Python 数据可视化的应用与价值

杨洁坤

2016.9.10

目录

当我们提到数据可视化的时候，
我们在说些什么？

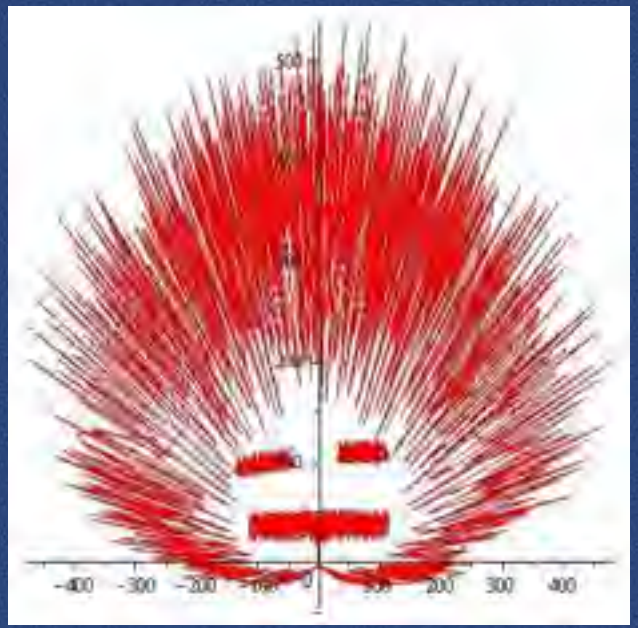

data visualization examples
GREAT EXAMPLES! #1
THE YEAR IN NEWS

或者，当一个艺术家.....

爱因斯坦 style

头发的原始函数

11 个函数的组合



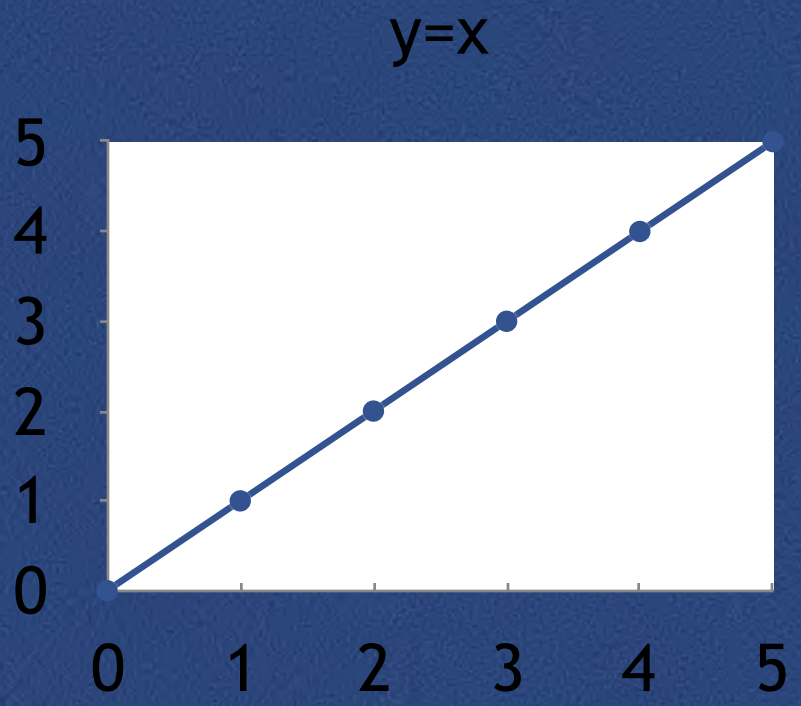
今天，我们聊算法

从 y=x 的函数图像开始

基础算法

中学时代
描点作图法

1. 取点，
2. 描点，
3. 连点成线。



基础算法

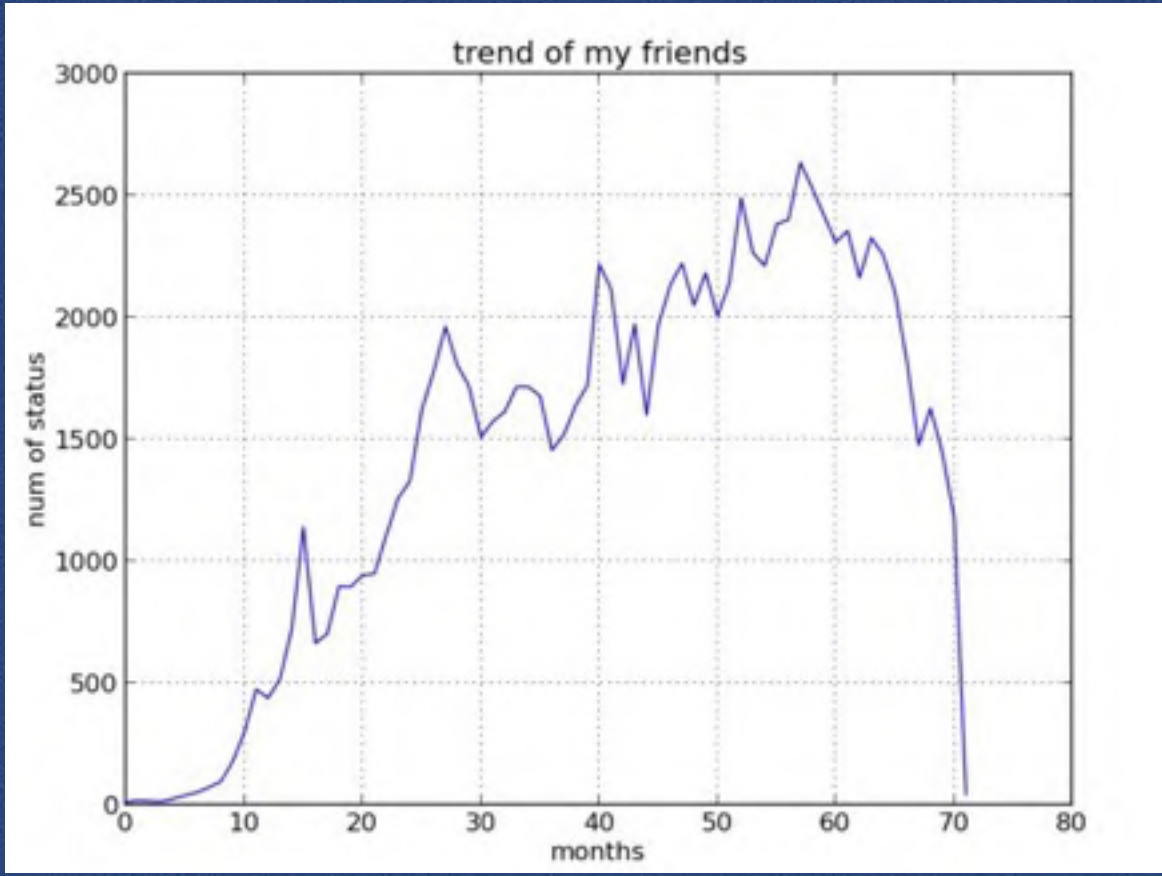
机器的算法
更加简单、粗暴.

大多数时候，中学的知识，就够了。

实践 1：SNS 网站，用户活跃度曲线

基础算法

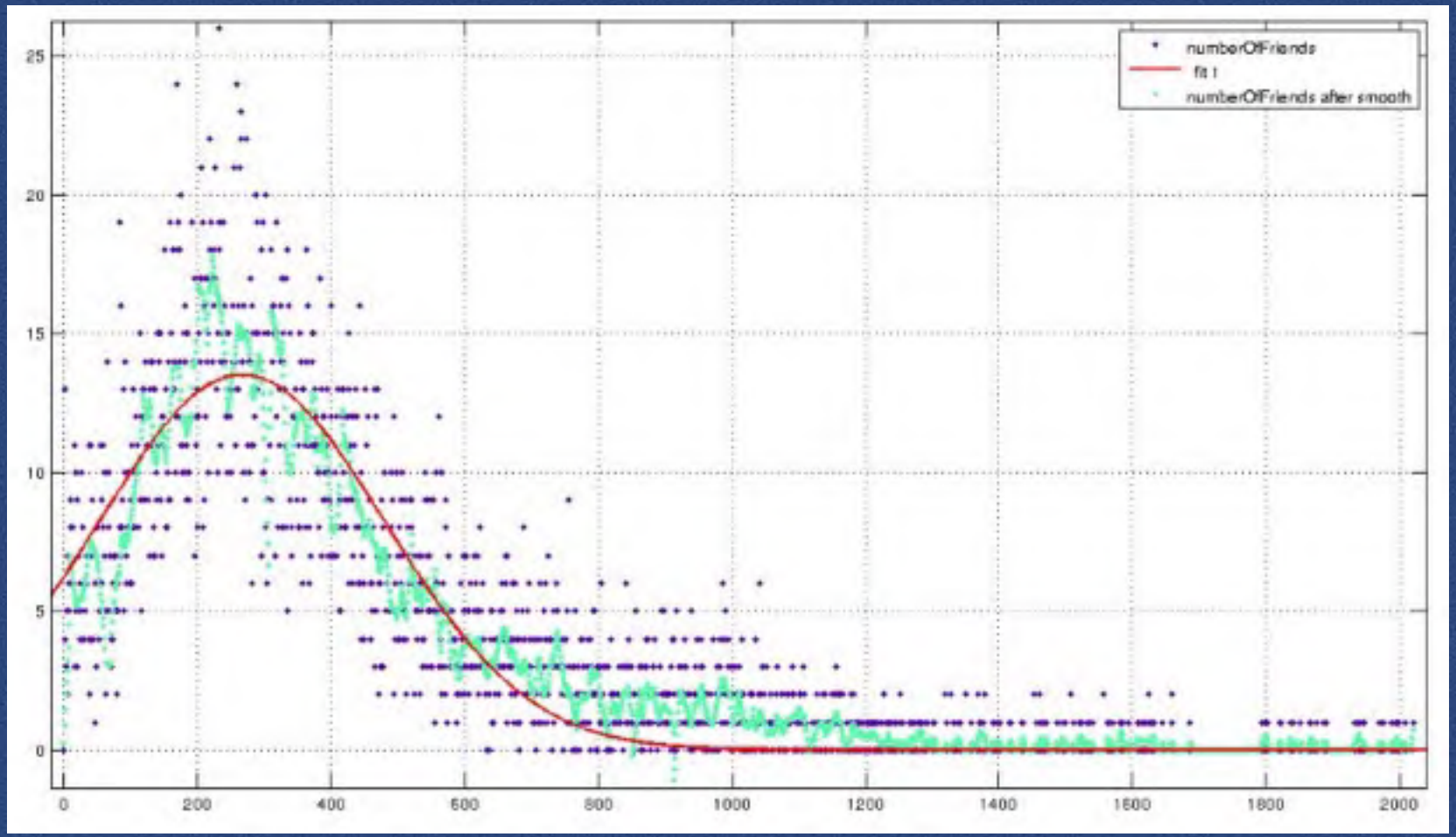
发布状态数



横坐标 60 处（单位：月）是毕业时间

滤波、拟合
——实践 2：好友数量分布曲线

基础算法



每个人的好友数量分布曲线

常见的滤波算法

Google
key words: 滤波算法

总结

1.滑窗式。n 个点一组，一组一组的滤。
2.计算最大差值，选择是否限幅。
3.返回中位数／平均值，实现消除扰动
4.加权，时间越靠后的，权重越高。

Case1: 性能监控曲线滤波

Case1: 性能监控

windows／mac／linux 任务管理器

1. 我们的系统真的很稳定
2. 峰值／波动／微扰，统统过滤掉

性能监控的目标

1. 曲线平滑。
2. 噪声性的扰动，过滤掉。
3. 相对较大的脉冲，保留。
4. 峰值的绝对数字，不失真。

talk is cheap, show me your code

Case1: 性能监控

```
15 def keep_peak(orig_data, step=7, peak_range=5):
16     """wave filtration
17
18     if max - min <= peak_range:
19         return [middle, middle + 1]
20     else
21         return [max, min]
22     """
23
24     res = []
25     for i in range(0, len(orig_data), step):
26         cut = orig_data[i: i + step]
27         cut.sort()
28         if float(cut[-1]) - float(cut[0]) < peak_range:
29             # small delta, take 2 middle values.
30             n = int(step / 2)
31             res.extend(cut[n - 1: n + 1])
32         else:
33             # keep the peak, keep the order
34             ordered = orig_data[i: i + step]
35             if ordered.index(cut[0]) < ordered.index(cut[-1]):
36                 res.extend([cut[0], cut[-1]])
37             else:
38                 res.extend([cut[-1], cut[0]])
39     return res
40
```

收益

1. 帮助测试组有证据的版本打回，版本延期 1 个月发布。
2. 某数据分析项目。加速了架构转型至 map-reduce。

Case2: 赚一点钱——彩票的 k 线图

这是一个充满误解的行业

彩票不就是
2 块钱中 500 万嘛

古典概型的
随机事件嘛！

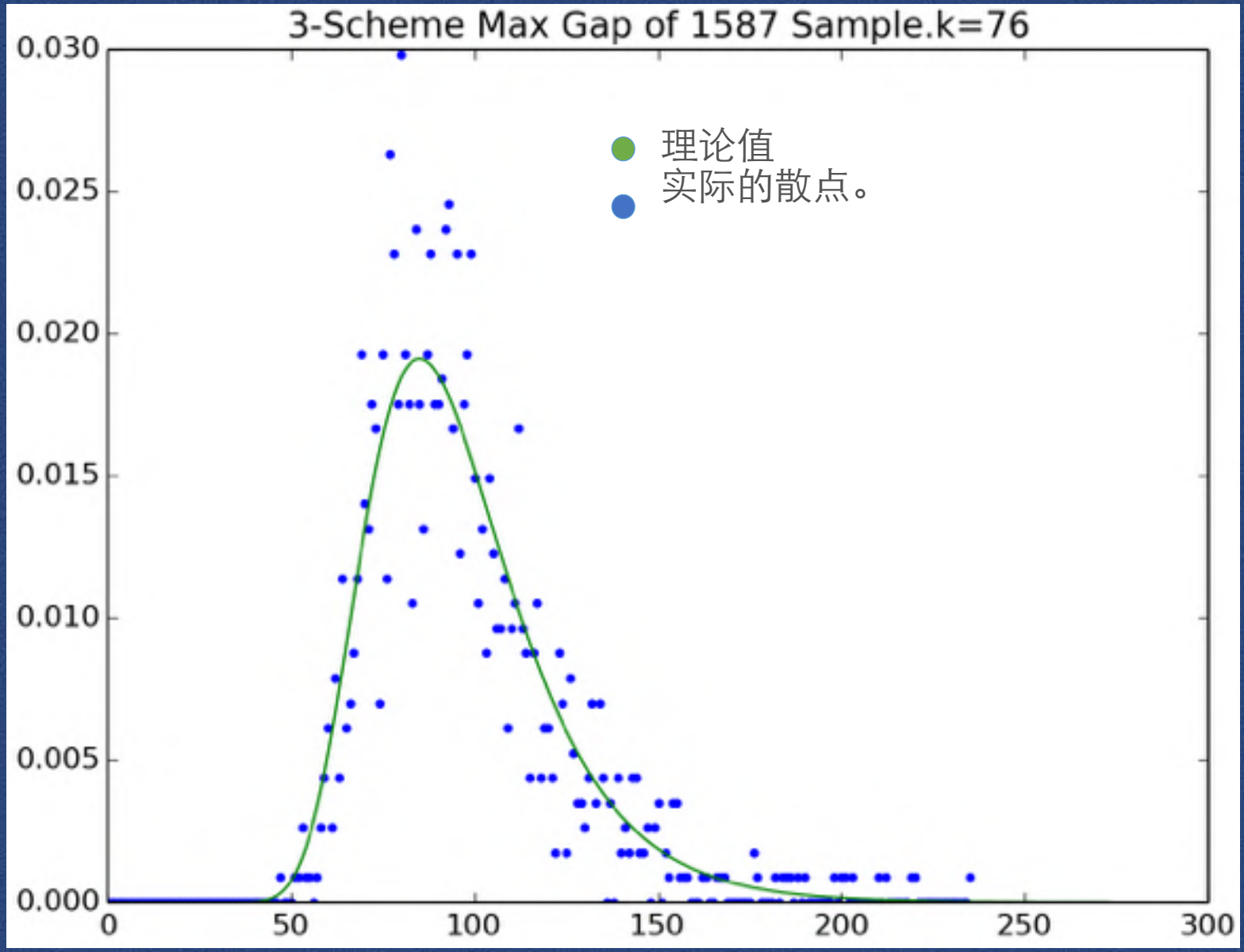
Case2: 彩票 K 线图

1.一般，单注奖金 < 1000 RMB
2.号版
3.心理学博弈
4.翻倍买，直到中奖，不会赔钱

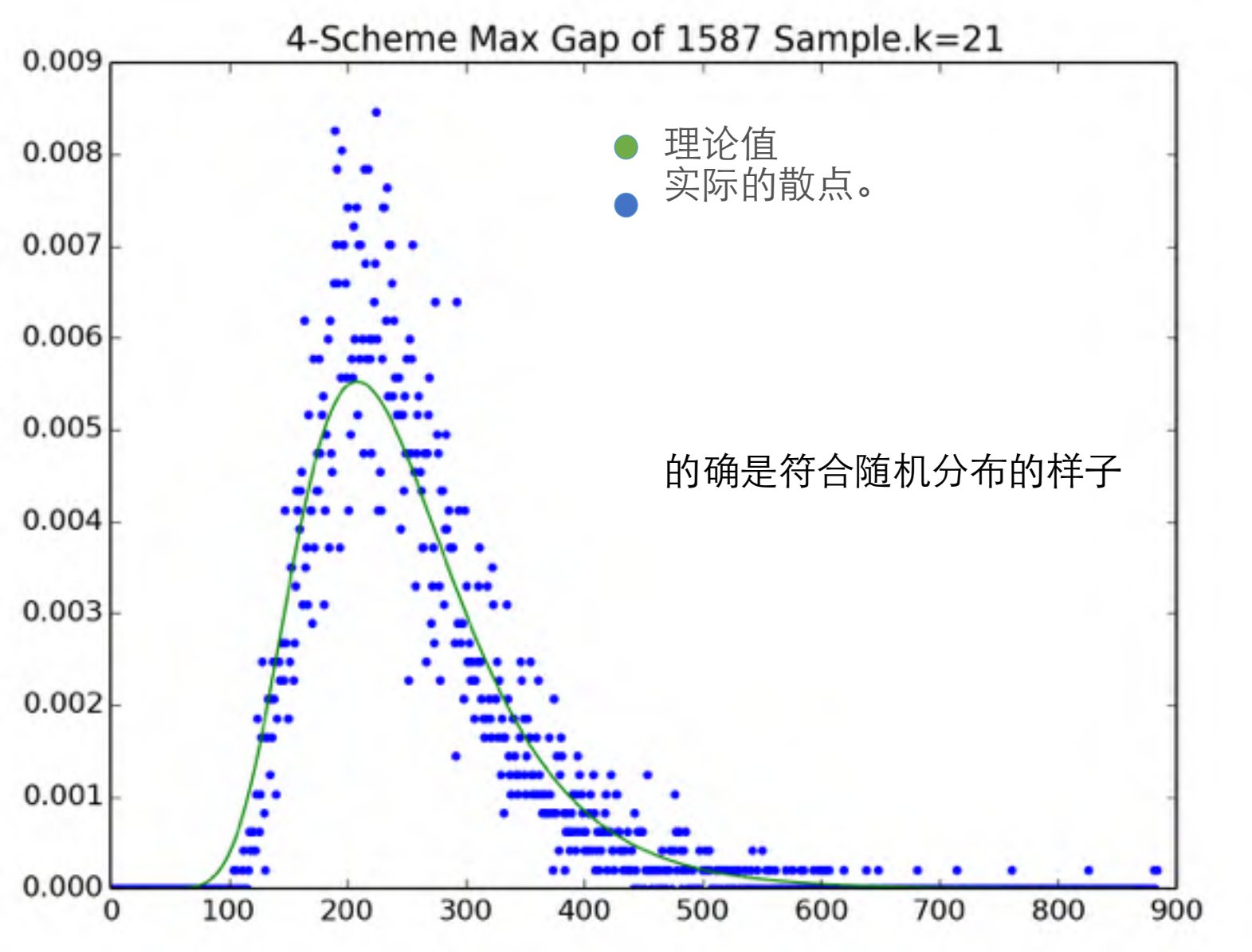
Case2: 彩票 K 线图

符合随机分布理论么?

Case2: 彩票 K 线图

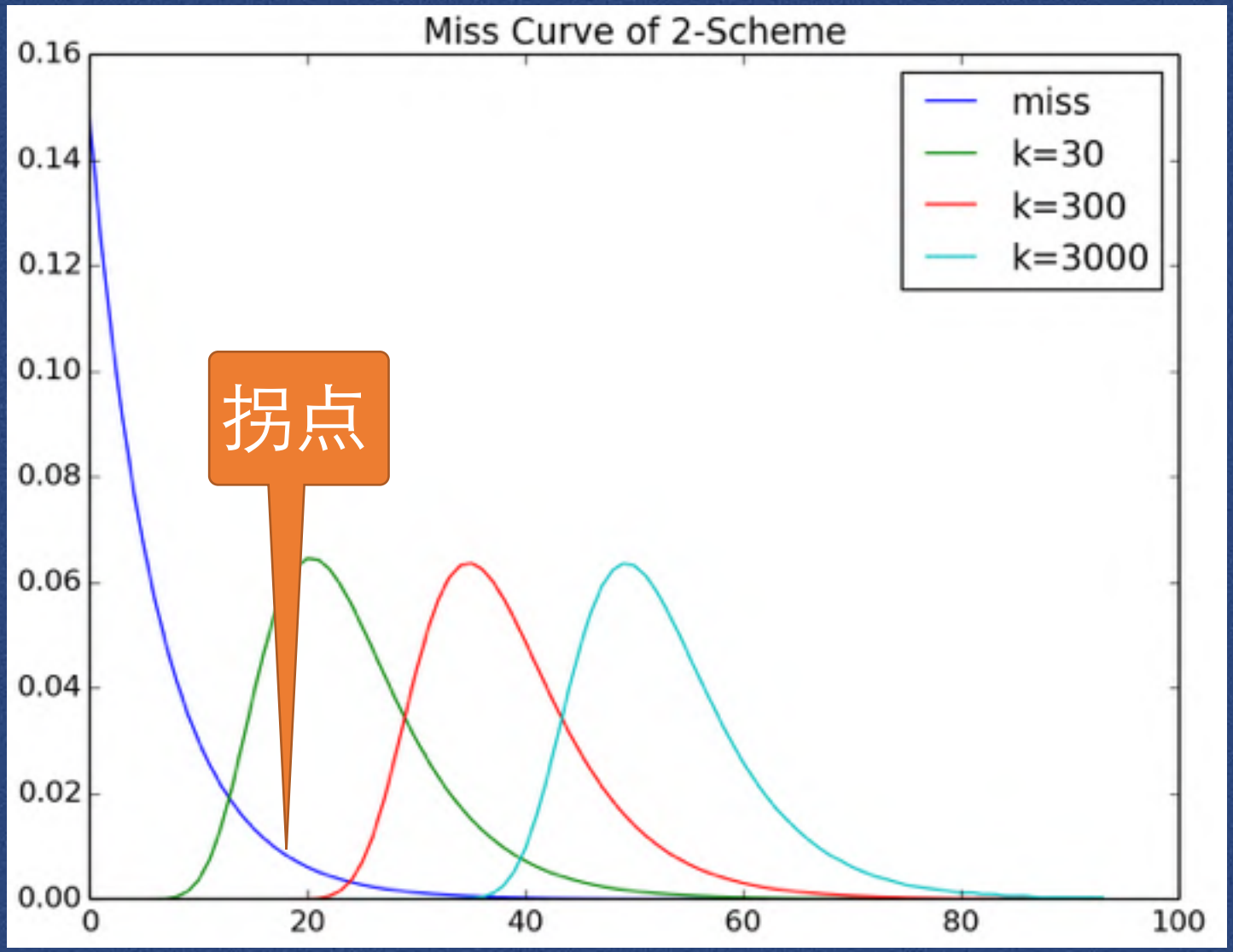

3-Scheme Max Gap of 1587 Sample.k=76
理论值
实际的散点。
0.030
0.025
0.020
0.015
0.010
0.005
0.000
0
50
100
150
200
250
300

Case2: 彩票 K 线图



的确是符合随机分布的样子

Case2: 彩票 K 线图



遗漏分布曲线

选号算法

1. 中长期出现频率在中位数附近
2. 长期热号会变冷
3. 长期冷号，这都是命
4. 普通彩民不会关注的指标

Case2: 彩票 K 线图

看图说话

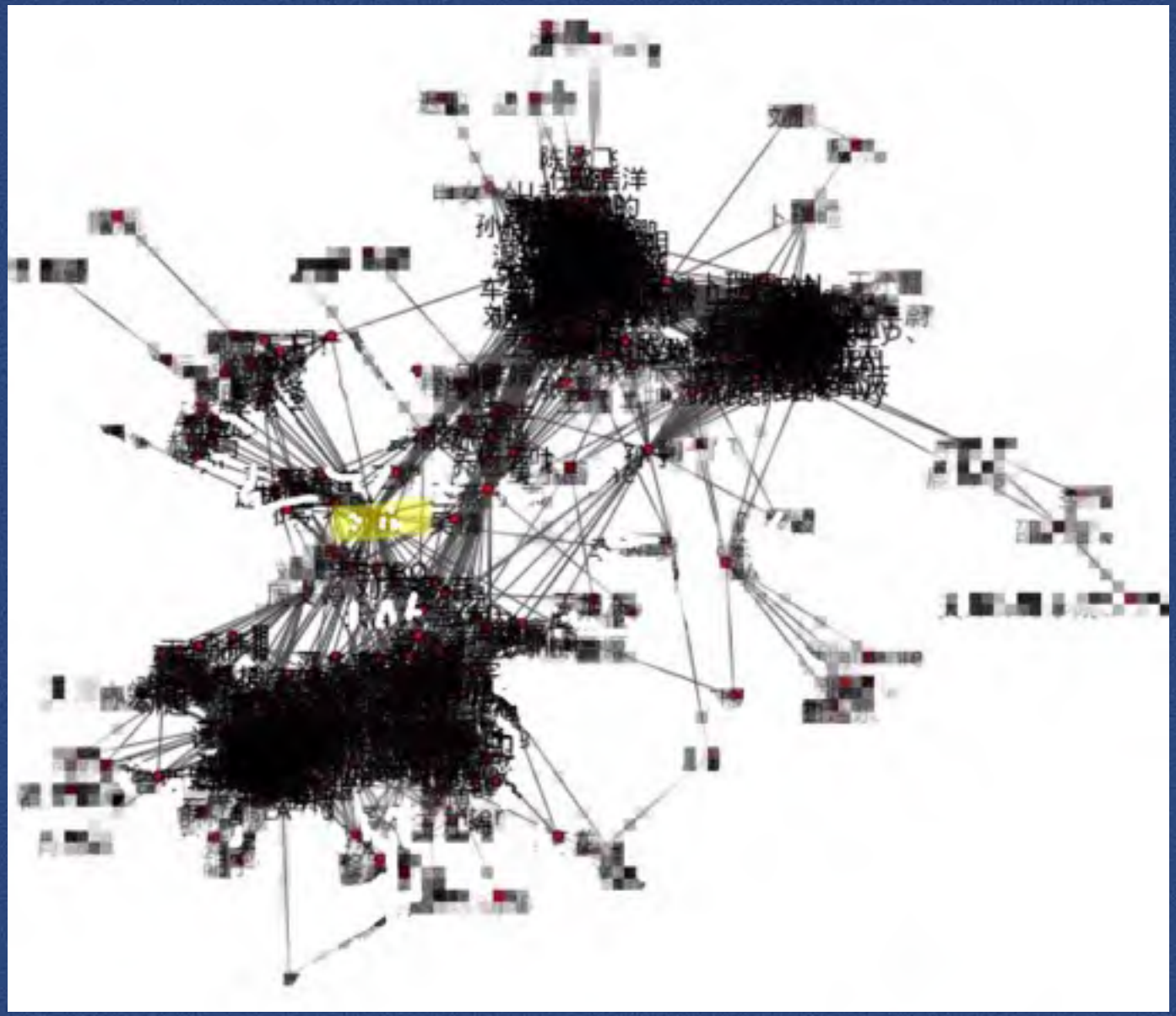
1.波动是有周期的
2.18-24 处，遗漏很小，连续出现。
3.60 以后，遗漏普遍高于平均值。
4.现在出现了遗漏超级大的值。

Case3: 找朋友，隐秘的异性朋友

Case3: SNS 好友分组

1. FR 算法 – – Fruchterman Reingold
2. 组群识别算法 GN & Fast GN

Case3: SNS 好友分组



中学同学，
依旧联系紧密，
与大学无差异。
—> 社交达人

左侧射线状。

黄色的顶点
-> 女朋友

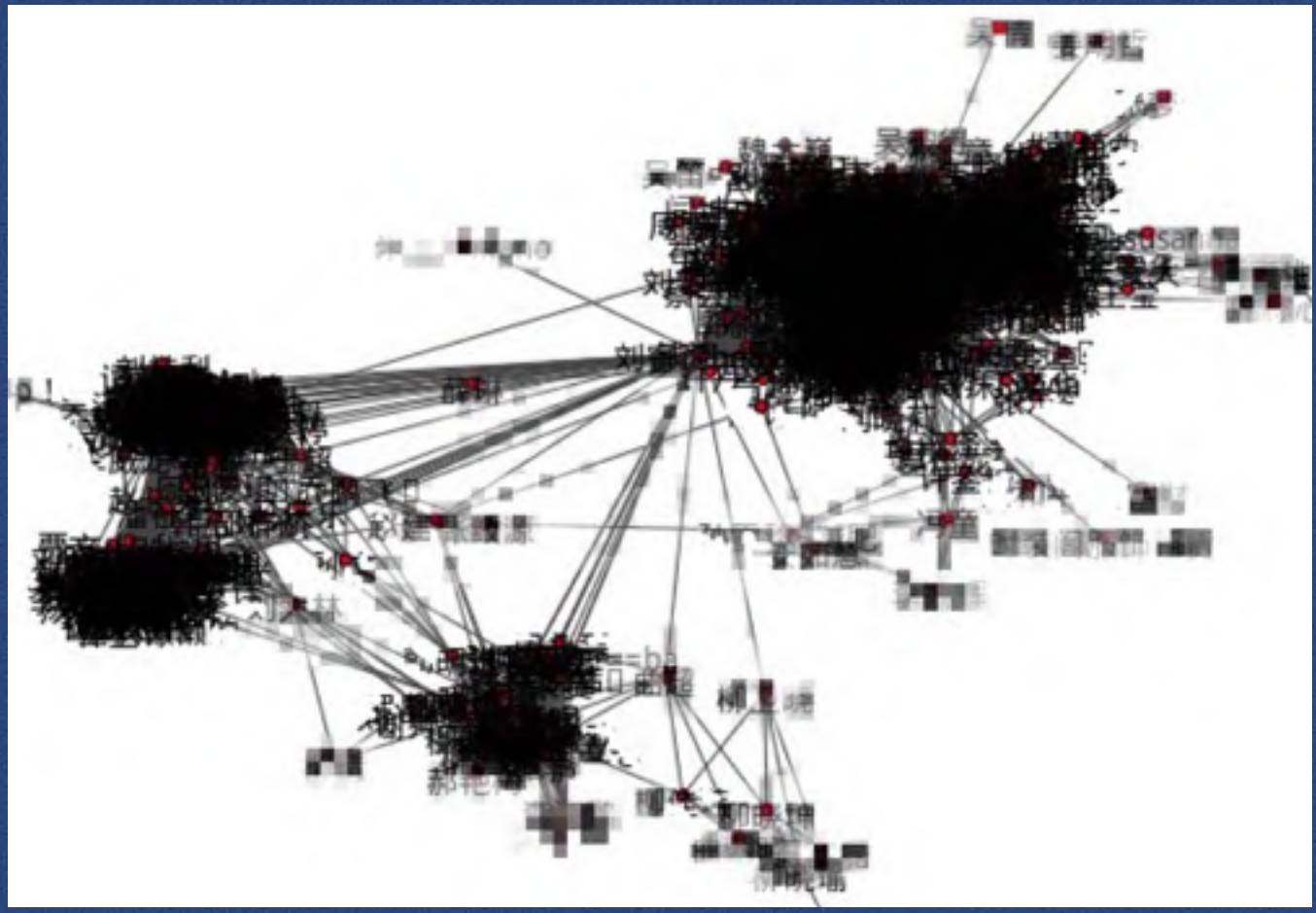
Case3: SNS 好友分组

高三复读过。

热衷于某项活动。

相邻年级，
联系紧密。

-> 活动达人



Case3: SNS 好友分组

```
def init_graph(data, orig_id):
        friends=data[orig_id]
        print(data.keys())
        #print(friends.keys())
        g=igraph.Graph(len(friends))
        i=0
        for rid, name in friends.items():
                g.vs[i]['rid']=rid
                g.vs[i]['name']=name
                #print(name)
                i += 1
        #add edges
        for rid in g.vs['rid']:
                for fid in set(data.get(rid,dict()).keys())&set(friends.keys()):
                        g.add_edges((g.vs['rid'].index(rid), g.vs['rid'].index(fid)))
        g.simplify()
        return g

def showGraph(graph,filename):
        ly=graph.layout('fr')
        visual_style = {}
        visual_style["vertex_size"] = 5
        visual_style['layout']=ly
        #visual_style["vertex_label"] = g.vs["uid"]
        #visual_style["vertex_label"] = graph.vs["name"]
        if filename is not None:
                igraph.plot(graph,"{}.png".format(filename),**visual_style)
        else:
                igraph.plot(graph,**visual_style)
```

总结

数学函数的可视化与 matplotlib。

寻找数据间的函数关系，
作出函数图像。

网络拓扑结构的可视化与 igraph。

数据不是孤立的，数据网络是有结构的。
网络的拓扑结构难以伪造、易于清洗。
可以发现很有价值的信息。

kunth002