Применения обобщённой статистики Хотеллинга

Анастасия Миллер

1 мая 2015 г.

1 Критерии, основанные на статистике Хотеллинга

1.1 Проверка гипотезы $H_0 : \mu = \mu_0$

Сгенерируем тестовые данные. Пусть это будет n наблюдений p независимых нормально распределённых величин с различными средними.

In[]:
```
import numpy as np
import scipy as sp
import pandas as pd
from scipy.stats import norm, f
from numpy.random import random_integers as rint
from numpy.linalg import inv
n = 1000
p = 7 # it's a good number
m = np.array([float(rint(-5, 5)) for i in range(p)])
data = np.array([norm(loc=mean).rvs(n) for mean in m]).T
```

Проверим гипотезы о том, что вектор средних равен вектору $\mu_1 =$ `m` и вектору $\mu_2 =$ `m + 0.7`.

In[]:
```
m1 = m
m2 = m + 0.7
```

In[]:
```
print "m1 = " + str(m1)
print "m2 = " + str(m2)
print "data average: " + str(data.mean(axis=0))
```

```
m1 = [ 3.  4.  3. -1.  3.  1. -4.]
m2 = [ 3.7  4.7  3.7 -0.3  3.7  1.7 -3.3]
data average: [ 2.96616197  3.93306745  3.02924759 -1.00465331  2.96818385  1.00289434
 -3.94421797]
```

Для построения требуемой статистики обозначим $\bar{X} = \left(\frac{1}{n}\sum_{i=1}^{n} x_{ij}\right)_{j=1}^{p}$, $\hat{\Sigma}$ – выборочная ковариационная матрица. Тогда, благодаря знанию о распределении обобщённой статистики Хотеллинга, мы можем определить

$$t(\mu) = \frac{n-p}{p(n-1)} n \left(\bar{X} - \mu\right)^T \hat{\Sigma}^{-1} \left(\bar{X} - \mu\right) \sim F(p, n-p)$$

In[]:
```
sigma = inv(np.cov(data, rowvar=False)) # inverted sample covariance matrix,
                                        # rowvar=False means that
                                        # variables are columns and observations are rows
t = lambda mu: (data.mean(axis=0) - mu).dot(sigma).dot(
               data.mean(axis=0) - mu)*n*(n-p)/(p*(n-1))
critical_value = f(p, n-p).ppf(0.95)
print "критическое значение: " + str(critical_value)
print "t(m1) = " + str(t(m1))
print "t(m2) = " + str(t(m2))
```

```
критическое значение: 2.01878444894
t(m1) = 1.47489934781
t(m2) = 545.57769242
```

Видно, что <> и <> варианты хорошо различаются критерием.

1.2 Проверка гипотезы $H_0 : \mu_1 = \mu_2$

Для демонстрации работы критерия используем данные об американских городах.

In[]:
```
data = pd.read_table("cities.txt", index_col=0, decimal="," )
data.describe()
```

	OLD	BLACK%	ASIAN%	HISP%	DEATH	POP_CH	\
count	77.000000	77.000000	77.000000	77.000000	77.000000	77.000000	
mean	11.772727	24.267532	5.212987	13.674026	12.038961	13.688312	
std	3.079116	18.471453	9.284726	16.180295	3.532971	24.027462	
min	3.600000	1.300000	0.600000	0.400000	4.900000	-18.600000	
25%	10.000000	9.600000	1.200000	2.600000	9.900000	-4.900000	
50%	11.900000	22.000000	2.000000	5.400000	11.300000	6.500000	
75%	13.700000	31.500000	4.800000	23.000000	13.900000	25.600000	
max	22.200000	75.700000	70.500000	69.000000	23.200000	94.600000	

	POPDEN	CRIME	INCOME	UNEMP	...	\
count	77.000000	77.000000	77.000000	77.000000	...	
mean	4914.038961	10255.584416	21674.662338	6.832468	...	
std	3995.573042	2782.124691	9559.022570	2.142824	...	
min	145.000000	5364.000000	3.000000	2.300000	...	
25%	2411.000000	8537.000000	20747.000000	5.400000	...	
50%	3546.000000	9958.000000	24819.000000	6.400000	...	
75%	6526.000000	11326.000000	27555.000000	7.700000	...	
max	23671.000000	18953.000000	32451.000000	13.100000	...	

	SCHOOL	DEGREE	ASSIST	GROSS	CONDOM	LAB_F	\
count	77.000000	77.000000	77.000000	77.000000	77.000000	77.000000	
mean	87.115584	22.212987	10.216883	445.831169	5.387013	58.157143	
std	6.028792	6.587435	4.990607	97.271785	4.501092	4.993402	
min	70.800000	8.100000	2.700000	308.000000	0.700000	48.100000	
25%	82.700000	17.800000	6.400000	379.000000	2.500000	54.400000	
50%	88.900000	22.000000	9.300000	422.000000	4.200000	58.000000	
75%	91.600000	26.600000	13.600000	476.000000	7.100000	61.300000	
max	96.100000	40.600000	26.100000	755.000000	29.700000	71.000000	

	MANLAB	TRANSP	TEMPER	PRECEP
count	77.000000	77.000000	77.000000	77.000000
mean	14.066234	9.462338	77.351948	32.763636
std	5.516211	10.372040	6.472403	14.912260
min	3.600000	0.200000	58.400000	4.100000
25%	9.900000	2.900000	73.500000	17.500000
50%	12.800000	4.700000	77.800000	36.300000
75%	16.700000	12.600000	82.000000	43.500000
max	31.300000	53.400000	93.500000	64.000000

[8 rows x 24 columns]

Предположим, что средний процент афроамериканского, азиатского и испанского населения в южных штатах и на среднем западе не отличается. Для проверки этого предположения разделим данные на относящиеся к северным и южным штатам:

In[]:
```
midwest_states = ["IL", "IN", "MI", "OH", "WI", "IA", "KS", "MN", "MO", "NE", "ND", "SD"]
south_states = ["DE", "FL", "GA", "MD", "NC", "SC", "WA", "VA", "WV"]
midwest_data = data[data["STATE"].apply(lambda state: state in midwest_states)]
south_data = data[data["STATE"].apply(lambda state: state in south_states)]
```

In[]:
```
BLACK = "BLACK%"
ASIAN = "ASIAN%"
HISP = "HISP%"
```

Задача сводится к проверке равенства векторов средних значений для различных групп штатов. Для проверки равенства $\mu_{midwest}$ и μ_{south} используем статистику Махаланобиса:

$$D^2 = \left(\bar{X}_m - \bar{X}_s\right)^T \hat{\Sigma}^{-1} \left(\bar{X}_m - \bar{X}_s\right)$$

для которой при объёмах выборок N_m и N_s известно распределение

$$\frac{N_m + N_s - p - 1}{p} \frac{N_m N_s}{(N_m + N_s)(N_m + N_s - 2)} D^2 \sim F(p, N_m + N_s - p - 1),$$

где p – размерность наблюдений, в нашем случае – 3.

In[]:
```
n_m = len(midwest_data)
n_s = len(south_data)
p = 3
nations_percent = [BLACK, ASIAN, HISP]
sigma = ((n_m - 1)*np.cov(midwest_data[nations_percent], rowvar=False) +
        (n_s - 1)*np.cov(south_data[nations_percent], rowvar=False)) / (n_m + n_s - 2)
D = (midwest_data[nations_percent].mean() - south_data[nations_percent].mean()).dot(
     inv(sigma)).dot(
     midwest_data[nations_percent].mean() - south_data[nations_percent].mean())
```

In[]:
```
t = (n_m + n_s - p - 1) * n_m * n_s * D / (p * (n_m + n_s) * (n_m + n_s - 2))
critical_value = f(p, n_m + n_s - p - 1).ppf(0.95)
print "критическое значение: " + str(critical_value)
print "t = " + str(t)
print "p-value = " + str(1-f(p, n_m + n_s - p - 1).cdf(t))
```

```
критическое значение: 3.02799838233
t = 0.828298961229
p-value = 0.491857608468
```

Значение статистики существенно ниже критического значения, поэтому мы не можем отвергнуть гипотезу о равенстве среднего числа афроамериканцев, азиатов и испанцев на Юге и Среднем Западе. Посмотрим на сами данные:

In[]:
```
print midwest_data[nations_percent].mean()
print south_data[nations_percent].mean()
```

```
BLACK%    29.406667
ASIAN%     2.073333
HISP%      4.033333
dtype: float64
BLACK%    33.433333
ASIAN%     2.625000
HISP%      8.241667
dtype: float64
```

Средний процент азиатов действительно совпадает, тогда как количество афроамериканцев и азиатов различается на 4%. Стоит ли считать существенным это различие?