习题 2.32

首先，读入数据。因为本题用到的数据仅为棒球选手的 ID、salary 和 POS，所以仅提取数据框中的 `player`、`salary` 和 `POS` 三列，读取的数据保存为变量名 `baseball`。

```
baseball = read.csv('baseball.csv', header=FALSE)
baseball = baseball[,c(3,4,5)]
colnames(baseball) = c('player', 'salary', 'POS')
head(baseball)
```

```
    player   salary POS
1 anderga0  6200000  CF
2 colonba0 11000000   P
3 davanje0   375000  CF
4 donnebr0   375000   P
5 eckstda0  2150000  SS
6 erstada0  7750000  1B
```

继续沿用 (a) 问中抽取的 SRS，结果保存在变量 `S` 中。

```
set.seed(0); set.seed(12345)
n = 150; N = dim(baseball)[1]
# 定义函数 get_SRS(n, N) 来从总体中得到一个 SRS
get_SRS = function(n, N){
  S = c(); k = 0 # 初始化变量
  while (k < n){
    x = runif(1)
    R = floor(N*x) + 1
    if (! R %in% S){
      S = append(S, R)
      k = k + 1
    }
  }
  return(S)
}
S = get_SRS(n, N)
```

c. `logsal` 的样本平均和 95% 置信区间

样本平均的计算公式为 $\bar{y} = \dfrac{1}{n}\sum_{i\in\mathcal{S}} y_i$ ；
总体均值 95% 置信区间的计算公式为 $[\bar{y} - z_{\alpha/2}SE(\bar{y}), \bar{y} + z_{\alpha/2}SE(\bar{y})]$.

使用程序计算 `logsal` 的样本平均和 95% 置信区间如下：

```
baseball_SRS = baseball[S,] # 用 baseball_SRS 储存抽样数据
salary = baseball_SRS$salary # 得到变量 salary
```

```
logsal = log(salary) # 得到变量 logsal
mean_logsal = mean(logsal) # logsal 的样本平均
s2_logsal = sum((logsal - mean_logsal)^2) / (n - 1) # logsal 的样本方差
se_logsal = sqrt((1-n/N)/n * s2_logsal) # logsal 的标准误 SE
CI_lb_logsal = mean_logsal - qnorm(0.975) * se_logsal # CI lower bound
CI_ub_logsal = mean_logsal + qnorm(0.975) * se_logsal # CI upper bound
# 输出结果
print(paste0('变量 logsal 的均值为', round(mean_logsal, 3)), quote=FALSE)
print(paste0('变量 logsal 的 95% 置信区间为',
             '[', round(CI_lb_logsal,3), ',', round(CI_ub_logsal,3), ']'), quote=FALSE)
```

```
[1] 变量logsal的均值为13.95
[1] 变量logsal的95%置信区间为[13.774,14.126]
```

d. pitcher 的总体比例的估计量

POS 列中为 P 的是 pitcher，在 baseball_SRS 中新添加 pitcher 列，以 1 代表是 pitcher，0 代表不是 pitcher。仍用 (c) 问中的公式计算 pitcher 总体比例的点估计及 95% 置信区间。

```
baseball_SRS$pitcher = rep(0, n) # 创建 pitcher 列
baseball_SRS$pitcher[baseball_SRS$POS == 'P'] = 1 # 是 pitcher 的选手赋值为 1
pitcher = baseball_SRS$pitcher
p_hat_pitcher = sum(pitcher) / n # pitcher 比例的样本估计值
s2_pitcher = sum((pitcher - p_hat_pitcher)^2) / (n - 1) # pitcher 比例的样本方差
se_pitcher = sqrt((1-n/N)/n * s2_pitcher) # pitcher 比例的标准误 SE
CI_lb_pitcher = p_hat_pitcher - qnorm(0.975) * se_pitcher # CI lower bound
CI_ub_pitcher = p_hat_pitcher + qnorm(0.975) * se_pitcher # CI upper bound
# 输出结果
print(paste0('pitcher 总体比例的样本估计值为', round(p_hat_pitcher, 3)), quote=FALSE)
print(paste0('pitcher 总体比例的 95% 置信区间为',
             '[', round(CI_lb_pitcher,3), ',', round(CI_ub_pitcher,3), ']'), quote=FALSE)
```

```
[1] pitcher总体比例的样本估计值为0.44
[1] pitcher总体比例的95%置信区间为[0.368,0.512]
```

e. logsal 的总体均值和 pitcher 的总体比例与其置信区间的比较

首先，计算 logsal 的总体均值和 pitcher 的总体比例：

```
mean_logsal_U = mean(log(baseball$salary)) # logsal 的总体均值
baseball$pitcher = rep(0, N)
baseball$pitcher[baseball$POS == 'P'] = 1
p_pitcher = mean(baseball$pitcher)
# 输出结果
```

```
print(paste0('logsal 的总体均值为', round(mean_logsal_U, 3)), quote=FALSE)
print(paste0('pitcher 的总体比例为', round(p_pitcher, 3)), quote=FALSE)
```

```
[1] logsal的总体均值为13.927
[1] pitcher的总体比例为0.472
```

然后，将 logsal 的总体均值和 pitcher 的总体比例与 (c) 问和 (d) 问中的置信区间进行比较：

```
logsal_in_CI = mean_logsal_U > CI_lb_logsal & mean_logsal_U < CI_ub_logsal
pitcher_in_CI = p_pitcher > CI_lb_pitcher & p_pitcher < CI_ub_pitcher
# 输出结果
print(paste0('logsal 的总体均值在其 CI 中的结果为', logsal_in_CI), quote=FALSE)
print(paste0('pitcher 的总体比例在其 CI 中的结果为', pitcher_in_CI), quote=FALSE)
```

```
[1] logsal的总体均值在其CI中的结果为TRUE
[1] pitcher的总体比例在其CI中的结果为TRUE
```

习题 2.32 补充

1. 从总体中反复、独立地抽取 1000 个容量为 150 的 SRS

对每个 SRS，计算变量 salary 和 logsal 的样本均值、样本方差，以及总体均值的 95% 置信区间，将计算得到的所有数据保存在数据框 SRS1000_df 中。

```
# 初始化所有变量
SRSmean_salary = c(); SRSmean_logsal = c()
SRSvar_salary = c(); SRSvar_logsal = c()
SRSlb_salary = c(); SRSub_salary = c()
SRSlb_logsal = c(); SRSub_logsal = c()

# 反复独立地抽取 1000 个容量为 150 的 SRS，并对每一个样本计算相关的统计量
set.seed(0); set.seed(54321)
for (i in 1:1000){
  S_temp = get_SRS(n, N) # SRS 的序号
  baseball_SRStemp = baseball[S_temp,] # 从原数据中提取 SRS 的信息
  salary_temp = baseball_SRStemp$salary # 得到样本的 salary
  logsal_temp = log(salary_temp) # 得到样本的 logsal

  SRSmean_salary = append(SRSmean_salary, mean(salary_temp)) # 更新 salary 样本均值
  SRSmean_logsal = append(SRSmean_logsal, mean(logsal_temp)) # 更新 logsal 样本均值
  SRSvar_salary = append(SRSvar_salary, var(salary_temp)) # 更新 salary 样本方差
  SRSvar_logsal = append(SRSvar_logsal, var(logsal_temp)) # 更新 logsal 样本方差

  se_salary_temp = sqrt((1-n/N)/n * var(salary_temp)) # salary 的标准误 SE
  se_logsal_temp = sqrt((1-n/N)/n * var(logsal_temp)) # logsal 的标准误 SE
```

```
  # 每一个样本 salary 的 95% 置信区间
  lb_salary_temp = mean(salary_temp) - qnorm(0.975) * se_salary_temp
  ub_salary_temp = mean(salary_temp) + qnorm(0.975) * se_salary_temp
  # 每一个样本 logsal 的 95% 置信区间
  lb_logsal_temp = mean(logsal_temp) - qnorm(0.975) * se_logsal_temp
  ub_logsal_temp = mean(logsal_temp) + qnorm(0.975) * se_logsal_temp

  SRSlb_salary = append(SRSlb_salary, lb_salary_temp) # 更新 salary 置信下限
  SRSub_salary = append(SRSub_salary, ub_salary_temp) # 更新 salary 置信上限
  SRSlb_logsal = append(SRSlb_logsal, lb_logsal_temp) # 更新 logsal 置信下限
  SRSub_logsal = append(SRSub_logsal, ub_logsal_temp) # 更新 logsal 置信上限
}

SRS1000_df = data.frame(SRSmean_salary, SRSvar_salary, SRSlb_salary,
                        SRSub_salary, SRSmean_logsal, SRSvar_logsal,
                        SRSlb_logsal, SRSub_logsal) # 将所有数据保存在数据框中
```

2. salary 和 logsal 的总体分布直方图

```
par(mfrow=c(1,2))
hist(baseball$salary, xlab='salary', main='Histogram of salary')
hist(log(baseball$salary), xlab='log(salary)', main='Histogram of logsal')
```

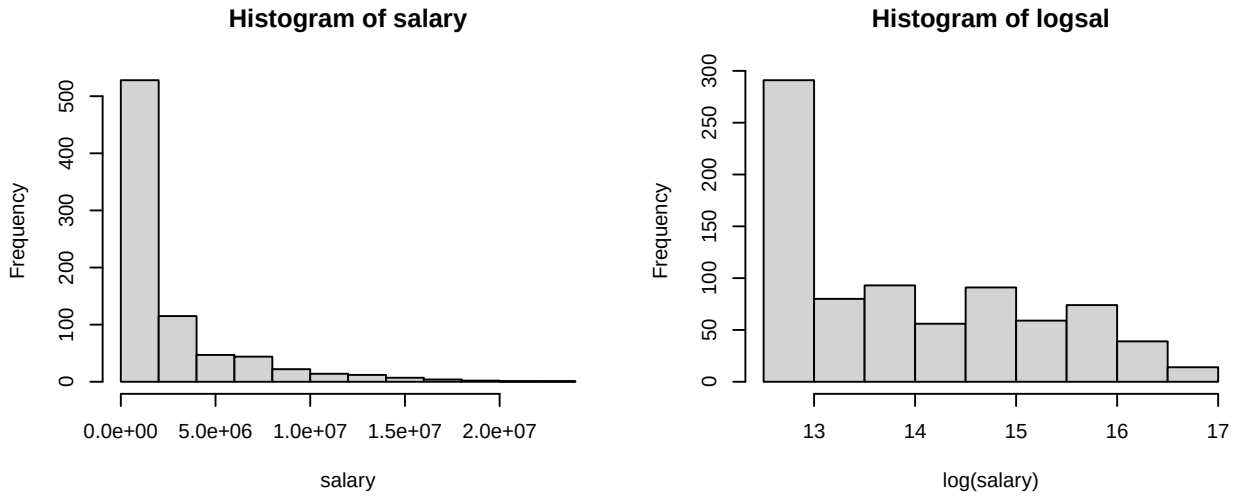

图 1: 总体分布直方图 (左：salary；右：logsal)

3. salary 和 logsal 的样本均值频率分布直方图

```
par(mfrow=c(1,2))
hist(SRS1000_df$SRSmean_salary, freq=FALSE,
```

```
     xlab='Sample Mean of salary', main='Histogram of salary sample mean')
hist(SRS1000_df$SRSmean_logsal, freq=FALSE,
     xlab='Sample Mean of log(salary)', main='Histogram of logsal sample mean')
```

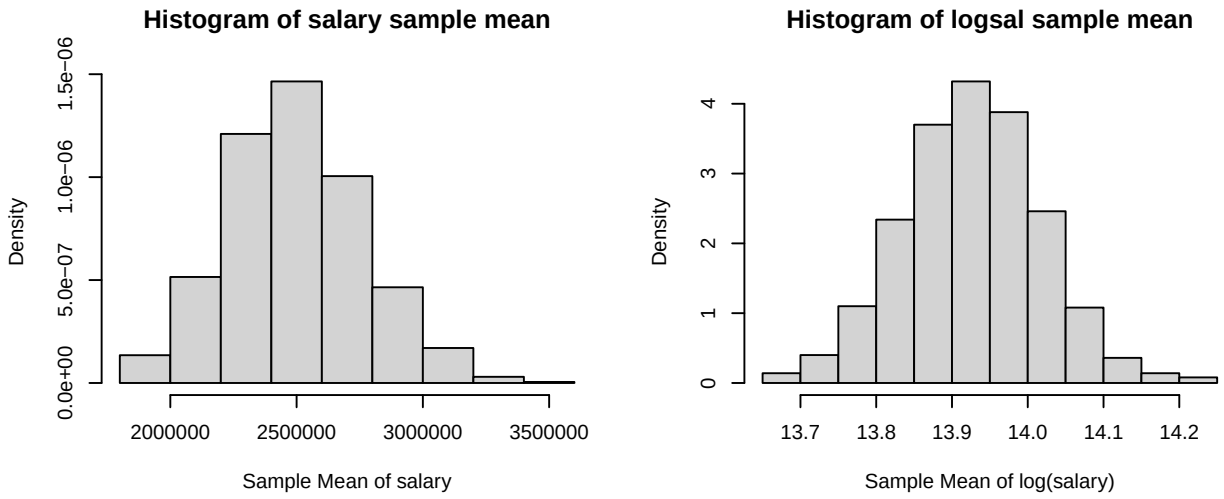

图 2: 样本均值频率分布直方图 (左：salary；右：logsal)

4. 样本均值的均值与方差、样本方差的均值

```
salary_meanmean = mean(SRS1000_df$SRSmean_salary) # salary 样本均值的均值
salary_meanvar = var(SRS1000_df$SRSmean_salary) # salary 样本均值的方差
salary_varmean = mean(SRS1000_df$SRSvar_salary) # salary 样本方差的均值
logsal_meanmean = mean(SRS1000_df$SRSmean_logsal) # logsal 样本均值的均值
logsal_meanvar = var(SRS1000_df$SRSmean_logsal) # logsal 样本均值的方差
logsal_varmean = mean(SRS1000_df$SRSvar_logsal) # logsal 样本方差的均值
# 输出结果
print(paste0('salary 样本均值的均值与方差、样本方差的均值分别为',
             format(salary_meanmean, scientific=TRUE, digits=5), ', ',
             format(salary_meanvar, scientific=TRUE, digits=5), ', ',
             format(salary_varmean, scientific=TRUE, digits=5)), quote=FALSE)
print(paste0('logsal 样本均值的均值与方差、样本方差的均值分别为',
             round(logsal_meanmean, 4), ', ', round(logsal_meanvar, 4),
             ', ', round(logsal_varmean, 4)), quote=FALSE)
```

```
[1] salary样本均值的均值与方差、样本方差的均值分别为2.4993e+06, 7.2612e+10, 1.2522e+13
[1] logsal样本均值的均值与方差、样本方差的均值分别为13.9272, 0.0084, 1.5356
```

根据定理 3.1，有：

(1) $E(\bar{y}) = \bar{y}_U$，即样本均值的均值为总体均值的无偏估计;

(2) $V(\bar{y}) = \frac{1-f}{n} S^2$

(3) $E(s^2) = S^2$, 即样本方差的均值为总体方差的无偏估计.

下面计算 salary 和 logsal 的总体均值、样本均值的方差、总体方差的真实值，以及估计值与真实值的相对误差。将各估计值与真实值进行比较，可以验证定理 3.1：

```
# salary 的各真实值的计算
salary_realmean = mean(baseball$salary) # 总体均值的真实值
salary_realvar = var(baseball$salary) # 总体方差的真实值
salary_real_sampleMeanVar = (1-n/N)/n * salary_realvar # 样本均值的方差的真实值
# salary 的各相对误差的计算
salary_mean_error = (salary_meanmean - salary_realmean) / salary_realmean * 100
salary_sampleMeanVar_error = (salary_meanvar - salary_real_sampleMeanVar) /
  salary_real_sampleMeanVar * 100
salary_var_error = (salary_varmean - salary_realvar) / salary_realvar * 100

# logsal 的各真实值的计算
logsal_realmean = mean(log(baseball$salary)) # 总体均值的真实值
logsal_realvar = var(log(baseball$salary)) # 总体方差的真实值
logsal_real_sampleMeanVar = (1-n/N)/n * logsal_realvar # 样本均值的方差的真实值
# logsal 的各相对误差的计算
logsal_mean_error = (logsal_meanmean - logsal_realmean) / logsal_realmean * 100
logsal_sampleMeanVar_error = (logsal_meanvar - logsal_real_sampleMeanVar) /
  logsal_real_sampleMeanVar * 100
logsal_var_error = (logsal_varmean - logsal_realvar) / logsal_realvar * 100

# 结果输出
print('========== salary ==========', quote=FALSE)
print(paste0('样本均值的均值为', format(salary_meanmean, scientific=TRUE, digits=5),
             ', 总体均值的真实值为', format(salary_realmean, scientific=TRUE, digits=5),
             ', 相对误差为', round(salary_mean_error, 4), '%'), quote=FALSE)
print(paste0('样本均值的方差的估计值为',
             format(salary_meanvar, scientific=TRUE, digits=5),
             ', 样本均值的方差的真实值为',
             format(salary_real_sampleMeanVar, scientific=TRUE, digits=5),
             ', 相对误差为', round(salary_sampleMeanVar_error, 4), '%'), quote=FALSE)
print(paste0('样本方差的均值为',
             format(salary_varmean, scientific=TRUE, digits=5),
             ', 总体方差的真实值为',
             format(salary_realvar, scientific=TRUE, digits=5),
             ', 相对误差为', round(salary_var_error, 4), '%'), quote=FALSE)

print('========== logsal ==========', quote=FALSE)
print(paste0('样本均值的均值为', round(logsal_meanmean, digits=5),
             ', 总体均值的真实值为', round(logsal_realmean, digits=5),
             ', 相对误差为', round(logsal_mean_error, 4), '%'), quote=FALSE)
print(paste0('样本均值的方差的估计值为', round(logsal_meanvar, digits=5),
             ', 样本均值的方差的真实值为', round(logsal_real_sampleMeanVar, digits=5),
```

```
            ', 相对误差为', round(logsal_sampleMeanVar_error, 4), '%'), quote=FALSE)
print(paste0('样本方差的均值为', round(logsal_varmean, digits=5),
            ', 总体方差的真实值为', round(logsal_realvar, digits=5),
            ', 相对误差为', round(logsal_var_error, 4), '%'), quote=FALSE)
```

```
[1] ========== salary ==========
[1] 样本均值的均值为2.4993e+06，总体均值的真实值为2.4977e+06，相对误差为0.0662%
[1] 样本均值的方差的估计值为7.2612e+10，样本均值的方差的真实值为6.7664e+10，相对误差为7.3122%
[1] 样本方差的均值为1.2522e+13，总体方差的真实值为1.2503e+13，相对误差为0.1546%
[1] ========== logsal ==========
[1] 样本均值的均值为13.92719，总体均值的真实值为13.92706，相对误差为0.001%
[1] 样本均值的方差的估计值为0.00836，样本均值的方差的真实值为0.0083，相对误差为0.6606%
[1] 样本方差的均值为1.53558，总体方差的真实值为1.53443，相对误差为0.0752%
```

5. 总体均值的置信区间覆盖真实值的比例

```
SRS1000_df$salaryMean_in_CI = rep(0, 1000)
SRS1000_df$logsalMean_in_CI = rep(0, 1000)
SRS1000_df$salaryMean_in_CI[salary_realmean>SRSlb_salary &
                            salary_realmean<SRSub_salary] = 1
SRS1000_df$logsalMean_in_CI[logsal_realmean>SRSlb_logsal &
                            logsal_realmean<SRSub_logsal] = 1
salary_confidenceLevel = sum(SRS1000_df$salaryMean_in_CI) / 1000 * 100
logsal_confidenceLevel = sum(SRS1000_df$logsalMean_in_CI) / 1000 * 100
print(paste0('salary 总体均值的置信区间覆盖真实值的比例为',
             salary_confidenceLevel, '%'), quote=FALSE)
print(paste0('logsal 总体均值的置信区间覆盖真实值的比例为',
             logsal_confidenceLevel, '%'), quote=FALSE)
```

```
[1] salary总体均值的置信区间覆盖真实值的比例为93.2%
[1] logsal总体均值的置信区间覆盖真实值的比例为94.6%
```

根据上述结果，`logsal` 总体均值的置信区间覆盖其真实值的比例更高，这是因为 `logsal` 的总体均值本身较为接近正态分布，能够更准确地构造置信区间。

习题 2.25

用 bootstrap 方法构造总体均值 $\bar{y}_U$ 的近似 95% 置信区间。采用从美国 $N = 3078$ 个县构成的总体中抽取的容量为 $n = 300$ 的 SRS WOR 样本数据 agsrs.csv，用 bootstrap 方法构造变量 y（1992 年农业用地面积）的总体均值 $\bar{y}_U$ 的近似 95% 区间。做法如下：

1) 从样本 agsrs.csv 的 n 个县中逐次有放回、等概率地抽取 N 个县，形成一个伪总体，变量 y 的值记为 $\mathcal{U}_j^* = \{y_{1,j}^*, ..., y_{N,j}^*\}$;

2) 从 $\mathcal{U}_j^*$ 中采用 SRS WOR 方法抽取一个容量为 n 的样本 $\mathcal{S}_j^*$，计算变量 y 的样本均值 $\bar{y}_j^*$;

3) 对 $j = 1, ..., 1000$，重复 1)-2)，得到 $\bar{y}$ 的 bootstrap 样本 $\bar{y}_1^*, ..., \bar{y}_{1000}^*$，计算该 bootstrap 样本的 0.025、0.975 样本分位点 $\bar{y}_L^*, \bar{y}_U^*$，得到总体均值 $\bar{y}_U$ 的近似 95% 置信区间 $[\bar{y}_L^*, \bar{y}_U^*]$.

```
agsrs_sample = read.csv('agsrs.csv') # 读入数据
acres92_sample = agsrs_sample$acres92
N_25 = 3078; n_25 = 300
set.seed(0); set.seed(1111)

acres92_sampleMeans = c() # 初始化 bootstrap 样本的样本均值
for (j in 1:1000){
  # 构造伪总体
  pseudo_popNum = c()
  for (i in 1:N_25){
    x = runif(1)
    R = floor(n_25*x) + 1
    pseudo_popNum = append(pseudo_popNum, R)
  }
  acres92_pseudoPop = acres92_sample[pseudo_popNum] # 一个伪总体 U^star_j

  # 从这一伪总体中采用 SRS WOR 方法抽取一个容量为 n 的样本
  pseudo_sampleNum = get_SRS(n_25, N_25)
  acres92_psedoSample = acres92_pseudoPop[pseudo_sampleNum]
  acres92_mean = mean(acres92_psedoSample) # 样本均值
  acres92_sampleMeans = append(acres92_sampleMeans, acres92_mean)
}
quantile(acres92_sampleMeans, c(0.025, 0.975)) # 近似 95% 置信区间
```

```
    2.5%    97.5%
256659.4 339165.0
```

用 Hájek 中心极限定理得到的 $\bar{y}_U$ 的置信区间为 $[260,856.08,\ 334,937.92]$，计算置信下限和置信上限的相对误差：

```
CI_lb_bootstrap = 256659.4; CI_ub_bootstrap = 339165.0
CI_lb_hajek = 260856.08; CI_ub_hajek = 334937.92
lb_error = (CI_lb_bootstrap - CI_lb_hajek) / CI_lb_hajek * 100
ub_error = (CI_ub_bootstrap - CI_ub_hajek) / CI_ub_hajek * 100
print(paste0('置信下限的相对误差为', round(lb_error,4), '%'))
print(paste0('置信上限的相对误差为', round(ub_error,4), '%'))
```

```
[1] "置信下限的相对误差为-1.6088%"
[1] "置信上限的相对误差为1.262%"
```

由上述计算结果可知，用 bootstrap 得到的置信下限和置信上限与 Hájek 置信下限和置信上限的相对误差大约在 ± 1.5 左右，所以用 bootstrap 来得到置信区间也是一种很好的方法。