

# 《数据可视化》



**马晓磊**

**交通科学与工程学院**

**2024年5月**

## 什么是可视化？

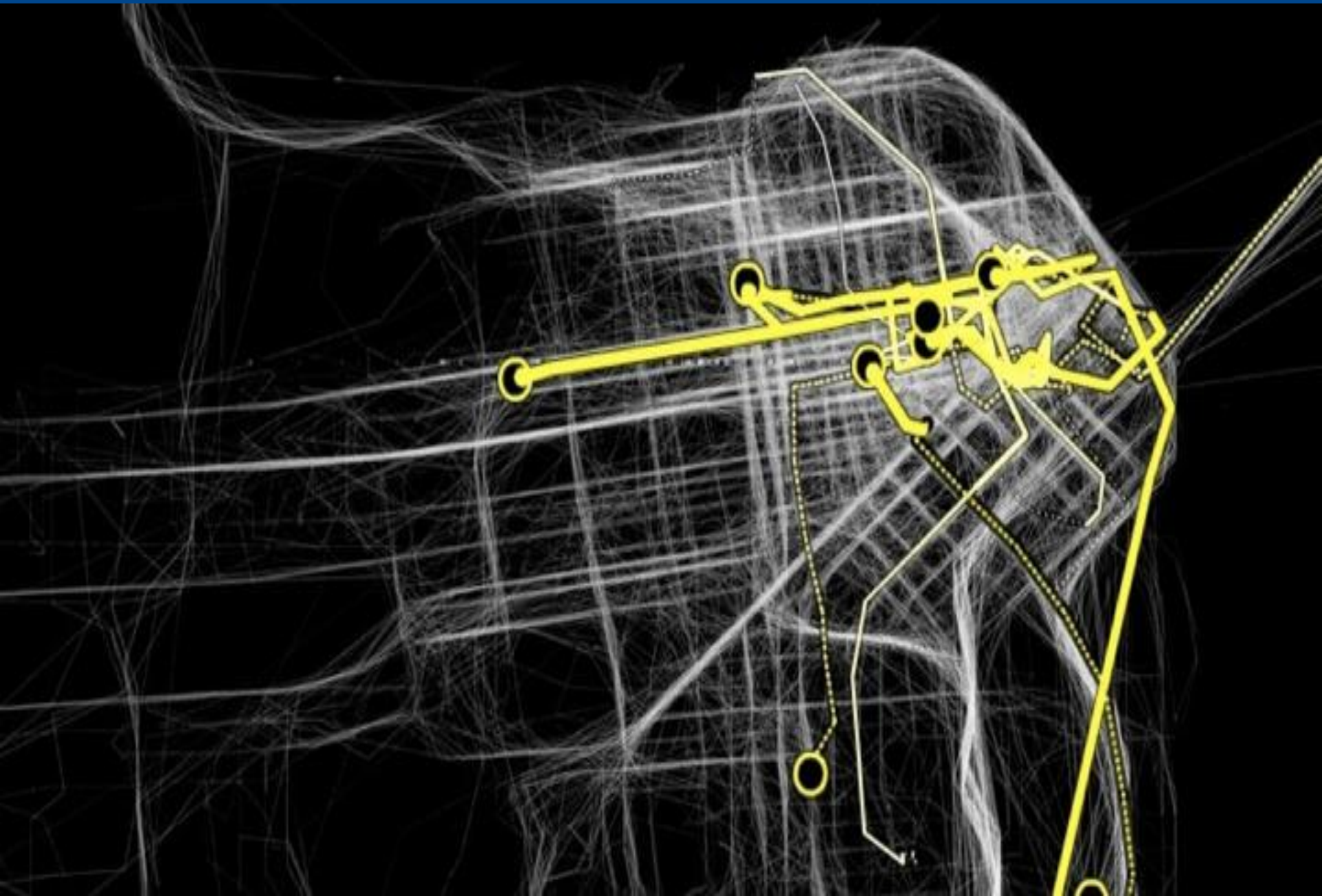
“从符号到几何的转换” [McCormick et al. 1987]

“...寻找最能支持我们自然感知方式的人工记忆 ” [Bertin 1967]

“使用计算机生成的、交互式的视觉数据表示法来增强认知能力  
” [Card, Mackinlay, & Shneiderman 1999]

利用计算机图形学和图像处理技术，将**数据**转换成**图形或图像**在屏幕上显示出来，再进行交互处理的理论、方法和技术。

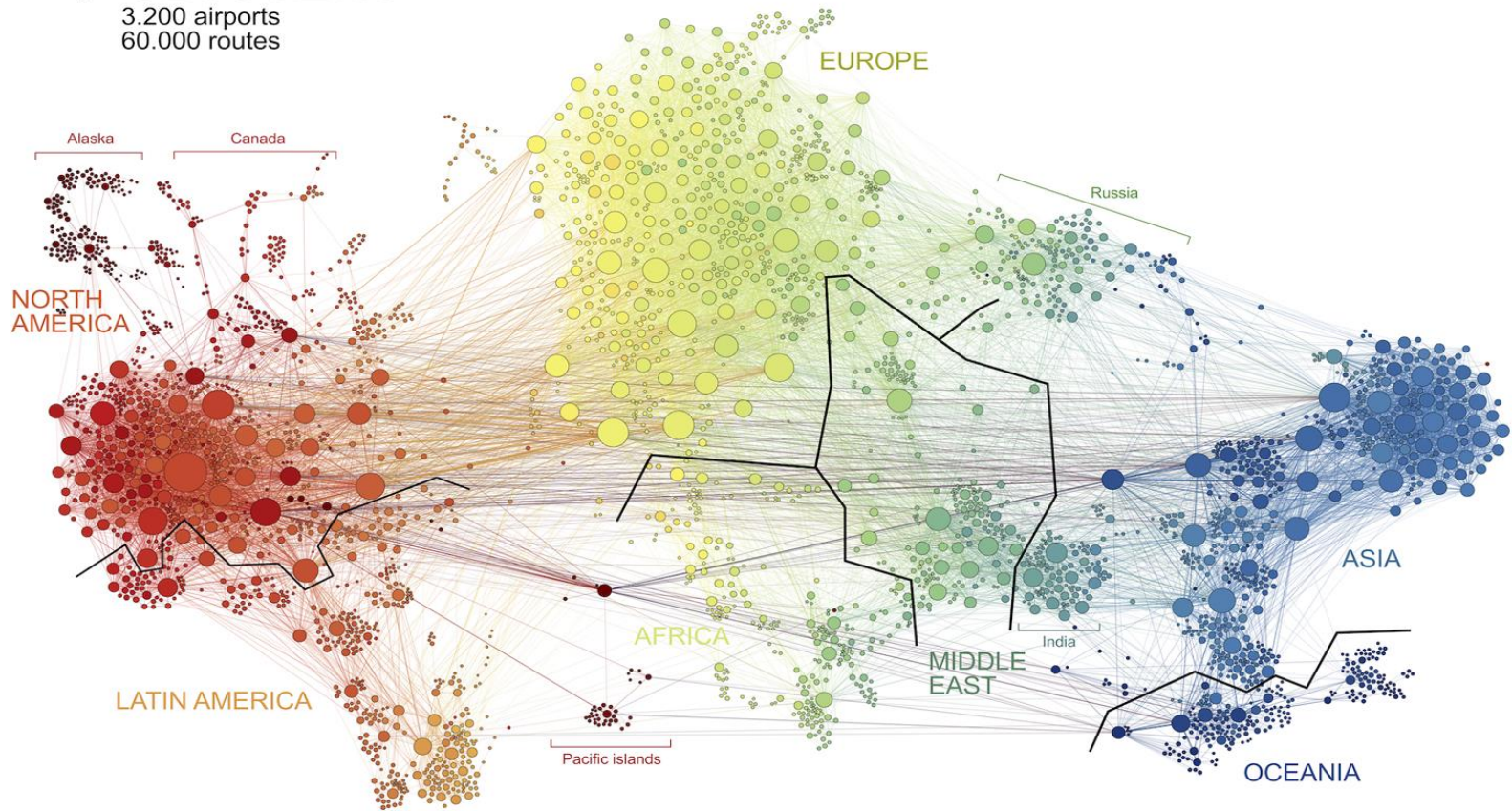
# 数据可视化案例



# 数据可视化案例

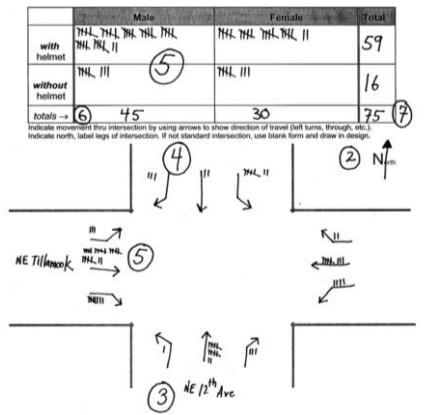
## TRANSPORTATION CLUSTERS

3.200 airports  
60.000 routes

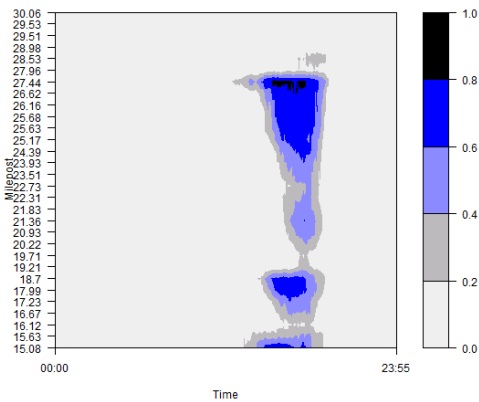


# 数据可视化的意义

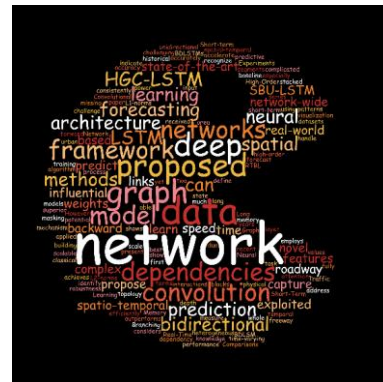
## 为什么数据需要可视化？



## 记录信息

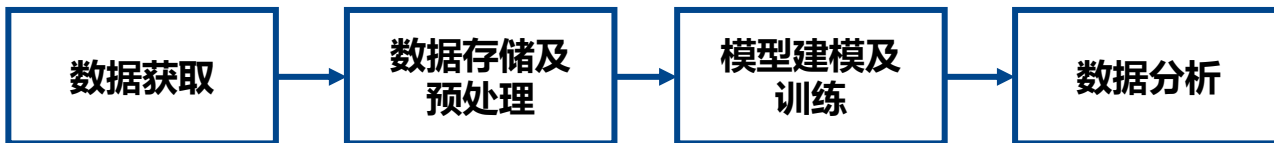


## 支持推论



## 交流信息

## 交通大数据技术需要可视化？





# 数据可视化整体框架

## 任务

问题，目标

- 假设

## 数据

- 物理数据类型
- 概念数据类型

## 域

- 元数据
- 语义
- 规定

处理算法

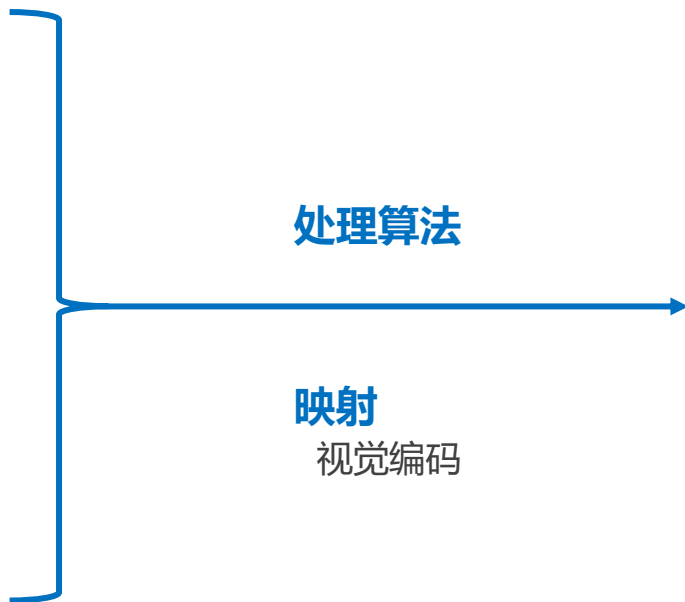
映射

视觉编码

图像

视觉通道

图形标记



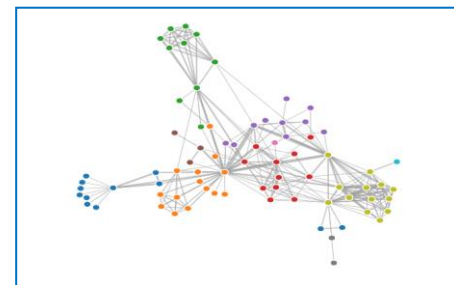
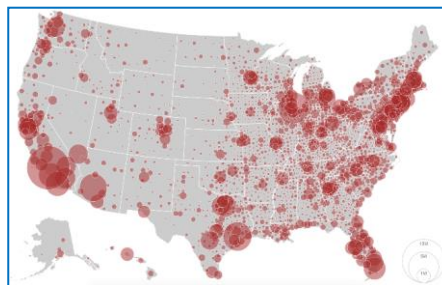
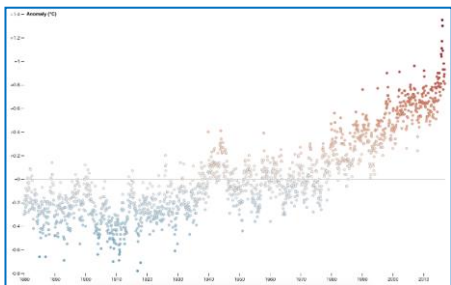
# 数据可视化主要技术

## 可视化的数据类型

1D ( 集合与序列 )、时间、2D ( 地图 )、3D ( 形状 )、nD ( 关系 )、树 ( 层次 )、网络 ( 图形 )、时空 ( 轨迹 ) 等。还能列举其他的数据类型吗？

## 可视化的类型

散点图、折线图、饼图、直方图、热力图、树状图、地图、曲线图等。还能列举其他的类型吗？



## 视觉编码设计

能否用一种可视化满足所有需求，回答所有问题？

很少有一种可视化能回答所有问题。相反，快速生成适当的可视化效果的能力是至关重要的！

## 视觉编码的设计要点

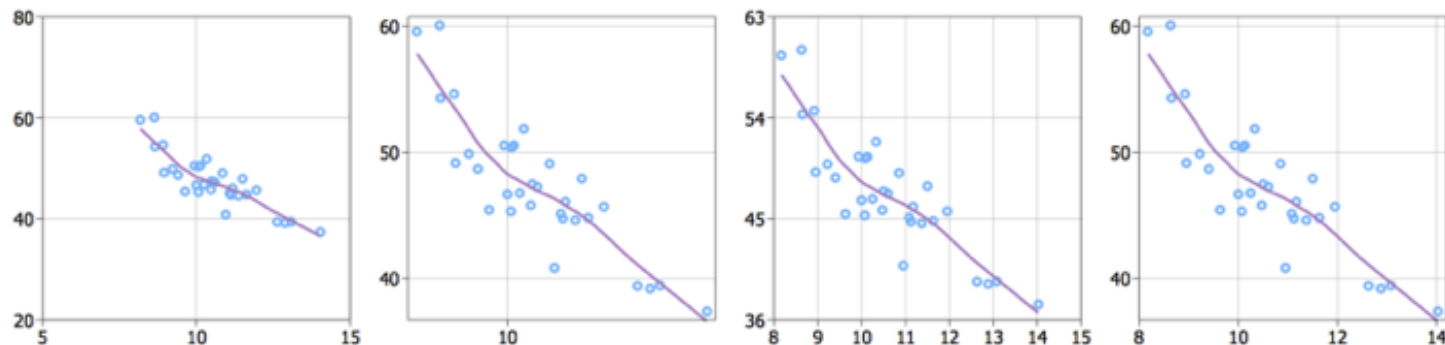
- 使用富有表现力和**有效的**编码、避免过度编码
- 智能地使用**空间**和**小而多图(small multiples)**、**减少**问题空间
- 使用**交互**来生成相关视图



# 数据可视化主要技术——有效编码

## 刻度和轴

下面各种表现方式分别不好在哪？



**要点：**

**简单**—数字是10、5、2的倍数

**覆盖率**—数据末尾附近的记号

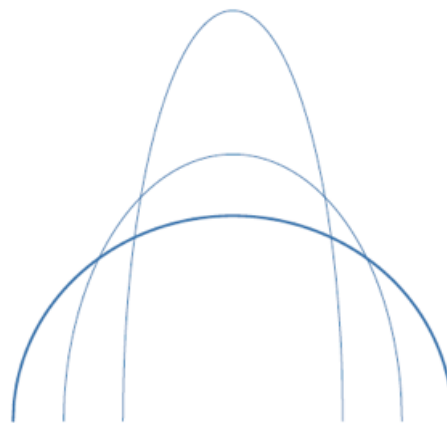
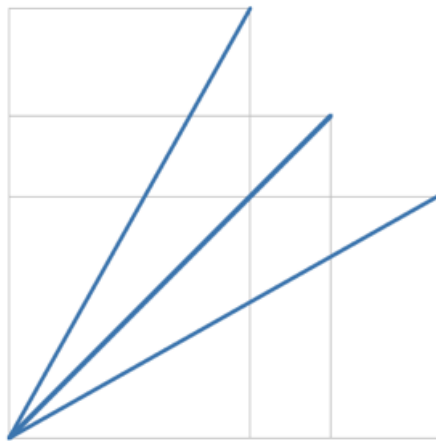
**密度**—不要太多，也要不太少

**易读性**—空白、水平文本、大小

# 数据可视化主要技术——有效编码

## 纵横比

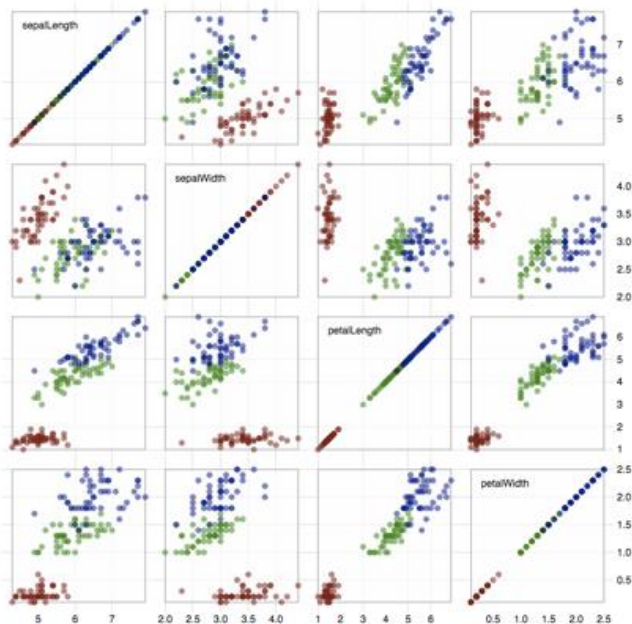
在下面各图的三条线中哪条线的表现方式最好？



**直线尽量接近45°，椭圆尽量接近圆**

# 数据可视化主要技术——小而多图

## 多维数据



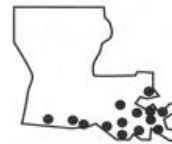
alfisol



entisol



histosol



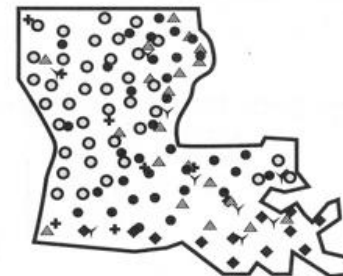
inceptisol



mollisol



ultisol



# 数据可视化主要技术——降维

## 多维数据的另一种处理方式——降维

### 多种降维还原技术：

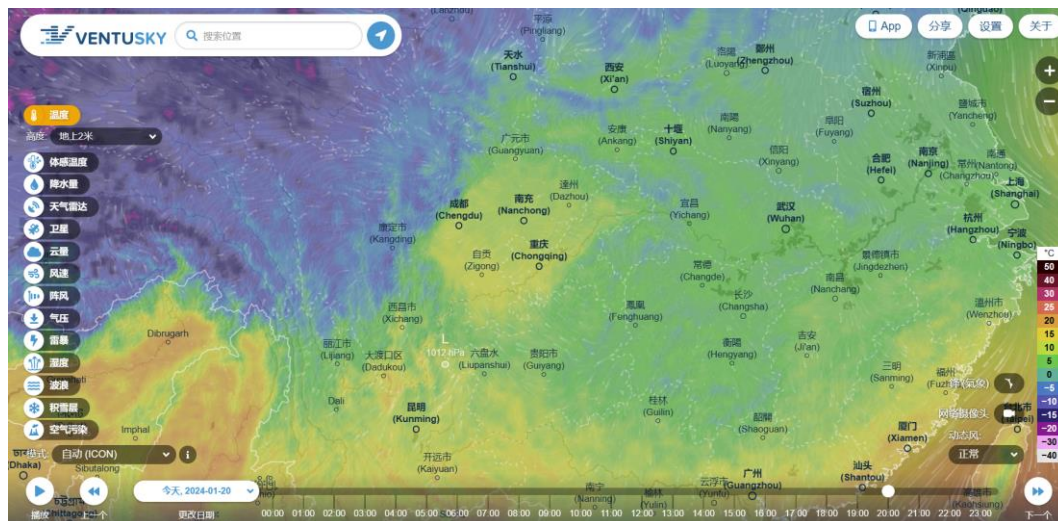
- 主成分分析 ( PCA )
- t—距离随机近邻嵌入 ( t-SNE )
- 统一流形逼近与投影 ( UMAP )
- 自动编码神经网络
- 多维尺度分析 ( MDS )
- ...



人与机器之间的交互需要**相互理解**或**共享理解**。

## 交互的分类

- 数据和视图规范
  - 可视化、过滤、排序、导出
- 视图操作
  - 选择、导航、协调、组织
- 过程和起源
  - 记录、注释、共享、指导

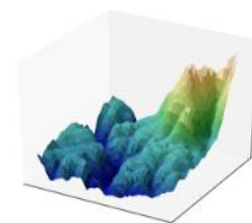
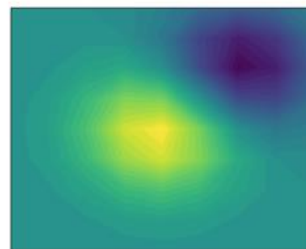
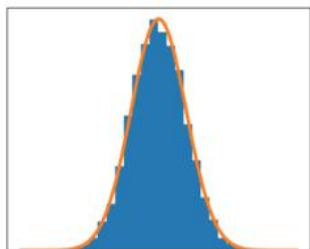
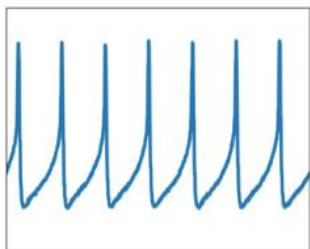


**交互式**可视化，甚至动画，对于数据可视化工具或数据分析平台非常有用

# 数据可视化的主要工具

## Matplotlib–Python (<https://matplotlib.org/>)

- Matplotlib是一个Python 2D绘图库，它以各种硬拷贝格式和跨平台的交互环境生成出版质量的图形。
- Matplotlib可用于Python脚本、Python和IPython shell、Jupyter notebook、web应用服务器和四个图形用户界面工具包。

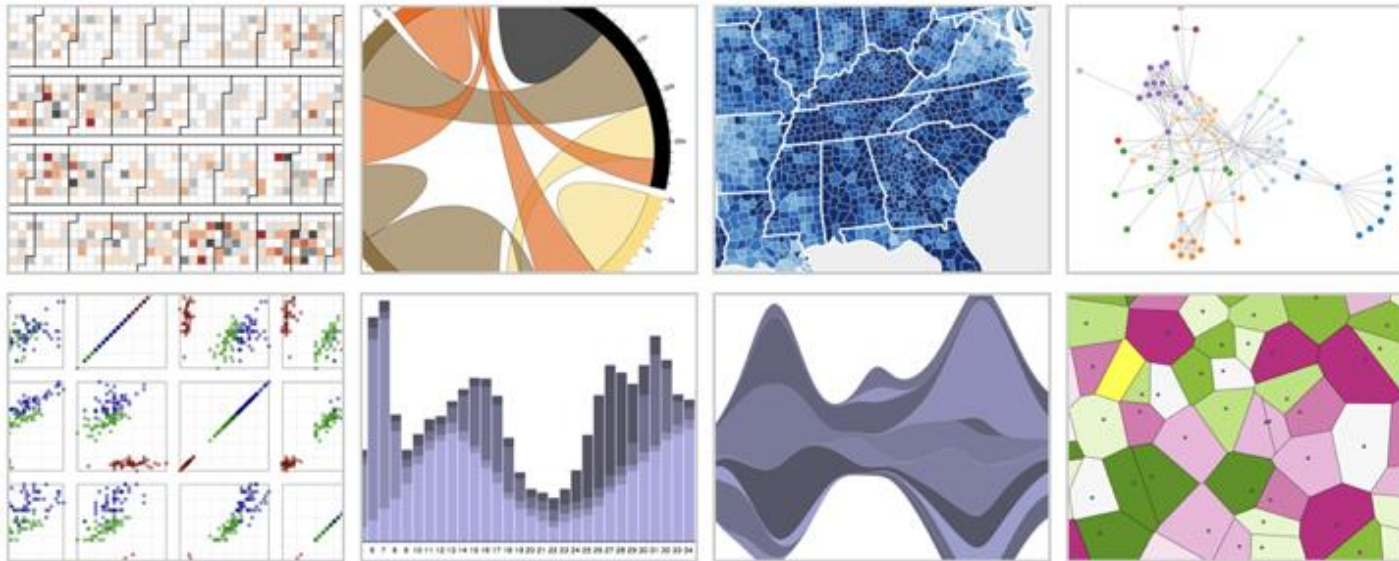


matplotlib



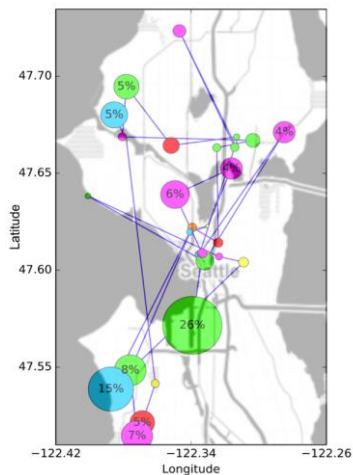
# 可视化主要工具

d3.js – JavaScript (<https://d3js.org/>)

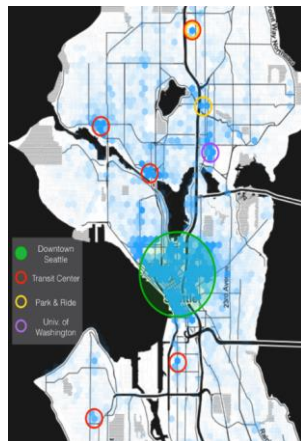


## 数据可视化也可以存在于任何部分

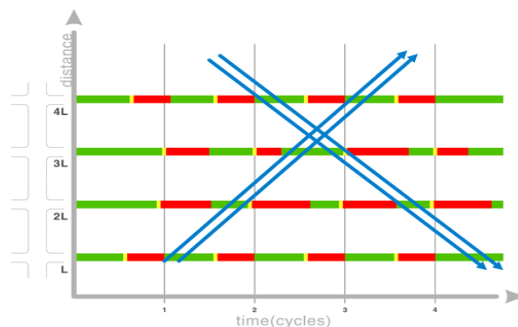
一般任务的流程是从**数据存储**到**数据处理**，再到**数据建模分析**，最后到**数据结果**。每一块都可以采用可视化。



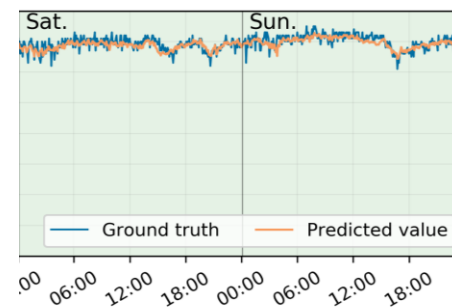
存储数据的可视化  
的可视化



数据处理中的可视化



数据建模分析的可视化



数据结果