统计学有两大主要分支，分别是描述性统计学和推断统计学。描述性统计学用于描述和概括数据的特征以及绘制各类统计图表。总体数据，往往因为数据量太大而难以被获取，所以就有了
通过较小的样本数据推测总体特性的推断统计学
。值得一提的是现今火热的“大数据”一词并不仅仅是指数据量大，在《大数据时代》一书中作者舍恩伯格强调“大数据”不是随机样本，而是所有数据，即总体，这与传统的统计研究方法是有很大区别的。

推断统计
学的一个研究方向
就是用样本数据估算总体的未知参数
，称之为参数估计
。如果是用一个数值进行估计，则称为点估计
；如果估计时给出的是一个很高可信度的区间范围，则称为区间估计。

本文先介绍了抽样分布和中心极限定理
，并用蒙特卡洛方法进行模拟；然后引入置信区间的概念，并将之用于分析BRFSS数据中的bmi指数上。

首先依旧是导入相关Python
模块和数据，其中brfss是专门用于读取和清理美国行为风险因素监控BRFSS调研数据的模块。

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import brfss  # ???????BRFSS??

%config InlineBackend.figure_format = 'retina'

df = brfss.ReadBrfss()  # ??BRFSS??
```

这里主要关注反应胖瘦
程度的BMI指数，并将这一数据存入BMI变量中，其数据量有40万之多。

```
bmi = df.bmi.dropna()  # ?????bmi????????
len(bmi)
```

405058

# 中心极限定理

如果我们将上述40万多份的BMI数据看成是总体，然后从中随机抽取n个数据组成一份样本，并计算该样本的均值。重复这一过程1000次，我们就得到了1000个样本的均值分布，即抽样分布。

抽样分布满足中心极限定理
，即在样本量n越来越大
时，均值的抽样分布将越来越接近正态分布
，该分布的均值等于总体的均值；标准差，在这里也称为标准误差SE满足公式：

$$SE = \frac{\sigma}{\sqrt{n}}$$ 其中$\sigma$是总体的标准差，$n$是样本大小



这里使用蒙特卡洛模拟的方法，在40万BMI数据中随机抽取n个数计算均值，并重复1000次，组成抽样分布。以下的sampling_distribution()函数用于实现这一模拟过程，并绘制抽样分布的直方图和ECDF图。

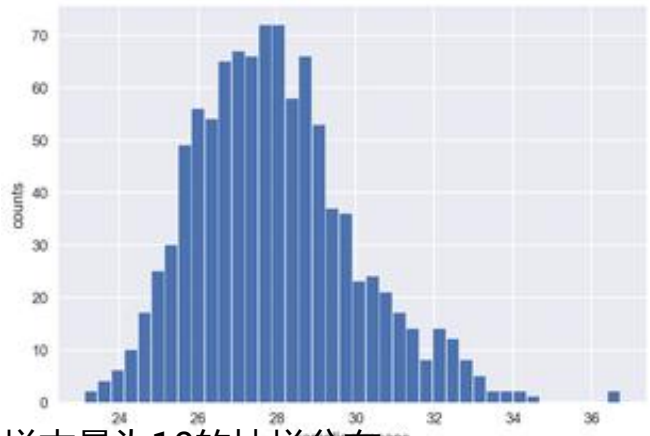
```
def sampling_distribution(data, sample_size=20, bins=40):
    '''??????????????????????ECDF?'''

    # ????
    sampling = [np.mean(np.random
.choice(data, size=sample_size, replace=False)) for _ in ran
ge(1000)]

    # ???????????????
    mu = np.mean(data)
    se = np.std(data) / np.sqrt(sample_size)
    print('mean of sample means: %.2f' % np.mean(sampling))
    print('population means: %.2f' % mu)
```

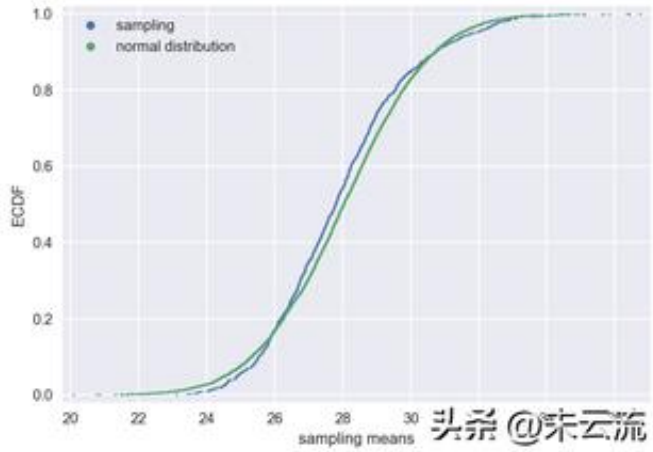
```
    print('Standard deviation of sample means: %.2f' % np.st
d(sampling))
    print('Standard Error: %.2f' % se)

    # ??????????ECDF?
    fig = plt.figure(figsize=(16,5))
    p1 = fig.add_subplot(121)
    plt.hist(sampling, bins=bins, rwidth=0.9)
    plt.xlabel('sampling means')
    plt.ylabel('counts')
    p2 = fig.add_subplot(122)
    plot_ecdf(sampling, xlabel='sampling means', label='samp
ling ')
    sample = np.random.normal(mu, se, size=10000)
    plot_ecdf(sample, xlabel='sampling means', label='normal
 distribution')
    plt.show()

def ecdf(data):
    '''??ECDF'''
    x = np.sort(data)
    y = np.arange(1, len(x)+1) / len(x)
    return (x,y)

def plot_ecdf(data, xlabel=None , ylabel='ECDF', label=None)
:
    '''??ECDF?'''
    x, y = ecdf(data)
    _ = plt.plot(x, y, marker='.', markersize=3, linestyle='
none', label=label)
    _ = plt.legend(markerscale=4)
    _ = plt.xlabel(xlabel)
    _ = plt.ylabel(ylabel)
    plt.margins(0.02)
```

下面我们将样本量n分别取为10、20、100，进行三次模拟。

```
sampling_distribution(bmi, sample_size=10)
```

```
mean of sample means: 27.95
population means: 28.04
Standard deviation of sample means: 2.04
Standard Error: 2.10
```

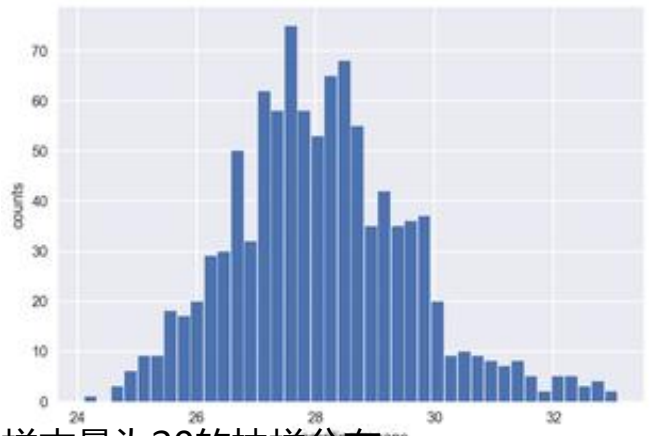


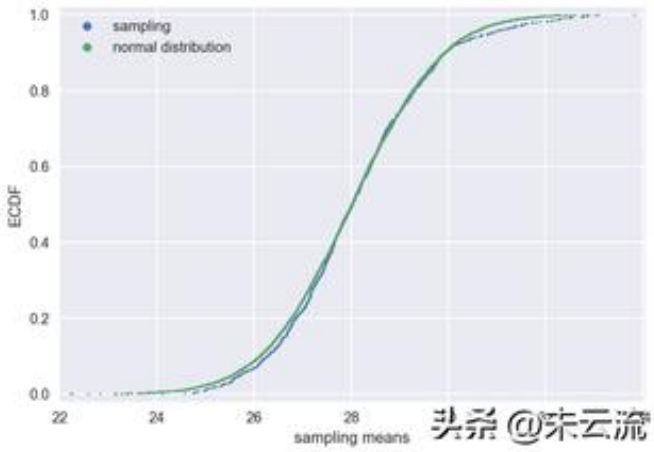




样本量为10的抽样分布

```
sampling_distribution(bmi, sample_size=20)
```

```
mean of sample means: 28.11
population means: 28.04
Standard deviation of sample means: 1.50
Standard Error: 1.49
```

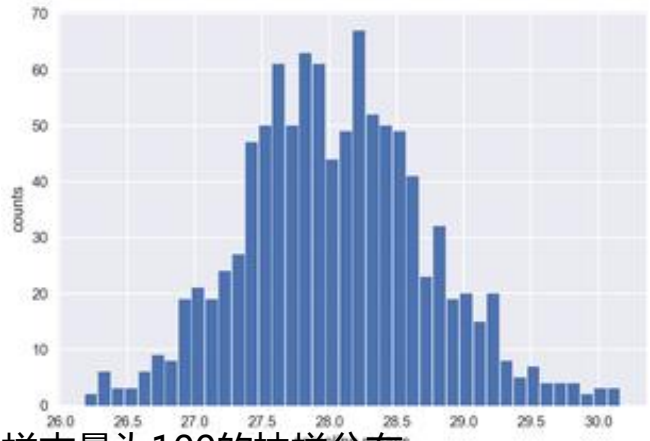


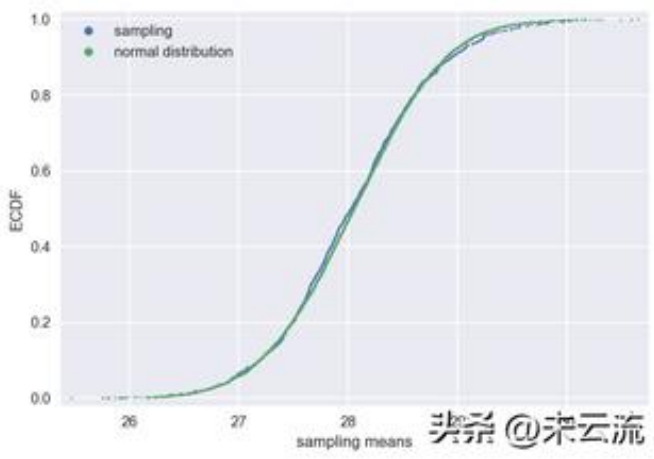




样本量为20的抽样分布

```
sampling_distribution(bmi, sample_size=100)
```

```
mean of sample means: 28.05
population means: 28.04
Standard deviation of sample means: 0.69
Standard Error: 0.67
```




样本量为100的抽样分布

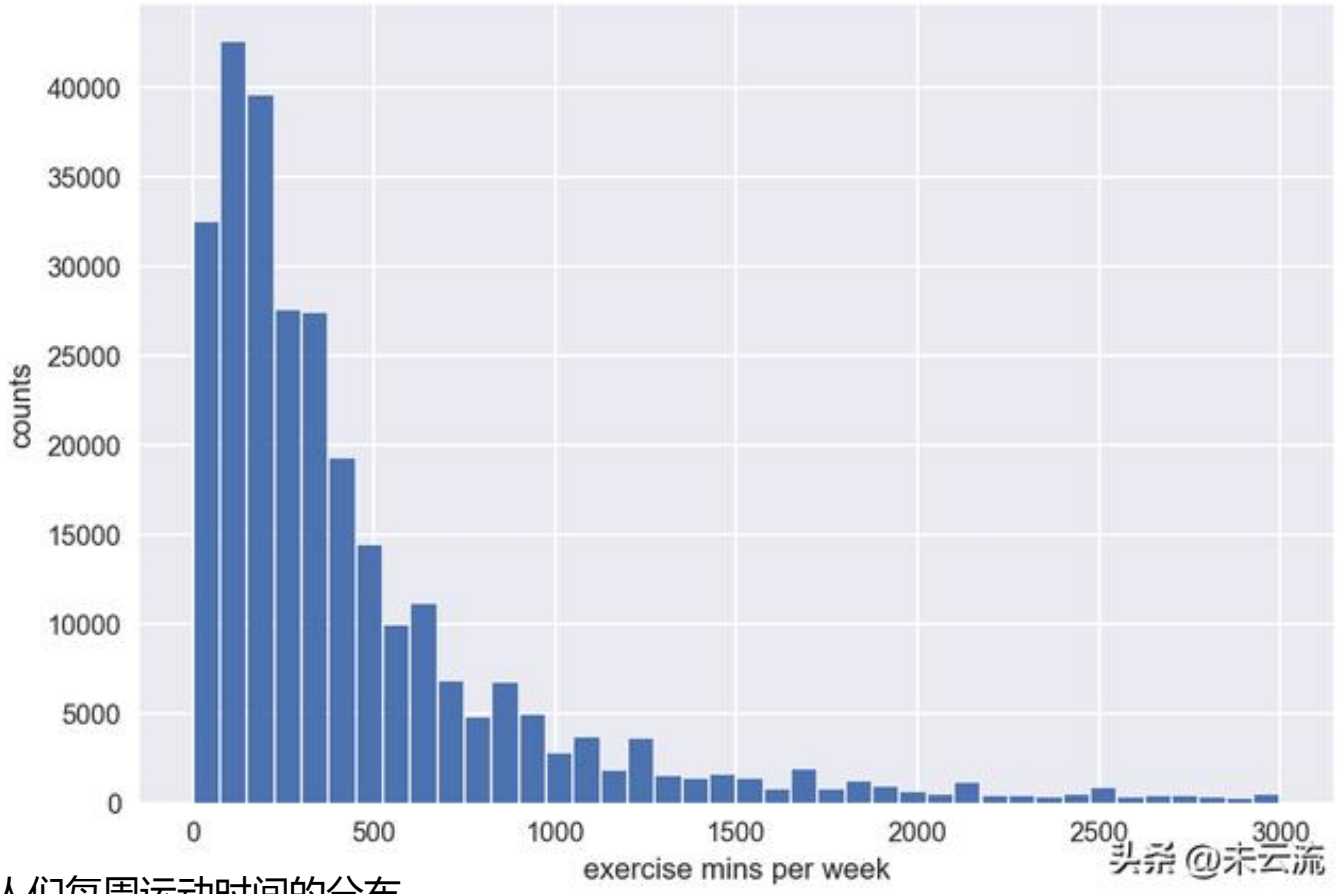
观察上面的输出结果和图形，我们发现随着样本量的递增，抽样分布越来越靠近正态分布，其均值和标准差也越来越符合中心极限定理中给出的关系。

一般当n大于等于30时，样本均值的抽样分布近似为正态分布。此时我们可以用样本的均值来估计总体的均值，这就是点估计的一种最简单的方式。但从上述分布也可以看出，样本均值其实是以一定概率在总体均值附近浮动的，所以这就有了后面将要讲的置信区间。

关于中心极限定理，还有一点需要强调的是，无论变量原来的分布是什么样的，其均值的抽样分布在n足够大时都会接近正态分布。比如我们研究BRFSS数据中人们每周运动的总时间（单位：分钟），大部分人每周运动的时间少于500分钟，而极少数人能达到3000分钟，其直方图
反应数据大部分集中在左侧，而右侧有一条长长的尾巴。

```
exemin = df[df.exemin != 0].exemin.dropna()   # ???????????0
?????
plt.hist(exemin,bins=40, range=(0,3000), rwidth=0.9)  # ????
```

```
?
plt.xlabel('exercise mins per week')
plt.ylabel('counts')
plt.show()
```

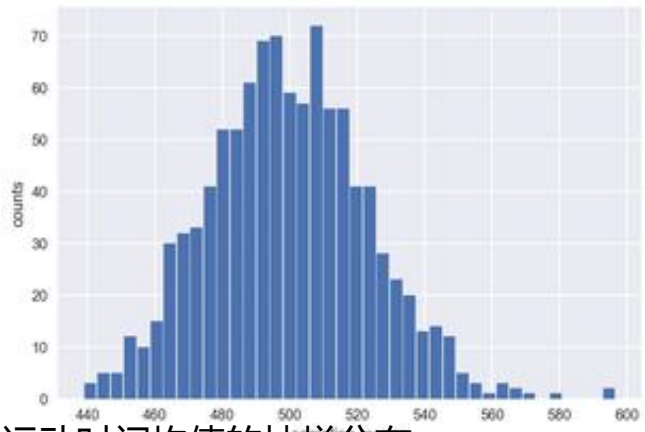



人们每周运动时间的分布

显然这一数据分布并不满足正态分布，但是我们采用上述相同的方法模拟其样本均值的抽样分布，在样本量n为1000时，抽样分布与正态分布符合的非常好。可见中心极限定理并不要求变量原来分布的样子，这也正是其魅力所在。

```
sampling_distribution(exemin, sample_size=1000)
```

```
mean of sample means: 499.54
population means: 499.37
Standard deviation of sample means: 23.60
Standard Error: 23.75
```

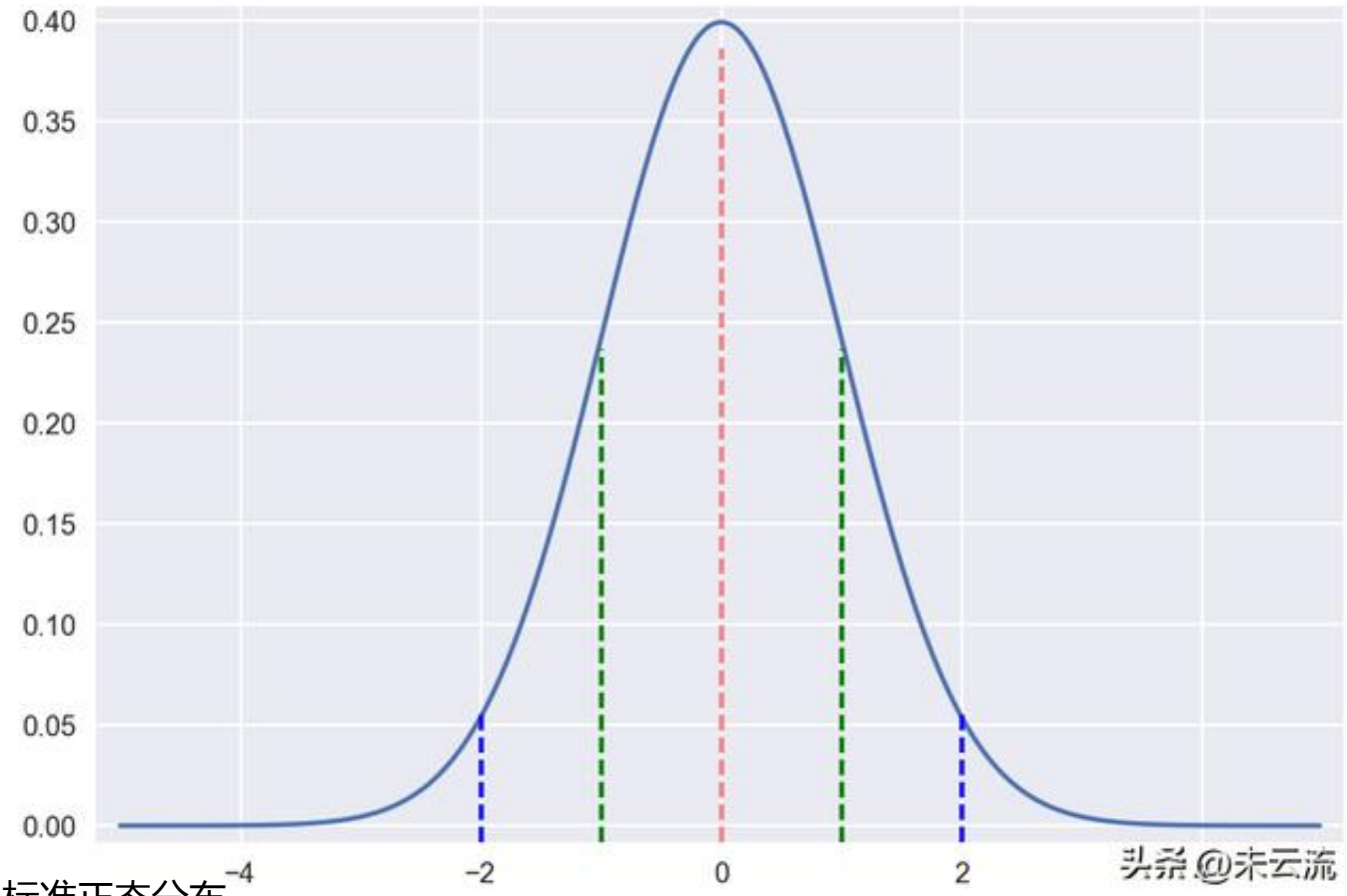


运动时间均值的抽样分布

# 正态分布的特性

既然中心极限定理中涉及了正态分布，我们就来看看其均值和标准差的一些性质。
这里导入scipy的统计模块，使用scipy.stats.norm()模拟
标准正态分布，即均值为0，标准差为1。使用norm.pdf()
计算概率密度，并绘制概率密度函数（PDF）图。

```
import scipy.stats
norm = scipy.stats.norm()  # ??????

x = np.arange(-5, 5, 0.02)
y = norm.pdf(x)  # ????
plt.plot(x,y)
plt.axvline(x=0,ymax=0.95, linestyle='--', color='red', alph
a=0.5)
plt.axvline(x=1,ymax=0.59, linestyle='--', color='green')
plt.axvline(x=-1,ymax=0.59, linestyle='--', color='green')
plt.axvline(x=2,ymax=0.16, linestyle='--', color='blue')
plt.axvline(x=-2,ymax=0.16, linestyle='--', color='blue')
plt.margins(0.02)
plt.show()
```




标准正态分布

PDF图中曲线下的面积代表了概率，
使用norm.cdf()
可计算这部分面积，即累积概率分布。于是我们就可以得到变量距离均值在1个标准差范围内的概率为0.68，2个标准差范围内的概率是0.95，3个标准差范围内的概率是0.997。可见在正态分布中，数据主要集中在3个标准差之内。

```
print('1 sigma : %.3f' % (norm.cdf(1) - norm.cdf(-1)))
print('2 sigma : %.3f' % (norm.cdf(2) - norm.cdf(-2)))
print('3 sigma : %.3f' % (norm.cdf(3) - norm.cdf(-3)))
```

```
1 sigma : 0.683
2 sigma : 0.954
3 sigma : 0.997
```

反过来，我们也可以

通过概率来求变量分布的区间，这里使用norm.interval()，比如95%的情况下变量分布在-1.96到1.96之间，99%的情况下分布在-2.58到2.58之间。

```
norm.interval(0.95)
```

```
(-1.959963984540054, 1.959963984540054)
```

```
norm.interval(0.99)
```

```
(-2.5758293035489004, 2.5758293035489004)
```

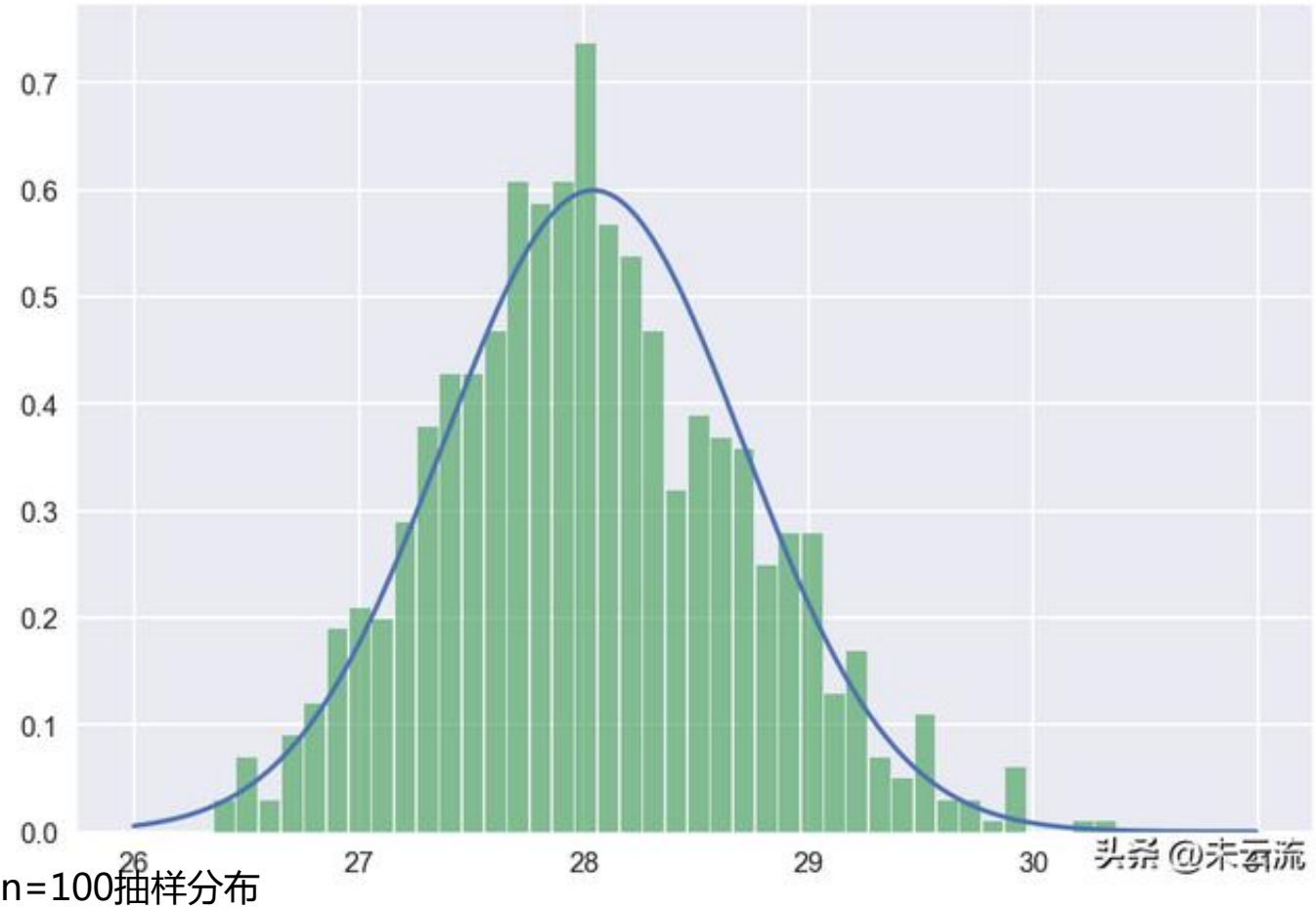
# 置信区间

在能够计算正态分布中一定概率下对应的变量区间后，我们再回到之前用样本均值估计总体均值时遗留的问题，即样本的均值围绕总体均值在一定范围内浮动的。我们需要估算总体均值在多大的概率下落在抽样的随机区间内，这就是置信区间。

我们仍然将40多万的bmi数据当成是总体，然后从中随机抽取样本量为100的数据，根据中心极限定理绘制抽样分布图如下。

```
sample_size = 100

# ??????????
mu = np.mean(bmi)
se = np.std(bmi) / np.sqrt(sample_size)
# ???????PDF
norm = scipy.stats.norm(mu, se)
x = np.arange(26, 31, 0.01)
y = norm.pdf(x)
plt.plot(x,y)

# ??????????
sample_size = 100
sampling
```

```
 = [np.mean(np.random.choice(bmi, size=sample_size, replace=
False)) for _ in range(1000)]
plt.hist(sampling, bins=40, rwidth=0.9, normed=True, alpha=0
.7)

plt.show()
```




n=100抽样分布

根据正态分布的性质，在95%的概率下，均值分布区间是(26.74, 29.35)。也就是说，在样本量为100时，我们有95%的信心相信总体均值落在26.74和29.35之间，这就是95%的置信区间。同理，99%的置信区间是(26.33, 29.76)。注意这是在大样本量
的情况下
，我们才能使用正
态分布，而如果样本量n小于30，则需要采用t分布，此处就不展开了。

```
norm.interval(0.95)
```

```
(26.738141245959351, 29.346706751112283)
```

```
norm.interval(0.99)
```

```
(26.328305902131977, 29.756542094939658)
```

# 区间估计的应用

回到本系列文章一直在探索的一个问题，即比较富人和普通人的BMI指数。此时，bmi数据不再当做总体看待，而是作为调查的样本，总体是BRFSS数据针对的全体美国人。首先将bmi数据按照收入等级分为两组，即富人bmi数据和普通人bmi数据。

```
df2 = df[['bmi', 'income']].dropna()  # ?????bmi????income?
?????????
bmi_rich = df2[df2.income == 8].bmi   # ?????8????????
bmi_ord = df2[df2.income != 8].bmi    # ?????1-7??????????
```

以下定义了mean_ci()
函数，根据置信区间的计算公式，计算95%置信度下均值所在的区间。

```
def mean_ci(data):
    '''???????????95%?????'''

    sample_size = len(data)
    std = np.std(data, ddof=1)  # ????????
    se = std / np.sqrt(sample_size)  # ??????
    point_estimate = np.mean(data)
    z_score = scipy.stats.norm.isf(0.025)  # ???95%
    confidence_interval = (point_estimate - z_score * se, po
int_estimate + z_score * se)

    return confidence_interval
```

于是得到富人bmi95%的置信区间为(27.42, 27.49),
普通人bmi95%的置信区间为(28.51, 28.57)。这两个区间间隔得还比较远，数值上差不多有1这么多。所以我们可以比较有信心地得出富人更瘦的结论。

```
mean_ci(bmi_rich)
```

```
(27.415906122294761, 27.485560606043915)
```

```
mean_ci(bmi_ord)
```

```
(28.509003170593907, 28.565637279855423)
```

但要注意了，以上之所以能得到这么肯定的结论，源于使用的样本数据量非常大，这大大缩小了置信区间的范围（这可以从中心极限定理中标准误差的公式看出）。现在让我们使用前500个数据，看看在样本较少时会发生什么情况。

```
mean_ci(bmi_rich[:500])
```

```
(27.849838839563304, 28.791561160436636)
```

```
mean_ci(bmi_ord[:500])
```

```
(28.200546441671069, 29.303493558328935)
```

此时富人bmi95%的置信区间为(27.85,
28.79)，而普通人bmi95%的置信区间为(28.20, 29.30)，很明显这两个区间都变大了。尽管富人的bmi指数仍有相对较小的趋势，但是这两个区间有部分重合，这时我们就无法得出非常肯定的结论了。可见样本量在做判断时起着非常重要的作用，样本越大，判断越准确，这也是与我们常识相符的。

# 小结

在这
一篇中，
我们了解了抽样分
布的概念，中心极限定理的含义，正
态分布的概率分布
，最重要的是使用置信区间的计算方法，通过样本数据估算总体的均值范围，至此我们进入了推断统计学的领域。

针对富人是否更瘦这个问题上，虽然使用了置信区间得出了较肯定的结论，但是仍然没有对富人更瘦这个假设做出明确的判断。在下一篇中我们将会讲到推断统计学的另一个领域：假设检验，即对参数的假设值进行决策，届时我们将和上述问题来个了断。