



- 1 Les diagrammes en bâtons
- 2 Les diagrammes en barres
- 3 Les histogrammes
- 4 Les polygones des effectifs
- 5 Les boîtes à moustaches
- 6 Les nuages de points
- 7 Les tiges et feuilles
- 8 Les résidus
- 9 Le graphe quantile normal



http://url.univ-irem.fr/fr36

Lorsqu'on utilise l'un des outils ,  ou  pour exporter un graphique vers la vue **Graphique**, GeoGebra traduit les réglages effectués par l'utilisateur à travers l'interface de la boîte de dialogue **Analyse des données** par une commande permettant de générer un objet de type « graphique ».

Dans la majorité des cas, l'interface proposée par GeoGebra s'avère suffisante pour obtenir le graphique souhaité, mais il peut parfois arriver que l'on ait besoin d'utiliser directement une commande de création de graphique.

1 Les diagrammes en bâtons

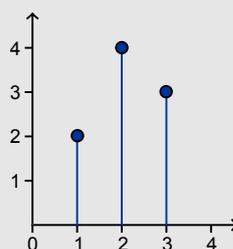
La commande **DiagrammeBâtons** permet de générer un diagramme en bâtons dans le repère de la vue **Graphique**. Plusieurs syntaxes sont possibles :

- **DiagrammeBâtons**[<liste d'abscisses>,<liste d'ordonnées>,<booléen pour bâtons horizontaux>] : en supposant que <liste d'abscisses> soit de la forme $\{x_1, x_2, \dots, x_n\}$ et que <liste d'ordonnées> soit de la forme $\{y_1, y_2, \dots, y_n\}$, cette commande crée des segments dont l'une des extrémités a pour coordonnées $(x_i; y_i)$ et dont l'autre extrémité a pour coordonnées $(x_i; 0)$ (dans le cas où <booléen pour bâtons horizontaux> a pour valeur **false**) ou bien $(0; y_i)$ (dans le cas où <booléen pour bâtons horizontaux> a pour valeur **true**).

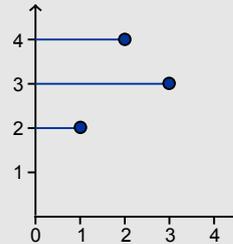
Le paramètre <booléen pour bâtons horizontaux> est optionnel et, dans le cas où il est omis, GeoGebra construit un graphique avec des segments verticaux.

Exemple(s)

 **DiagrammeBâtons** [{1, 2, 3}, {2, 4, 3}] produit le graphique :



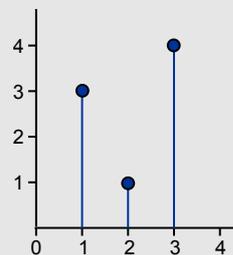
 **DiagrammeBâtons** `[{1,2,3},{2,4,3},true]` produit le graphique :



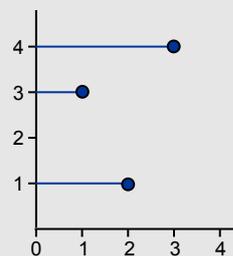
- **DiagrammeBâtons** [`<liste de points>`, `<booléen pour bâtons horizontaux>`] : cette commande permet de créer des bâtons verticaux (si le paramètre optionnel `<booléen pour bâtons horizontaux>` est omis ou bien s'il a pour valeur `false`) ou des bâtons horizontaux (si `<booléen pour bâtons horizontaux>` a pour valeur `true`)

Exemple(s)

 **DiagrammeBâtons** `[{(1,3),(2,1),(3,4)}]` produit le graphique :



 **DiagrammeBâtons** `[{(1,3),(2,1),(3,4)},true]` produit le graphique :



L'objet créé à l'aide de la commande **DiagrammeBâtons** est un objet de type « numérique » et sa valeur est égale à la somme des longueurs de tous les bâtons, ce qui correspond, dans le cadre d'une étude statistique, à l'effectif total de la population.

Ainsi, en reprenant l'exemple précédent, si $a = \text{DiagrammeBâtons} \{ \{(1,3), (2,1), (3,4)\} \}$, alors $a = 3 + 1 + 4 = 8$.

Remarque :

2 Les diagrammes en barres

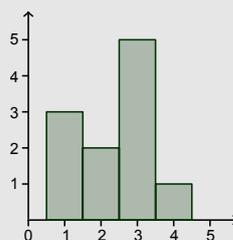
Pour produire des diagrammes en barres, GeoGebra propose la commande **Barres** qui offre plusieurs syntaxes possibles.

- **Barres** [`<données brutes>`, `<largeur des barres>`] : cette commande permet, à partir des données brutes de la série fournies sous forme d'une liste, de produire le diagramme en barres correspondant. Le paramètre, non optionnel, `<largeur des barres>` indique la largeur de chacune de barres dans l'unité relative à celle choisie pour l'axe des abscisses.

Il est possible d'indiquer une largeur des barres égale à zéro, et, dans ce cas, le diagramme en barres à l'allure d'un diagramme en bâtons.

Exemple(s)

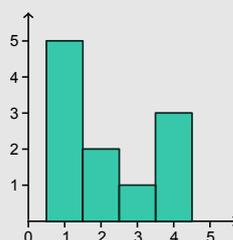
 `Barres[{1,1,1,2,2,3,3,3,3,4},1]` produit le graphique :



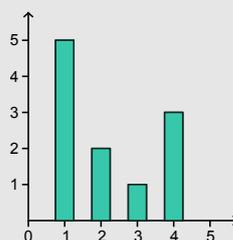
- `Barres[<liste des données>,<liste des effectifs>,<largeur des barres>]` : cette commande produit un diagramme en barres dont la hauteur est déterminée par la liste <liste des effectifs> et dont la largeur peut être précisée à l'aide du paramètre optionnel <largeur des barres>. Les listes <liste des données> et <liste des effectifs> doivent posséder le même nombre d'éléments.

Exemple(s)

 `Barres[{1,2,3,4},{5,2,1,3}]` produit le graphique :



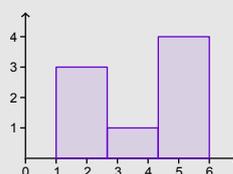
 `Barres[{1,2,3,4},{5,2,1,3},0.5]` produit le graphique :



- `Barres[<borne inf>,<borne sup>,<liste hauteurs>]` : cette commande permet de créer un diagramme dont le nombre de barres est égal au nombre d'éléments de la liste <liste hauteurs> et ce, dans l'intervalle qui a pour borne inférieure <borne inf> et pour borne supérieure <borne sup>. La largeur des barres est automatiquement calculée par le logiciel.

Exemple(s)

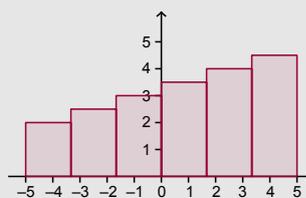
 `Barres[1,6,{3,1,4}]` produit le graphique :



- `Barres[<borne inf>,<borne sup>,<expression>,<variable>,<valeur initiale>,<valeur finale>]` : cette commande permet de créer un diagramme en barres dans l'intervalle qui a pour borne inférieure <borne inf> et pour borne supérieure <borne sup>. La liste des hauteurs résulte d'un calcul fourni par le paramètre <expression> dans lequel <variable> est un nombre qui varie entre <valeur initiale> et <valeur finale> avec un pas de un.

Exemple(s)

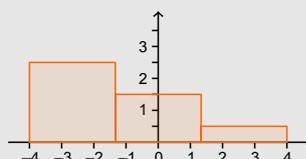
 **Barres**[-5,5,0.5*n+2,n,0,5] produit le graphique :



- **Barres**[<borne inf>,<borne sup>,<expression>,<variable>,<valeur initiale>,<valeur finale>,<pas>] : avec cette syntaxe, la variable <variable> varie entre <valeur initiale> et <valeur finale> avec des incréments de <pas>.

Exemple(s)

 **Barres**[-4,4,5-n/2,n,5,10,2] produit le graphique :



Remarque :

L'objet créé à l'aide de la commande **Barres** est un objet de type « numérique » et sa valeur est égale au produit de la largeur de chaque barre par la somme des hauteurs de toutes les barres. Il s'agit donc de l'aire totale des rectangles dessinés.
Si la largeur des barres est nulle, la variable numérique associée au diagramme en barres est également nulle.

3 Les histogrammes

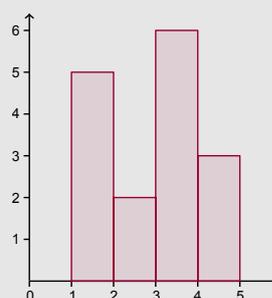
Pour produire des histogrammes, il faut utiliser la commande **Histogramme**. Celle-ci possède plusieurs syntaxes possibles :

- **Histogramme**[<bornes des classes>,<hauteurs>] : cette commande permet de construire un histogramme dont les classes ont pour bornes les valeurs fournies dans la liste <bornes des classes>. Si cette liste n'est pas ordonnée dans l'ordre croissant, GeoGebra ne renvoie pas d'erreur, néanmoins l'histogramme résultant ne saurait être satisfaisant.

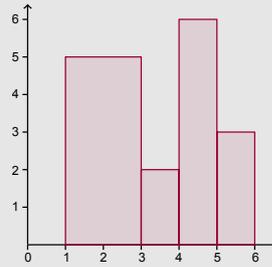
Les hauteurs des rectangles sont déterminées par la liste <hauteurs>. Cette liste doit donc contenir un élément de moins que la liste <bornes des classes> (si ce n'est pas le cas, l'histogramme n'est pas construit).

Exemple(s)

 **Histogramme**[{1,2,3,4,5},{5,2,6,3}] produit le graphique :



 **Histogramme**[{1,3,4,5,6},{5,2,6,3}] produit le graphique :



- **Histogramme**[<booléen pour cumul>,<bornes des classes>,<données brutes>,<booléen pour densité>,<échelle>] : cette commande permet de générer différents types d'histogrammes selon la valeur des paramètres.

<booléen pour cumul> est un paramètre optionnel (par défaut il vaut `false`). Si ce paramètre est fixé à `true`, l'histogramme représente les effectifs (ou les fréquences) cumulé(e)s croissant(e)s. Attention, en fixant ce paramètre, le paramètre <booléen pour densité> devient obligatoire et devra être explicitement fourni.

Le paramètre <bornes des classes> représente la liste des bornes des différentes classes (de préférence, dans l'ordre croissant).

<données brutes> est la liste (non nécessairement ordonnée) des valeurs prises par la série statistique. Si l'une des valeurs n'appartient pas à l'intervalle défini par le minimum et le maximum des bornes alors l'histogramme n'est pas construit.

Le paramètre <booléen pour densité> est un paramètre optionnel (par défaut, il vaut `true`). Lorsque ce paramètre est égal à `false`, la hauteur des rectangles est égale à l'effectif de la classe (si les classes n'ont pas toutes la même amplitude, l'aire des rectangles n'est alors pas proportionnelle à l'effectif des classes).

Lorsque <booléen pour densité> vaut `true`, l'aire des rectangles devient proportionnelle à l'effectif de chaque classe. En particulier, la hauteur des rectangles est alors déterminée par la formule :

$$\text{Hauteur} = \text{Échelle} \times \frac{\text{Effectif de la classe}}{\text{Largeur de la classe}}$$

où « Échelle » désigne la valeur de paramètre <échelle>.

Le paramètre <échelle> est également optionnel (il est cependant nécessaire de fournir explicitement la valeur du paramètre <booléen pour densité>, même si celui-ci est optionnel, pour pouvoir modifier l'échelle) qui vaut 1 par défaut. C'est un paramètre de type « numérique » qui permet d'indiquer le facteur d'agrandissement ou de réduction selon l'axe des ordonnées. Si <booléen pour densité> vaut `false`, la modification de l'échelle reste sans effet.

Si n désigne l'effectif total de la série, alors, attribuer la valeur $\frac{1}{n}$ au paramètre <échelle> permet d'obtenir un histogramme dont l'aire totale vaut 1.

Exemple(s)

On considère la liste $L = \{1, 2, 5, 6, 7, 8, 11, 12, 13, 20\}$ utilisée dans les exemples ci-dessous.

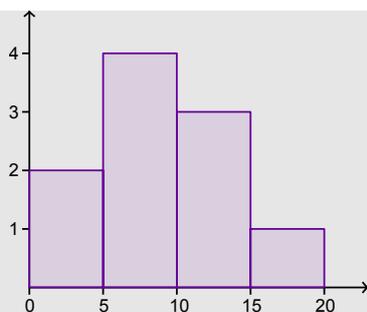
 **Histogramme**[{0,5,10,15,20},L] produit le graphique ci-contre.

Le premier rectangle a pour hauteur

$$\frac{\text{Effectif de la première classe}}{\text{Largeur de la première classe}} = \frac{2}{5} = 0,4$$

Les hauteurs des autres rectangles sont calculées de la même façon (en remarquant que la dernière classe est un intervalle fermé à droite).





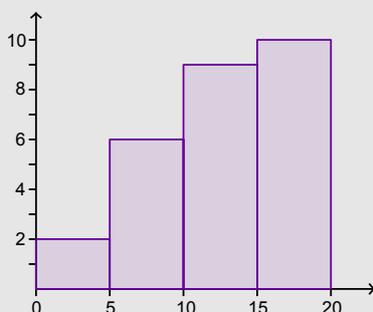
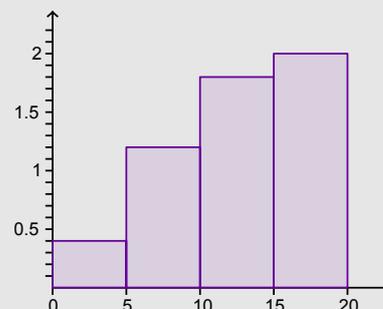
`Histogramme[{0,5,10,15,20},L,false]`
produit le graphique ci-contre.

Lorsque <booléen pour densité> vaut `false`, les hauteurs des rectangles sont les effectifs des classes.

`Histogramme[true,{0,5,10,15,20},L,true]`
produit le graphique ci-contre.

Dans cet histogramme des effectifs cumulés croissants, la hauteur du dernier rectangle est égale à

$$\frac{\text{Effectif total}}{\text{Largeur de la dernière classe}} = \frac{10}{5} = 2$$

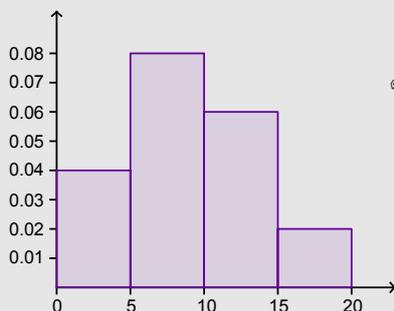
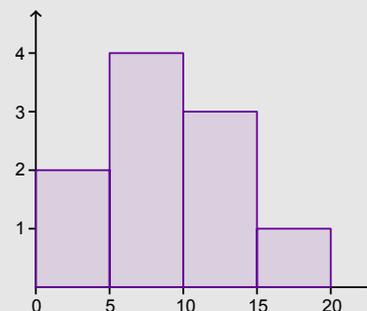


`Histogramme[true,{0,5,10,15,20},L,false]`
produit le graphique ci-contre.

Ici, <booléen pour densité> vaut `false`, donc les hauteurs des rectangles sont les effectifs cumulés croissants des classes.

`Histogramme[{0,5,10,15,20},L,true,5]`
produit le graphique ci-contre.

En affectant l'amplitude des classes au paramètre <échelle>, on obtient un histogramme dont les hauteurs des rectangles sont les effectifs des classes (à condition que toutes les classes aient la même amplitude).



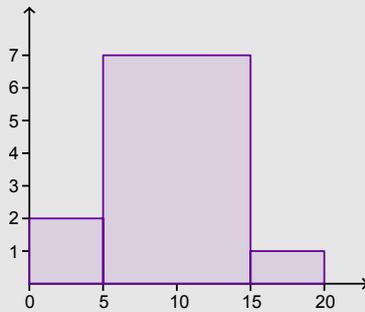
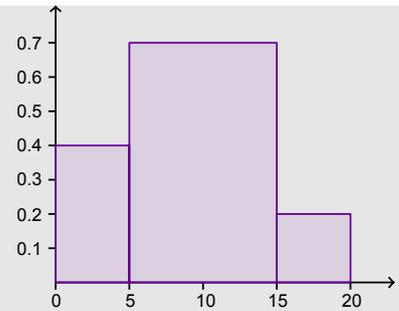
`Histogramme[{0,5,10,15,20},L,true,1/Longueur[L]]`
produit le graphique ci-contre.

Lorsque le paramètre <échelle> vaut $\frac{1}{\text{Effectif total}}$, on obtient un histogramme dont l'aire vaut 1.

 **Histogramme**[{0,5,15,20},L,true]
produit le graphique ci-contre.

Dans cet exemple, les classes n'ont pas la même largeur. La hauteur du deuxième rectangle est égale à

$$\frac{\text{Effectif de la classe}}{\text{Largeur de la classe}} = \frac{7}{10} = 0,7$$

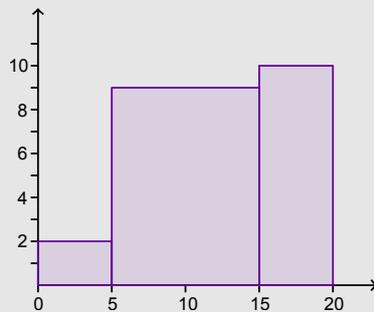
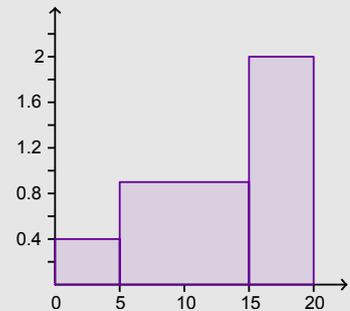


 **Histogramme**[{0,5,15,20},L,false]
produit le graphique ci-contre.

Les classes n'ont pas la même largeur, mais, puisque <booléen pour densité> vaut **false**, les hauteurs des rectangles sont les effectifs des classes.

 **Histogramme**[true,{0,5,15,20},L,true]
produit le graphique ci-contre.

Cet histogramme représente les effectifs cumulés croissants de la série regroupée en classes de largeurs différentes.



 **Histogramme**[true,{0,5,15,20},L,false]
produit le graphique ci-contre.

Les classes n'ont pas la même amplitude, mais, lorsque <booléen pour densité> vaut **false**, les hauteurs des rectangles sont les effectifs cumulés croissants des classes.

Remarque :

La syntaxe de la commande **Histogramme** peut parfois prêter à confusion. En effet, si l'on écrit **Histogramme**[{0,5,10,15},{3,6,11}], cela signifie-t-il que l'on souhaite construire un histogramme dont les hauteurs des rectangles sont 3, 6 et 11 ou bien un histogramme à partir des données brutes 3, 6 et 11 ?

Si la seconde liste contient exactement un élément de moins que la première liste, GeoGebra considère alors que la seconde liste désigne les hauteurs des rectangles, sinon, il traite la seconde liste comme la liste des données brutes de la série (ce qui peut amener à des incohérences dans le cas où cette liste est dynamique).

Pour que le logiciel traite systématiquement la seconde liste comme celle des données brutes de la série, il suffit de fournir explicitement le paramètre <booléen pour densité> en écrivant, par exemple, **Histogramme**[{0,5,10,15},{3,6,11},false].

Remarque :

L'objet créé à l'aide de la commande **Histogramme** est un objet de type « numérique ». La « valeur » de l'histogramme est égale à son aire totale. Ainsi, si le paramètre <densité> vaut **true**, l'aire de l'histogramme est égale à l'effectif total multiplié par l'échelle.

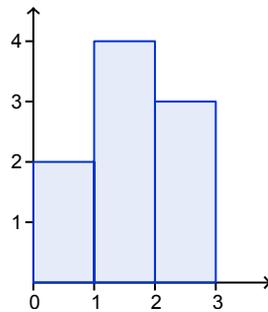
Remarque :

Avec la commande **Histogramme** les classes sont des intervalles fermés à gauche et ouverts à droite, excepté le dernier qui est fermé. On peut utiliser la commande **HistogramDroite** si on désire des intervalles ouverts à gauche et fermés à droite pour les classes (excepté pour le premier qui est fermé).

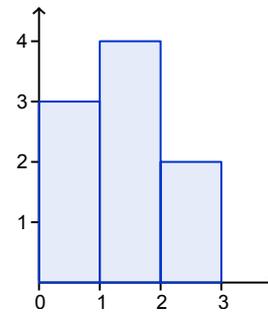
La syntaxe de la commande **HistogramDroite** est rigoureusement identique à celle de la commande **Histogramme**.

Si $L = \{0, 0.5, 1, 1.2, 1.5, 1.7, 2, 2.6, 3\}$:

Histogramme[{0,1,2,3},L,false]



HistogramDroite[{0,1,2,3},L,false]



4 Les polygones des effectifs

La commande **PolygoneEffectifs** permet le tracé du polygone des effectifs. Cette commande suit rigoureusement la même syntaxe que celle de la commande **Histogramme**, à savoir :

- **PolygoneEffectifs**[<bornes des classes>,<hauteurs>] pour obtenir le tracé du polygone des effectifs en fournissant la liste des hauteurs;
- ou
- **PolygoneEffectifs**[<booléen pour cumul>,<bornes des classes>,<données brutes>,<booléen pour densité>,<échelle>] pour obtenir le tracé du polygone des effectifs en fournissant la liste des données brutes de la série.

Lorsque <booléen pour cumul> vaut **false**, le polygone des effectifs est une ligne brisée reliant les points ayant pour abscisse le centre des classes et pour ordonnée l'effectif de la classe. GeoGebra crée également deux points fictifs d'ordonnée nulle aux extrémités de cette ligne brisée : si la première classe est de la forme $[x_1 ; x_2[$ et si la dernière classe est de la forme $[x_{n-1} ; x_n]$, ces deux points ont pour coordonnées $(x_1 - \frac{x_2 - x_1}{2}; 0)$ et $(x_n + \frac{x_n - x_{n-1}}{2}; 0)$.

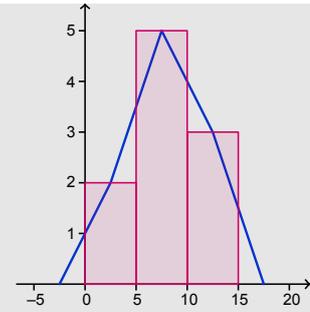
Lorsque <booléen pour cumul> vaut **true**, le polygone des effectifs est une ligne brisée reliant les points ayant pour abscisse la borne supérieure de chaque classe et pour ordonnée l'effectif cumulé de la classe. GeoGebra crée également, à l'extrémité gauche de cette ligne brisée, un point fictif d'ordonnée nulle et d'abscisse la borne inférieure de la première classe.

Exemple(s)

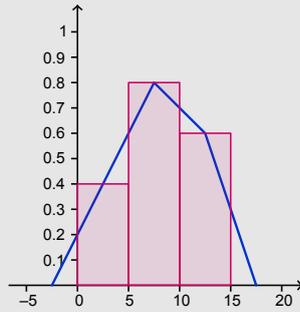
Dans chacun des exemples proposés ci-dessous, on a représenté la ligne brisée produite par la commande **PolygoneEffectifs** ainsi que l'histogramme produit par la commande **Histogramme** employée avec des paramètres identiques.

 `PolygoneEffectifs[{0,5,10,15},{2,5,3}]`
produit le graphique ci-contre.

Les éléments de la seconde liste désignent ici les hauteurs des rectangles.



Posons $L=\{2,2,7,7,7,7,14,14,14\}$.



 `PolygoneEffectifs[{0,5,10,15},L]`
produit le graphique ci-contre.

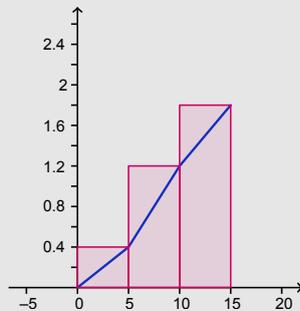
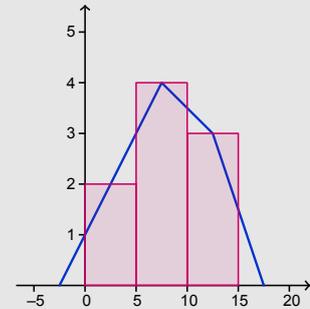
Le premier rectangle a pour hauteur

$$\frac{\text{Effectif de la première classe}}{\text{largeur de la première classe}} = \frac{2}{5} = 0,4$$

Les hauteurs des autres rectangles sont calculées de la même façon.

 `PolygoneEffectifs[{0,5,10,15},L,true,5]`
produit le graphique ci-contre.

En affectant l'amplitude des classes au paramètre <échelle>, les rectangles ont pour hauteur l'effectif des classes.

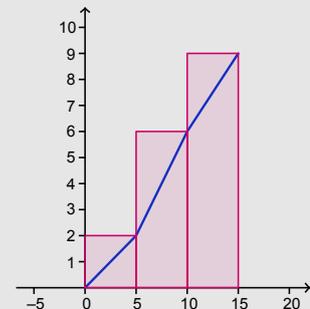


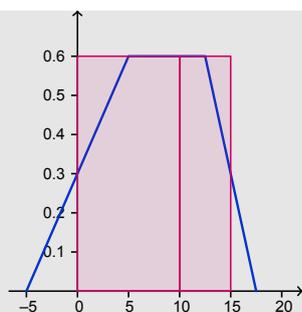
 `PolygoneEffectifs[true,{0,5,10,15},L,true]`
produit le graphique ci-contre.

On représente ici les effectifs cumulés croissants.

 `PolygoneEffectifs[true,{0,5,10,15},L,true,5]`
produit le graphique ci-contre.

En affectant l'amplitude des classes au paramètre <échelle>, les rectangles ont pour hauteur l'effectif cumulé croissant des classes.





`PolygoneEffectifs` $\{0, 10, 15\}$, L , $true$ produit le graphique ci-contre.

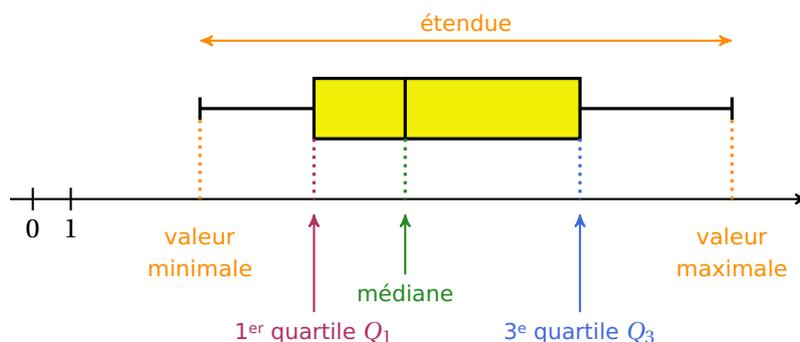
Dans cet exemple, les classes n'ont pas la même largeur.

L'objet créé à l'aide de la commande `PolygoneEffectifs` est un objet de type « numérique ». Sa valeur est toujours nulle.

Remarque :

5 Les boîtes à moustaches

GeoGebra rend possible la création de boîtes à moustaches (encore appelées « diagrammes en boîte » ou bien « boîtes de TUKEY »). Il existe plusieurs manières de résumer une série statistique à l'aide d'une boîte à moustaches et GeoGebra utilise le mode de représentation suivant :



Pour construire une boîte à moustaches, on utilise la commande `BoiteMoustaches` qui accepte les syntaxes suivantes :

- `BoiteMoustaches` [\langle ordonnée \rangle , \langle demi hauteur \rangle , \langle données brutes \rangle , \langle booléen pour valeurs aberrantes \rangle] : cette commande permet de construire une boîte à moustache d'ordonnée \langle ordonnée \rangle dans le repère de la vue **Graphique**.

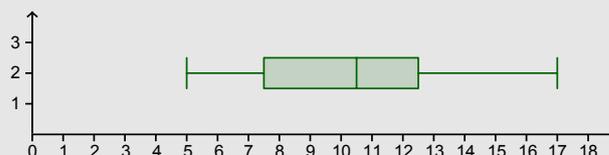
La moitié de la largeur du rectangle représentatif de l'intervalle $[Q_1 ; Q_3]$ est donnée par le paramètre \langle demi hauteur \rangle .

Les valeurs prises par la série sont fournies sous forme d'une liste (paramètre \langle données brutes \rangle).

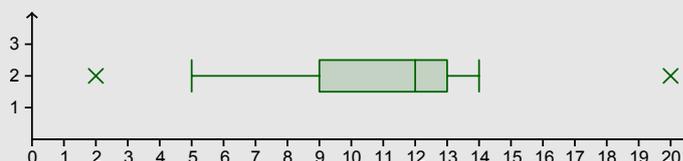
\langle booléen pour valeurs aberrantes \rangle est un paramètre optionnel (par défaut, il vaut `false`). Lorsque ce paramètre vaut `true`, les valeurs aberrantes de la série sont représentées par des croix. GeoGebra considère qu'une valeur est aberrante lorsqu'elle est plus petite que $Q_1 - 1,5 \times EI$ ou lorsqu'elle est plus grande que $Q_3 + 1,5 \times EI$ (où EI est l'écart interquartile $Q_3 - Q_1$).

Exemple(s)

 `BoiteMoustaches[2,0.5,{5,5,7,8,9,10,11,11,12,13,14,17}]` produit le graphique :



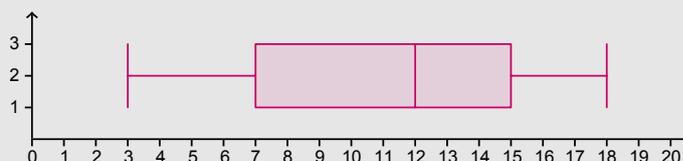
 `BoiteMoustaches[2,0.5,{2,5,9,12,12,12,12,13,13,14,20},true]` produit le graphique :



- `BoiteMoustaches` [<ordonnée>, <demi hauteur>, <valeur minimale>, <Q1>, <médiane>, <Q3>, <valeur maximale>] : cette commande permet de représenter une boîte à moustaches en fournissant directement les différentes caractéristiques de la série nécessaires à la construction du graphique.

Exemple(s)

 `BoiteMoustaches[2,1,3,7,12,15,18]` produit le graphique :



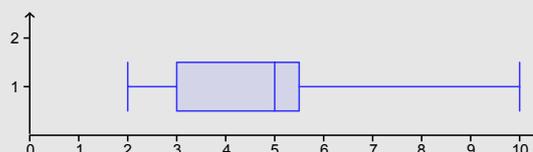
- `BoiteMoustaches` [<ordonnée>, <demi hauteur>, <liste des valeurs>, <liste des effectifs>, <booléen pour valeurs aberrantes>] : lorsque les données de la série ont été dépouillées et classées, on peut utiliser cette syntaxe pour construire la boîte à moustaches correspondante.

Les listes des valeurs de la série et celle des effectifs doivent être de même longueur.

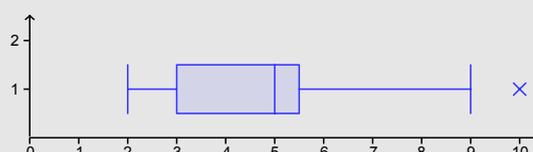
Le paramètre <booléen pour valeurs aberrantes> n'est pas optionnel dans le cadre de cette syntaxe et doit donc être explicitement précisé.

Exemple(s)

 `BoiteMoustaches[1,0.5,{2,3,4,5,6,9,10},{2,4,2,5,2,1,1},false]` produit le graphique :



 `BoiteMoustaches[1,0.5,{2,3,4,5,6,9,10},{2,4,2,5,2,1,1},true]` produit le graphique :



Remarque :

- Contrairement à ce que les exemples précédents laissent supposer, la liste des données brutes de la série peut être passée en paramètre de la commande **BoiteMoustaches** sans être ordonnée.
- L'objet créé à l'aide de la commande **BoiteMoustaches** est un objet de type « numérique » et sa valeur est égale à la médiane de la série statistique.

6 Les nuages de points

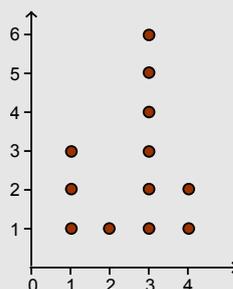
Dans un graphique en nuage de points, chaque point représente une valeur prise par la variable. Lorsqu'une valeur apparaît à plusieurs reprises, les points sont placés les uns au-dessus des autres de telle sorte que la hauteur de la colonne obtenue représente l'effectif de la valeur de la variable considérée.

Pour créer un nuage de points avec GeoGebra, on utilise la commande **NuagePoints** dont la syntaxe est la suivante :

- **NuagePoints**[<données brutes>] : cette commande permet de construire le nuage de points représentatif de la série dont les valeurs sont fournies sous la forme d'une liste, non nécessairement ordonnée (paramètre <données brutes>).

Exemple(s)

 **NuagePoints** [{1, 1, 1, 2, 3, 3, 3, 3, 3, 4, 4}] produit le graphique :



Remarque :

- L'objet créé à l'aide de la commande **NuagePoints** est un objet de type « liste » : GeoGebra retourne la liste des points (définis par leurs coordonnées) qui constituent le graphique.
- On peut également utiliser la commande **NuagePoints** avec une série dépouillée. En supposant que les listes Valeurs et Effectifs contiennent respectivement les valeurs prises par la variable et les effectifs correspondants (ces deux listes doivent donc être de même longueur), la commande **Aplatir**[**Séquence**[**Séquence**[**Elément**[Valeurs, n], i, 1, **Elément**[Effectifs, n]], n, 1, **Longueur**[Valeurs]]] permet de générer la liste des données brutes de la série.

7 Les tiges et feuilles

Dans un diagramme en tige et feuilles, on décompose les valeurs prises par la variable en deux parties : la partie principale, qui forme la tige, est constituée des premiers chiffres du nombre (dizaines, centaines, milliers, ...) tandis que la partie restante (le dernier chiffre du nombre) est ventilée sur les feuilles du diagramme, en face de sa partie principale. Les valeurs sur les feuilles sont classées dans l'ordre croissant ainsi que les valeurs sur la tige.

Pour créer un diagramme en tiges et feuilles avec GeoGebra, on utilise la commande **TigeFeuilLes** dont la syntaxe est la suivante :

- **TigeFeuilles**[<données brutes>, <ajustement>] : cette commande permet de construire le diagramme en tige et feuilles représentatif de la série dont les valeurs sont fournies sous la forme d'une liste, non nécessairement ordonnée (paramètre <données brutes>).

Le paramètre <ajustement> est optionnel et peut prendre les valeurs -1, 0 ou 1 (par défaut, il vaut 0) :

- si <ajustement>=-1, les valeurs sur la tige sont comprises entre 1 et 99;
- si <ajustement>=0, les valeurs sur la tige sont comprises entre 0 et 9;
- si <ajustement>=1, la tige ne comporte que la valeur 0.

Exemple(s)

 **TigeFeuilles** [{12, 23, 19, 6, 10, 7, 15, 25, 21, 12}] produit le graphique :

```

0 | 6 7
1 | 0 2 2 5 9
2 | 1 3 5

```

Lecture : 3|1 signifie 31

 **TigeFeuilles** [{45.2, 45.6, 46.3, 46.7, 46.9, 47, 47.5, 48.2}, -1] produit le graphique :

```

45 | 2 6
46 | 3 7 9
47 | 0 5
48 | 2

```

Lecture : 3|1 signifie 3.1

 **TigeFeuilles** [{0.1, 0.2, 0.2, 0.3, 0.5}, 1] produit le graphique :

```

0 | 1 2 2 3 5

```

Lecture : 3|1 signifie 3.1

Remarque :

- L'objet créé à l'aide de la commande **TigeFeuilles** est un objet de type « texte ».
- Dans GeoGebra, la tige ne peut excéder une certaine longueur et le logiciel regroupe automatiquement les valeurs isolées.

Ainsi, **TigeFeuilles** [{7, 28, 63, 64, 64, 71, 77, 82, 85, 89, 93, 135}] produit le graphique :

bas : 7.0, 28.0

```

6 | 3 4 4
7 | 1 7
8 | 2 5 9
9 | 3

```

haut : 135.0

Lecture : 3|1 signifie 31

- GeoGebra peut procéder à des arrondis sur les valeurs de la série selon l'ajustement choisi. Par exemple, avec **TigeFeuilles** [{12, 12, 13, 14, 15.5, 16}], GeoGebra arrondi la valeur 15,5 à 16 (il faut alors choisir un ajustement égal à -1 pour faire apparaître les parties décimales).

8 Les résidus

Étant donné une fonction f et un ensemble de points de coordonnées $(x_i; y_i)$, la commande **Résidus** retourne une liste de points de coordonnées $(x_i; y_i - f(x_i))$.

Sa syntaxe est la suivante :

- **Résidus**[<liste de points>,<fonction>] : <liste de points> et <fonction> désignent respectivement la liste des points et la fonction par rapport auxquels on désire calculer les résidus.

Exemple(s)  **Résidus**[(2,3),(-1,-2),(4,7)],x^2] retourne la liste {(2; -1),(-1; -3),(4; -9)}.

9 Le graphe quantile normal

Pour déterminer si un échantillon provient d'une population distribuée selon une loi normale, on peut utiliser un graphe quantile normal.

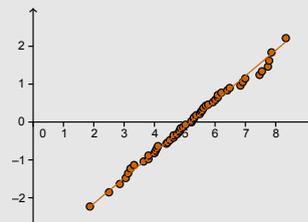
Pour réaliser un tel type de graphique, on utilise la commande **NormaleQuantile** dont la syntaxe est la suivante :

- **NormaleQuantile**[<données brutes>] : <données brutes> représente la liste de valeurs de l'échantillon.

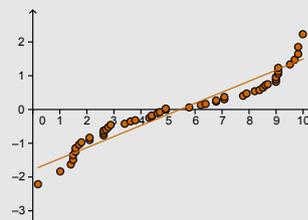
Exemple(s)

On suppose que L désigne la liste des valeurs d'un échantillon.

-  Si la population est distribuée selon une loi normale, la commande **NormaleQuantile**[L] produit un graphique du type suivant :



-  Dans le cas où la population n'est pas distribuée selon une loi normale, la commande **NormaleQuantile**[L] produit un graphique du type suivant :



Remarque :

L'objet créé à l'aide de la commande **NormaleQuantile** est un objet de type « liste ». Il s'agit de la liste des points qui ont pour abscisses les valeurs de la série et pour ordonnées leur score normal attendu (score z).