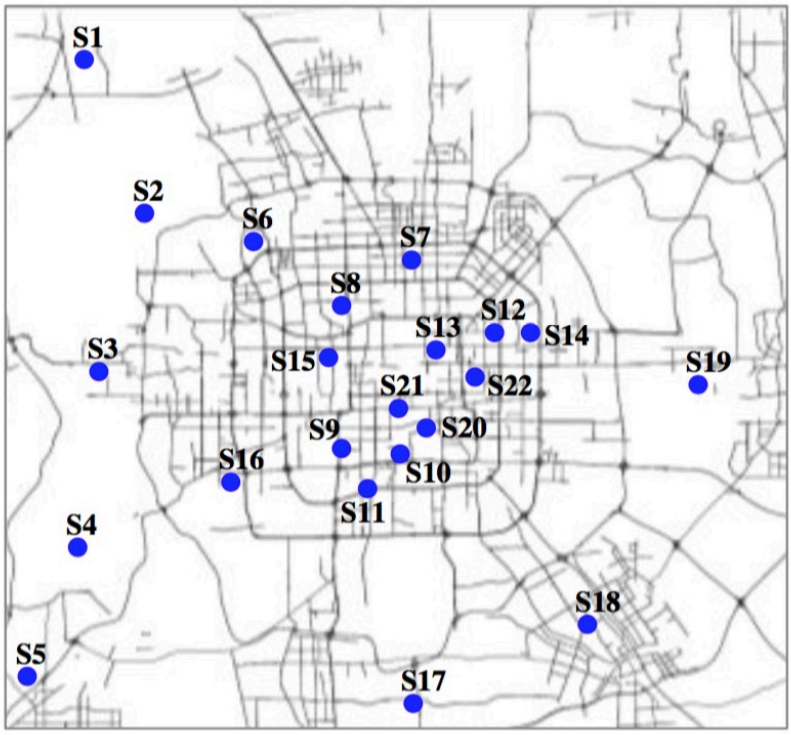

图 4　城市监测站点分布

0.3　问题的提出和挑战

区别于传统空气质量估计，我们希望能利用移动车采集回来的大量实时数据，再加上辅助的气候数据，来进行实时的空气质量估计。在这一过程，主要有如下几个挑战：

- 除了移动数据，我们除了利用移动数据，还可以利用原有的固定基站数据进行辅助，如何融合多源数据，进行统一建模是一个新的挑战
- 传统的空气质量估计，预测时间粒度很大，以月为单位。它们在处理过程中，会把大量原始数据聚合，失去了很多信息，无法进行实时估计。在移动监测的场景下，我们如何利用采样样本的时空信息。
- 高斯过程为基础的回归模型都面临着时间复杂度高的问题，如何利用问题的特点，来提高估计的效率是实时估计的保证。

因此我们的问题是：如何利用多源的采样数据，对城市空气进行空间时间上小尺度的空气污染状况估计。

0.4 本文主要工作

本文的整理了多台移动车数个月采集的移动数据以及北京对应几个月的气候数据。并在此基础上做了如下工作：

- 在高斯回归过程的基础上扩展了其核函数，用于描述在一个时间窗口内，多个采样样本之间的距离协方差大小。
- 为了克服模型的训练复杂度过高问题。我们利用 k-d 对空间进行了划分，在每一层，选取代表样本作为这个区域的代表，用以简化协方差矩阵的计算，从而提高模型效率。
- 通过实验验证，本文提出方法的误差率均低于传统的方法，并在时间效率上有若干个量级提升。

0.5 文章组织结构

从结构上, 本文一共分为六部分, 安排如下:

引言部分，介绍了本文的研究背景、研究意义以及问题的提出。并说明了本文的研究内容以及论文的组织结构。

第一章主要介绍了和空气质量有关的工作，包括基于污染物浓度时间序列的回归方法，基于气象扩散模型的污染物传播模拟方法和基于静止基站的空间回归方法。

第二章首先介绍如何基于高斯过程进行扩展，引入对时空信息的支持。

第三章详细介绍如何利用 k-d 树简化协方差矩阵的计算，利用代表样本的距离近似于普通样本间的距离。

第四章进行了相关的实验，比较了我们方法在误差和运行时间上和其他方法的差异。

结论部分对本文的研究工作进行了总结, 分析目前模型和实验中的不足, 并对未来工作提出展望。

第一章　相关工作

大气颗粒物 (PM, particulate matter), 是悬浮在地球大气中的固液小颗粒物质, 这些颗粒物可以由人类生产生活制造, 也可以由大自然的运转中生成。科学已经表明, 这些颗粒物无论对天气和降水都有影响, 并且会最终将影响人类的健康。大气颗粒物由多种物质构成, 相比于从自然生成的 (如火山爆发、沙尘暴等), 我们更关心人工造成的。因为后者的危害往往更大。正因为如此, 空气质量的相关课题一直是城市计算 [4, 5] 这一大框架下的重要领域。

1.1　数据的来源

传统的空气污染颊侧主要都是利用放置在固定监测站的昂贵设备, 这些设备互相之间距离非常远, 监测的空间粒度也非常大。近年来, 得益于科技的进步, 越来越多的途径能获得 $PM_{2.5}$ 的值。

1.1.1　遥感 (remote sensing)

通过卫星遥感技术, 可以获得地表空气污染的具体状况 [7]。在 [8] 中, 文章通过从卫星上检测气溶胶光学深度 (AOD, aerosol optical depths) 数据, 并利用一种叫全球化学转换模型 (global chemical transport model) 去分析 AOD 和 PM2.5 的关系, 从而得到 PM2.5 的测量值。

1.1.2　监测站 (monitor station)

空气质量监测站的功能主要是对空气中的常规污染因子和气象参数进行 24 小时连续在线的监测, 将分析出的数据提供给环保局作为空气质量好坏参考, 并辅助环保决策, 其中待监测因子包括：污染极细颗粒物 (PM2.5, PM10), 臭氧, 二氧化硫, 一氧化碳, 硫化氢, 氮氧化物, 挥发性有机污染物, 总悬浮颗粒物, 铅, 苯, 气象参数, 能见度等。这是最常见的获取空气污染信息的方式 [9]

1.1.3　移动传感器 (mobile sensor)

用于空去污染监测的移动传感器最初是被设计安装在公共交通设施上的 [10], 这些公共交通往往具有比较固定的路线。后来, 这些设备又更多地安装在了私人汽车上, 这

样，汽车在行驶过程中就可以采集大量数据。覆盖面比更为广泛。因为低廉，容易携带和维护，近年来越来越多研究关注这种数据采集方式 [11, 12]，甚至随着传感器进一步变小，开始有人在手机等手持移动设备进行数据的采集 [13, 14]

1.1.4 外部数据

外部数据主要分两大类，一类是和空气质量直接相关的工业排放和交通流量数据。它们都是污染物的直接造成原因；第二类数据就是人类活动的相关数据，如常见的天气数据，天气的具体条件会间接影响空气中细颗粒污染物的浓度大小。这些外部数据从侧面反映了当前的空气污染状况，研究 [15, 16, 17] 表明它们对城市空气质量估计有提升的作用。目前有部分外部数据集可以通过公开的渠道 [2] 获取。

1.2 研究的方法

空气质量监测一直是研究的热点问题，有许多相关工作已经实现了对空气质量进行初步估计，下面我们介绍几大类常见的空气污染物浓度估计的方法。

1.2.1 自回归模型 (autoregressive model)

自回归模型 (Autoregressive Model) 是用自身做回归变量的过程，即利用前期若干时刻的随机变量的线性组合来描述以后某时刻随机变量的线性回归模型，它是时间序列中的一种常见形式。

考虑一个时间序列 $(y_1, y_2, ..., y_n)$, 它的 p 阶自回归模型 (记为 AR(P)) 表明序列中的 y_t 是前 p 个序列的线性组合及误差项的函数，一般形式为：

$$y_t = \phi_0 + \phi_1 y_{t-1} + \phi_2 y_{t-2} + ... + \phi_p y_{t-p} + e_t$$

其中, $\phi_i, i = 0, 1, ..., p$ 为模型参数,e_t 是一个均值为 0 的，方差为 σ 的白噪声。

但由于现实中时间序列多不是平稳的，因此人们提出了一种适用性更广泛的自回归积分滑动平均模型 (ARIMA)，是由博克思 (Box) 和詹金斯 (Jenkins) 于 70 年代初提出的一著名时间序列预测方法，所以又称为 box-jenkins 模型、博克思-詹金斯法。其中 ARIMA(p，d，q) 称为差分自回归移动平均模型，AR 是自回归, p 为自回归项; MA 为移动平均，q 为移动平均项数，d 为时间序列成为平稳时所做的差分次数。

ARIMA 模型的基本思想是：将预测对象随时间推移而形成的数据序列视为一个随机序列，用一定的数学模型来近似描述这个序列。这个模型一旦被识别后就可以从时间序列的过去值及现在值来预测未来值。

ARIMA(p,d,q) 可以形式化的表示为：

$$(1-\sum_{i=1}^{p}\phi_i L^i)(1-L)^d X_t = (1+\sum_{i=1}^{q}\theta_i L_i)\epsilon_t$$

其中 p 是自回归项数，q 是滑动平均想输，d 为使之成为平稳序列所做的差分次数 (阶数)。

ARIMA 模型将非平稳时间序列转化为平稳时间序列，然后将因变量仅对它的滞后值以及随机误差项的现值和滞后值进行回归所建立的模型。

1.2.2　扩散模型方法 (atmospheric diffusion model)

大气污染物扩散模式是模拟大气污染物的输送、扩散、迁移过程，预测在不同污染源条件、气象条件及下垫面条件下某污染物浓度时空分布的数学模型，是低层大气中污染物迁移和扩散规律的、简单化的数学描述。根据不同的建模理论体系、污染物迁移、扩散过程以及不同的描述对象，模式的形式也各不相同。一般使用得最广泛、适用于小尺度、定常流场中连续高架点源污染物浓度估算的为高斯烟流模式 (Gauss plume model)。

高斯烟羽模式是计算释入大气中的气载污染物下风向浓度的应用最广的方法 [18]。此模式假定烟羽中污染物浓度在水平方向和垂直方向都遵循高斯分布。对于在恒定气象条件 (指风向、风速、大气稳定度不随时间而改变) 高架点源的连续排放，在考虑了烟羽在地面的全反射后，下风向任一点的污染物浓度 $C(x,y,z)$ 有：

$$C(x,y,z)=\frac{\sigma}{2\pi\sigma_y\sigma_z u}e^{-\frac{y^2}{2\sigma_y^2}}(e^{-\frac{(z-He)^2}{2\sigma_z^2}}+e^{-\frac{(z+He)^2}{2\sigma_z^2}})$$

其中 $C(x,y,z)$ 为下风处某点 (x,y,z) 处的空气污染物浓度，其中：

- x – 下风向距离
- y – 横截风向距离
- z – 距下风地面高度
- u – 排放高度处的平均风速
- He – 有效排放高度
- σ_y ,σ_z – 水平方向和垂直方向扩散参数

该类模型适用于小尺度范围的污染物扩散，并且改模型只适用于单源污染物的建模。另外，该类模型对大气条件有比较严苛的限制，因此适用范围比较窄。对于范围较广的城市空气质量监测，这类方法不适合适用。

1.2.3 空间插值方法 (spatial interpolation)

空间插值 (spatial interpolation) 是一种通过部分地区的已观察数据去估计未采样地区数值的一类方法。一般而言，观察数据为数值型的特征。空间插值是很多空间场景下的重要解决手段，如我们的空气质量监测场景就是一典型空间插值问题。通过已经采集到的数据，去估计没有监测到区域的空气质量。

空间插值方法是用来解决大量空间采样场景下，采样成本 (时间、金钱上的) 过高，无法获得覆盖面较广的采样数据的方法。例如空气质量监测中，如果我们用传统的基站去收集空气质量数据，只能覆盖城区少数的若干个“点”的数据。

空间插值的理论基础是“地理学第一定律”这一基本假设。也就是说，空间上距离越靠近的点，应该有相似的特征值；而相反，空间上距离越远的店，它们拥有相似的特征值可能性越小。例如空气质量监测场景下的污染物 $PM_{2.5}$ 的监测就大体服从这一假设。我们可以通过采集而来的空气质量数据，利用“地理学第一定律”来求得没有监测区域的 $PM_{2.5}$ 的值。

根据空间插值的基本假设，其方法思路在于，找到描述空间中任意两点之间的插值权重的函数，从而通过已有的观察数据加权回归出任意一个点的目标值。不同权重函数会有不同的插值效果，所以不同的空间插值方法目标都是希望基于问题场景的理解找到一个适合的权重函数。

空间离散数据经过插值后主要用途有：可以用于绘制数据的等高线平面图；可以用于计算具体空间某点的目标值；可以便于人们基于空间插值图来进行相应的决策。

下面我们将介绍几种常见的空间插值方法：

反距离加权法 (IDW, Inverse Distance Weighted)

反距离加权法是一种最简单和常用的空间插值方法，它直接通过插值点和样本点的距离的反比来作为权重加权。也就是说，距离插值点越远的样本点权重越小，而距离插值点越近的样本点的权重越大。

假设空间里有一组点 $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x_i}, y_i)|i = 1, 2, ..., n\}$, 我们定义距离函数为 $d(\mathbf{x_i}, \mathbf{x_j})$。给定一个要插值点 $\mathbf{x}_*$, 那么按照假设，它和已有样本的权重应该满足：

$$\omega_i \propto \frac{1}{d(\mathbf{x}_*, \mathbf{x_i})}$$

其中 $i = 1, 2, ..., n$

为了确保权重有意义，我们对所有权重进行归一化，因此有：

$$\omega_i = \frac{1/d(\mathbf{x}_*, \mathbf{x_i})}{\sum_{j=1}^{n} 1/d(\mathbf{x}_*, \mathbf{x_j})}$$

最后按照定义，我们给出插值点 $\mathbf{x}_*$ 的回归值 y_* 的表达式：

$$y_* = \sum_{i=1}^{n} \omega_i y_i$$

反距离加权法一般而言准确率不高，但是算法复杂度只有 $\mathrm{O}(n)$, 时间效率非常高。而且实现起来也非常方便。

样条函数法 (spline function)

一类分段光滑，并且在各段交界处也有一定光滑性的函数称为样条函数。样条函数法是利用样条函数进行差值的方法，它常常采用一阶导数和二阶导数的连续分段多项式，去逼近已知的样本点，产生平滑的插值曲线。这种方法很适合根据密集的点内插等值线，可用于逐渐变化的表面。方法易操作，计算量不大，运算速度较快，可以带来较好的视觉效果。但是，难以估计内插时的误差，点稀的时候效果不好。

趋势面法 (trend surface)

趋势面法属于统计方法插值的一种, 前提假设是一系列相互相关的空间数据, 插值点的趋势和周期是与它相关的其他变量的函数。它也是一种比较常用的整体插值方法，可以用一个平滑的数学平面来描述某种在空间连续变化的地理属性。它的核心思想是, 先用现有的已知点数据拟合出一个平滑的数学平面方程，再根据该方程计算未知点的数据值。也就是说，它根据样本点的属性数据和地理坐标的关系，进行多元回归分析得到平滑的数学平面方程，即趋势面是个平滑函数。而实际上，除非出现数据点少且恰好被曲面通过等特殊情况，趋势面一般很难正好通过已知数据点，所以趋势面法是一种近似插值的方法。

土地利用回归 (land-use regression)

土地利用回归 (LUR, land-use regression) 是一种数据驱动的空间回归模型 [19]。LUR 利用从地理信息系统获取的可解释变量作为模型的输入，这些可解释变量包含了不同区域的土地利用信息，例如交通流量、人口密度和海拔等。它通过描述污染程度 p 和

解释变量集合 $\{A_1, ..., A_n\}$:

$$\ln(p) = a + s_1(A_1) + s_2(A_2) + ... + s_n(A_n) + \epsilon$$

这个模型完全没有利用空气污染浓度作为输入，单纯地利用这些土地利用信息作为输入。

克里金方法 (Kriging)

克里金方法是一种空间自协方差的方法，最早用于矿物勘测的问题当中。一方面考虑空气污染物变量的随机性，也考虑到样本点之间的相关性。克里金方法本质上是为了在满足插值方法最小的条件下，给出最佳的线性无偏估计，同时还给出插值的方差值。

给定空间里有一组点 $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x_i}, y_i)|i = 1, 2, ..., n\}$，对于给定插值点 $\mathbf{x}_*$ 和其目标值 y_*, 我们假设满足关系：

$$y_* = \sum_{i=1}^{n} \omega_i y_i$$

其中权重 $\{\omega_i\}|_{i=1,2...,n}$ 应该满足以下无偏和最小方差约束：

$$\begin{aligned} \sum_{i=1}^{n} \omega_i &= 1 \\ \sum_{i=1}^{n} \omega_i \mathrm{Cov}(\mathbf{x_i}, \mathbf{x_j}) &= \mathrm{Cov}(\mathbf{x}_*, \mathbf{x_j})(j = 1, 2, ..., n) \end{aligned}$$

其中 $\mathrm{Cov}(\mathbf{x_i}, \mathbf{x_j})$ 是表示 y_i 和 y_j 的协方差函数。对于任意两个随机变量 X 和 Y，它们的协方差可定义为：

$$\mathrm{Cov}(X, Y) = \mathbb{E}[(X - \mathbb{E}(X))(Y - \mathbb{E}(Y)]$$

通过求解如上方程组，我们就可以最终求得 y_* 的值。

克里金方法通过约束各个观察样本所代表变量方差的方差，来实现一个局部最优的无偏估计，广泛适用于各种空间插值问题，并且插值的结果可信度很高。但是缺点是它的计算复杂度很高，时间效率低下。

1.2.4 其他研究方法

相对于直接使用回归插值等方法估计空气污染状况，一些研究 [9, 1, 20, 21, 22, 6] 从别的角度切入，利用近年来兴起地从机器学习、统计等领域的方法对空气质量进行估计。又或者，有些研究 [11, 9] 会利用许多外部数据源去辅助估计。

1.3　目前研究工作的不足

前面介绍了目前主流的几大类空气中污染物浓度的估计方法，现在我们来分析下他们的不足之处。

从方法上来说，这些方法不足之处都有：

- 基于扩散模型的方法：模型建模复杂，涉及太多领域知识，难以应用
- 基于自回归模型的方法：没有基于问题场景特点建模的能力
- 空间插值方法：考虑到了空间建模的特性，是目前最常用、效果最好的一类方法

而关于空间插值本身，各类方法优劣不同，其中基于克里金的方法由于其模型本身描述能力强，是当中效果最好的模型。

从研究内容上来说，首先，过往的相关工作都倾向于研究估计较长时间内的空气污染浓度平均值，也就是估计的时空粒度过大 (至少一周时间)，缺乏现实的人类日常出行指导意义。由于粒度过大，模型对于局部的预测精度也往往不如人意。其次，往往通过这些方法得到的污染浓度值都是一个准确的值，无法体现空气污染浓度的不确定性，也没法体现模型收到噪音数据干扰下的预测置信区间。最后，有些方法的原始数据采集粒度非常小，但是因为目的在于预测大尺度的空气质量，往往将原始数据聚合，使得丧失了许多小粒度的时空信息，无法再进行实时和小尺度的预测。

在数据上，本文采用了移动监测的方式，使得采样的数据覆盖时空面比较广泛，使得我们可以进行更细粒度的估计。在空气质量估计的场景下，污染物在时空分布上有很强的局部性 (locality)，即在时空上接近的样本应该有相似的空气污染物浓度。本文引入了高斯过程对这一点进行数学建模，构建了核函数去描述了不同样本污染物浓度协方差和它们在时空上分布的关系。利用高斯过程这一工具，得出了空气质量的估计值以及其方差。

1.4　本章小结

空气质量估计是一个交叉学科的热点问题。本章主要总结了污染物监测的数据来源，和几类常见的空气质量估计分析方法，其中包括扩散模型估计、时间序列回归和空间插值回归等，尤其详细介绍了和本文最相关的空间插值方法。并且分析了它们的优缺点和针对我们目前研究问题的不足之处。

第二章　基于高斯过程的时空回归模型

这一章主要研究的是如何对空气质量实时估计这一问题进行建模，围绕着从采样而来的样本数据进行训练，然后对未监测区域进行污染物浓度输出的这一过程。整个研究框架如图 2.1所示：

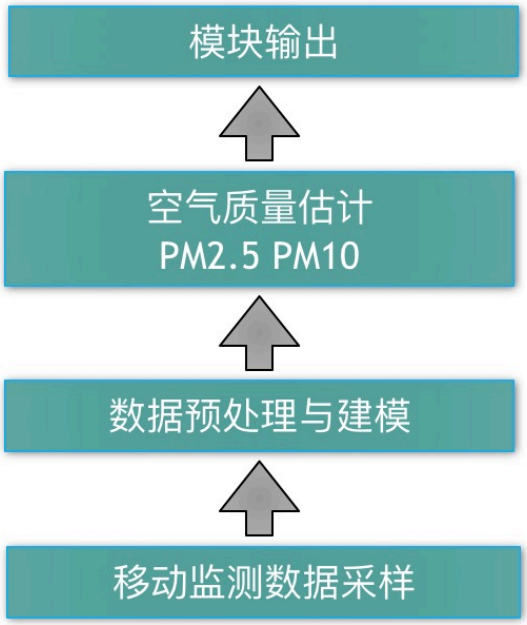

图 2.1　空气质量回归研究框架

2.1　数据集

2.1.1　天气数据

本文中使用的数据为安装在汽车上的移动采集数据和城区的气候数据。数据中记录了该城市约两个月内的十多辆汽车的移动采集数据。如图 2.2和 2.3是多辆车某一天内和某两小时内的覆盖城区范围。

我们可以看出来，移动车在一天时间内基本覆盖了整个北京城城区。相对而言，在两小时内，移动车则无法做到全面覆盖，在部分区域路段，会缺乏数据。

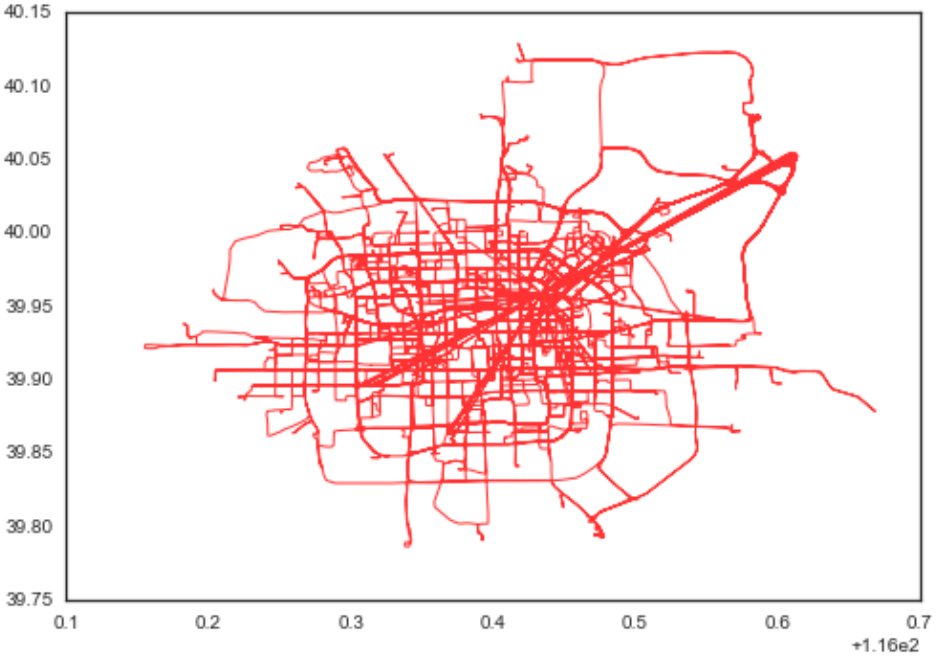

图 2.2　移动采样车某一天的覆盖采样范围

图 2.3　移动采样车某两小时的覆盖采样范围

接下来，我们对各部分数据进行介绍：

首先我们介绍气候数据，气候数据采集的是城区各个区域关于气候的数据，入气温、湿度等，表 2.1记录了气象数据库的各字段和说明：

字段名	说明	示例
FTIME	采集的时间（世界时间）	2015-06-03-12:00:00
FID	网格编号（范围 04355，66*66）:	901
U	风场在 X 轴（正东）方向的分量，单位 m/s	3.235
V	风场在 Y 轴（正北）方向的分量，单位 m/s	3.235
TEMPARATURE	温度，单位：摄氏度	25.421
RAIN	降水量，单位：毫米	0
PRESSURE	大气压，单位：百帕（hPa）	976.664
HUMIDITY	相对湿度，0 100%	24.072
FLOORS	垂直层高，近地面层	0
DISTS	网格区域，目前为北京市区域	D03

表 2.1　气象模型数据说明

我们抽取某一个小时的数据，并绘制 PM2.5 浓度和各个天气因素之间的相关关系图，如图 2.4 我们可以看到 PM2.5 的浓度和各个天气因素并不是强的线性相关关系。特别地，PM2.5 和 PM10 浓度有比较强的线性关系，真是因为一般而言，在稳定的采样条件下，PM10 都是包含 PM2.5 的。

为了更具体的了解各个因素之间的关系，我们绘制的更为详尽的相关矩阵，如图 2.5。对角线上的图反映的是横轴上特征和 PM2.5 分布的情况。其余的图为各个特征两两组合下，PM2.5 浓度分布的情况。

我们可以大致看出一些结论，例如高气温和低湿度的情况下，空气污染浓度会较低。而高气压和低温度的情况下，PM2.5 的浓度约会比较小。因此，我们可以知道天气因素在特定的情况下会影响空气质量，是一个可以利用的外部数据。

2.1.2　移动采样数据

这两类数据连同原有的固定基站数据构成了我们整个数据集。

2.2　数据预处理

原始数据由两个表组成，而且对应字段不一，需要我们进行接合的处理。另外，数据存在许多缺失、异常等情况，需要我们进行建模前的清洗。

对应气象数据而言，没有记录准确的经纬度，而是记录了其所处的方格位置，而这个方格位置的中心经纬度我们是可以推倒得到的。因此，气象数据记录的是 4356 个

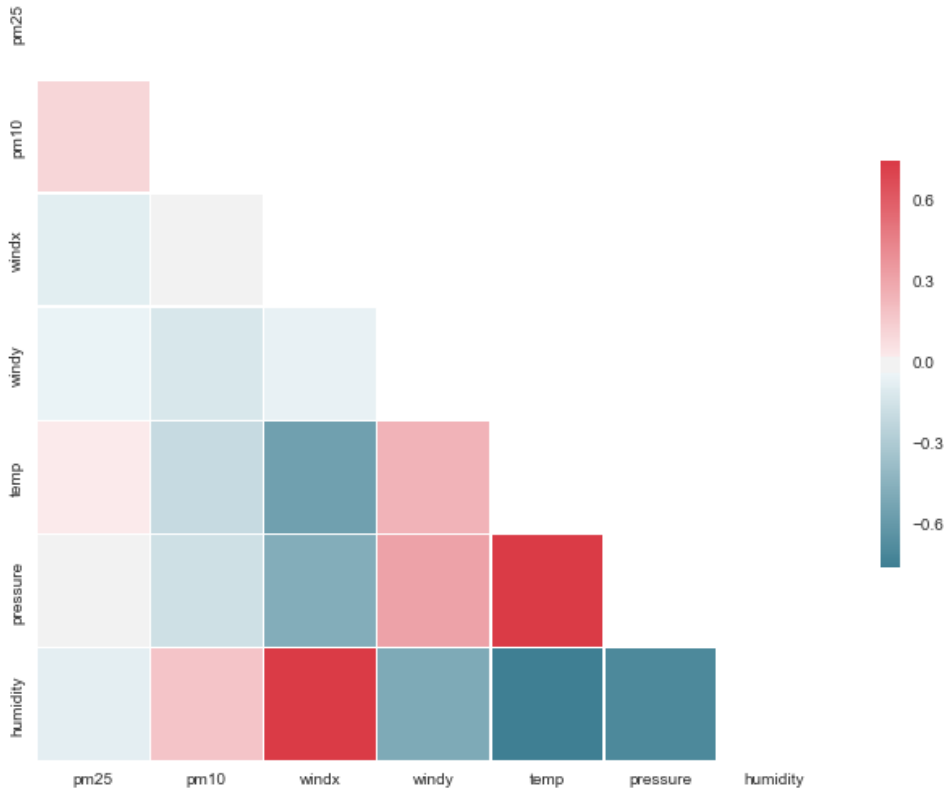

图 2.4　PM2.5 和各个天气因素相关关系图。图中 windx 表示东风；windy 表示北风；temp 表示气温；pressure 表示气压；humidity 表示湿度

字段名	说明	示例
ID	采集设备编号	001
longitude	经度（东经）	116.22662
latitude	纬度（北纬）	40.297644
height	设备采样高度，单位 m	1.5
speed	设备移动速度，单位：m/s	3.2
direction	设备移动方向，0 360	90
pm10	pm10 浓度, 单位：毫克/平方米	30
pm25	pm2.5 浓度，单位：毫克/平方米	24.072
sample_time	采样时间	2015-06-03-12:00:00

表 2.2　移动采样数据说明

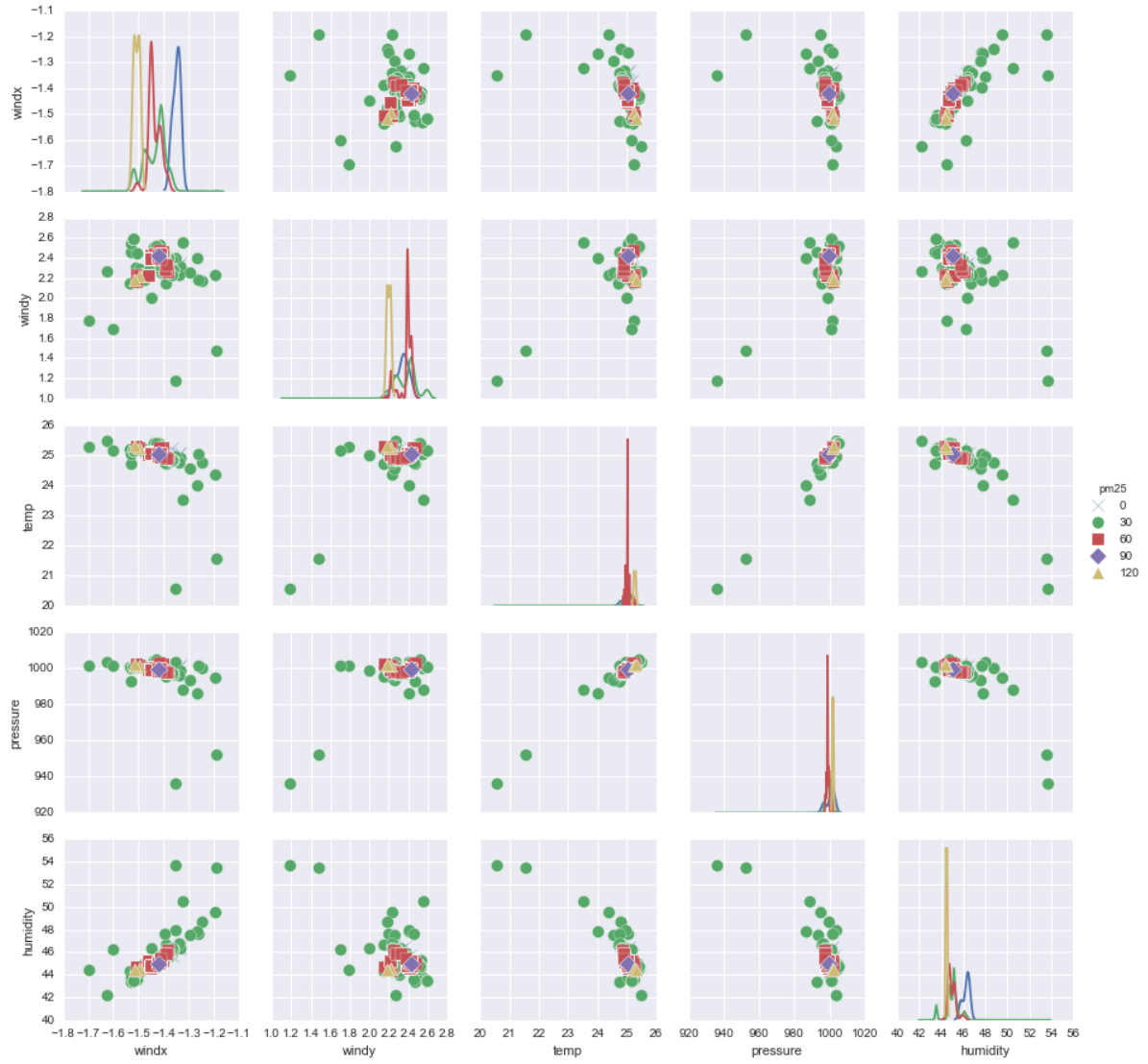

图 2.5　PM2.5 和各个天气因素相关关系矩阵。图中 windx 表示东风；windy 表示北风；temp 表示气温；pressure 表示气压；humidity 表示湿度

不同方格的天气状况。

而对于移动采集数据，我们有精确的经纬度坐标，这些坐标通通都落在天气数据的方格里面，因此，每个方格将会对应多个移动采集样本。我们需要将不同的移动采集数据按照方格所属经纬度范围划分，然后就可以求得该移动数据样本的气候状况了。

对于缺失值，由于数据量较大，我们直接舍弃掉含缺失值的样本。而对于异常值，如果该字段值超过正常范围，我们同样将该样本舍弃。

这样，我们将两张表合并，得到用于我们问题建模的表 2.3，如下：

字段名	说明	示例
longitude	经度（东经）	116.22662
latitude	纬度（北纬）	40.297644
pm10	pm10 浓度, 单位：毫克/平方米	30
pm25	pm2.5 浓度，单位：毫克/平方米	24.072
sample_time	采样时间	2015-06-03-12:00:00
U	风场在 X 轴（正东）方向的分量，单位 m/s	3.235
V	风场在 Y 轴（正北）方向的分量，单位 m/s	3.235
TEMPARATURE	温度，单位：摄氏度	25.421
RAIN	降水量，单位：毫米	0
PRESSURE	大气压，单位：百帕（hPa）	976.664
HUMIDITY	相对湿度，0 100%	24.072

表 2.3　移动采样数据说明

数据从设备、基站等地采集回来后，可以分为三类特征，即气象特征、空间特征和时间特征。有如下关系：

$$x_i = (x_i^{weather}, x_i^{spatio}, x_i^{time})$$

分别刻画了该样本的气候特征、空间特征 (经纬度) 和时间特征。

对于时间窗口 $(t-h, t]$, 我们可以构造一个该时间窗口内的训练样本 $(\mathbf{X}_h(t), \mathbf{y}_h(t)) = \{(\mathbf{x_i}, y_i) \mid x_i^{time} \in (t-h, t], i = 1, ..., n\}$

从图 2.6中我们可以看到，我们将不同来源的数据统一描述为不同时间下的采样数据。这些采样数据共同构成了数据流。通过对窗口大小的选择，我们可以构造出当前时刻 t 所对应的训练样本。但为了控制窗口内采样数据的规模，本文选取 h 大小，使得窗口内训练样本数目等于给定值 m，即

$$h = \arg\min_h \{\mathrm{card}(\mathbf{X}_h(t)) = m\}$$

也就是相当于在数据流中，采样记录是“进一个出一个的”。

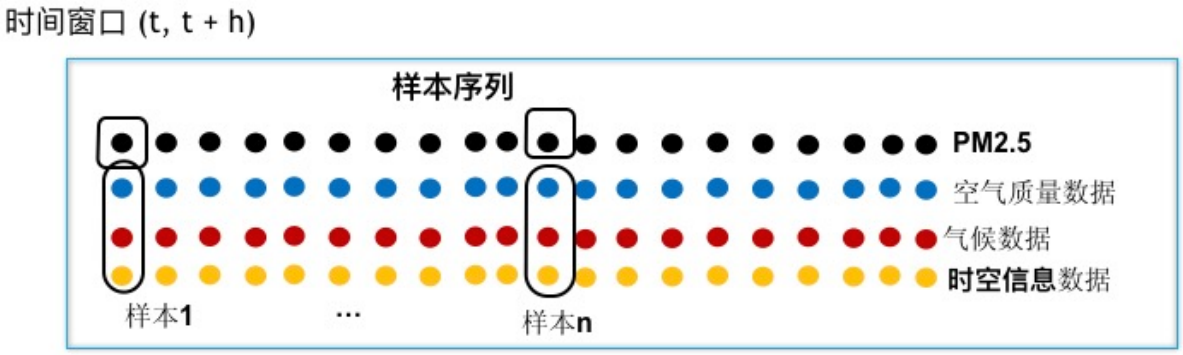

图 2.6　时间窗口下的样本流数据

2.3　问题形式化定义

问题按照如下形式化为数学语言。对于给定训练数据集 $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x_i}, y_i)|i = 1, ..., n\}$。在这里，$y_i$ 为我们的输出目标，代表了第 i 个样本的 $\mathrm{PM}_{2.5}$ 的大小，而 $\mathbf{x_i}$ 则表示第 i 个样本的特征向量。这里的特征向量包括三个部分，有：

$$x_i = (x_i^{weather}, x_i^{spatio}, x_i^{time})$$

分别刻画了该样本的气候特征、空间特征 (经纬度) 和时间特征。

对于给定时间窗口 $(t-h, t]$, 我们可以构造一个该时间窗口内的训练样本 $(\mathbf{X}_h(t), \mathbf{y}_h(t)) = \{(\mathbf{x_i}, y_i) \mid x_i^{time} \in (t-h, t], i = 1, ..., n\}$

那么我们的问题转化为, 已知一个时间窗口的训练数据集 $(\mathbf{X}_h(t), \mathbf{y}_h(t))$, 现在给一个新的测试样本 $\mathbf{X}_*$ (应该满足 $X_*^{time} == t$, 要我们求出它对应的输出 $\mathbf{f}^*$ 满足的概率分布:

$$\mathbf{f}^*|\mathbf{X}_*, \mathbf{X}_h(t), \mathbf{y}_h(t) \sim \mathcal{N}(\bar{\mathbf{f}}_*, \mathrm{cov}(\mathbf{f}^*))$$

2.4　高斯过程

高斯过程 (Gaussian Process, **GP**) 是定义在函数上的一种分布 (distribution) [23]。它用一组 (可能无限多个) 随机变量 (random variables) 来描述。这些随机变量本身都满足一个正态分布 (normal distribution)，并且这些随机变量任意组合的分布都满足多元高斯分布 (multivariate normal distribution)。

我们可以通过给定高斯过程的均值函数 (mean function) $m(\mathbf{x})$ 和协方差函数 (co-

variance function) $k(\mathbf{x}, \mathbf{x}')$ 来唯一确定一个高斯过程，即：

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})] \\ k(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] \end{aligned}$$

我们一般可以吧高斯过程写作如下形式：

$$f(\mathbf{x}) \sim \mathbf{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$$

为了简化需要，我们可以直接取均值函数未 **0**-函数，即 $m(\mathbf{x}) \triangleq \mathbf{0}$，这并不会影响建模的结果。

高斯过程回归

高斯过程回归是利用了高斯过程的特点，用来进行机器学习当中的回归任务。假设我们有如下的训练样本集合：

$$(\mathbf{X}, \mathbf{f}) = \{(\mathbf{x_i}, f_i) | i = 1, ..., n\}$$

那么给定测试样本 $\mathbf{X}^*$ 以及其对应输出 $\mathbf{f}^*$，根据高斯过程的性质，它们的联合分布应该满足：

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}\ , \quad \begin{bmatrix} K(\mathbf{X}, \mathbf{X}) & K(\mathbf{X}, \mathbf{X}^*) \\ K(\mathbf{X}^*, \mathbf{X}) & K(\mathbf{X}^*, \mathbf{X}^*) \end{bmatrix}\right)$$

其中 $K(X, X^*)$ 表示样本间的协方差矩阵。那么，通过限定训练样本的取值，我们可以给出测试样本输出 $\mathbf{f}^*$ 的高斯分布：

$$\begin{aligned} \mathbf{f}^* | \mathbf{X}^*, \mathbf{X}, \mathbf{f} \ \sim\ & \mathcal{N}(K(\mathbf{X}^*, \mathbf{X})K(\mathbf{X}, \mathbf{X})^{-1}\mathbf{f}, \\ & K(\mathbf{X}^*, \mathbf{X}^*) - K(\mathbf{X}^*, \mathbf{X})K(\mathbf{X}, \mathbf{X})^{-1}K(\mathbf{X}, \mathbf{X}^*)) \end{aligned}$$

但是现实中，一般的样本输出都是带观察噪声的，我们一般假设：

$$y = f(\mathbf{x}) + \epsilon$$

其中，$\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma_n^2)$, 满足一个均值为 $\mathbf{0}$ 的高斯分布。根据高斯过程的定义，我们有：

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}\ , \quad \begin{bmatrix} K(\mathbf{X},\mathbf{X}) + \sigma_n^2\mathbf{I} & K(\mathbf{X},\mathbf{X}^*) \\ K(\mathbf{X}^*,\mathbf{X}) & K(\mathbf{X}^*,\mathbf{X}^*) \end{bmatrix} \right)$$

同理，我们给出测试样本输出 $\mathbf{f}^*$ 的分布：

$$\mathbf{f}^*|\mathbf{X}^*, \mathbf{X}, \mathbf{y} \sim \mathcal{N}(\bar{\mathbf{f}}^*, \mathrm{cov}(\mathbf{f}^*))$$

其中

$$\begin{aligned} \bar{\mathbf{f}}^* &\triangleq \mathbb{E}[\mathbf{f}^*|\mathbf{X}^*, \mathbf{X}, \mathbf{y}] = K(\mathbf{X}^*,\mathbf{X})[K(\mathbf{X},\mathbf{X} + \sigma_n^2\mathbf{I}]^{-1}\mathbf{y} \\ \mathrm{cov}(\mathbf{f}^*) &= K(\mathbf{X}^*,\mathbf{X}^*) - K(\mathbf{X}^*,\mathbf{X})[K(\mathbf{X},\mathbf{X}) + \sigma_n^2\mathbf{I}]^{-1}K(\mathbf{X},\mathbf{X}^*) \end{aligned}$$

这样，我就就通过已有的样本来回归估计出测试样本的输出。如图 2.7是一个在二维平面上回归的例子。

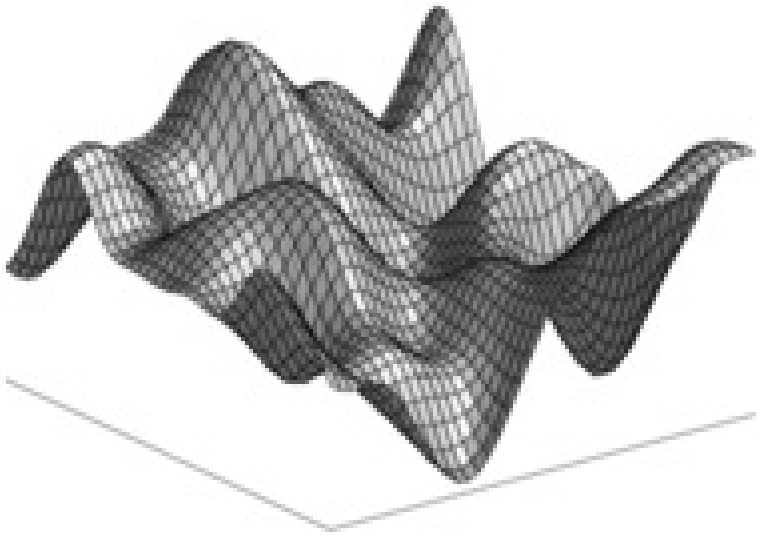

图 2.7　高斯过程回归示例

2.5 高斯回归在时空维度的延拓

一般的空间插值问题，都会有一个基本假设，两个样本的空间距离越近，则应该有相似的输出。空气质量估计问题不同于一般的回归问题，每个样本除了空间属性，还带有了许多气候特征和采样时间。基于空气污染物在大气中的时空连续地扩散这一认知，我们可以加强这一空间插值的基本假设。也就是说，不仅是空间相近的样本有相似的输出，在时间上相近，在气候条件相似的样本上，它们也应该有相似的污染物浓度。

高斯过程是一个非参过程，只能通过调整超参数来改变模型。高斯回归过程允许建模的时候，通过设计核函数来反映我们对问题的先验知识。核函数的大小反映了样本在映射后空间的距离远近，也就是样本见取值的协方差大小。因此，我们直接给出空气质量估计问题场景下的核函数：

$$\begin{aligned} k_f(\mathbf{x},\mathbf{x}' &= \sigma_w^2 \exp\{-\frac{1}{2} D_{weather}^T M D_{weather}\} + \\ & \sigma_s^2 \exp\{-\frac{||x^{spatio} - x'^{spatio}||^2}{l_s}\} + \sigma_t^2 \exp\{-\frac{||x^{time} - x'^{time}||^2}{l_t}\} \end{aligned}$$

其中，$D_{weather} = x^{weather} - x'^{weather}$，$M = \mathrm{diag}(\mathbf{l_w}^{-2})$

考虑到我们观察有高斯噪声，我们将最终的核函数写成：

$$K(X, X') = K_f(X, X') + \sigma_n^2 \mathbf{I}$$

而高斯过程的另一个特征函数，均值函数我们取为 0 函数，这不会影响我们的回归结果。

这样，我们就定义了一个空气质量估计场景下的高斯过程回归模型。这个模型涉及了许多的超参数:

$$\theta = (\sigma_s^2, \sigma_t^2, \sigma_w^2, l_s, l_t, l_w, \sigma_n^2)$$

我们观察可以发现，这个核函数有三个不同的核函数构成，虽然它们具体的形式不一样，都可以归类为平方指数 (squared exponential) 核函数，它的一般形式为：

$$\exp(-\frac{\mathbf{r}^2}{2\mathbf{l}^2})$$

其中 r 为两个特征向量的距离 $||\mathbf{x} - \mathbf{x}'||$，$\mathbf{l}$ 是尺度 (scale) 参数。我们可以画出他的函数示意图，如图 2.8。其中横轴为两个特征向量的距离 r，纵轴为 $f(x)$ 和 $f(x')$ 的协方差。

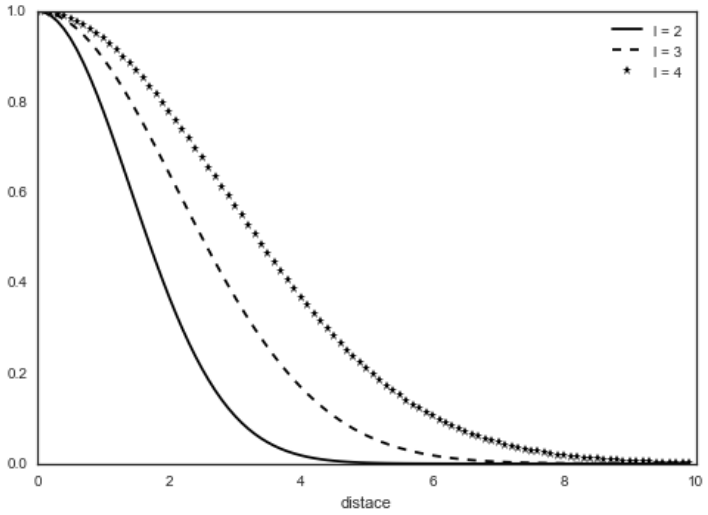

图 2.8　平方指数核函数的函数图像

我们可以观察到随着两个样本的距离 r 越小，它们对应的在核空间的函数值协方差越大，表明对应的函数值越接近。而尺度参数在这里的作用是使适应不同的特征向量的尺度，当它越大的时候，对特征向量的距离越不敏感，变化越慢。

2.6　模型的训练

在本文涉及的移动数据和气候数据是随着时间变化的，也就是说随着时间窗口的滑动，训练样本是变化了。但是考虑到这些数据是同质的，仅需要学习一次模型的超参数即可。通过预设一份训练样本用于训练超参数，之后就不需要再重新学习。一般而言，我们会设置用于训练超参数的样本数目为两倍于正常时间窗口下样本的两倍，保证模型的训练效果。

由于不同的超参数设置，会有不同的模型表现。这里，我们需要一种学习的方法来寻找一个最优的超参数设置。通过给出的核函数，我们可以推倒出模型的似然表达式（推倒过程省略）：

$$\mathbf{p}(y|X,\theta) = -\frac{1}{2}y^T K^{-1} y - \frac{1}{2}\log|K| - \frac{n}{2}\log 2\pi$$

通过对模型的超参数求偏导，我们可以得到如下公式：

$$\begin{aligned}\frac{\partial}{\partial\theta_j}\log\mathbf{p}(y|X,\theta) &= \frac{1}{2}y^TK^{-1}\frac{\partial K}{\partial\theta_j}K^{-1}y-\frac{1}{2}\mathrm{tr}(K^{-1}\frac{\partial K}{\partial\theta_j})\\ &= \frac{1}{2}\mathrm{tr}((\alpha\alpha^T-K^{-1})\frac{\partial K}{\partial\theta_j})\end{aligned}$$

其中 $\alpha=K^{-1}y$

为了得到具体的梯度方向，我们将 $\frac{\partial K}{\partial\theta_j}$ 的表达式分别取出来，罗列如下：

$$\begin{aligned}\frac{\partial K}{\partial\sigma_s} &= 2\sigma_s\exp\{-\frac{||x^{spatio}-x'^{spatio}||^2}{l_s}\}\\ \frac{\partial K}{\partial\sigma_t} &= 2\sigma_t\exp\{-\frac{||x^{time}-x'^{time}||^2}{l_t}\}\\ \frac{\partial K}{\partial\sigma_w} &= 2\sigma_w\exp\{-\frac{1}{2}D^T_{weather}MD_{weather}\}\\ \frac{\partial K}{\partial\sigma_s} &= \sigma_s^2l_s^{-2}\exp\{-\frac{||x^{spatio}-x'^{spatio}||^2}{l_s}\}||x^{spatio}-x'^{spatio}||\\ \frac{\partial K}{\partial\sigma_t} &= \sigma_t^2l_t^{-2}\exp\{-\frac{||x^{time}-x'^{time}||^2}{l_t}\}||x^{time}-x'^{time}||\\ \frac{\partial K}{\partial\sigma_{w,i}} &= \sigma_w^2l_{w,i}^{-3}\exp\{-\frac{1}{2}D^T_{weather}MD_{weather}\}D^T_{weather}D_{weather}\\ \frac{\partial K}{\partial\sigma_n} &= 2\sigma_n\mathbf{I}\end{aligned}$$

通过梯度下降的方法，我们不断迭代，直至整个似然表达式前后之差收敛，我们就完成了对模型的训练。其中梯度下降表达式为：

$$\theta_j^{(i+1)}=\theta_j^{(i)}-\eta\frac{\partial}{\partial\theta_j}\log\mathbf{p}(y|X,\theta)$$

其中 η 为学习率，控制梯度下降的步长。

对于训练而言，我们由它的超参数偏导公式可以看到，整个偏导当中主要的计算量在于计算协方差矩阵 K 的逆，根据一般的求逆过程，其复杂度为 $\mathrm{O}(n^3)$，其中 n 为训练样本的数目。因此模型的效率将会非常差，我们将会在第三张介绍如何去加速训练过程。

2.7　模型估计

2.7.1　静态估计

完成了模型的学习，给定一个新的测试样本 X_*（或者查询），我们根据高斯过程的特性，有：

$$\begin{bmatrix}\mathbf{y}\\ \mathbf{f}^*\end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}\ ,\ \begin{bmatrix}K_f(\mathbf{X},\mathbf{X})+\sigma_n^2\mathbf{I} & K_f(\mathbf{X},\mathbf{X}^*)\\ K_f(\mathbf{X}^*,\mathbf{X}) & K_f(\mathbf{X}^*,\mathbf{X}^*)\end{bmatrix}\right)$$

因此，我们就可以得到我们最终的空气污染浓度的期望值和方差值：

$$\begin{aligned}\bar{\mathbf{f}}^* &= K_f(\mathbf{X}^*,\mathbf{X})[K_f(\mathbf{X},\mathbf{X})+\sigma_n^2\mathbf{I}]^{-1}\mathbf{y}\\ \mathrm{cov}(\mathbf{f}^*) &= K_f(\mathbf{X}^*,\mathbf{X}^*)-K_f(\mathbf{X}^*,\mathbf{X})[K_f(\mathbf{X},\mathbf{X})+\sigma_n^2\mathbf{I}]^{-1}K_f(\mathbf{X},\mathbf{X}^*)\end{aligned}$$

我们给出空气质量估计的具体算法 1

Algorithm 1 Input : $\mathbf{X}$(inputs), $\mathbf{y}$(targets), k_f(covariance function), σ_n^2(noise level), $\mathbf{x}_*(testinput)$

1: $L \leftarrow \mathrm{cholesky}(K+\sigma_n^2\mathbf{I})$
2: $\alpha \leftarrow L^{\mathrm{T}} \setminus (L \setminus \mathbf{y})$
3: $\bar{f}_* \leftarrow \mathbf{k}_*^{\mathrm{T}}\alpha$
4: $\mathbf{v} \leftarrow L \setminus \mathbf{k}_*$
5: $\mathbb{V}[f_*] \leftarrow k(\mathbf{x}_*,\mathbf{x}_*)-\mathbf{v}^{\mathrm{T}}\mathbf{v}$
6: **RETURN** : $\bar{f}_*$ (mean) , $\mathbb{V}[f_*]$ (variance)

模型估计的主要计算量也在于对求协方差矩阵 K 的逆上,但是我们注意到 $K_f(\mathbf{X}^*,\mathbf{X})$ 和 $[K_f(\mathbf{X},\mathbf{X})+\sigma_n^2\mathbf{I}]^{-1}$ 其实在训练过程中已经求出来了，我们可以预先保留他们的计算结果和相乘结果，并不需要再次计算。因此估计的时间复杂度为 $\mathrm{O}(n)$

2.7.2　动态估计

随着时间窗口的向前滑动，训练样本是在不断发生变化的。同时，它们对应的协方差矩阵 K 也是在不断变化的。对于静态估计中，直接复用 $[K_f(\mathbf{X},\mathbf{X})+\sigma_n^2\mathbf{I}]$ 的计算结果变得不可行。假如重新计算的话，时间复杂度又高达 $\mathrm{O}(n^3)$。下面介绍一种近似技术，去动态更新协方差矩阵。

因为时间窗口内的样本数是一定的，那么更新前后的协方差矩阵就可以表述为：

$$K_f' = K_f + \Delta K_f$$

对于只有一个样本改变的情况，更新的复杂度为 O(n)

利用矩阵逆的二阶近似，我们可以近似求得新时间窗口下的协方差矩阵的逆：

$$\begin{aligned} K'^{-1} &= (\sigma_n^2\mathbf{I} + K_f + \Delta K_f)^{-1} \\ &\approx K^{-1} - (\Delta K_f - \delta K \cdot K_f - K_f \cdot \Delta K - \Delta K_f \cdot \Delta K_f) \end{aligned}$$

由于 ΔK_f 非常稀疏，有许多零元素。那么这个更新公式可以非常高效地更新，时间复杂度为 O(n)。

2.8 模型复杂度分析

按照上文描述，我们模型的训练复杂度为 O(n^3)，但是只要模型启动的时候学习一次即可。而我们模型的估计的复杂度为 O(n) 而动态更新协方差矩阵相关结果的时间复杂度为 O(n)。因此整个估计过程的时间复杂度为 O(n)

可以观察到，模型的训练复杂度非常高，需要我们进一步优化。在下一章将会介绍优化的方法。

2.9 城市空气质量估计

我们将市区划分大小均等的网格，利用两个小时内的采样数据作为输入，求得每一个网格中心点的 PM2.5，绘制成图 2.9和图 2.10：

在图 2.9中，平面坐标系为经纬坐标系，z 轴为 PM2.5 的浓度。红色部分为移动车采样的路径。我们可以观察到，大部分区域污染浓度都比较低，只有局部地区污染浓度突出。对于采样数据密集的部分，显然会有更好的效果。而对于采样数据稀疏的边缘部分，回归效果则不尽人意。

图 2.10是图 2.9在经纬坐标平面的投影。颜色越靠近黄色，污染浓度越高。从这张图我们同样可以得到类似结论，采样数据越密集的地区，会有更精细的回归效果。而对于采样稀疏的地区，由于提供回归的训练样本少，贡献的权重小，导致没有办法得到确信的结果。

2.10 本章小结

本章分为了若干部分，首先我们介绍了数据的基本情况；其次我们介绍了如何对数据进行预处理；然后，我们基于高斯过程的扩展对我们的问题进行建模。并给出了模型的训练方法。最后我们给出了如何预测空气质量的方法。

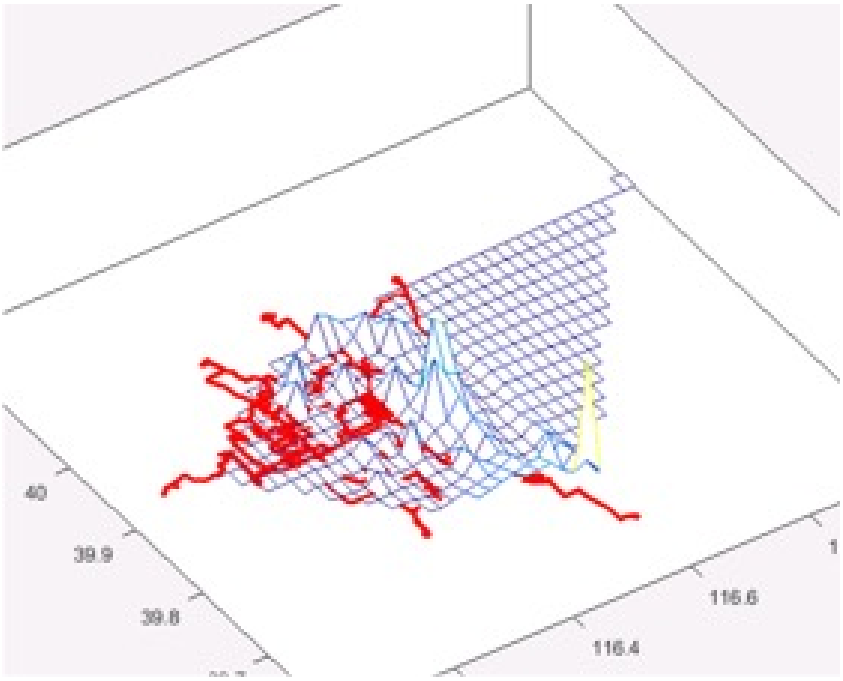

图 2.9　城市空气质量估计 3-d 示意图

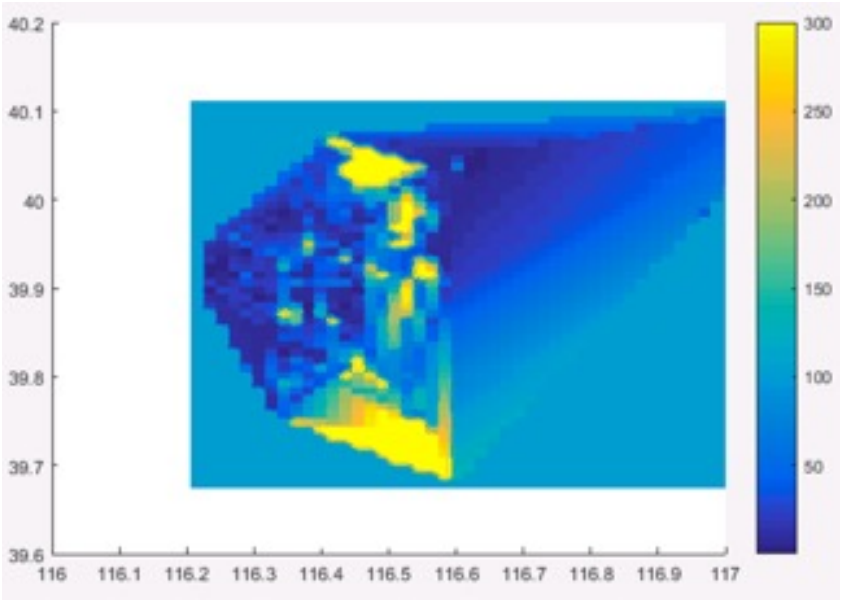

图 2.10　城市空气质量估计平面热力图

第三章　基于 k-d 树的模型训练和估计加速

在上一章，模型的训练复杂度高达到 $O(n^3)$，而在动态数据时间窗下估计的复杂度也高达 $O(n)$。一旦在采样数据增多时，模型的训练时间就和估计时间会和空气质量估计的实时性发生冲突。这就需要我们根据问题的特点提出优化方法。一个非常直观的想法，就是减少实际用于训练和估计模型的样本数目，因此我们将会引入 k-d 树来达成这个目标。

在本章，将会首先介绍 k-d 树的相关知识；其次，介绍如何和模型相结合，通过空间划分的方式选取训练样本中的有代表性的样本，简化计算中涉及的冗余样本。最后，具体阐述如何利用上面的结果加速训练模型和加速估计。

3.1　k-d 树

k-d 树（是 k-dimensional 树的缩写）是一颗特殊的二叉树，是一种对 k 维空间进行划分管理的数据结构，k-d 树通过对数据进行空间划分，可以高效地进行于范围查询和最近邻查询等操作。

k-d 树的构建

k-d 树的结点 (node) 包含了所有的 k 维点 (k-dimentinal point)，构建思路是通过每一次选择一个超平面（hyperplane)，将一个非叶子结点 (non-leaf node) 划分成两半，放入左右两个儿子节点当中。重复这一过程，对其儿子节点进行划分，直至满足一定条件为止。

超平面的选取有很多种方法，最常见的是选取某一维度的中位 (median) 取值作为划分依据，这样确保了左右两颗子树 (subtree) 的规模是平衡的 (balanced)。使得最终的 k-d 树是一棵平衡树 (balanced tree)。

k-d 树的增删操作

和其他搜索树 (search tree) 类似，k-d 树的增删操作如下所述：

- **增加 (Adding)**：从树的根部开始遍历 (traverse)，根据结点的划分依据，选择进入结点的左子树或右子树。重复这一过程，直到遍历到叶子结点为止。增加结点会导致 k-d 树变得不平衡 (unbalanced)，一般当增加一定量结点之后，需要重新对树进行再平衡 (re-balanced) 操作。

- **删除 (Removing)**: 查找到要删除元素所处的子树，删除该元素，并重新构建这棵子树成为一棵新的平衡子树。同理，删除操作会导致整个 k-d 树无法保持平衡，当 k-d 树查询性能变差时，需要再平衡。

k-d 树的复杂度

通过“中位数的中位数算法”(median of medians algorithm), k-d 树的构造复杂度可以为 O($n$log$n$), 而相应的，k-d 树的增删操作复杂度为 O(n)。如图 3.1是一个利用 k-d 树对二维平面进行划分的例子。

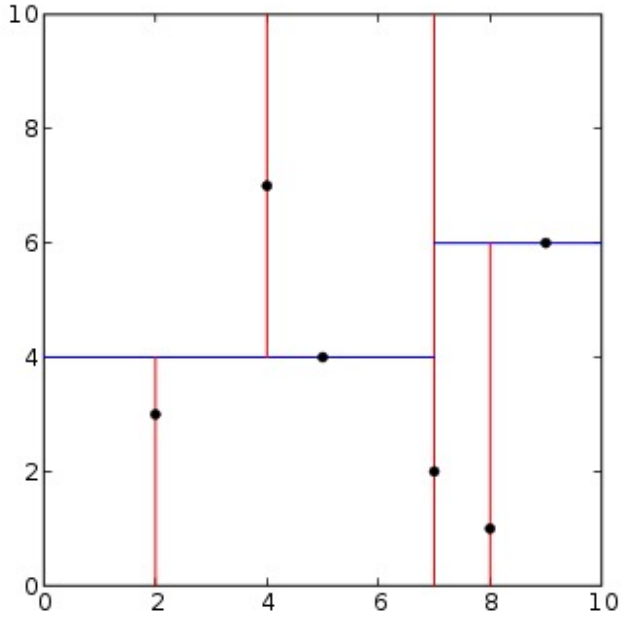

图 3.1　k-d 树空间划分示例

3.2　问题的特点与分析

基于移动采样数据的空气质量估计场景下，空气质量估计的空间和时间局部性相对比较强，空间距离较远的样本往往贡献较小，同一时间窗口下的时间较远的两个样本关联性比较小。这是有污染物传播的机理可以造成的，空气污染物在空间连续分布，受各种因素影响其值的大小。因此，相对于气候条件相似而言，空间和时间的相似性来得更重要。

在模型的训练过程中，需要计算两两样本间的距离，其协方差矩阵 $K(X,X)$ 是一个大小为 $n \times n$ 的矩阵。我们需要对其求逆的话，复杂度为 $\mathrm{O}(n^3)$。但是基于我们上述的对问题的认识，对于空间距离较远的样本，由于贡献很小，因此可以通过用归约样本的方式，来减小协方差矩阵的大小。

我们通过引入 k-d 树来对空间进行划分，这里注意仅仅对样本中的 x_{spatio} 维度进行划分，并通过限制叶子结点内包含样本的最小数目。思路主要是着重考虑同一个结点内的样本间的计算，对于结点和结点之间的距离计算，我们将简化为它们代表结点的距离计算。

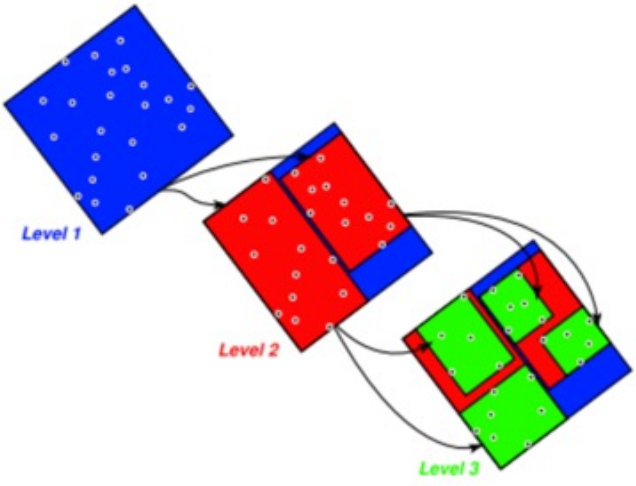

图 3.2　利用 k-d 树对样本空间进行层次划分

3.3　空间的层次划分

在这一节将具体阐述如何根据我们上面的设定进行空间的层次划分。为了避免叶子结点只有一个样本，我们设置叶子结点的最小样本数（如最少 20 个样本）。根据一般的 k-d 树构建方法，我们对样本所处的地理空间进行划分。图 3.2是空间中样本按照层次划分的一个示意图。通过这种划分方式，得到了一个多层结构的训练集。对于每一个结点而言，都包含了所有属于该结点的所有样本。

我们记所有结点的中心位置为该结点的代表样本 c_i，是一个虚拟样本。它的取值却取决于同结点内所有样本的均值，即：

$$\mathbf{c_i} = \frac{\mathrm{sum}_{x \in TreeNode(i)}\{x\}}{|TreeNode(i)|}$$

其中 $TreeNode(i)$ 表示树中第 i 个结点的样本集合。

除了正常的 k-d 树构造外，在构造过程中，我们还需要计算样本间的距离，并分两类情况：

- 对于非叶子结点：计算其子结点的代表结点的距离 $r = ||\mathbf{c} - \mathbf{c}'||$
- 对于叶子结点：计算叶子结点内部样本两两的距离 $cr = ||\mathbf{x} - \mathbf{x}'||$

通过以上流程，就构造出一棵应用于空气质量回归的 k-d 树。

3.4 层次样本空间的维护

无论是插入还是删除,我们先根据样本的位置 $s = x_{spatio}$ 找到其所处位置 $TreeNode(i, j)$，在增加或删除后，首先按照 k-d 树规则更新树结构，然后对于以下两种情况分别操作：

- 在插入（删除）到非叶子节点过程中，重新计算子结点代表样本间的距离 $r = ||\mathbf{c} - \mathbf{c}'||$
- 在插入 (删除) 到叶子节点时，更新它和叶子节点内其他样本的距离 $r = ||\mathbf{x} - \mathbf{x}'||$

通过维护叶子和非叶子结点当中的距离信息，我们就可以随着时间窗口的滑动，在大量样本重复的时候，不需要进行冗余计算。

3.5 模型的加速

通过前面对如何构建一棵应用于空气质量回归的 k-d 树后，接下来将具体阐述如何通过这一数据结构加速模型的训练和估计。首先我们先定义一些术语，对于每一个空间位置 s 而言，通过确认它所处的结点，都可以给出它两部分的关联样本：

- 内部关联样本：同处于一个叶子节点的样本 x, 其规模为 $\mathrm{O}(L)$
- 外部关联样本：其余叶子结点的代表样本 c，其规模近似为 $\mathrm{Olog}(n/L)$

其中 L 为叶子结点样本个数。

接下来我们重新定义了样本间两两距离的距离矩阵:

$$r_{i,j} = \begin{cases} ||x_i - x_j||, \text{if } x_i, x_j \text{ in a same leaf node} \\ ||c_{2t+1} - c_{2t}||, TreeNode(t) \text{ is the ancestor of } TreeNode(i), TreeNode(j) \end{cases}$$

相比于原始的距离矩阵，对于在叶子节点外部的值，都用代表样本的距离代替了，这样为协方差矩阵和后续的计算带来便利。相应地，我们也可以重新计算出协方差矩阵 $K_{induction}(\mathbf{r})$。示意如图 3.3所示，绿色部分代表叶子结点内样本两两间的距离，交叉部分是代表样本的距离，其余空白部分都用相应的代表样本距离代替。

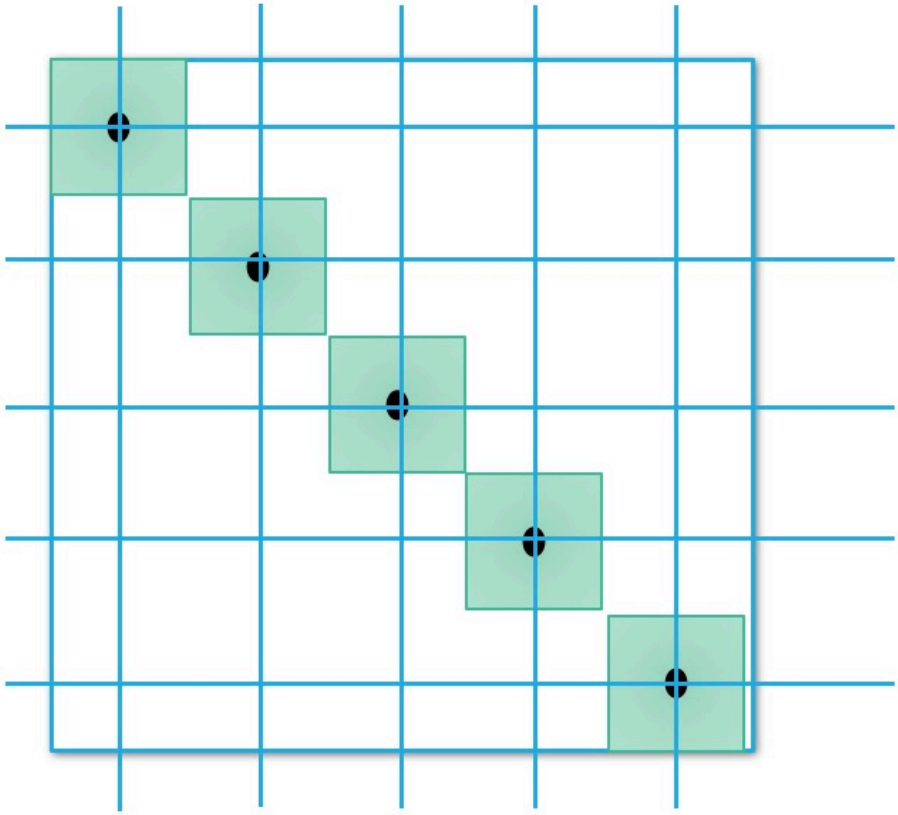

图 3.3　协方差矩阵的归约示意图

这样对于所有的空间位置 s 和一个给定的训练集合 $(\mathbf{X}, \mathbf{y})$，可以给出和它相关，用于训练和估计计算的样本集合：

$$X_s = \{(\mathbf{x}, y) \mid (\mathbf{x}, y) \in TreeNode(t)\} \bigcup \{(\mathbf{c_i}, y_{c_i}) \mid i \neq t\}$$

其中 t 表示 s 所处的叶子结点编号。

3.5.1　训练的加速

通过上面对协方差矩阵的归约，我们只要维护了 k-d 树，从而就维护了距离矩阵、协方差矩阵和它的逆。但是可以观察到此时协方差矩阵的规模仍然是 $n \times n$，接下来将利用多次采样的方式来加速训练参数。我们选取一组位置 $s_1, s_2, ..., s_p$，使得它们分别落在各个叶子结点所处的平面空间里。这样它们对应的相关样本集合为 $X_{s_1}, X_{s_2}, ..., X_{s_p}$。相比直接利用全样本信息去训练模型，我们通过利用归约后的样本集重复去训练超参数。

通过这种方式,每次训练的样本规模为 $O(L+\log(n/L))$, 叶子结点的个数为 $O(n/L)$, 这样模型的我们训练时间复杂度降为 $O((L+\log(n/L))^3 n$。在 $L << n$ 的情况下，这个复杂度是远小于 $O(n)$ 的。

3.5.2 模型估计的加速

给定一个测试样本 x，我们根据它的位置 s 查询出它所在的叶子结点，然后选出和它相关的样本 X_s，从而减少矩阵的规模，打到加快计算的摸底。如图 3.4是一个估计过程的示意图，具体如下：如测试样本落在叶子结点 C1 内，我们首先计算它和 C1 内样本间的距离。然后，我们考虑它和外部关联样本的关系。它和 C2 中的样本的距离为 C1 的代表样本和 C2 代表样本的距离，存储于 B1 中。类似的，和 C3,C4 的距离则为 B1 和 B2 代表样本的距离，存储于 A1 之中。这样我们将协方差矩阵的规模从 $n \times n$ 近似

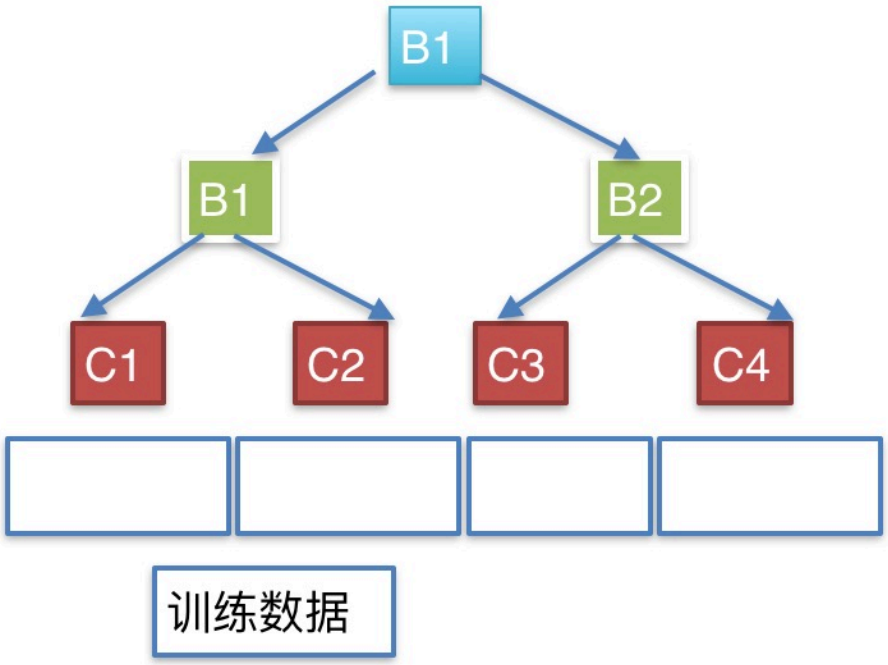

图 3.4 基于 k-d 树的空气质量查询示意图

减少为 $(L+\log(n/L)) \times (L+\log(n/L))$。估计的时间复杂度也变为 $O(L+\log(n/L))^3$, 在 $L << n$ 的情况下，这个复杂度是远小于 $O(n)$ 的

3.6　本章小结

因为随着样本数量增加，模型的复杂度非常高。我们给出了一种空间划分方法，对样本所处空间进行层次划分。这样对于我们要估计的位置，我们只需要保留和它相关的两部分样本。一方面保留了我们要估计区域附近的局部相关样本。另一方面，对于较远的样本，我们挑选代表性的留下。通过这个办法，我们是模型的训练和估计时间复杂度均有了下降，从而实现了动态时间窗口场景下的高效城市空气质量估计。

第四章　实验结果和分析

在本章，我们将会从多个角度去验证算法模型的有效性和可行性，并在真实的数据集上运行实验。我们分别介绍以下几项内容：其一，数据集的具体情况，如何利用数据集进行试验；其二，介绍实验的具体设置，如平台、对比算法等设置；其三，从多个角度进行试验，分析实验结果。

4.1　实验数据集

4.1.1　实验数据描述

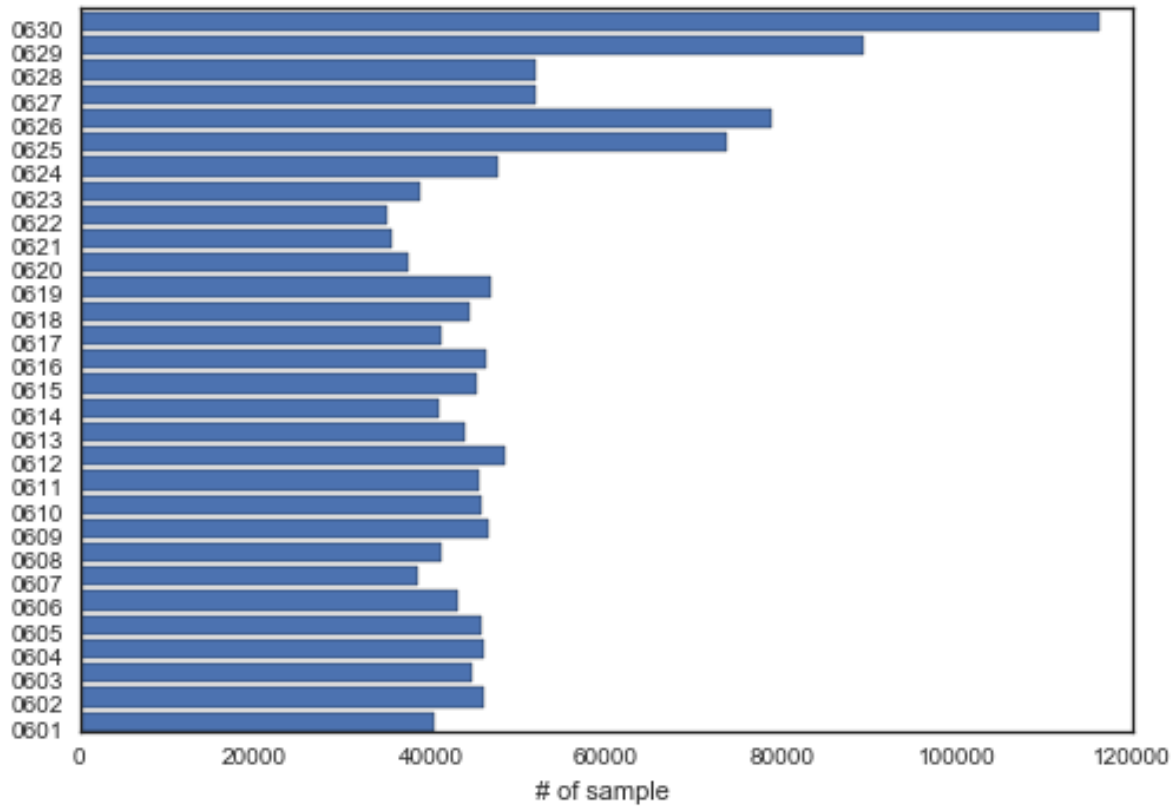

图 4.1　6 月份移动车采样情况

我们的数据是从 2015 年 6 月 1 日到 9 月 29 日，共 121 天的数据。在这期间有 51 台不同的采样设备在采样（它们可能不同时在路上行驶)。如图 4.1，描述了 6 月份每一

天移动车采样记录的总数。其中纵轴为 6 月的日期，其格式形如“06xx”，表示 6 月的第“xx”天；横轴为当天的采样总量。我们可以看到大多数时候采样的数目是相对稳定的，如果出现数目大量增加的日子，如月底，则是因为当天投入的采样设备增多的原因。

我们除了要了解每一天的采样总量情况，同时也要了解一天里不同时段的采样情况。我们将将近 4 个月的数据聚合后，将他们在一天 24 小时分布的情况描绘出来，如图 4.2。其中横轴为时间戳，每一个单位表示 30s 时间，共 2880 个时间戳，用来表示一天 24 小时；而纵轴则为对应时间戳内的采样总量。我们可以发现，移动车采样的情况和人类的作息时间是一致的，在白天最活跃，在中午左右的两三个小时，采样量达到最大。而到了夜间，随着投入设备减少，采样数目也急剧下降，知道日出未知。24 小时采样量的状况和人类活动一致的原因主要是因为我们的装置是安装在私人汽车上的，我们是被动地跟着他们的轨迹去采样。

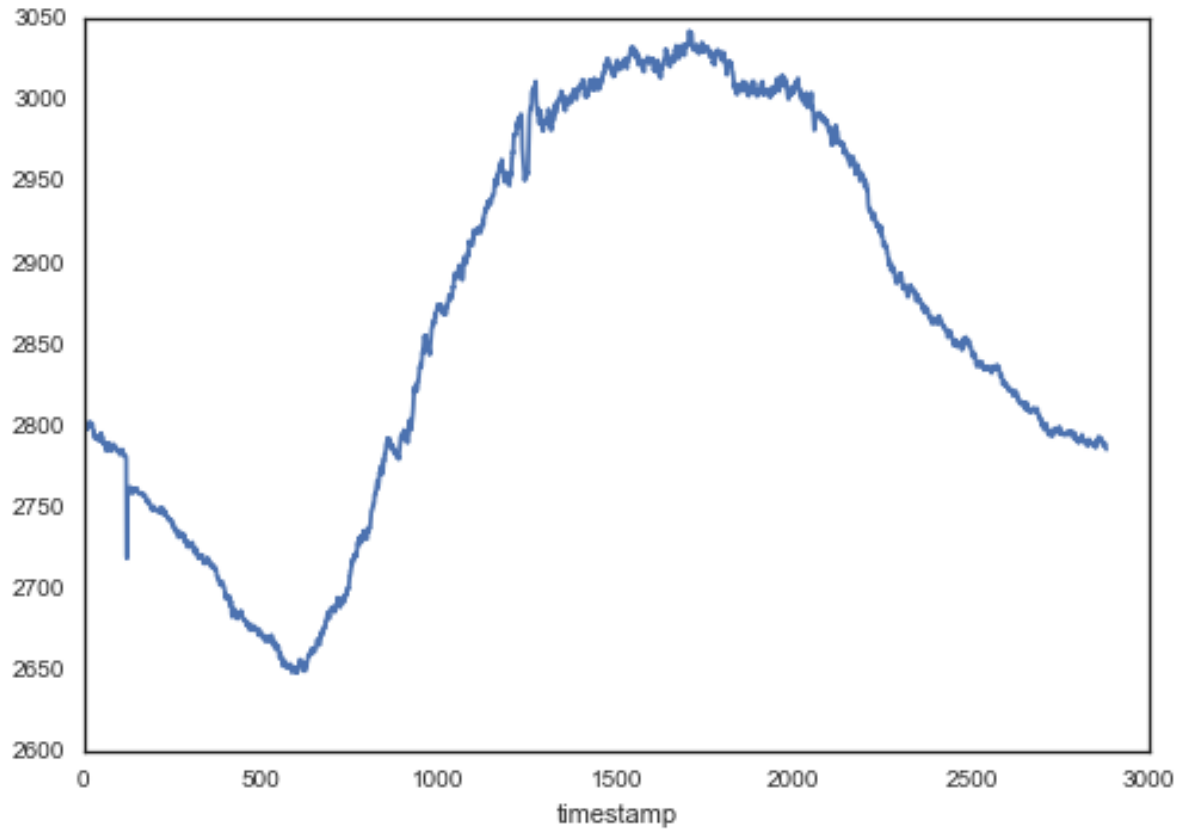

图 4.2　一天内移动车采样数量情况

我们还需要了解，这些移动车主要是采样了哪些区域。如图 4.3，描述了移动车采样的空间分布情况。横轴是经度，纵轴是纬度。上方的直方图表示的是采样数目随经度的变化，右边的直方图则表示采样数目随纬度的变化。我们可以清楚的发现，移动

车主要采样了 5 环内的空气质量状况。特别地，移动车还大量地采集了西五环外的区域。我们同样可以推测，这些数据的分布实际是受汽车移动所造成的。因此数据的空间分布情况符合我们的直觉情况。

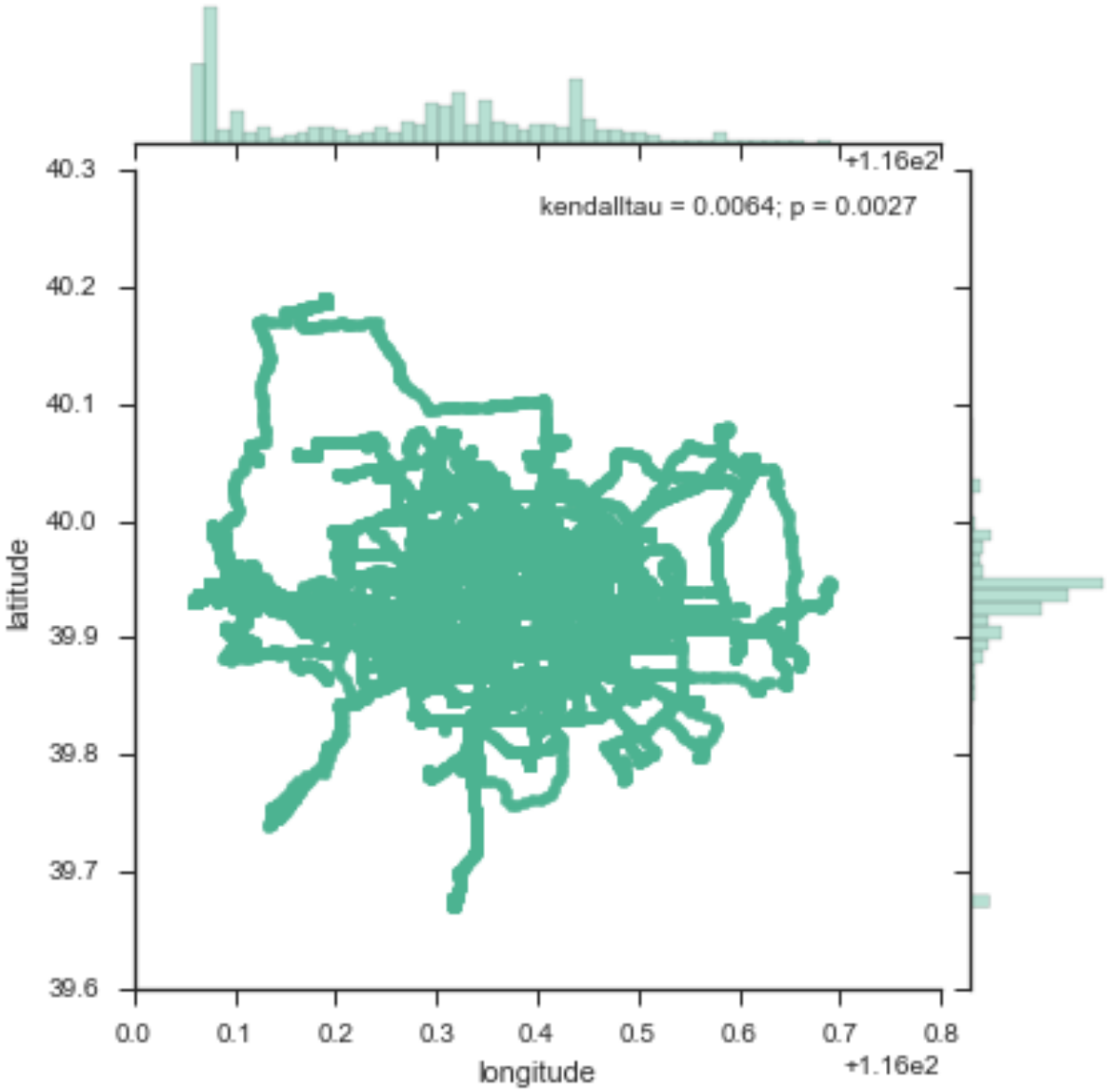

图 4.3　移动车采样空间分布状况

4.1.2　实验数据集生成

通过以上的数据图表和分析，我们对数据的情况有了大体的了解。实际试验中，我们无法对所有日期的数据进行实验展示，我们通过以下方法来生成方便我们实验的数据集。

特征选取

根据第二章的处理结果，数据集有将近百万条数据记录，十几个字段。特别地，这里可以给出每一条记录的表示形式：

$$x_i = (x_i^{weather}, x_i^{spatio}, x_i^{time})$$

这些特征分为 3 类，第一类是天气特征，我们选取了如下字段：

- 东风
- 北风
- 温度
- 气压
- 湿度

而对于第二类特征，也就是空间特征，我们直接选了了每条记录代表的经纬度。而对于第三类时间特征，我们选取了采样时间。

对于第三类特征，时间特征需要做一个特别的处理，我们将所有时间转化为一个相对时间。例如，我们将 2015 年 6 月一日第一条记录的时间记为 0，其余采样记录的时间记录为他们真实采样时间的差（以秒作单位）。

另外，我们还需要明确我们的估计目标。空气质量具体由很多不同污染物的状况所决定，在这里，我们出于数据的实际情况，只考虑如下两种污染物浓度的估计：

- PM2.5
- PM10

时间窗数据

选取了实验所需的特征和预测目标之后，我们需要考虑，我们估计当前空气质量的时刻，所涉及到的采样记录有哪些。考虑到离当前时刻很远的数据 (如一天以前)，并不会给估计带来正贡献，甚至会使得模型误差变大。因此我们通过设置一个时间窗口来限定哪些样本记录和当前时刻相关，用于估计当前时刻的空气质量。

给定一个窗口大小 h, 对于当前时刻 t 而言, 它对应的时间窗口为区间 $(t-h, t]$。我们记整个数据集为 $(\mathbf{X}, \mathbf{y})$, 那么我们就可以给出 t 时刻的相关训练样本集 $(\mathbf{X}_h(t), \mathbf{y}_h(t))$, 满足：

$$(\mathbf{X}_h(t), \mathbf{y}_h(t)) = \{(\mathbf{x_i}, y_i) \mid x_i^{time} \in (t-h, t], i = 1, ..., n\}$$

那么给定一个新的测试样本 x_*，其中 $x_*^{timee} = t$，我们就可以利用训练数据集 $(\mathbf{X}_h(t), \mathbf{y}_h(t))$ 对 f_* 进行估计

实验选取 h 大小，都满足恰好使窗口内训练样本数目等于给定值 m，即

$$h = \arg\min_{h} \{\text{card}(\mathbf{X}_h(t)) = m\}$$

4.2　实验设置和评价指标

4.2.1　对比算法

空气质量估计是一个典型的空间插值问题，我们在第二章相关工作中介绍了各种空气质量估计方法。考虑到我们实际的数据集情况，问题场景适用于常见的空间插值算法。因此，我们选取了两个有代表性的空间插值算法作为我们主要的对比算法，分别是克里金方法和逆距离权重插值法。前者是目前公认最佳的空间插值算法，它一般而言效果佳、模型表示力强，是我们主要的对比对象。而后者是一个带有朴素假设的空间插值算法，虽然在大多数情况下，它的算法效果可能不够好，但是它速度快，是一个非常合理的基线 (baseline) 算法。

下面我们将介绍着两种算法的设置和实现。克里金方法是一种特殊的高斯过程回归，它选取线性核作为其协方差函数，即：

$$k(x, x') = c||\mathbf{x} - \mathbf{x}'||$$

我们将除了时间外的特征作为克里金方法的输入特征，通过梯度下降的方法，学习其超参数 c. 克里金核函数关于其超参数 c 的偏导数为：

$$\frac{\partial k}{\partial c} = c||\mathbf{x} - \mathbf{x}'||$$

逆距离权重插值方法则是根据测试样本和训练样本距离的反比来加权插值，它的表达式为：

$$f_* = \frac{\sum \frac{y_i}{d_i^2}}{\sum \frac{1}{d_i^2}}$$

其中 $d_i = ||\mathbf{x}_* - \mathbf{x_i}||$

4.2.2　设置验证数据集

我们前面一直在讨论如何处理训练集，并没有明确交待如何生成测试集。一个直接的做法是就是将训练集划分成两部分，一部分用作训练，另一个部分用来作为测试集和我们的估计结果进行对比。

但是某时间窗下对应训练集合内样本有时间差的问题, 单纯的随机划分两部分, 会造成训练数据的时间后于测试数据。因此为了贴合实际的估计状况，我们应该选取时间窗口里靠近 t 时刻的样本作为测试集。这样给定时刻 t , 我们随机选取 $(t-\epsilon, t]$ 时刻内的一部分样本作为测试集。具体有：

$$(\mathbf{X}_{\text{test}}(t), \mathbf{y}_{\text{test}}(t)) = \{(\mathbf{x_i}, y_i) \mid x_i^{time} \in (t-\epsilon, t] \text{ and random}() < p_{threshold}, i = 1, ..., n\}$$

通常而言, 我们选取 $\epsilon << h$。

我们就可以得到 t 时刻的训练数据 $(\mathbf{X}_{train}(t), \mathbf{y}_{train}(t))$

$$(\mathbf{X}_{train}(t), \mathbf{y}_{train}(t)) = (\mathbf{X}_h(t), \mathbf{y}_h(t)) \setminus (\mathbf{X}_{\text{test}}(t), \mathbf{y}_{\text{test}}(t))$$

这样通过给出时间 t, 窗口大小 h、测试窗口 ϵ 和测试选择概率 $p_{threshold}$，我们就可以给出了训练数据和测试数据的生成方法。接下来，我们将可以进行实际的实验。

4.2.3 算法实现和平台

实验数据的清洗、预处理和连接等操作均用 Python 完成。所有的算法均以 Java 实现, 特别地，我们的算法实现只用到了一个 Java 的第三方线性代数库 EJML① 。算法均为单线程实现，全部实验均在 2.4GHz Intel Core i5 4G 内存的机器下运行。

4.2.4 评价指标

对于如何评价不同算法之间的准确度，我们用平均绝对误差来衡量预测结果和真实样例间的差距，它的计算方法如下：

$$\text{MAE} = \frac{\sum |y_i - \hat{y}_i|}{m}$$

4.2.5 实验目标

本文实验的目的有如下三个：其一，验证我们算法的建模的有效性；其二，验证我们算法和其他算法的性能对比。其三，验证我们算法在各种场景下的运行情况。

- 验证我们算法的有效性，而这
- 对比算法之间的性能实验主要在两方面进行比较:
  - 对比常见回归方法的时间表现和误差表现

① http://code.google.com/efficient-java-matric-library/

– 对比常见回归方法的时间表现

4.3　实验结果分析

4.3.1　核函数选取的评价

在本文，我们提出了一个由三部分构成的核函数，那么三部分核函数各自起到的作用是怎样。他们是不是都是模型的有效部分，我们将进行实验验证。

我们将三个核分别表示如下：

$$\begin{aligned} K_{weather} &= \sigma_w^2 \exp\{-\frac{1}{2} D_{weather}^T M D_{weather}\} \\ K_{spatio} &= \sigma_s^2 \exp\{-\frac{||x^{spatio} - x'^{spatio}||^2}{l_s}\} \\ K_{time} &= \sigma_t^2 \exp\{-\frac{||x^{time} - x'^{time}||^2}{l_t}\} \end{aligned}$$

通过对它们所有组合枚举，我们随机选取了 10 个时间点进行实验，计算它的 MAE 值的平均值，对他们其结果显示在了表格 4.1中。

特征 | MAE	PM2.5	PM10
$K_{weather}$	19.52	20.13
K_{spatio}	9.81	10.23
K_{time}	40.32	45.67
$K_{weather} + K_{spatio}$	8.67	8.89
$K_{weather} + K_{time}$	16.87	17.01
$K_{spatio} + K_{time}$	8.87	9.01
$K_{weather} + K_{spatio} + K_{time}$	7.69	7.93

表 4.1　核函数选择组合实验

我们可以看出如下几点：其一，单独使用时间特征的核函数效果很差，但是当和别的特征的核函数结合会有效果上的提升；其二，三个特征当中，空间特征对应的核函数贡献最大，单独实用的情况下已经有很好的效果；其三，当三者全部组合在一起时，此时效果是最好的，验证了我们模型的有效性和正确性。

4.3.2　空气质量估计误差比较与分析

我们随机选取 10 个时间戳下的数据分别对 PM2.5 和 PM10 进行了实验，最终求得算法误差和训练样本规模的关系，如图 4.4 和 4.5。其中它们的横轴为每次实验的平均训练样本数, 纵轴则为 MAE 值；而对比的算法为克里金方法和 IDW。

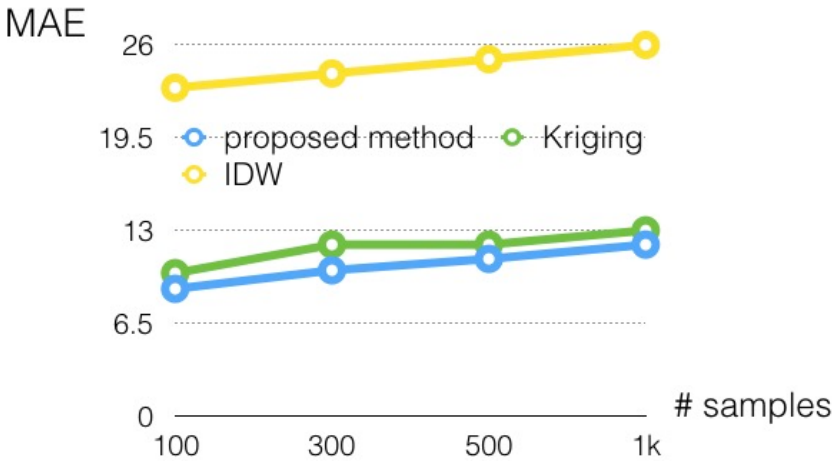

图 4.4　PM2.5 回归误差比较

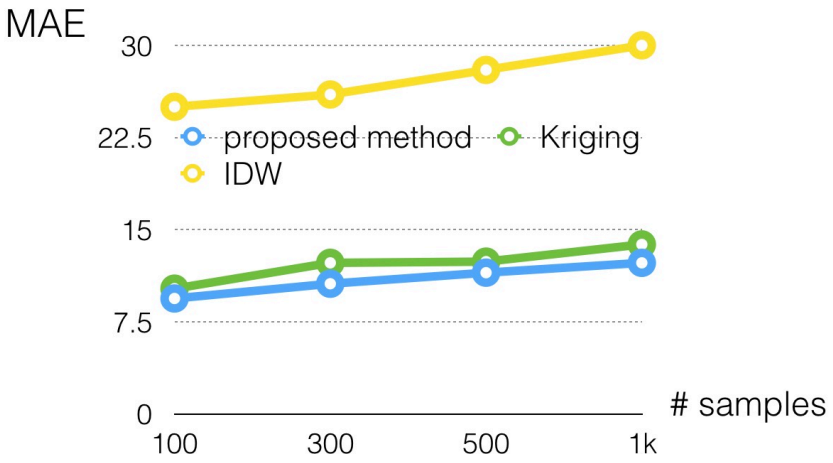

图 4.5　PM10 回归误差比较

从图 4.4 和 4.5中，我们可以看出 IDW 算法效果最不好，远不如其他两种算法。而对于我们的算法和克里金方法比较，我们的算法在误差上略优于克里金方法。总得而言，我们的算法取得最佳表现。考虑到空气污染的浓度是有非常复杂而且充满噪音的模型产生，任何空气质量估计的算法，其误差下界实际主要都受数据本身所影响。可以看出在本文所涉及的数据集的情况下，同时在克里金方法的对比下，我们可以知道，我们的算法的效果已经达到一个比较优的结果

4.3.3　空气质量估计效率比较与分析

同样的，我们随机选取 10 个时间戳下的数据分别对 PM2.5 和 PM10 进行了实验，最终求得算法平均运行时间和样本规模的关系，如图 4.6 和 4.7。其中它们的横轴为每次实验的平均训练样本数，纵轴则为 MAE 值；而对比的算法为克里金方法和 IDW。

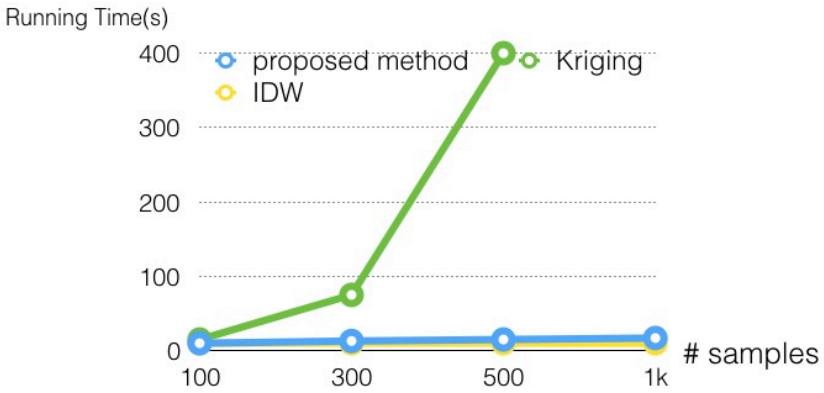

图 4.6　PM2.5 回归的时间性能

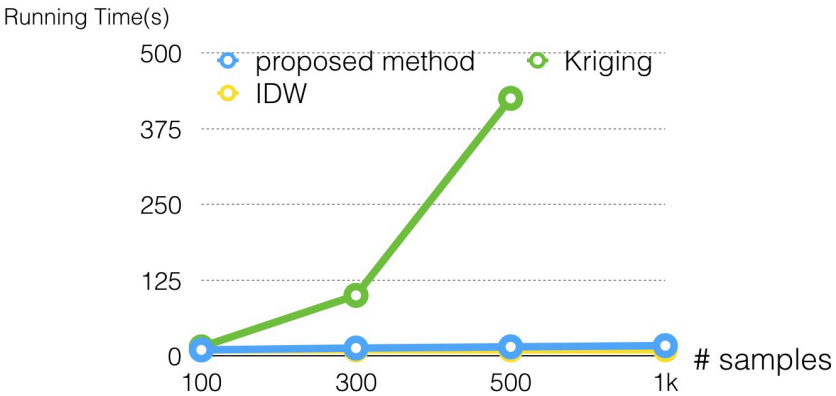

图 4.7　PM10 回归的时间性能

从图 4.6和 4.7中，我们可以看出，克里金方法复杂度非常高，当样本超过 500 个以后，机器已经无法跑出结果。而 IDW 算法本身是一个线性复杂度的算法，由于数据规模不大，其运行时间几乎没有多大变化。而我们提出的算法在时间均略高于它，但是几乎同样是线性增长。我们的算法优于克里金方法两到三个两集。通过我们之前的分析可以知道，算法的复杂度虽然是超线性的，但是在实际应用的场景下（如数千样本），其运行时间已经足以在分钟内跑出，已经符合实际场景需求。

综合其误差表现来看，总的而言，我们算法的综合评价是最高的。

4.3.4 不同空间粒度下的空气质量估计

实际需求当中，我们不需要对城市任意一点回归计算其值，这样做代价非常大。一般而言，我们会将城市划分为大小一致的网格。我们只需要求出网格中心点的空气污染状况，就可以代表整个网格的空气质量了。当然对于不同大小的网格划分，造成的效果会有所不同。下面，我们用实验来展示。

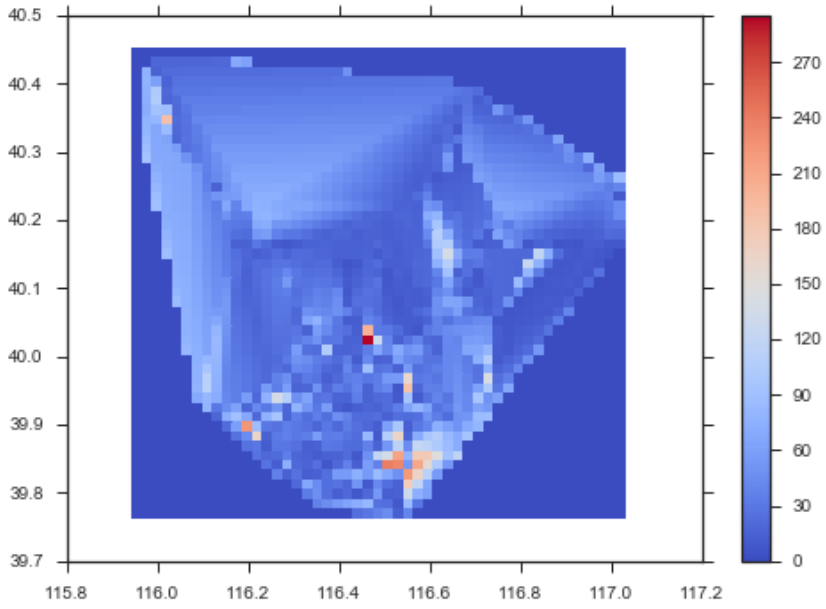

图 4.8 按照 2km 方格大小估计城市空气质量（PM2.5）

如图 4.8、 4.9和 4.10，是将城市按照 2km、4km 和 10km 边长方格划分，并求出其空气污染状况的实验结果。它们的横轴都是经度，纵轴是纬度，颜色的深浅表示污染浓度的大小。蓝色代表污染状况较轻，红色代表空气污染状况较为严重。

我们可以看到，随着划分粒度的越来越大，图标描述能力越差。相邻网格实际上做了一个“合并”。按照 2km 网格划分，可以看到更多的受污染区域细节。而到了 10km

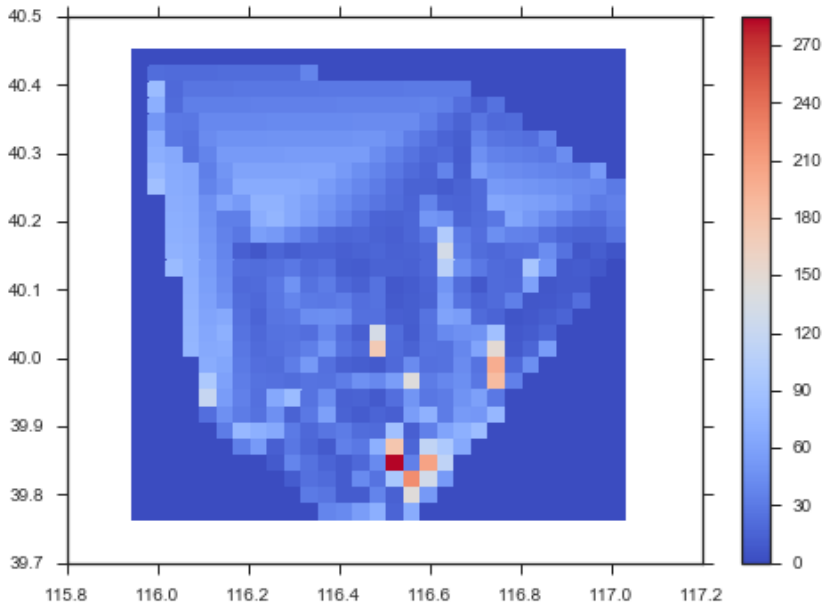

图 4.9　按照 4km 方格大小估计城市空气质量（PM2.5）

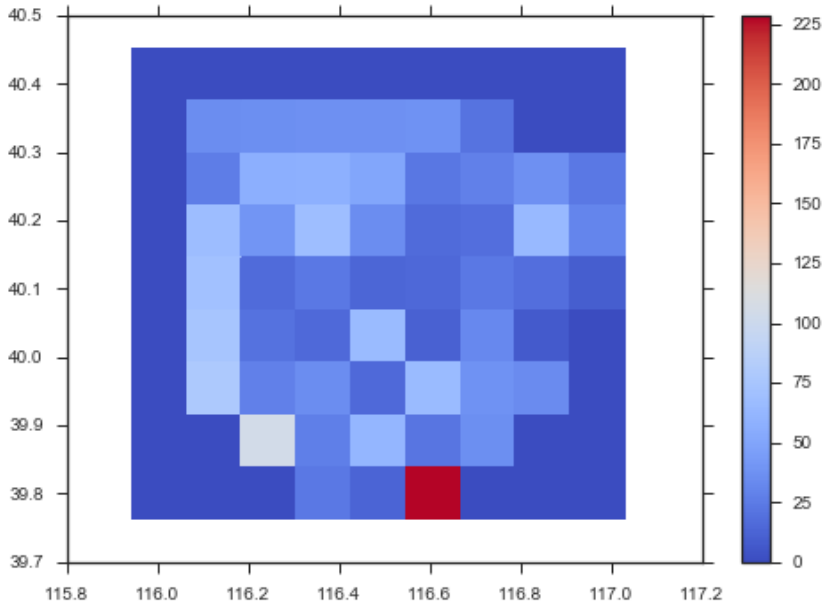

图 4.10　按照 10km 方格大小估计城市空气质量（PM2.5）

的时候，细节消失，只能知道各个大区域的空气状况。具体而言，人们可以根据实际的需求选择恰当的网格划分方式。

与此同时，观察这些图，不难看出，我们的采样数据集中在 5 环内的部分，因此 3 个图中央偏下的部分的方格污染浓度多变，不想其他部分是一个连续变化的“色块”。所以本质上，空气质量估计如果想要效果好，需要我们去采集时空覆盖面较广的数据。

4.4 本章小结

本章介绍了实验的数据情况和实验的基本设置。选取了两种最有代表性的空间插值方法作为对比方法。对比了我们提出的方法和它们在误差表现和时间性能表现两方面的对比。实验表明，我们的算法，和目前公认最优的克里金方法的误差近似，略优于它。并且我们在时间性能上的表现远远优于克里金方法，效率提高了 2-3 个量级。

总结与未来工作

4.5 总结

论文工作旨在通过对移动采样数据和天气数据建模，从而去估计城市的空气质量。总结本文工作如下：

- 结合移动监测数据和气候数据，提出了一种基于高斯过程回归的新回归方法，通过引入新的核函数来描述采样数据在气候特征、时空特征等特征上的距离，用来估计没有监测到区域的当前空气污染状况。
- 针对移动车采样数据量大的情况，给出一种基于 k-d 树的空间划分和采样方法，用来提高我们空气污染估值的效率。
- 实验表明，我们的方法准确度优于传统的插值方法，并在时间效率上有了多个量级的提升。

4.6 未来工作

文本主要研究了如何估计城市中的空气质量，未来还有许多可以进一步研究的地方：

- 污染物的传播扩散是一个非常复杂的模型，我们通过数据驱动的方法固然可以发现当中的部分规律和关联。但是如果我们如果想进一步提高实用性，可以考虑引入更多的外部数据。除了移动采样数据和天气数据，我们可以融入城市的 POI 信息，工业排放量和汽车行驶情况等信息，来进一步优化我们的结果。
- 本文主要对城市空气质量进行估计。但是数据本身还蕴含了许多信息，我们是否能用来进行更多的关于城市状况的计算。基于对下一个十分钟、一小时、两小时等未来的估计。而不仅仅是对未检测区域的污染物浓度进行回归分析。
- 如今移动传感器是单纯安装在了私人汽车上，汽车的移动是没有事先规划的，往往出现某些时刻，采样数据集中在某些区域。在将来，如果有专门采样的移动车，我们是否可以设计出相应的路径规划算法，使得各辆汽车互相配合，让采样的效率达到最大化。

环境计算是城市计算未来最重要的课题之一，和我们人类生活直接相关。如何更好地利用分布在城市中各色的“传感器”来进行计算，从而为人类带来便利，是一个非常有意义的挑战。

参考文献

[1] Julie Yixuan Zhu, Chenxi Sun, Victor OK Li. Granger-Causality-based air quality estimation with spatio-temporal (ST) heterogeneous big data[C]. Computer Communications Workshops (INFOCOM WKSHPS), 2015 IEEE Conference on. IEEE, 2015, 612–617

[2] Jason Jingshi Li, Boi Faltings, Olga Saukh, David Hasenfratz, Jan Beutel. Sensing the air we breathe-the opensense zurich dataset[C]. Proceedings of the National Conference on Artificial Intelligence. 2012, vol. 1, 323–325

[3] Carl Edward Rasmussen, Hannes Nickisch. Gaussian processes for machine learning (GPML) toolbox[J]. The Journal of Machine Learning Research. 2010, **11**:3011–3015

[4] 王静远, 李超, 熊璋, 单志广. 以数据为中心的智慧城市研究综述 [J]. 计算机研究与发展. 2014, **51**(2):239–259

[5] Yu Zheng, Licia Capra, Ouri Wolfson, Hai Yang. Urban computing: concepts, methodologies, and applications[J]. ACM Transactions on Intelligent Systems and Technology (TIST). 2014, **5**(3):38

[6] David Hasenfratz, Olga Saukh, Christoph Walser, Christoph Hueglin, Martin Fierz, Lothar Thiele. Pushing the spatio-temporal resolution limit of urban air pollution maps[C]. Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on. IEEE, 2014, 69–77

[7] Randall V Martin. Satellite remote sensing of surface air quality[J]. Atmospheric Environment. 2008, **42**(34):7823–7843

[8] Aaron Van Donkelaar, Randall V Martin, Rokjin J Park. Estimating ground-level PM2. 5 using aerosol optical depth determined from satellite remote sensing[J]. Journal of Geophysical Research: Atmospheres. 2006, **111**(D21)

[9] Yu Zheng, Furui Liu, Hsun-Ping Hsieh. U-Air: When urban air quality inference meets big data[C]. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013, 1436–1444

[10] Srinivas Devarakonda, Parveen Sevusu, Hongzhang Liu, Ruilin Liu, Liviu Iftode, Badri Nath. Real-time air quality monitoring through mobile sensing in metropolitan areas[C]. Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing. ACM, 2013, 15

[11] Arnaud Jutzeler, Jason Jingshi Li, Boi Faltings, et al. A Region-Based Model for Estimating Urban Air Pollution.[C]. AAAI. 2014, 424–430

[12] David Hasenfratz, Olga Saukh, Christoph Walser, Christoph Hueglin, Martin Fierz, Tabita Arn, Jan Beutel, Lothar Thiele. Deriving high-resolution urban air pollution maps using mobile sensor nodes[J]. Pervasive and Mobile Computing. 2015, **16**:268–285

[13] Yifei Jiang, Kun Li, Lei Tian, Ricardo Piedrahita, Xiang Yun, Omkar Mansata, Qin Lv, Robert P Dick, Michael Hannigan, Li Shang. MAQS: a personalized mobile sensing system for indoor air quality monitoring[C]. Proceedings of the 13th international conference on Ubiquitous computing. ACM, 2011, 271–280

[14] David Hasenfratz, Olga Saukh, Silvan Sturzenegger, Lothar Thiele. Participatory air pollution monitoring using smartphones[J]. Mobile Sensing. 2012:1–5

[15] Hsun-Ping Hsieh, Shou-De Lin, Yu Zheng. Inferring air quality for station location recommendation based on urban big data[C]. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015, 437–446

[16] 郑宇. 城市计算与大数据 [J]. 中国计算机学会通讯. 2013, **9**(8):6–16

[17] Ravinder K Jain, Jose MF Moura, Constantine E Kontokosta. Big Data+ Big Cities: Graph Signals of Urban Air Pollution [Exploratory SP][J]. Signal Processing Magazine, IEEE. 2014, **31**(5):130–136

[18] Donald L Ermak. An analytical model for air pollutant transport and deposition from a point source[J]. Atmospheric Environment (1967). 1977, **11**(3):231–237

[19] Gerard Hoek, Rob Beelen, Kees De Hoogh, Danielle Vienneau, John Gulliver, Paul Fischer, David Briggs. A review of land-use regression models to assess spatial variation of outdoor air pollution[J]. Atmospheric environment. 2008, **42**(33):7561–7578

[20] Yu Zheng, Xiuwen Yi, Ming Li, Ruiyuan Li, Zhangqing Shan, Eric Chang, Tianrui Li. Forecasting fine-grained air quality based on big data[C]. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015, 2267–2276

[21] Lixue Xia, Rong Luo, Bin Zhao, Yu Wang, Huazhong Yang. An accurate and low-cost PM 2.5 estimation method based on Artificial Neural Network[C]. Design Automation Conference (ASP-DAC), 2015 20th Asia and South Pacific. IEEE, 2015, 190–195

[22] Sotiris Vardoulakis, Bernard EA Fisher, Koulis Pericleous, Norbert Gonzalez-Flesca. Modelling air quality in street canyons: a review[J]. Atmospheric environment. 2003, **37**(2):155–182

[23] Christopher KI Williams, Carl Edward Rasmussen. Gaussian processes for machine learning[J]. the MIT Press. 2006, **2**(3):4

在学期间研究成果

已发表论文

Xiaodong Chen, Guojie Song, Xinran He and Kunqing Xie. On Influential Nodes Tracking in Dynamic Social Networks. SDM, 2015.

参与课题

863 子课题：面向社交网络的搜索方法与群体行为分析

横向项目：NEC 城市空气质量估计、预测以及可视化研究

致谢

感谢帮助过我的所有人

北京大学学位论文原创性声明和使用授权说明

原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含任何其他个人或集体已经发表或撰写过的作品或成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本声明的法律结果由本人承担。

论文作者签名：　　日期：　　年　　月　　日

学位论文使用授权说明

（必须装订在提交学校图书馆的印刷本）

本人完全了解北京大学关于收集、保存、使用学位论文的规定，即：

- 按照学校要求提交学位论文的印刷本和电子版本；
- 学校有权保存学位论文的印刷本和电子版，并提供目录检索与阅览服务，在校园网上提供服务；
- 学校可以采用影印、缩印、数字化或其它复制手段保存论文；
- 因某种特殊原因需要延迟发布学位论文电子版，授权学校在 □ 一年 / □ 两年 / □ 三年以后在校园网上全文发布。

（保密论文在解密后遵守此规定）

论文作者签名：　　导师签名：　　日期：　　年　　月　　日