Université de Liège
Faculté des Sciences Appliquées

Cours de statistique
Projet 1 : Rapport

Mormont Romain
3ème bachelier ingénieur civil, orientation ingénieur civil
Options *informatique* et *électricité et électronique*
s110940

Année académique 2013-2014

Table des matières

1 Analyse descriptive

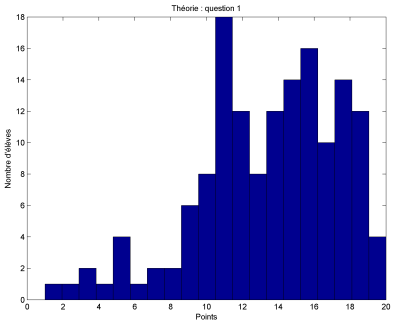
1.1 Point (a)

Les histogrammes des fréquences pour les questions 1, 2 et 3 de théorie sont respectivement donnés sur les Figures 1(a), 1(b) et 1(c). On constate qu'une majorité des étudiants ont eu plus de la moitié pour la question 1. La cote apparaissant le plus souvent (le mode) pour cette même question est le 11 (18 étudiants).

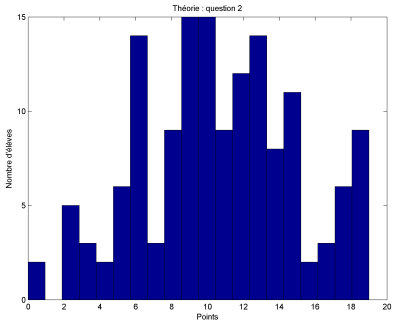
Pour la question 2, la majorité des étudiants a une cote dans l'intervalle $[6, 15]$. Les notes apparaissant le plus souvent sont 9 et 10 (15 étudiants pour chacune de ces cotes).

Pour la question 3, la répartition des cotes est plus ou moins uniforme et tourne autour de 7 élèves par cote. On constate néanmoins que 17 étudiants ont eu 0 et qu'aucun étudiant n'a eu 9.

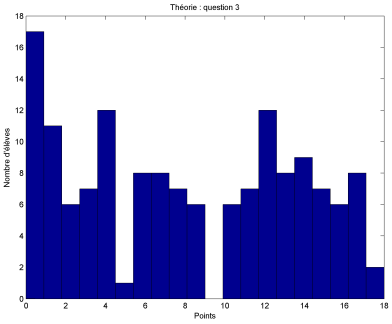
De manière générale, la répartition des fréquences pour les différentes questions nous indique que la Question 1 a été, dans l'ensemble, mieux réussie que la Question 2 qui a elle-même été mieux réussie que la Question 3.



(a) Théorie 1



(b) Théorie 2



(c) Théorie 3

FIGURE 1 – Fréquences pour les questions de théorie

1.2 Point (b)

Les moyennes, modes, médianes et écarts types sont donnés dans la Table 1. On constate que l'**exercice 2** a été **mieux réussi** que les deux autres. En effet, plus de la moitié des élèves ont eu 16 sur 20 ou plus et la cote la plus représentée est 20.

L'**exercice 3** a été quant à lui le **moins bien réussi** car plus de la moitié des élèves ont un moins de 5 sur 20 (ou 5 sur 20) avec une majorité d'élèves ayant eu 2.

En ce qui concerne les intervalles *normaux* $[\mu_i - \sigma_i, \mu_i + \sigma_i]$, on trouve :
- pour la **question 1** : 64.86% des étudiants dans l'intervalle $[3.5549, 14.0937]$ (96 étudiants).
- pour la **question 2** : 57.43% des étudiants dans l'intervalle $[10.3998, 19.6948]$ (85 étudiants)
- pour la **question 3** : 73.65% des étudiants dans l'intervalle $[1.4892, 9.6730]$ (109 étudiants)

Exercice	**Moyenne**	**Médiane**	**Mode**	**Écart-type**
Question 1	8.8243	8	6	5.2694
Question 2	15.0473	16	20	4.6475
Question 3	5.5811	5	2	4.0919

TABLE 1 – Exercices d'examen

1.3 Point (c)

Les boîtes à moustaches mettent en évidence des résultats aberrants pour les trois projets mais pas pour la question d'examen sur le projet (voir Figure 2). Les valeurs aberrantes sont reprises dans la Table 2.

Les premier et troisième quartiles des résultats des projets et de la question d'examen sur le projet 3 sont donnés dans la Table 3.

Projet	**Notes aberrantes**
1	0 (× 4), 5.5, 7, 9
2	0 (× 4), 7, 11, 11.5, 12
3	0 (× 4), 8.11, 10.33, 10.5, 12 (× 3)

TABLE 2 – Résultats aberrants pour les projets

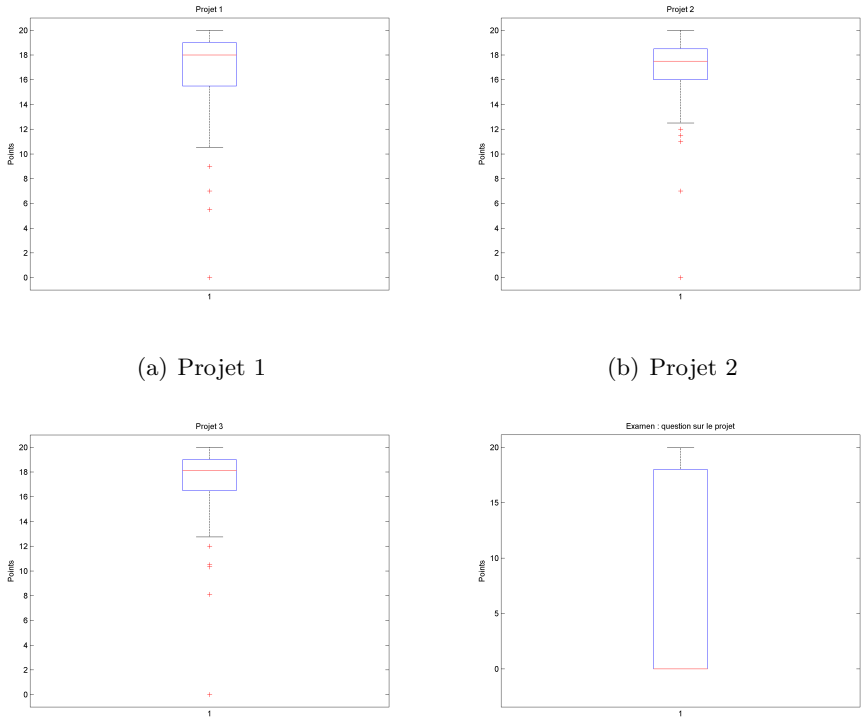
Quartile	**Projet 1**	**Projet 2**	**Projet 3**	**Examen**
1^{er}	15.5	16	16.5	0
$3^{\text{ème}}$	19	18.5	19	18

TABLE 3 – Premier et troisième quartiles

1.4 Point (d)

Les graphiques des fréquences relatives cumulées pour les moyennes des questions de théorie et des questions d'exercice sont donnés respectivement sur les Figures 3(a) et 3(b). La proportion d'étudiant ayant obtenu une cote dans un certain intervalle $[a, b]$ est obtenue à l'aide de la fonction des fréquences relatives cumulées F :

$$F(a \leq x_i \leq b) = F(x_i = b) - F(x_i = a)$$



(a) Projet 1

(b) Projet 2

(c) Projet 3

(d) Examen : question sur le projet

FIGURE 2 – Résultat des projets et de la question sur le projet

Les proportions obtenues pour l'intervalle $[12, 15]$ pour les moyennes théorie et l'exercice sont respectivement **18.24**% et **22.30**%. On constate, de plus, que la forme des graphes est similaire à la forme du graphe théorique de la fréquence relative cumulée pour une loi normale (surtout pour la moyenne de théorie).

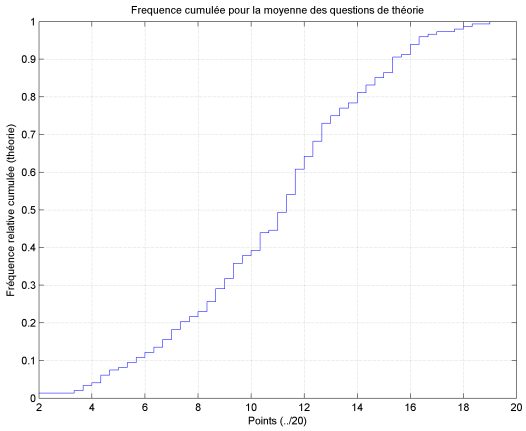
1.5 Point (e)

Le scatterplot entre les résultats du projet 3 et de la question sur le projet 3 est donné sur la Figure 4. Le coefficient de corrélation obtenu est **0.2106**. En se basant sur ce coefficient de corrélation, on ne peut pas tirer de conclusion de l'influence de la réussite ou non du projet 3 sur la réussite ou non de la question sur le projet 3. En effet, on observe qu'une même proportion de personne ayant réussi le projet 3 (cote > 10) a réussi et a raté la question sur le projet 3.

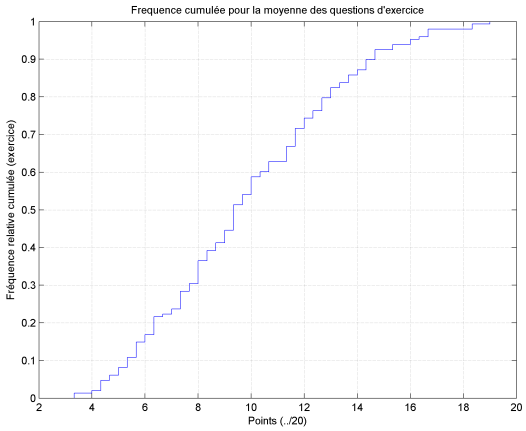
2 Calcul de statistiques sur échantillons

2.1 Remarque sur `datasample`

Les échantillons sont calculés avec une implémentation propre de la fonction `datasample`. Le choix d'implémenter cette fonction est du au fait qu'elle n'est pas présente sur la version R2010a de Matlab.



(a) Théorie



(b) Exercice

FIGURE 3 – Fréquences relatives cumulées

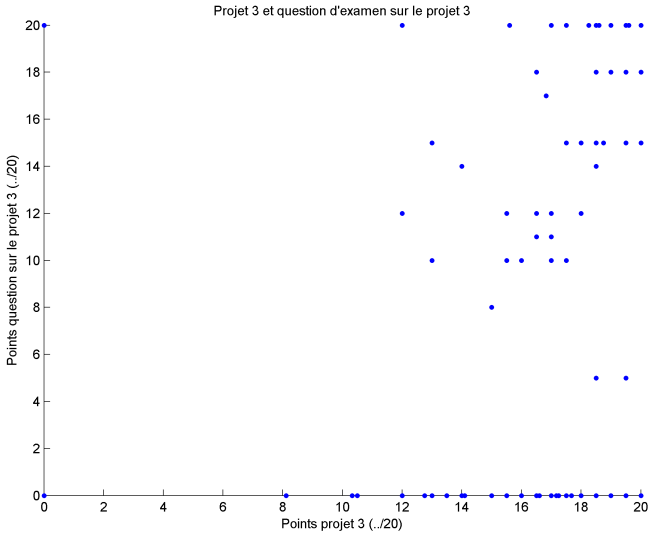


FIGURE 4 – Scatterplot entre les résultats du projet 3 et de la question sur le projet 3

Cette implémentation utilise la fonction `randi` qui effectue bien un tirage *aléatoire*[1] avec remise et permet donc, comme espéré, d'obtenir un échantillon i.i.d.

2.2 Point (a)

(i) Un échantillon généré aléatoirement à l'aide de `datasample` est donné dans la Figure 5 et les moyennes, médianes et écart types pour les résultats des exercices sont données dans la Table 4.

On observe une légère imprécision sur les statistiques calculées sur base d'un échantillon comparées à celles calculées sur base de la population. En effet, la sélection d'un échantillon

1. Le tirage étant en fait **pseudo aléatoire** et basé sur un *seed*, il pourrait être nécessaire d'initialiser ce seed avec une autre valeur que la valeur par défaut afin de ne pas générer des séries de valeurs identiques à chaque ouverture de Matlab. Néamoins, cette opération ne me semble pas nécessaire étant donné l'ampleur de ce projet.

15, 19, 21, 24, 42, 63, 72, 81, 94, 118, 119,
121, 135, 136, 136, 142, 142, 143, 143, 144

FIGURE 5 – Index des individus de l'échantillon étudié

	Exercice	**Moyenne**	**Médiane**	**Ecart type**
Échantillon	1	9.7500	9.0000	5.6557
	2	15.600	17.000	4.7617
	3	4.6000	3.5000	4.1218
Population	1	8.8243	8.0000	5.2694
	2	15.0473	16.000	4.6475
	3	5.5811	5.0000	4.0919

TABLE 4 – Moyennes, médianes et écart types pour les exercices

provoque une **perte d'information** par rapport à la population. On constate aussi, en comparant aux statistiques de la population, que l'écart-type varie moins que les autres statistiques.

(ii) Dans la continuité de ce qui a été dit au point précédent, on constate que la perte d'information liée à la sélection d'un échantillon entraîne des différences (moins ou plus de données aberrantes, déplacement des quartiles,...) entre les boîtes à moustaches tracées pour la Question 1 et celle donnée sur la Figure 6. On peut néanmoins observer que, malgré ces variations évidentes, les boîtes sont positionnées de la même manière qu'à la question 1. Cette observation n'est pas surprenante étant donné que l'échantillon a été tiré de la population qui a donné les premières boîtes.

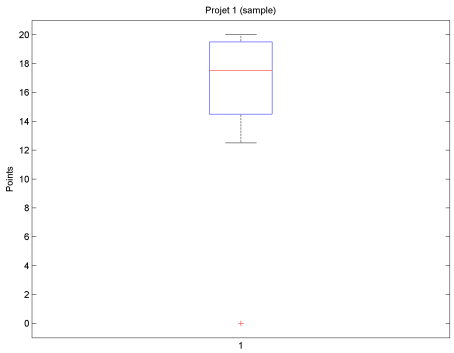
(iii) Les courbes des fréquences cumulées pour la moyenne des questions de théorie pour l'échantillon et la population sont données sur la Figure 7. On constate que la fonction relative à l'échantillon contient moins de marches que celle de la population. Encore une fois, cette observation n'est pas surprenante étant donné que le nombre d'individus est beaucoup plus petit dans le premier cas. De ce fait, la fonction est moins proche de la courbe théorique (distribution normale) que la courbe de la population.

La distance de Kolmogorov-Smirnov entre les deux courbes est calculée à l'aide de la fonction `kstest2`. La distance de K-S entre les fonctions pour l'échantillon et la population est **0.0986**. Cette valeur, qui est relativement faible, indique que les deux distributions suivent probablement de la même loi de probabilité.

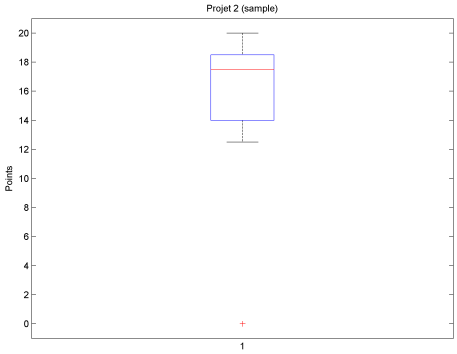
2.3 Point (b)

Les histogrammes pour les sous-questions (*i*), (*ii*) et (*iii*) sont donnés sur la Figure 8.

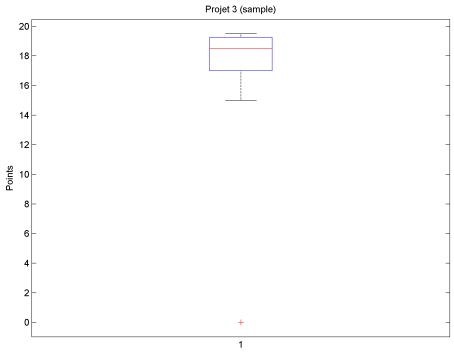
(i) La moyenne des moyennes (pour l'exercice 1) est **8.8645**. Cette moyenne est **proche** de la moyenne pour la population et bien plus précise que la valeur obtenue avec un seul échantillon. On peut expliquer cette précision accrue par le fait que l'augmentation du nombre d'échantillon a *atténué la perte d'information.* En effet, pour 100 échantillons le nombre d'individus différents sondés a fortement augmenter.



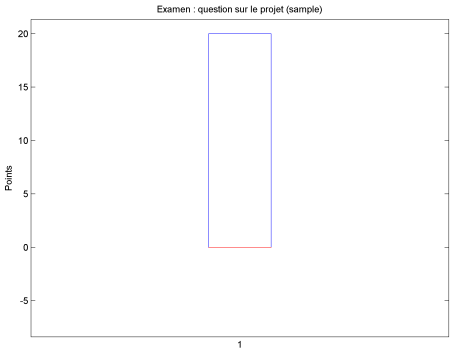
(a) Projet 1



(b) Projet 2



(c) Projet 3



(d) Question projet 3

FIGURE 6 – Résultat des projets pour l'échantillon

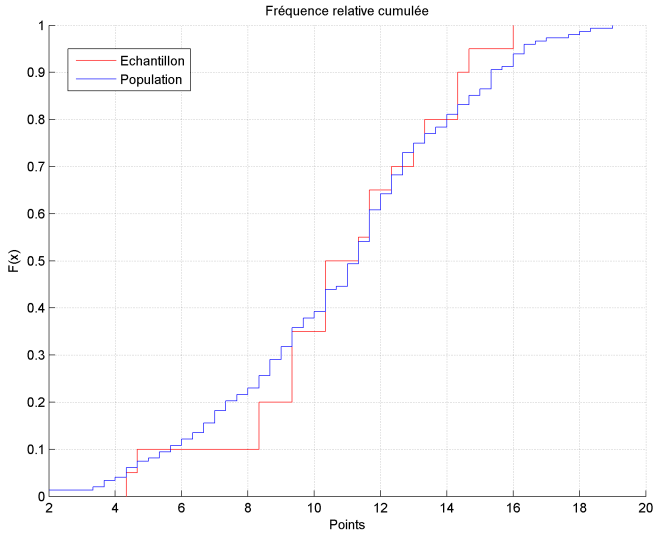


FIGURE 7 – Fréquences cumulées pour la moyenne des questions de théorie

L'allure de l'histogramme rappelle la **loi normale**. En effet, on constate une accumulation d'individus autour de la moyenne de cette variable et une décroissance de la

fonction de part et d'autre de cette moyenne.

(ii) La moyenne des médianes est **8.2250**. Cette valeur est, comme au point $2.b.(i)$, plus précise que celle trouvée pour un seul échantillon.

On constate, à nouveau, l'accumulation des médianes autour de la moyenne des médianes mais la décroissance de part et d'autre de la moyenne est moins évidente que pour la moyenne des moyennes.

(iii) La moyenne des écart-types est **5.3321**. Encore une fois, cette valeur est plus précise que pour un seul échantillon.

Pour ce qui est du graphe, on retrouve une loi normale sur base des mêmes observations que celles énoncés aux points précédents.

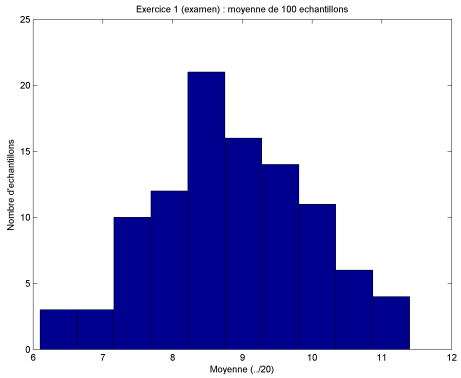
(iv) et (v) Les histogrammes des distances de Kolmogorov-Smirnov entre les fonctions de fréquences cumulées des exercices pour 100 échantillons par rapport à la population sont donnés sur la Figure 9.

L'allure des histogrammes rappelle une loi normale. La distance de K-S pour un échantillon aléatoire et la population dont l'échantillon est extrait évolue selon une loi normale.

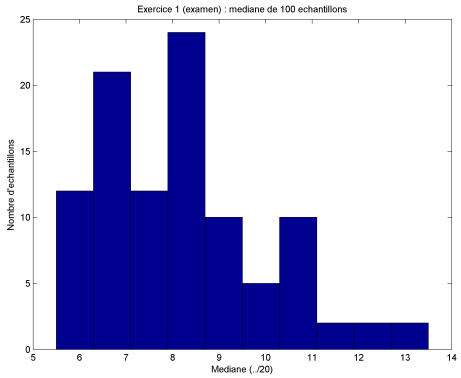
3 Annexe

3.1 Code

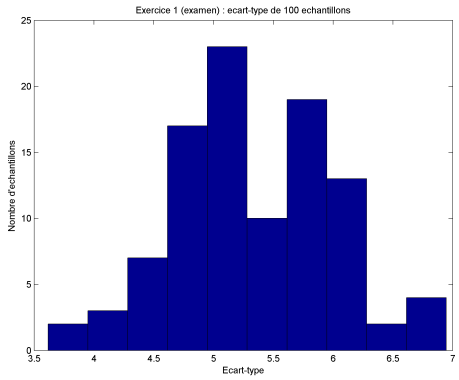
```
clear all;
close all;

% Load data
s_score = load('stat_data.mat');
score = s_score.m;
clear s_score;
% Splitting different columns into different variables
exam_exer = score(:,8:10);
exam_theo = score(:,5:7);
exam_proj = score(:,4);
projects = score(:,1:4);
n_students = length(score(:,1));

%% Question 1
% Point a)
figure; hist(exam_theo(:,1), 20); % theorie 1
title('Theorie : question 1');
ylabel('Nombre d''eleves');
xlabel('Points');

figure; hist(exam_theo(:,2), 20); % theorie 2
title('Theorie : question 2');
ylabel('Nombre d''eleves');
xlabel('Points');
```



(a) Moyenne



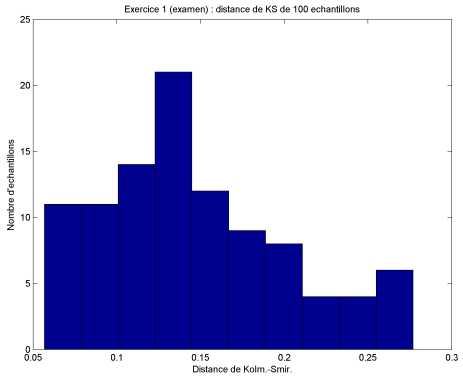
(b) Médiane



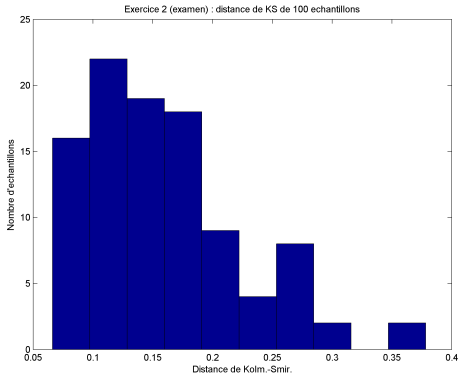
(c) Écart-type

FIGURE 8 – Calculs statistiques pour l'exercice 1 (100 échantillons)

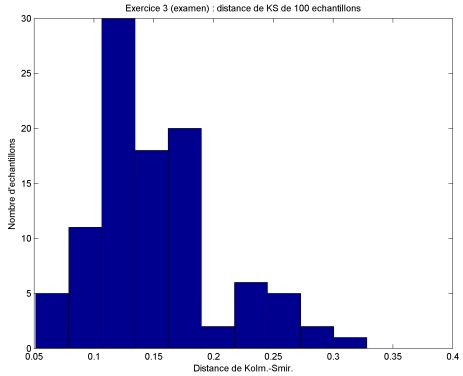
```
26
27 figure; hist(exam_theo(:,3), 20); % theorie 3
28 title('Theorie : question 3');
29 ylabel('Nombre d''eleves');
30 xlabel('Points');
31
32 % Point b)
33 ex_mean  = mean(exam_exer);
34 ex_median = median(exam_exer);
35 ex_mode  = mode(exam_exer);
36 ex_ectype = std(exam_exer, 1);
37
38 norm_min_threshold = ex_mean - ex_ectype;
39 norm_max_threshold = ex_mean + ex_ectype;
40
41 n_students_normal_exer = [length(find(exam_exer(:,1) >= norm_min_threshold(1) &
       exam_exer(:,1) <= norm_max_threshold(1) )) ...
42         length(find(exam_exer(:,2) >= norm_min_threshold(2) & exam_exer(:,2) <=
               norm_max_threshold(2) )) ...
43         length(find(exam_exer(:,3) >= norm_min_threshold(3) & exam_exer(:,3) <=
               norm_max_threshold(3) ))];
44
45 % Point c)
46 figure; boxplot(projects(:,1));
```



(a) Exercice 1



(b) Exercice 2



(c) Exercice 3

FIGURE 9 – Distance de Kolmogorov-Smirnov de 100 échantillons par rapport à la population

```
47  title('Projet 1');
48  ylabel('Points');
49
50  figure; boxplot(projects(:,2));
51  title('Projet 2');
52  ylabel('Points');
53
54  figure; boxplot(projects(:,3));
55  title('Projet 3');
56  ylabel('Points');
57
58  figure; boxplot(exam_proj);
59  title('Examen : question sur le projet');
60  ylabel('Points');
61
62  proj_Q1 = quantile(projects, 0.25);
63  proj_Q3 = quantile(projects, 0.75);
64
65  min_threshold = proj_Q1 - 1.5 * (proj_Q3 - proj_Q1);
66  max_threshold = proj_Q3 + 1.5 * (proj_Q3 - proj_Q1);
67
68  % Finding outliers indexes
```

```
outlier_index_p1 = [find(projects(:,1) < min_threshold(1)); find(projects(:,1) >
    max_threshold(1))];
outlier_index_p2 = [find(projects(:,2) < min_threshold(2)); find(projects(:,2) >
    max_threshold(2))];
outlier_index_p3 = [find(projects(:,3) < min_threshold(3)); find(projects(:,3) >
    max_threshold(3))];
outlier_index_exam_p = [find(projects(:,4) < min_threshold(4)); find(projects(:,4) >
    max_threshold(4))];

outliers_p1 = projects(outlier_index_p1, 1).';
outliers_p2 = projects(outlier_index_p2, 2).';
outliers_p3 = projects(outlier_index_p3, 3).';
outliers_ex_proj = projects(outlier_index_exam_p, 4).';

clear outlier_index_p1 outlier_index_p2 outlier_index_p3 outlier_index_exam_p;

% Point d)
theo_mean = mean(exam_theo .').';
exer_mean = mean(exam_exer .').';

figure; cdfplot(theo_mean);
title ('Frequence cumulee pour la moyenne des questions de theorie');
xlabel('Points (../20) ');
ylabel('Frequence relative cumulee (theorie)');

figure; cdfplot(exer_mean);
title ('Frequence cumulee pour la moyenne des questions d''exercice');
xlabel('Points (../20) ');
ylabel('Frequence relative cumulee (exercice)');

[ecdf_theo x_ecdf_theo] = ecdf(theo_mean);
[ecdf_exer x_ecdf_exer] = ecdf(exer_mean);

%close all;
res_min = find(x_ecdf_exer >= 12, 1);
res_max = find(x_ecdf_exer <= 15, 1, 'last');
score12_15_exer = (ecdf_exer(res_max) - ecdf_exer(res_min));

res_min = find(x_ecdf_theo >= 12, 1);
res_max = find(x_ecdf_theo <= 15, 1, 'last');
score12_15_theo = (ecdf_theo(res_max) - ecdf_theo(res_min));

clear res_min res_max;
% Point e)
figure; scatter(projects(:,3), exam_proj, 4*pi, 'blue', ' fill ');
title ('Projet 3 et question d''examen sur le projet 3');
xlabel('Points projet 3 (../20) ');
ylabel('Points question sur le projet 3 (../20) ');

corr_proj3 = corrcoef(projects(:,3), exam_proj);
corr_coef_p3 = corr_proj3(1,2);

clear corr_proj3;
```

Listing 1 – `Question1.m`

```
clear all;
```

```
close all;

% Load data
s_score = load('stat_data.mat');
score = s_score.m;
clear s_score;
% Splitting different columns into different variables
exam_exer = score(:,8:10);
exam_theo = score(:,5:7);
exam_proj = score(:,4);
projects = score(:,1:4);
n_students = length(score(:,1));

%% Question 2.a
[sample_2a sample_index_2a] = datasample(score, 20);

% Question 2.a.i
ex_mean  = mean(sample_2a(:,8:10));
ex_median = median(sample_2a(:,8:10));
ex_ectype = std(sample_2a(:,8:10));

% Question 2.a.ii
figure; boxplot(sample_2a(:,1));
title ('Projet 1 (sample)');
ylabel('Points');

figure; boxplot(sample_2a(:,2));
title ('Projet 2 (sample)');
ylabel('Points');

figure; boxplot(sample_2a(:,3));
title ('Projet 3 (sample)');
ylabel('Points');

figure; boxplot(sample_2a(:,4));
title ('Examen : question sur le projet (sample)');
ylabel('Points');

% Question 2.a.iii
theo_mean_sample_2a = mean(sample_2a(:,5:7).').';
theo_mean = mean(exam_theo.').';

ks = ksdist(theo_mean_sample_2a, theo_mean);
[~, ~, ks_f] = kstest2(theo_mean_sample_2a, theo_mean);
figure; cdfplot(theo_mean_sample_2a);
figure; cdfplot(theo_mean);

%% Question 2.b
n_sample = 100;
sample_size = 20;
sample_2b = zeros(sample_size, n_sample);

for i=1:n_sample
        sample_2b(:,i) = datasample(exam_exer(:,1), sample_size);
end

% Resultat de la question 1 (population)
```

```
ex1_mean = mean(exam_exer(:,1));
ex1_median = median(exam_exer(:,1));
ex1_std = mean(exam_exer(:,1));

% Question 2.b.i
ex1_mean_s = mean(sample_2b).';

figure; hist(ex1_mean_s);
title('Exercice 1 (examen) : moyenne de 100 echantillons');
ylabel('Nombre d''echantillons');
xlabel('Moyenne (../20)');

mean_ex1_mean_s = mean(ex1_mean_s);

% Question 2.b.ii
ex1_median_s = median(sample_2b).';

figure; hist(ex1_median_s);
title('Exercice 1 (examen) : mediane de 100 echantillons');
ylabel('Nombre d''echantillons');
xlabel('Mediane (../20)');

mean_ex1_median_s = mean(ex1_median_s);

% Question 2.b.iii
ex1_std_s = std(sample_2b).';

figure; hist(ex1_std_s);
title('Exercice 1 (examen) : ecart-type de 100 echantillons');
ylabel('Nombre d''echantillons');
xlabel('Ecart-type');

mean_ex1_std_s = mean(ex1_std_s);

% Question 2.b.iv

ks_ex1 = zeros(100,1);

for i = 1:100
        ks_ex1(i) = ksdist(exam_exer(:,1), sample_2b(:,i));
end

figure; hist(ks_ex1, 10);
title('Exercice 1 (examen) : distance de KS de 100 echantillons');
ylabel('Nombre d''echantillons');
xlabel('Distance de Kolm.-Smir.');

% Question 2.b.v

ks_ex2 = zeros(100,1);
ks_ex3 = zeros(100,1);

for i = 1:100
        [~, ~, ks_ex2(i)] = kstest2(exam_exer(:,2), datasample(exam_exer(:,2),
            sample_size));
        [~, ~, ks_ex3(i)] = kstest2(exam_exer(:,3), datasample(exam_exer(:,3),
            sample_size));
```

```
114 end
115
116 figure; hist(ks_ex2, 10);
117 title('Exercice 2 (examen) : distance de KS de 100 echantillons');
118 ylabel('Nombre d''echantillons');
119 xlabel('Distance de Kolm.-Smir.');
120
121 figure; hist(ks_ex3, 10);
122 title('Exercice 3 (examen) : distance de KS de 100 echantillons');
123 ylabel('Nombre d''echantillons');
124 xlabel('Distance de Kolm.-Smir.');
```

Listing 2 – `Question2.m`

```
1  function [sample index_perm] = datasample(data, k)
2          % datasample(data, k)
3          %     This function returns a random sample from the matrix
               $data$.
4          %     It randomly selects $k$ lines in the matrix.
5          %
6          %     /!\ $k$ must be less than the numbers of row in $data$
7          %
8          %     PARAMETERS :
9          %           - data : the matrix from which a sample must be
               extracted
10         %           - k : the size of the sample
11         %
12         %     RETURN :
13         %           - sample : the sample extracted
14         %           - index_perm : the (row) indexes of the values
               extracted in $data$
15
16         [len_x ~] = size(data);
17
18         if(k >= len_x)
19                 error('The sample size ''k'' must be less than or equal to the number of
                       line in the matrix');
20         else
21                 index_perm = randi(len_x, k, 1);
22                 index_perm = sort(index_perm);
23                 sample = data(index_perm,:);
24         end
25 end
```

Listing 3 – `datasample.m`