

# 当数据遇到骑行

野兽骑行 高嘉阳

# 提纲

- 公司介绍
- 数据matters
- 数据存储
- 数据可视化



# 野兽运动自行车

科技、竞技、美



# 野兽智能中控

将任何自行车变成智能自行车



# 野兽骑行APP

最好的骑行者社区

# 10项骑行数据的自动收集及专业可视化分析



XCoach智能骑行

# 教练系统

## 实时骑行指导

在APP设定为完成骑行教练后，  
你的数据将同步到智能中控系统，  
中控系统将实施进行骑行指导反  
馈，让你更专业的骑行，更优秀  
的达成健身目标。







## LEOPARD PRO

顶配级碳纤维智能公路车

- T1000军工级碳纤维车架
- SpeedX 智能中控系统
- 全隐藏式走线、刹车设计
- 空气动力学设计
- Shimano UT Di2电子变速
- 全碳纤维轮组
- 碳纤维车把

¥14,999



## LEOPARD

专业级碳纤维智能公路车

- T1000军工级碳纤维车架
- SpeedX 智能中控系统
- 全隐藏式走线、刹车设计
- Shimano 105 套件
- 空气动力学设计
- 车架终身包换

¥7,999



## LEOPARD AL

发烧级铝合金智能公路车

- 超轻、超强、“无焊点”铝合金车架
- SpeedX Lite 智能中控系统
- Shimano公路车碟刹
- Shimano Tiagra 20速套件
- 空气动力学设计

¥4,999



## MUSTANG

专家级碳纤维智能山地车

- 东丽T800碳纤维车架
- SpeedX 智能中控系统
- SHIMANO DEORE 30速变速套件
- Rockshox 前叉
- Frog Design

¥3,999



## MUSTANG AL

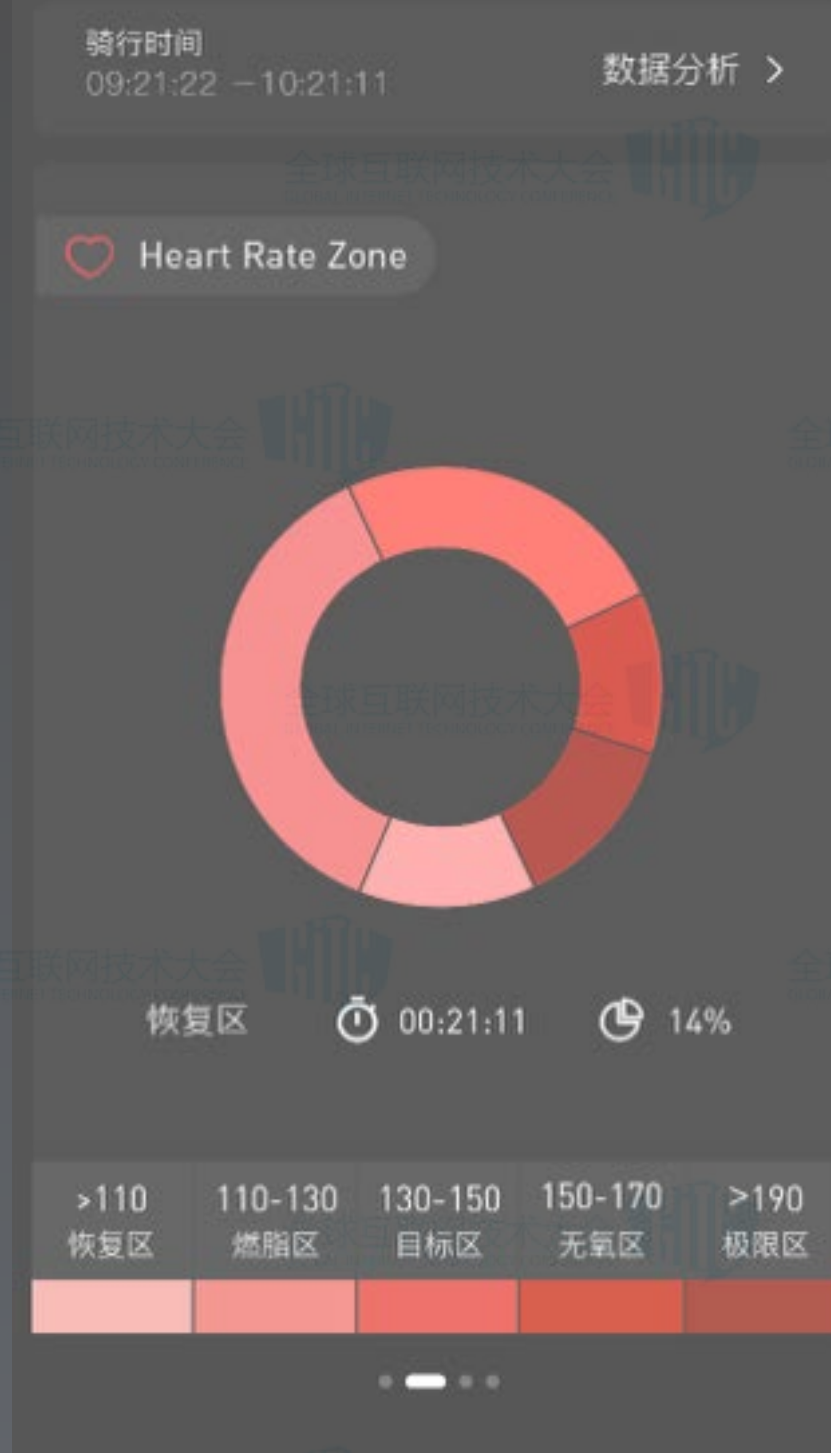
发烧级铝合金智能山地车

- 超轻、超强、“无焊点”铝合金车架
- SpeedX Lite智能中控系统
- SHIMANO DEORE 30速变速套件
- Rockshox 前叉
- 油压碟刹

¥2,999

# 最专业的骑行数据分析

对心率深入的分析，可以准确了解每次运动后的效果，多维度数据对比，可以更方便的对运动进行分段分析，清晰了解自己的短板。





## Apple Watch Series 2

概览

表盘

购买



## 苹果官网推荐的唯一运动类APP

### Apple Pay

走进商铺，选好商品，然后只需连按两下侧边按钮，就能用手表安全地支付。

### 提醒事项

让 Siri 提醒你接下来该做什么。当你结束一项任务时，可轻点“已完成”，当需要往后推一推时，可轻点“稍后提醒”。

### 野兽骑行

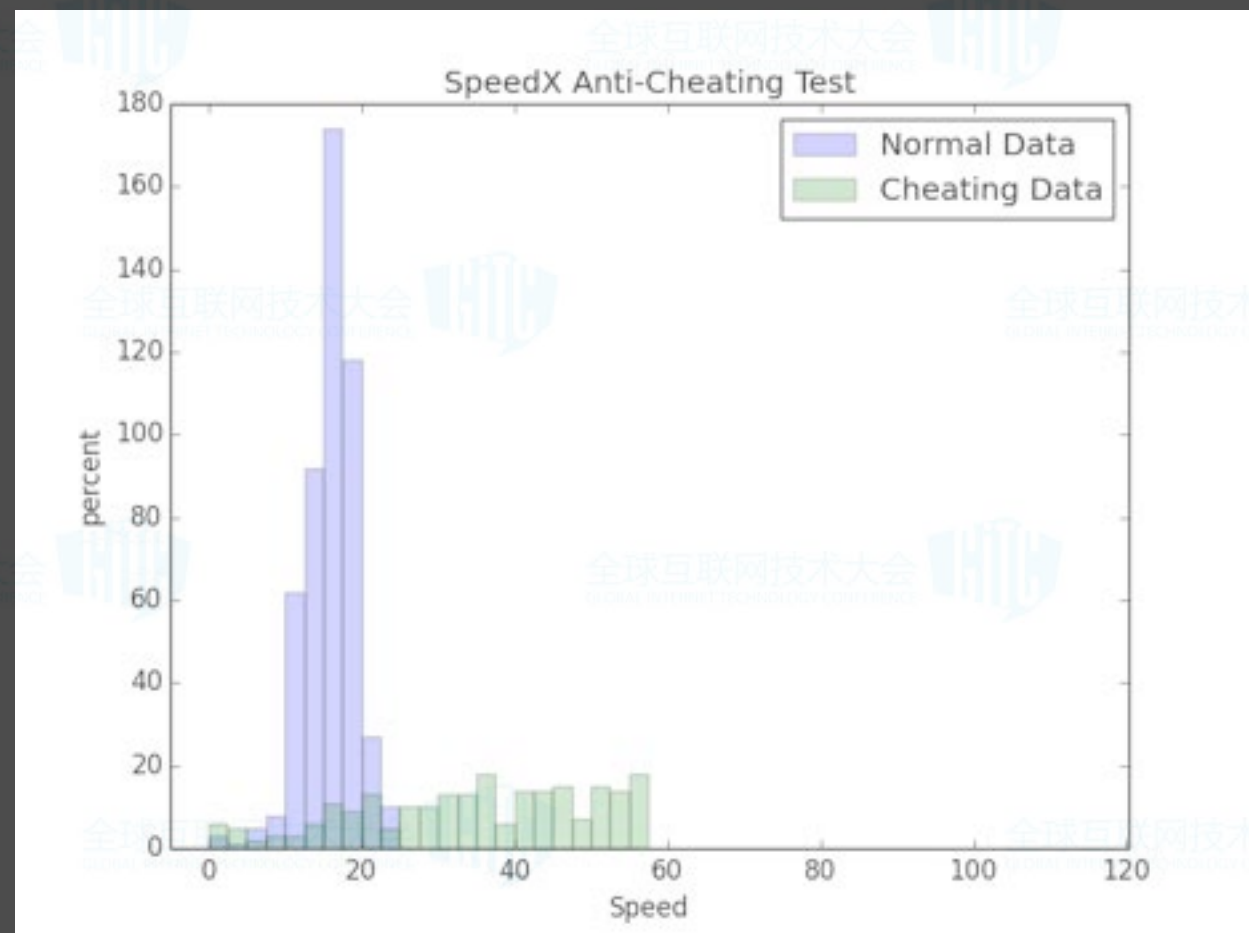
实时记录速度、里程、时间、卡路里、海拔等骑行数据，为你建立云端骑行中心，打造数字化的骑行生活。

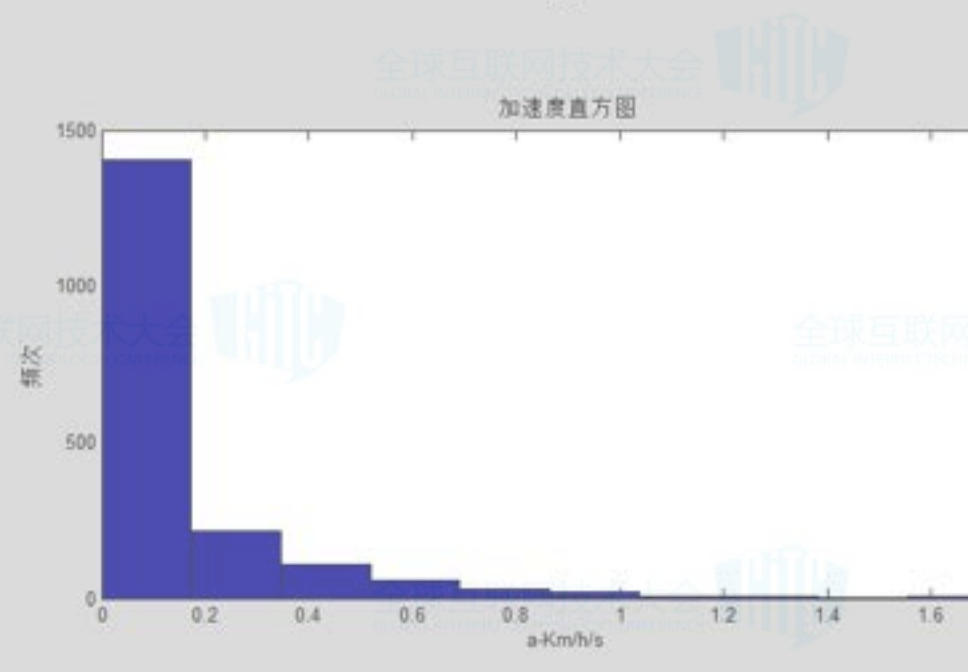
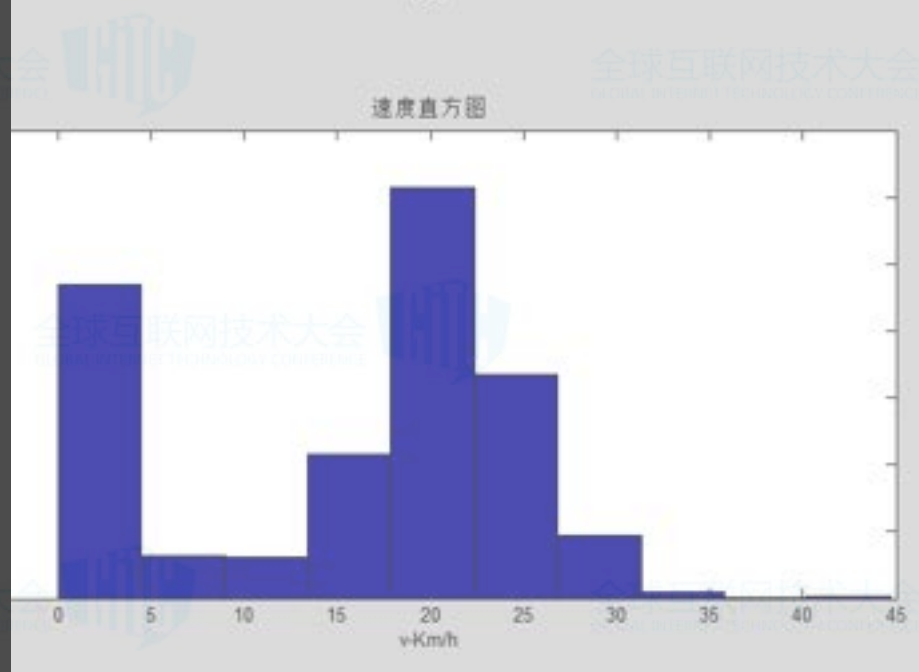
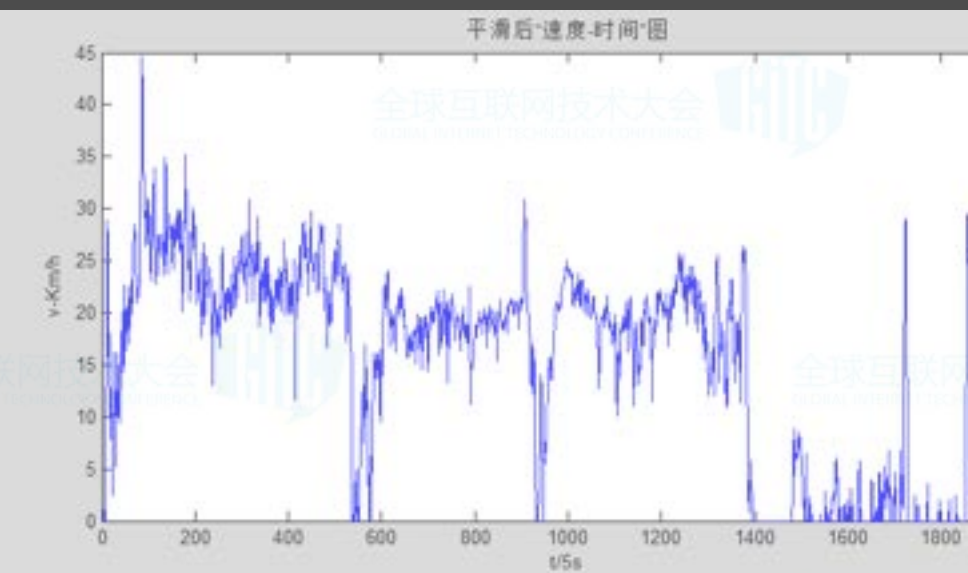
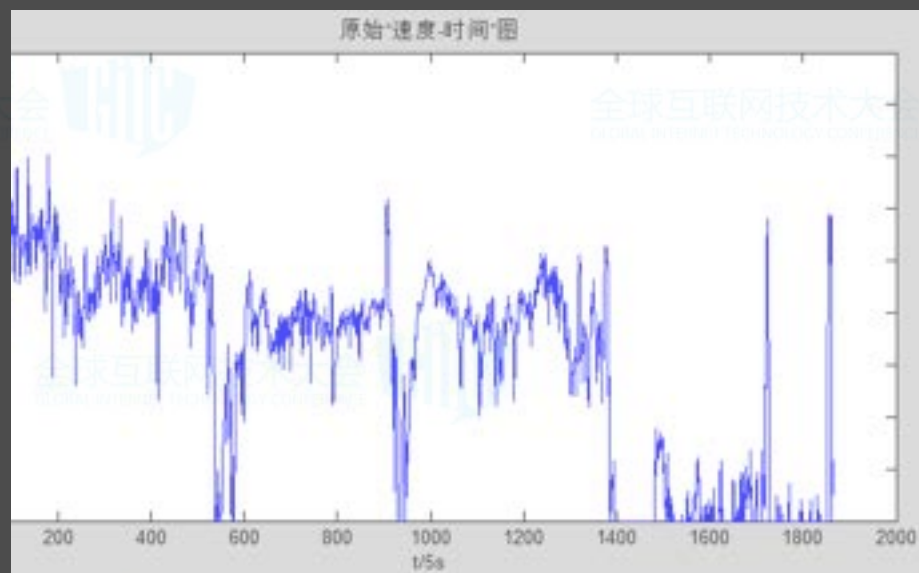
# 数据Matters

- 反作弊系统
- 俱乐部推荐
- 赛段系统

# 反作弊系统

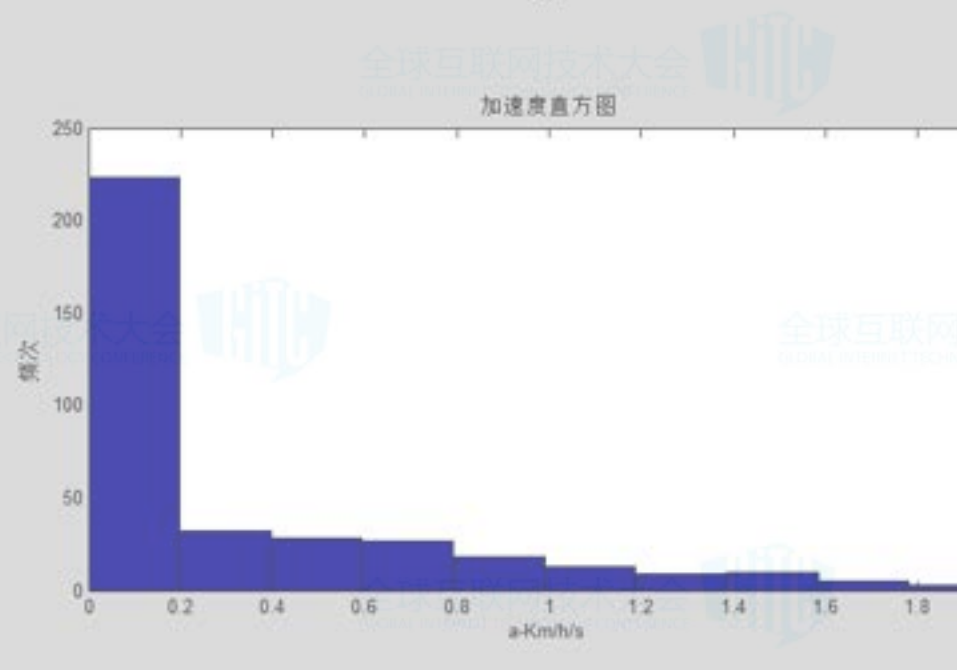
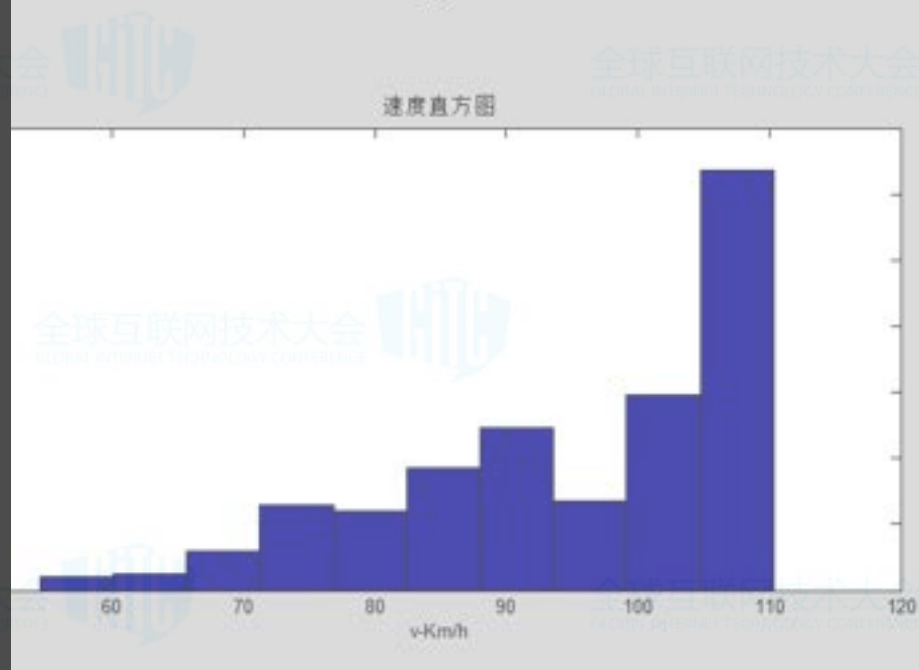
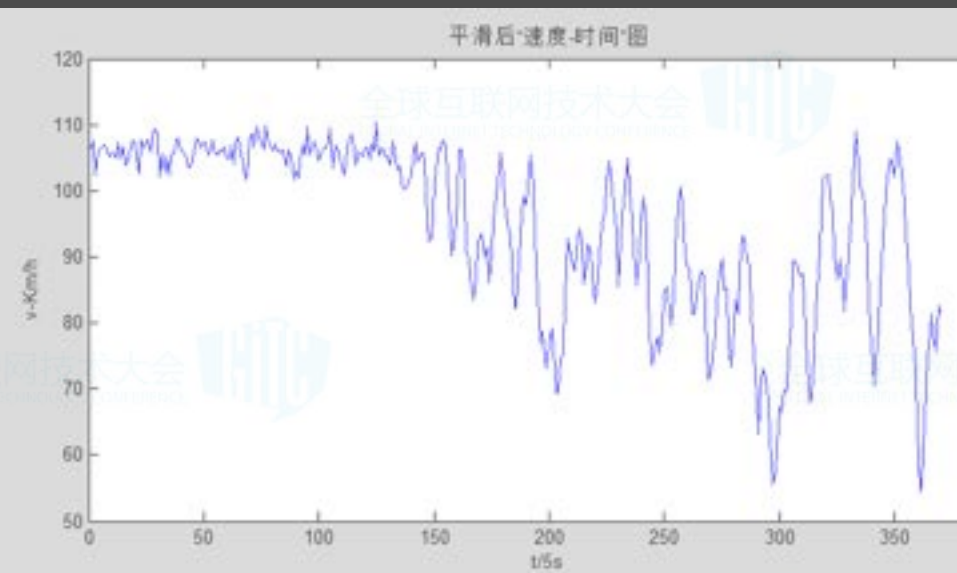
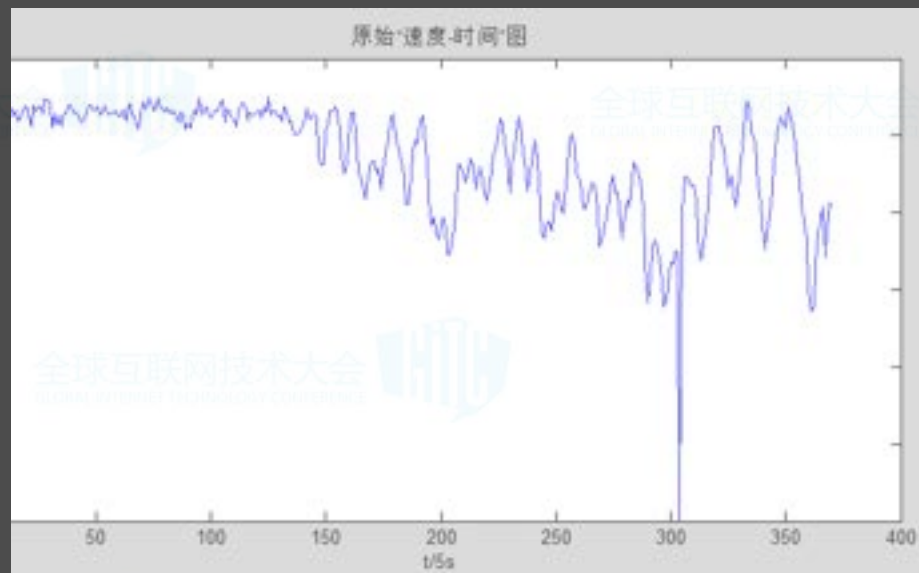
- 速度直方图
- 加速度直方图
- 标准化
- 神经网络训练
- 特殊样本优化



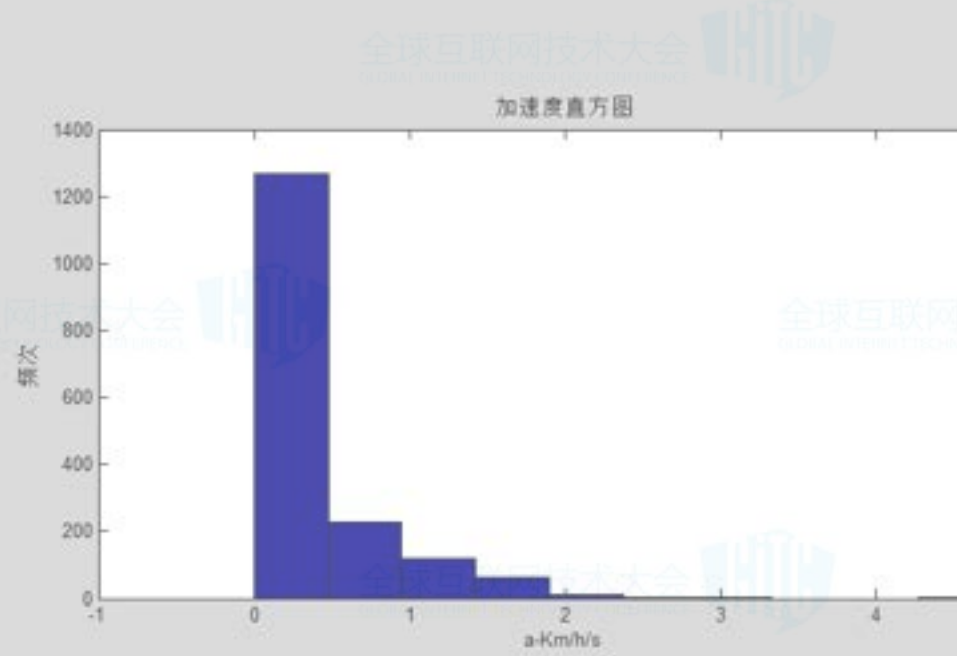
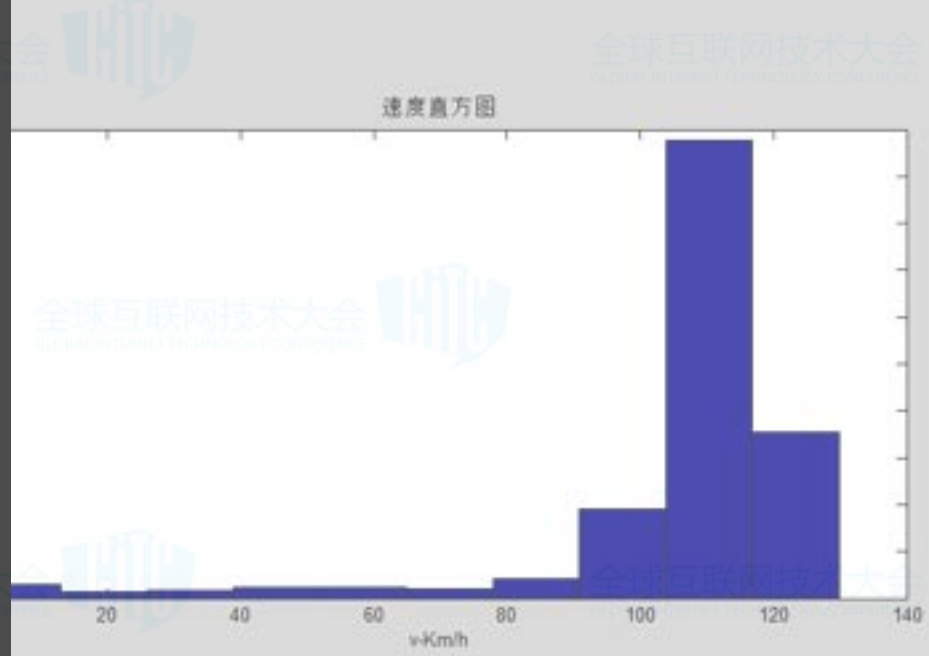
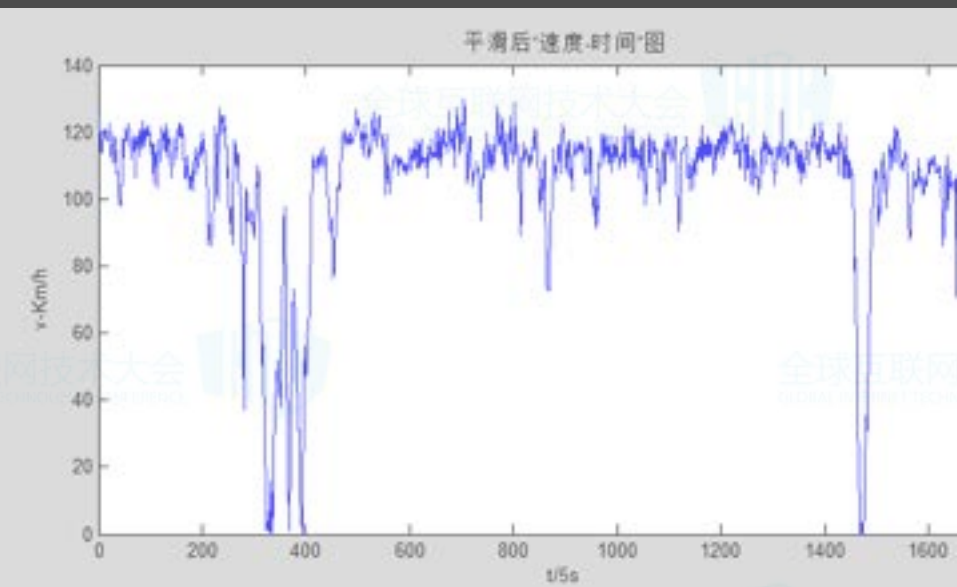
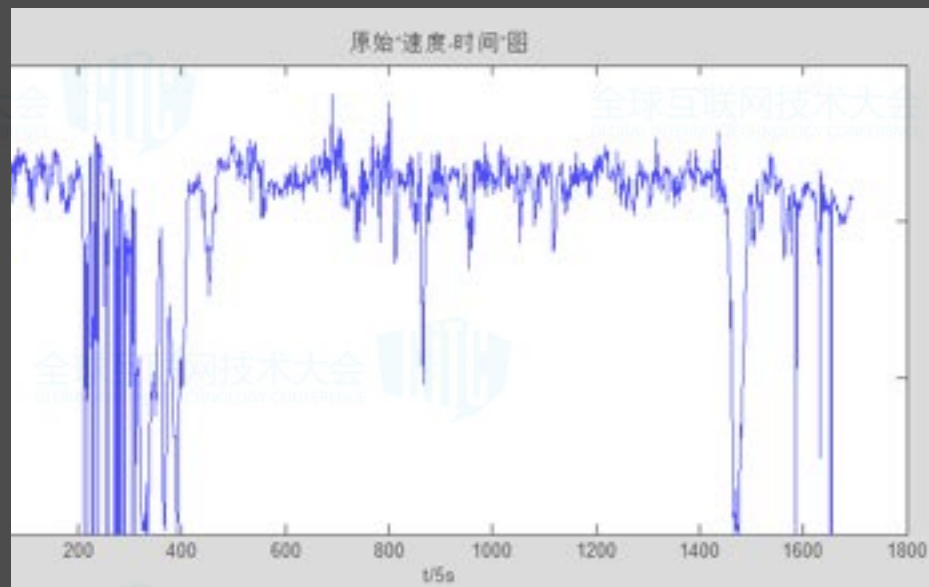


# 正常数据分布

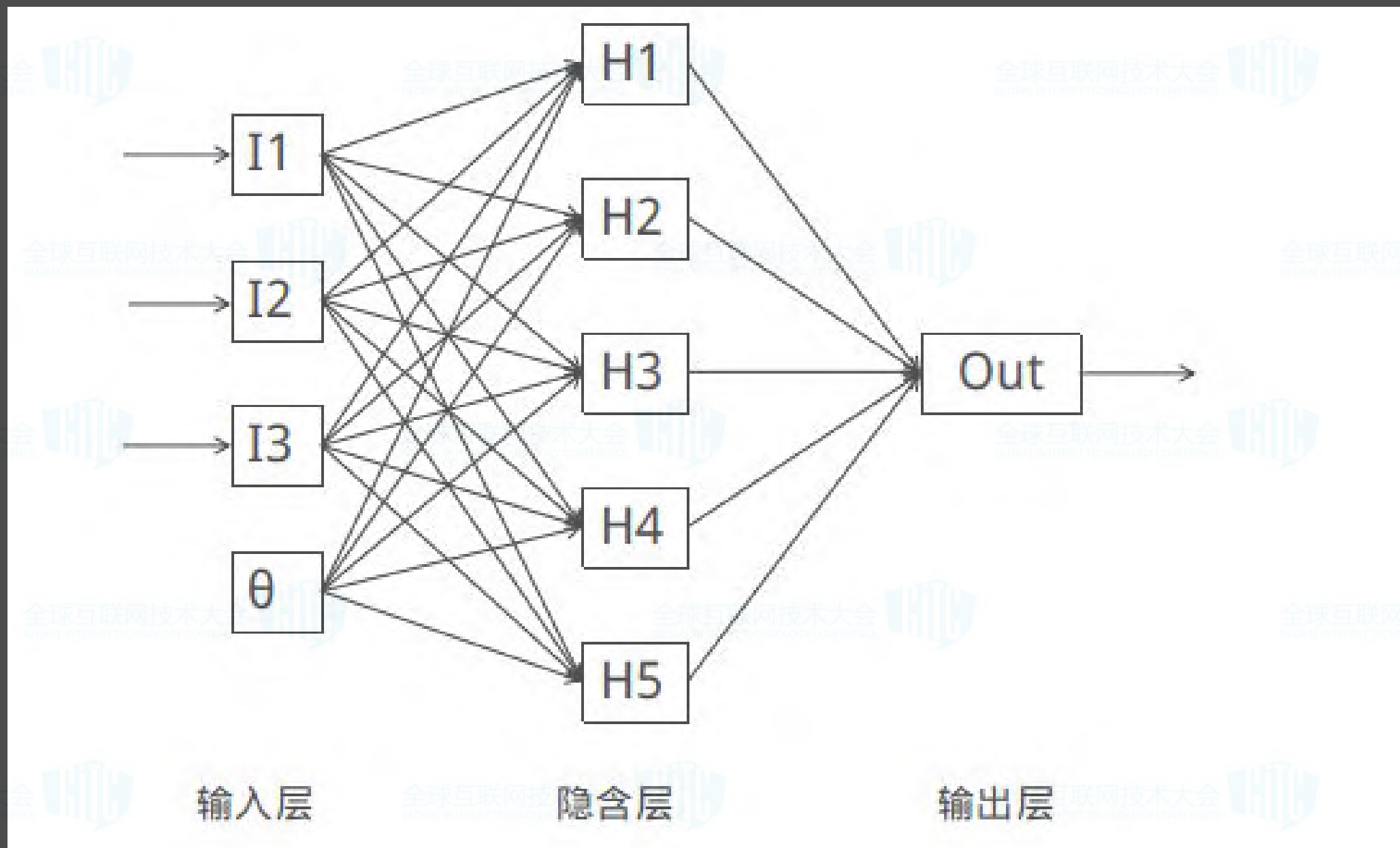




# 异常数据分布



# 异常数据分布



# 神经网络学习特征

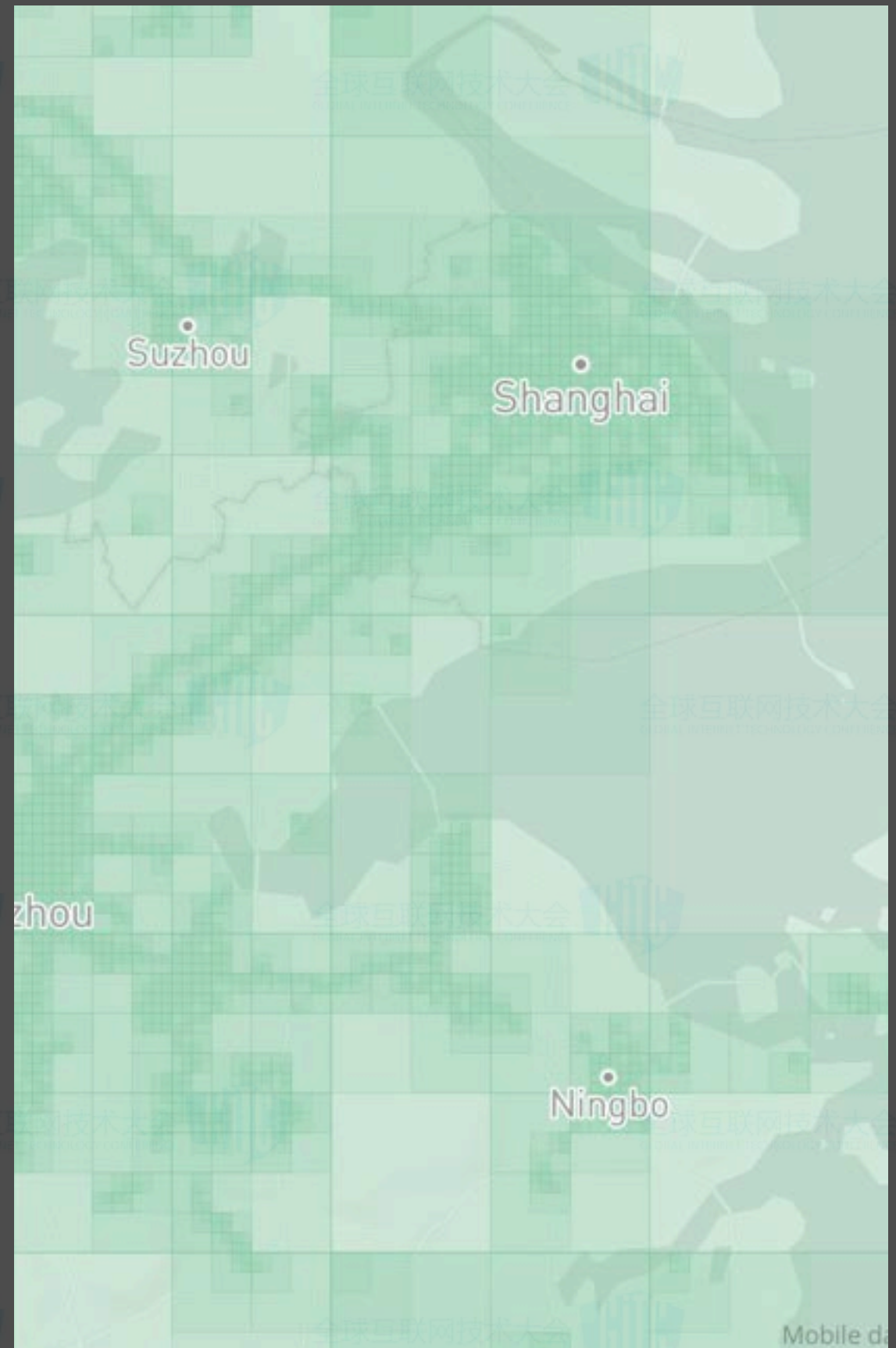
# 俱乐部推荐

- 格子计算系统
- 附近活跃的才是最重要的



# 格子计算系统

- 解决
  - 点和点之间之间不容易匹配
  - 路线数据量大，查询计算复杂



# 赛段推荐

- 格子计算系统
- GeoHash
- 正则匹配

# 地理数据存储

- PostgreSQL
- 路线数据无损压缩
- 路线数据有损压缩

# PostgreSQL PostGIS

- 传统方案
  - MySQL
  - GeoHash 基于MySQL



# MySQL

- `SELECT id, ( 6371 * acos( cos( radians(37) ) *  
cos( radians( lat ) ) * cos( radians( lng ) -  
radians(-122) ) + sin( radians(37) ) *  
sin( radians( lat ) ) ) ) AS distance FROM  
places HAVING distance < 25 ORDER BY  
distance LIMIT 0 , 100;`
- `SELECT * FROM places WHERE ((lat  
BETWEEN ? AND ?) AND (lng BETWEEN ?  
AND ?));`

# GeoHash

- 是一种地址编码，通过切分地图区域为小方块（切分次数越多，精度越高），它能把二维的经纬度编码成一维的字符串。也就是说，理论上geohash字符串表示的并不是一个点，而是一个矩形区域，只要矩形区域足够小，达到所需精度即可
- 如：wtw3ued9m就是目前我所在的位置，降低一些精度，就会是wtw3ued，再降低一些精度，就会是wtw3u。（[点击查看坐标编码对应Google地图的位置](#)）
- 所以这样一来，我们就可以在MySQL中用LIKE 'wtw3u%'来限定区域范围查询目标点，并且可以对结果集做缓存。更不会因为微小的经纬度变化而无法用上数据库的Query Cache。

# PostgreSQL

- 性能卓越
- 方便的地理信息查询
- 大数据查询瓶颈高
- 百万级以上数据首选

10W	20W	50W	100W	260W	1000w
1.2ms	1.4ms	1.8ms	2.5ms	4.2ms	15.3ms

# 路线无损压缩

## Encoded Polyline Algorithm

- (39.99682, 116.47880)
- (39.99318, 116.47471)
- (39.99466, 116.47248)
- (39.99262, 116.47055)





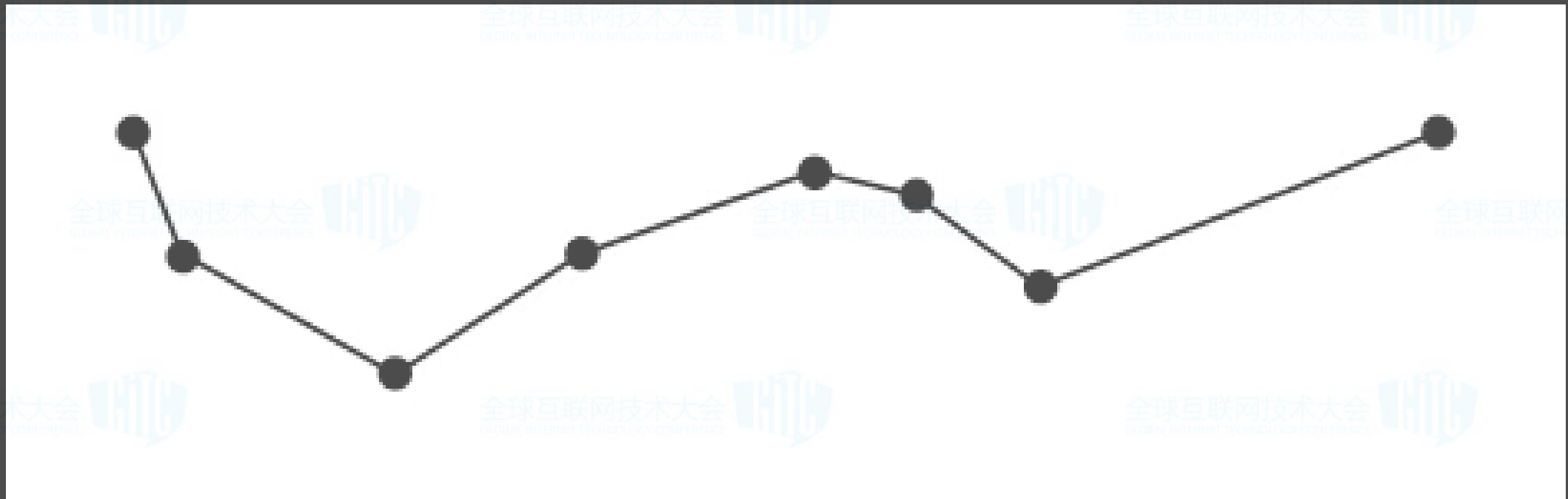
# 路线无损压缩

## Encoded Polyline Algorithm

- (39.99682,116.47880)(39.99318,116.47471)  
(39.99466,116.47248)(39.99262,116.47055)
- c{bsFow|eUvUpXgH|LvK`K
- 81 to 22 (73%)

# 路线有损压缩

## Ramer Douglas Peucker Algorithm



- Complexity  $\Theta(n^2)$

- [https://en.wikipedia.org/wiki/Ramer%E2%80%93Douglas%E2%80%93Peucker\\_algorithm](https://en.wikipedia.org/wiki/Ramer%E2%80%93Douglas%E2%80%93Peucker_algorithm)

# 路线有损压缩效果

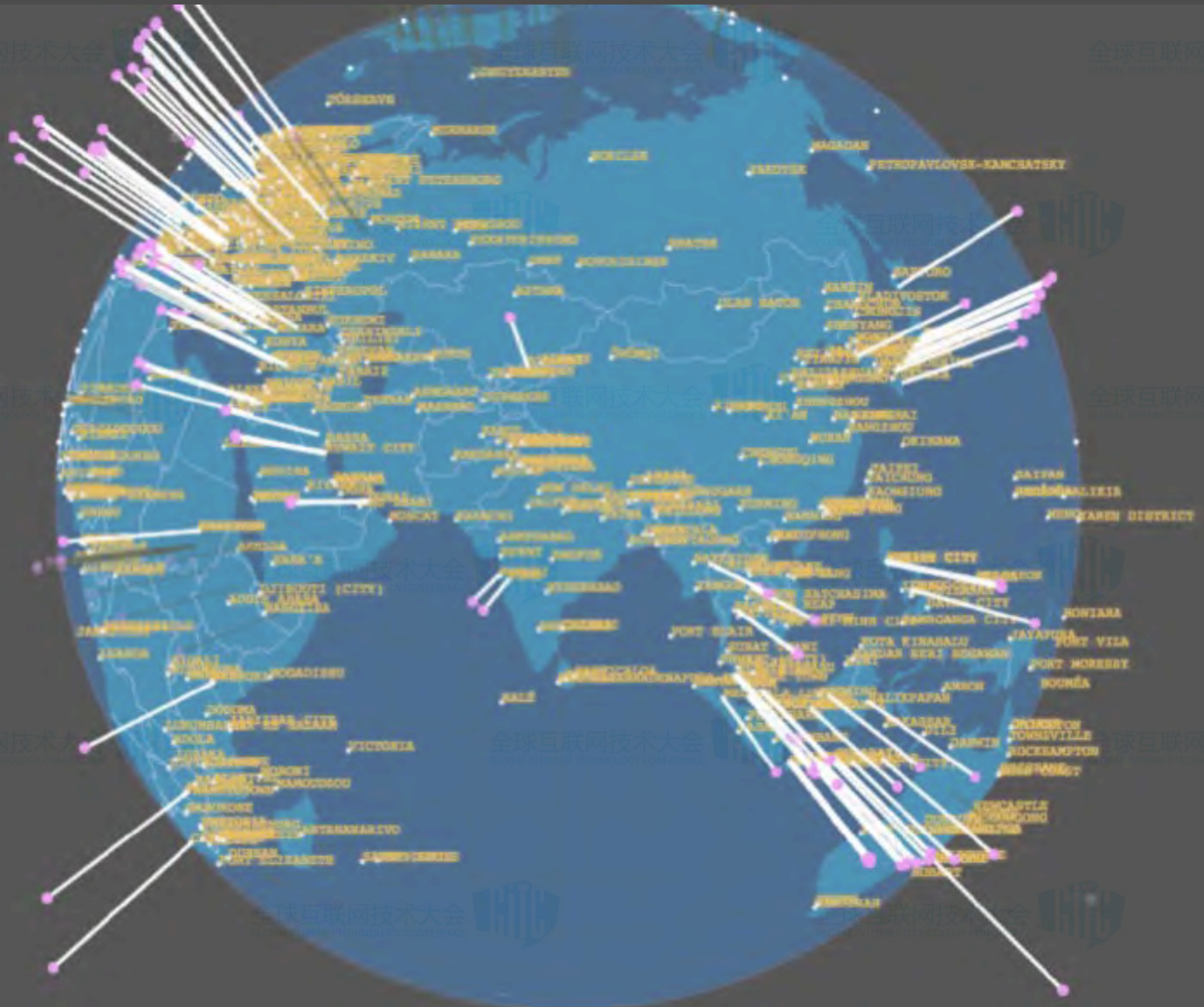


# 数据可视化

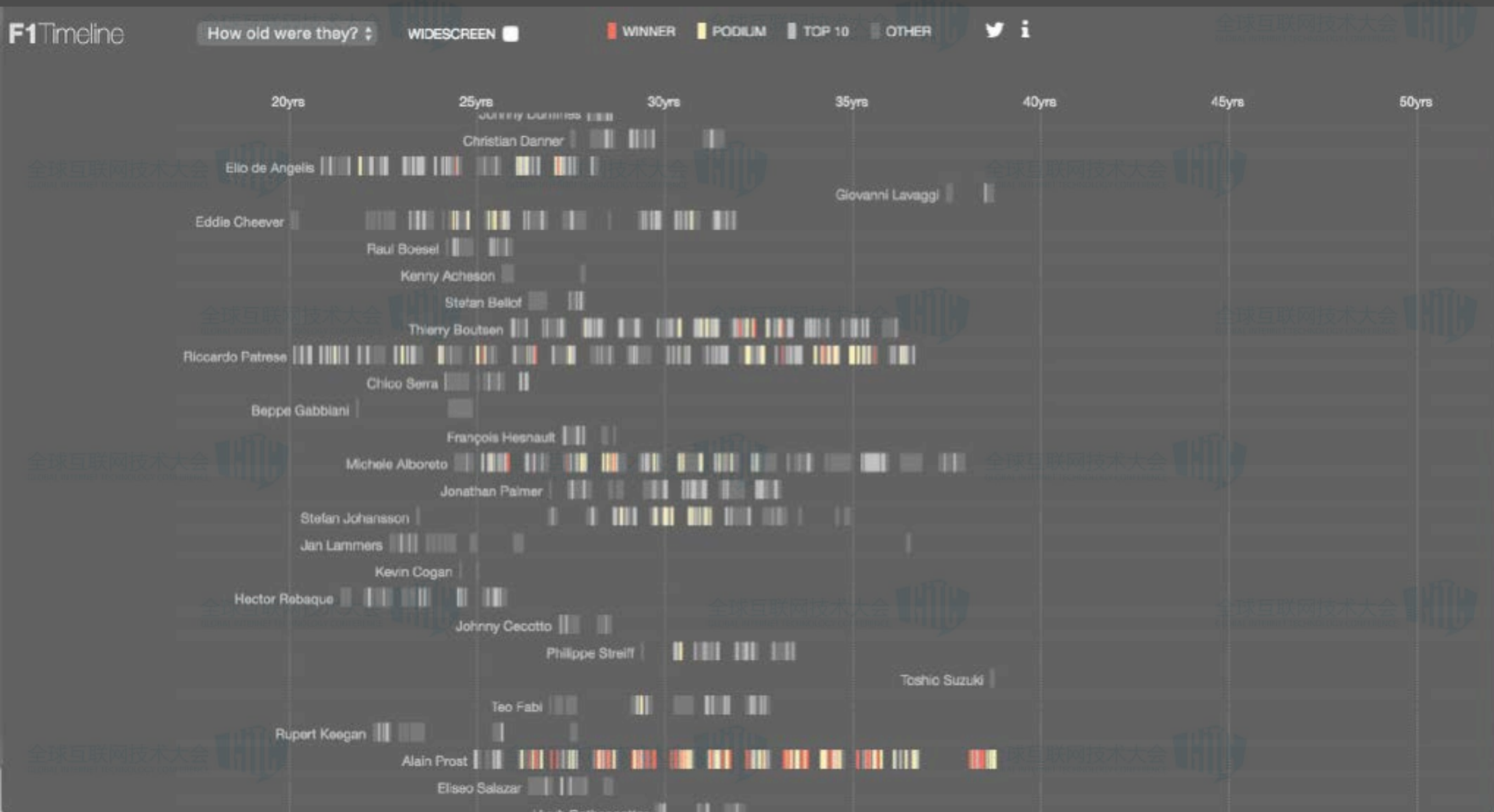
- JS工具
- 自建 Python Matplotlib
- Mapbox



# D3.js



# D3.js





# D3.js

## UK Temperature History

Explore the temperature in the United Kingdom since 1910 by scrolling the page up and down. Visit years marked in red or blue that have significant weather events. You can also sort by year, maximum, minimum or mean temperature

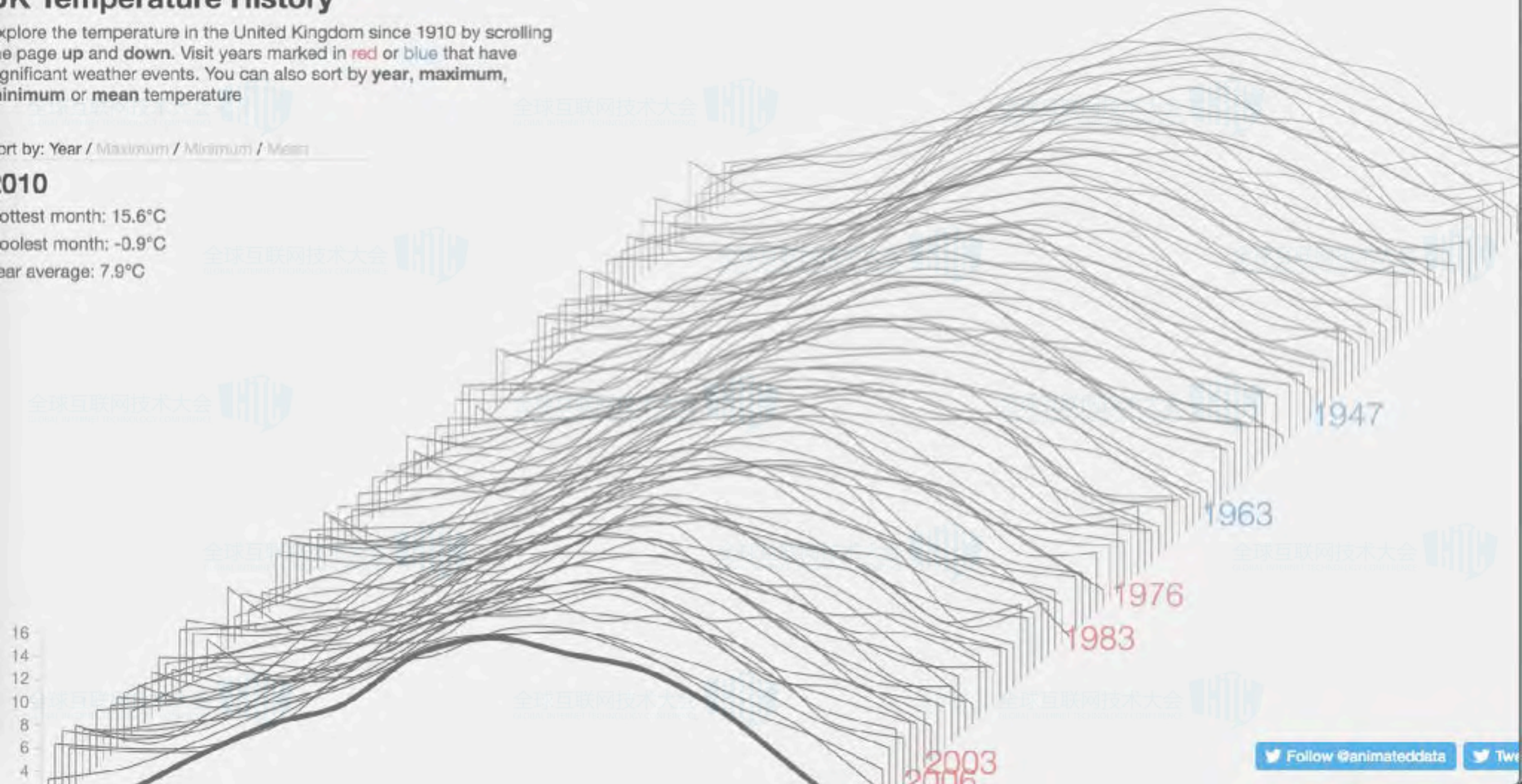
Sort by: Year / Maximum / Minimum / Mean

2010

Hottest month: 15.6°C

Coolest month: -0.9°C

Year average: 7.9°C



# D3.js

## Animated UK Wind Chart


This chart shows wind speed and direction in the UK. The more the wind blows, the faster the lines go!

The original inspiration came from the beautiful [Wind Map](#) by hint.fm. Weather data is from the Met Office's [DataPoint](#) service and the UK map was taken from Mike Bostock's [mapping tutorial](#).

Although real data, this visualisation is just for demonstration.

Built by [Peter Cook](#).

 Follow @animateddata

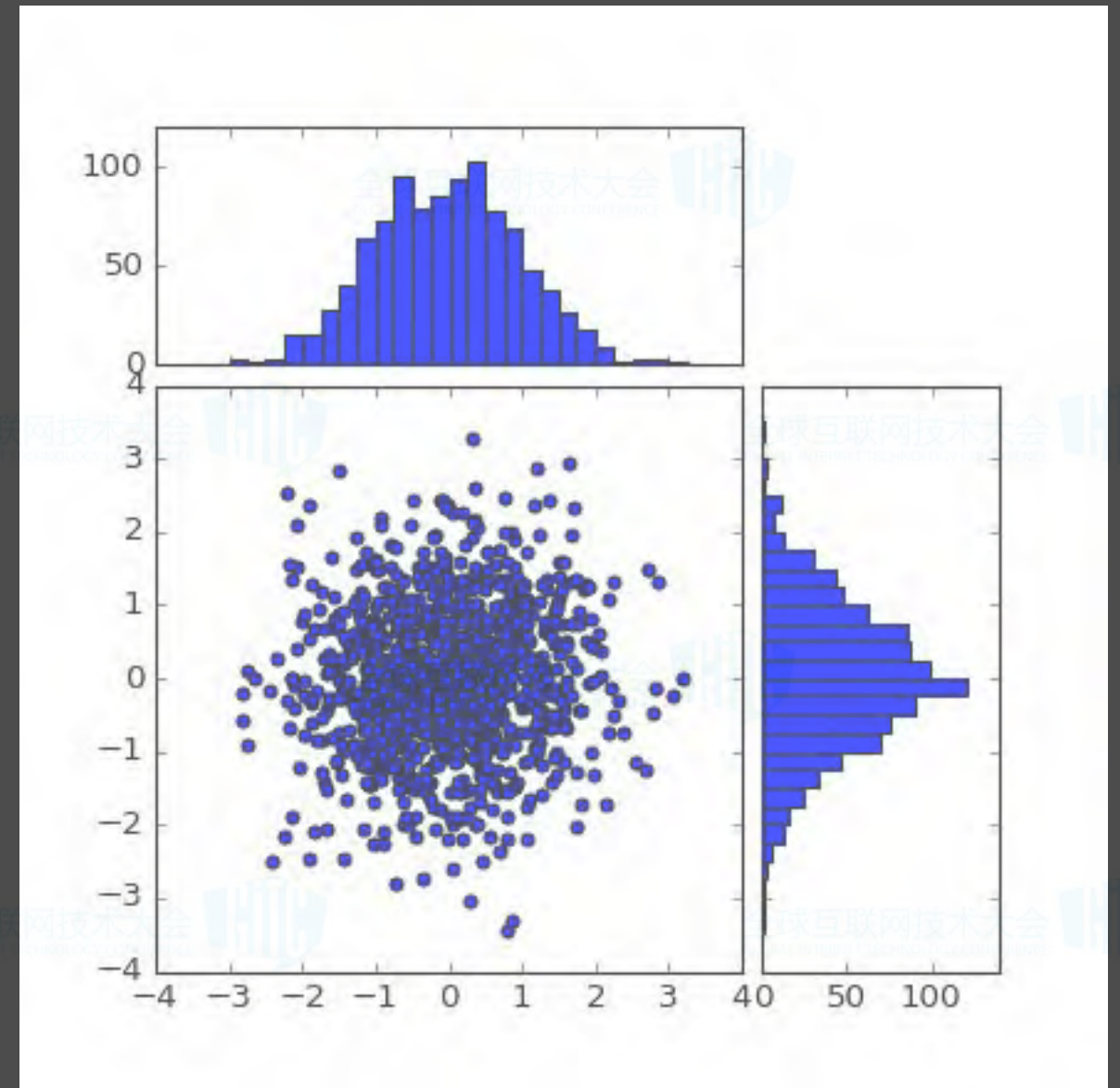
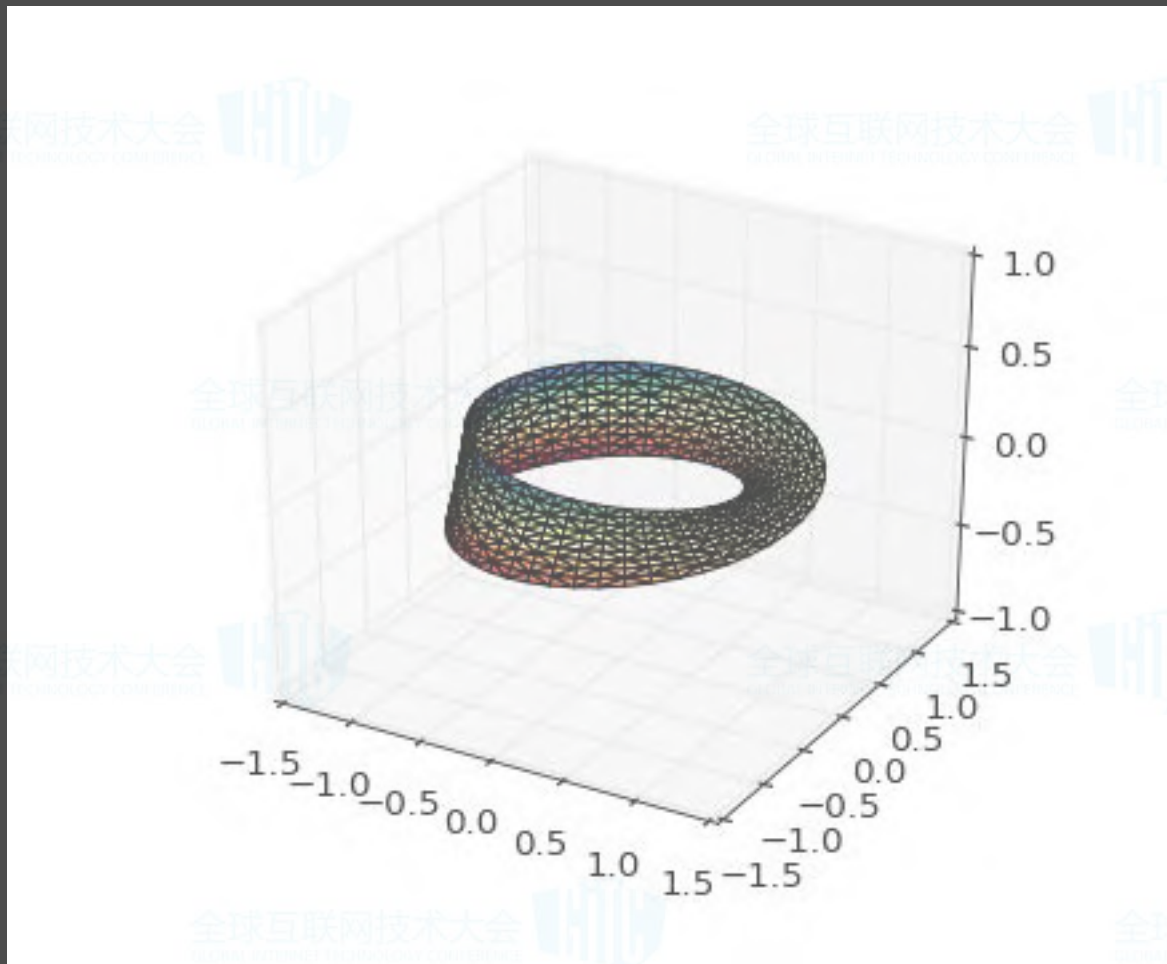
 Tweet

 Like 7

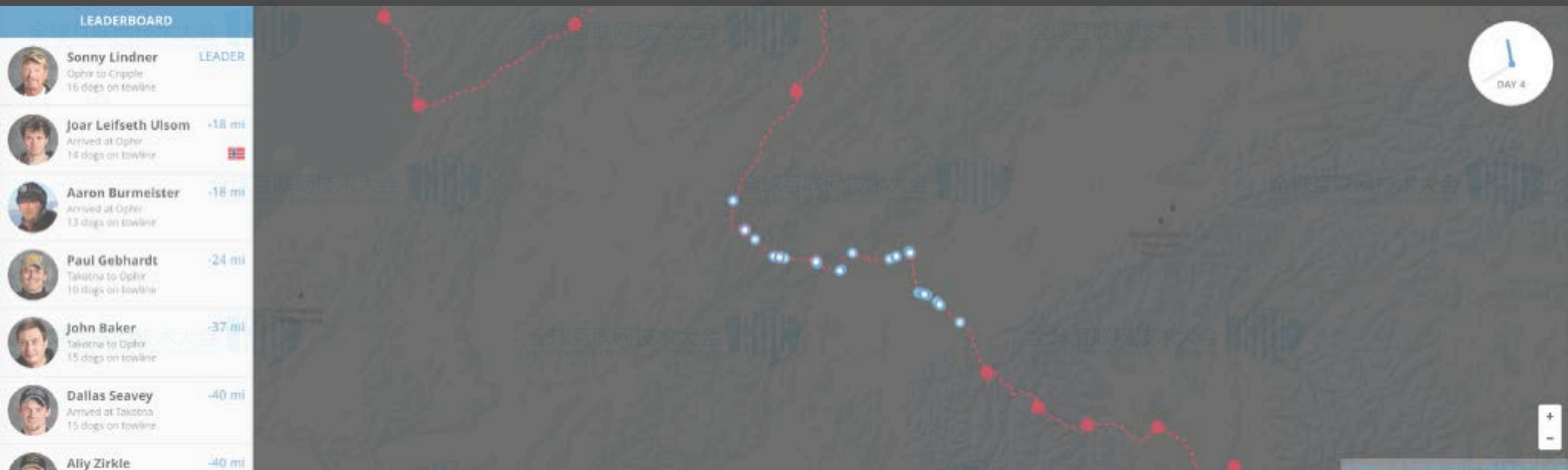




# Matplotlib



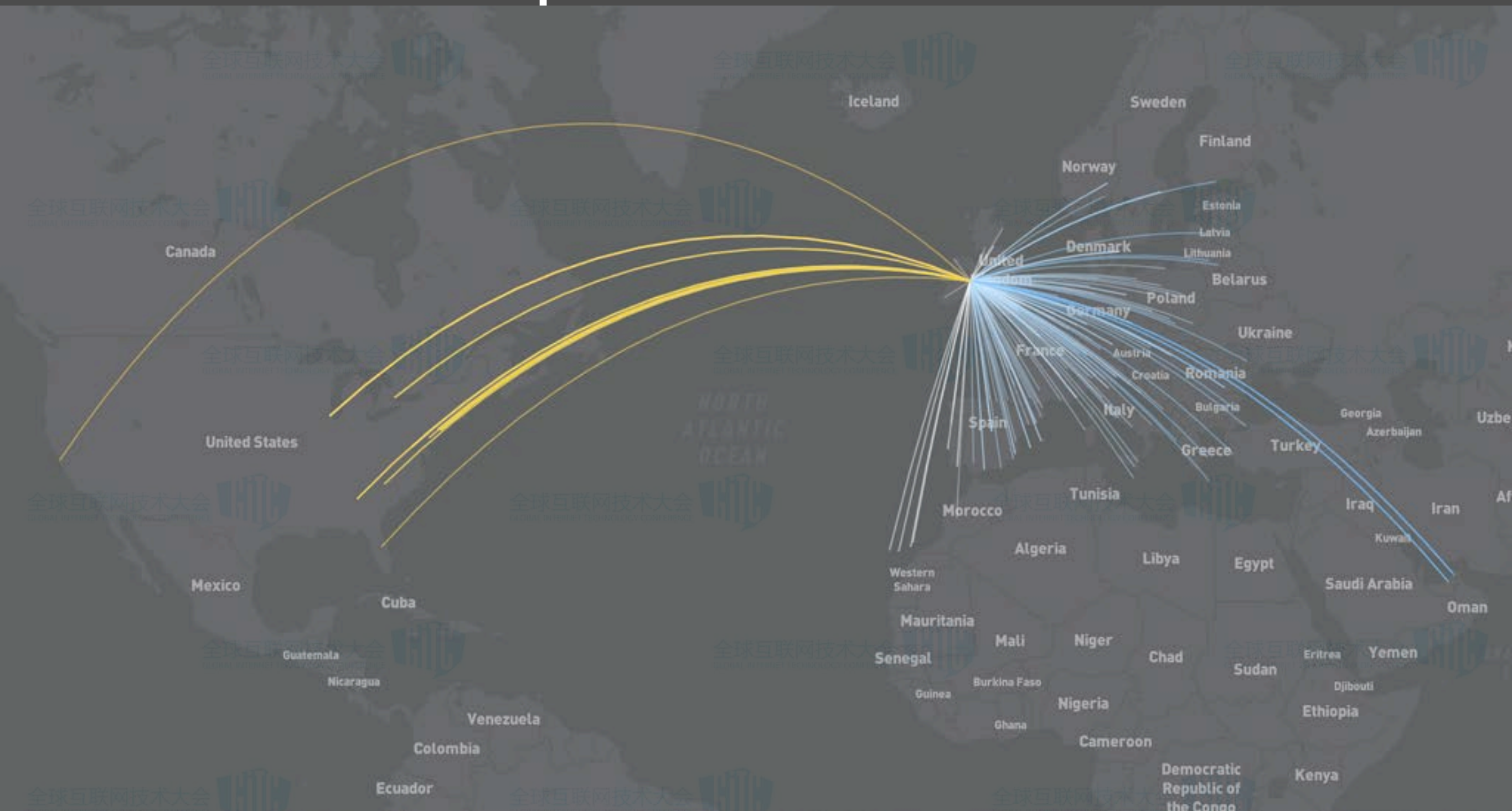
# Mapbox Demo



- <https://www.mapbox.com/blog/playback-the-iditarod-with-turf/>

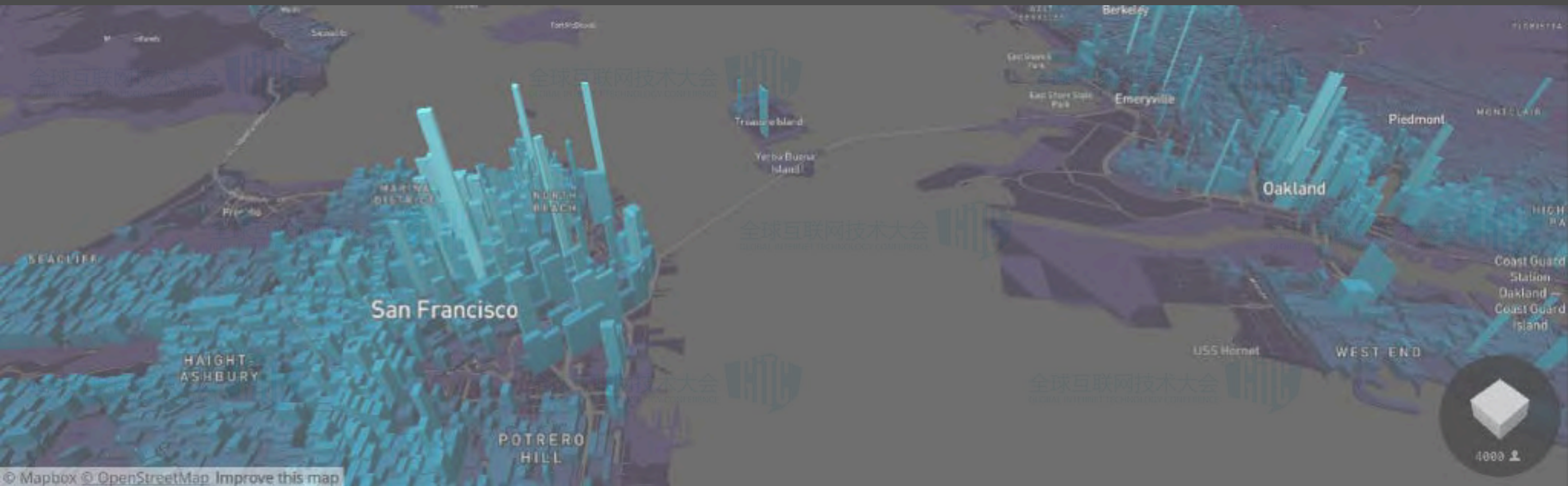


# Mapbox Demo



- <https://www.mapbox.com/blog/track-your-jetlag-with-data-driven-line-color/>

# Mapbox Demo



- <https://www.mapbox.com/blog/population-inspector/>