Faculté des sciences appliquées

MATH0487-2
Eléments de statistique

Partie 2 du projet personnel

Antoine Wehenkel

Année académique 2015-2016

22 octobre 2015

1 Estimation

a

Après calcul des moyennes des 100 échantillons on calcule le biais en faisant la différence de la moyenne des moyennes moins la valeur de la moyenne de la population. On obtient la valeur 0,0406. Pour calculer la variance de l'estimateur on utilise la formule habituel de la variance, on obtient la valeur 0,3319.

b

En appliquant des méthodes similaires au point précédent(sur les même échantillons) sur les médianes on obtient un biais de -0,0587 et une variance de 0,4358. On voit que comme la théorie le prévoit les biais sont proches(et quasiment nulle) mais la variance de l'estimateur médiane est plus élevée.

c

En répétant l'expérience pour des échantillons de 50 étudiants on obtient les résultats suivants :

- Médiane
  - biais : -0,0698
  - variance : 0,2264
- Moyenne
  - biais : -0,0537
  - variance : 0,1451

On observe que les variances ont diminuées et que les biais restent plûtot identiques. En effet cela est logique les échantillons étant plus grands les estimateurs sont en moyenne plus proches de la vraie valeur de la moyenne et sont donc plus proches les uns des autres. On voit ici que la principale différence entre les deux estimateurs est la variance qui est plus faible pour la moyenne(comme la théorie le prévoit).

d

En utilisant premièrement une loi de student(avec 19 degrés de liberté) pour créer nos intervalles de confiance les intervalles sont donnés par la formule suivante : $m_X - t_{1-\alpha/2}\frac{\sigma}{\sqrt{n}} \leq \mu \leq m_X + t_{1-\alpha/2}\frac{\sigma}{\sqrt{n}}$
Où n vaut 20 et α vaut 5. On obtient 94 intervalles contenant la vraie valeur de la moyenne cela est proche des 95 attendus par définition de la construction de l'intervalle de confiance. Pour la loi normale on utilise les même intervalles en remplaçant simplement t par le u adéquat. On obtient 91 intervalles contenant la valeur de la population. Ceci est plus éloigné des 95 attendus en effet nous construisons notre intervalle par une loi normale alors que la taille des échantillons est inférieure à 30, dans un cas pareil l'intervalle donné par une loi de student est meilleur. On peut donc conclure que les intervalles sont assez correcte et que donc les notes moyennes suivent une loi normale. De plus si l'on augmente la taille des échantillons les intervalles donnés par la loi normale contiennent 95 fois sur 100 la vraie valeur de la moyenne. Il est donc bien raisonable de supposer la variable parente gaussienne.

2 Tests d'hypothèse

a

Pour tester l'hypothèse H_0 que plus d'un cinquième des étudiants réussissent le cours de statistiques on utilise un intervalle de confiance sur une proportion. Ici on rejete H_0 uniquement si le taux d'échec est supérieur à 20% avec un seuil de signification α qui vaut 5%. Un institut de sondage considerera donc l'hypothèse H_0 à rejeter si et seulement si le taux d'échec est supérieur au seuil donné par la formule suivante : $seuil = f + u_{95}\sqrt{\frac{f(1-f)}{n}}$ où f est le taux d'échec donné par H_0 c'est à dire 0,2. On obtient en moyenne que sur 100 échantillons les autorité de l'université rejette H_0 environ 9 ou 10 fois. Or nous savons que H_0 est vraie et que donc avec un seuil α de 5% l'université ne devraient se tromper que 5 fois sur 100. Cependant nous utilisons ici un intervalle de confiance qui n'est normalement valable que quand le minimum de $\{nf, n(1-f)\}$ est plus grand que 5 or ici il vaut au minimum 4. Une partie de l'erreur provient donc de celà.

b

En moyenne dans 51% des cas il y a eu un article dans la gazette locale en effet nous avons observé à la question précédente que notre test était correct avec une probabilité de 0,91 ici nous faisons 7 tests. La probabilité pour qu'aucun ne se trompe ou qu'il soit tous corrects est donc donnée par $0,91^7 = 0,52$ et donc la probabilité d'avoir au moins un test faux est de $1 - 0,52 = 0,48$ ce qui est très proche de la valeur obtenue. On voit ici tout le problème que pose les tests mutliples on a presque une chance sur deux de dire quelque chose de faux dans la gazette locale.

c

Premièrement les instituts pourraient utiliser un seuil de signification α plus petit afin d'avoir un risque qu'au moins un des instituts se trompe vale 5%. Les instituts pourraient également travailler en collaboration sur un même échantillon, étant plus nombreux ils pourraient prendre des échantillons plus grands.

A Q1A

```
%Cette fonction renvoie le biais et la variance de l'estimateur m_x sur
%"nbr_sample" échantillons iid de "size_sample" notes finales et sauvegarde les é
    chantillons créé dans sample.mat.
function [biais_mean, variance_mean] = Q1Af(nbr_sample, size_sample)
%% Import the data
[~, ~, raw] = xlsread('probalereSession20142015.xls','Données');
raw = raw(2:end,:);

%% Create output variable
data = reshape([raw{:}],size(raw));
notes_moyennes = sum(data, 2)/9;
pop_mean = mean(notes_moyennes);%moyenne de la population
sample_mat = generateSample(nbr_sample, size_sample);
sample_mean = mean(sample_mat, 2);
save('sample.mat', 'sample_mat');
biais_mean = mean(sample_mean) - pop_mean;
variance_mean = var(sample_mean, 1);
end
```

```
%Ce script donne le biais et la variance de l'estimateur m_x sur 100
%échantillons iid de 20 notes finales.
[biais, variance] = Q1Af(100, 20)
```

B Q1B

```
%Cette fonction renvoie le biais et la variance de l'estimateur m_x sur
%"nbr_sample" échantillons iid de "size_sample" notes finales à partir des échantillons
    sauvegardés dans sample.mat.
function [biais_median, variance_median] = Q1Bf(nbr_sample, size_sample)
%% Import the data
[~, ~, raw] = xlsread('probalereSession20142015.xls','Données');
raw = raw(2:end,:);

%% Create output variable
data = reshape([raw{:}],size(raw));
notes_moyennes = sum(data, 2)/9;
pop_mean = mean(notes_moyennes);
load('sample.mat');
sample_median = median(sample_mat, 2);
biais_median = mean(sample_median) - pop_mean;
variance_median = var(sample_median, 1);
end
```

```
%Ce script ecrit le biais et la variance de l'estimateur median_x sur 100
%échantillons iid de 20 notes finales.
[biais, variance] = Q1Bf(20,100)
```

C Q1C

```
%Cet script ecrit le biais et la variance de l'estimateur m_x et de l'estimateur median_x
    sur 100
%échantillons iid de 50 notes finales.
[biais_mean, variance_mean] = Q1Af(100, 50)
[biais_median, variance_median] = Q1Bf(100, 50)
```

D Q1D

```
function [nbre_normale, nbre_student] = Q1D
%% Import the data
[~, ~, raw] = xlsread('probalereSession20142015.xls','Données');
raw = raw(2:end,:);
%% Create output variable
data = reshape([raw{:}],size(raw));
notes_moyennes = sum(data, 2)/9;
nbr_sample = 100;
size_sample = 20;
pop_mean = mean(notes_moyennes);
t = 2.093;
u = 1.96;
sample = generateSample(nbr_sample, size_sample);
mean_sample  = mean(sample, 2).';
sn_1 = std(sample.', 1);
borne_student = [mean_sample - t*sn_1/(size_sample)^(1/2); mean_sample + t*sn_1/(
    size_sample)^(1/2)];
borne_normale = [mean_sample - u*sn_1/(size_sample)^(1/2); mean_sample + u*sn_1/(
    size_sample)^(1/2)];
nbre_student  = 0;
nbre_normale = 0;
%On compte le nombre de notes dans les intervalles.
for i = 1 : nbr_sample
    if(pop_mean > borne_student(1, i) && pop_mean < borne_student(2, i))
        nbre_student = nbre_student + 1;
    end
    if(pop_mean > borne_normale(1, i) && pop_mean < borne_normale(2, i))
        nbre_normale = nbre_normale + 1;
    end
end


end
```

E Generation de sample

```
function [sample] = generateSample(nbre_sample, size_sample)
%% Import the data
[~, ~, raw] = xlsread('probalereSession20142015.xls','Données');
raw = raw(2:end,:);

%% Create output variable
data = reshape([raw{:}],size(raw));
notes_moyennes = sum(data, 2)/9;
%On génère les moyennes de n échantillons de k élèves pour l'exercice 1
sample = zeros(nbre_sample, size_sample);

for j = 1 : nbre_sample
    sample_tmp = randsample(length(notes_moyennes), size_sample, true);
    for i = 1 : size_sample
        sample(j, i) = notes_moyennes(sample_tmp(i));
    end
end
end
```

F Q2A

```
%Retourne le nombre de rejet de l'hypothèse
function [nbre_rejet] = Q2A
```

```
val_crit = 0.2 + 1.645*(0.2*0.8/20)^(1/2);
proportion_rate = zeros(7, 100);
rejet = zeros(100,1);
%On fait 100 tests pour les 7 instituts
for i = 1 : 100
    sample = generateSample(7, 20);
    for j = 1 : 7
        proportion_rate(j, i) = length(find(sample(j, :) < 10))/20;%Pourcentage de ratage
            de l'échantillon.
        %On ne regarde que la premiere gazette.
        if proportion_rate(1, i) > val_crit
            rejet(i) = 1;
        end
    end
end
nbre_rejet = length(find(rejet > 0));
end
```

G Q2B

```
%% Retourne le nombre de rejets.
function [nbre_rejet] = Q2B
val_crit = 0.2 + 1.645*(0.2*0.8/20)^(1/2);
proportion_rate = zeros(7, 100);
rejet = zeros(100,1);
for i = 1 : 100
    sample = generateSample(7, 20);
    for j = 1 : 7
        proportion_rate(j, i) = length(find(sample(j, :) < 10))/20;
        if proportion_rate(j, i) > val_crit
            rejet(i) = 1;
        end
    end
end
nbre_rejet = length(find(rejet > 0))
end
```