



#### Unité 5 : Utiliser la bibliothèque TI System

#### Compétence 2 : Modélisation

Dans cette seconde leçon de l'unité 5, vous allez découvrir comment importer les résultats d'une modélisation en utilisant la bibliothèque **TI System**.

#### Objectifs :

- Effectuer une modélisation linéaire.
- Importer les résultats de cette modélisation dans un script Python.

Vous allez dans cette leçon effectuer une modélisation linéaire à partir de données préalablement inscrites dans les listes de la calculatrice. Ensuite, vous écrirez un script afin d'importer les résultats de cette modélisation afin de les utiliser pour une représentation graphique, une interpolation ou extrapolation...

**Le problème :** Dans une journée, le pic de consommation d'électricité est atteint vers 19 h. Au niveau national, on a enregistré un pic de consommation de 96 350 mégawatts le mercredi 15 décembre 2019 à 19h02. Vous désirez prévoir les pics de consommation du prochain week-end pour la zone directement raccordée à la centrale.

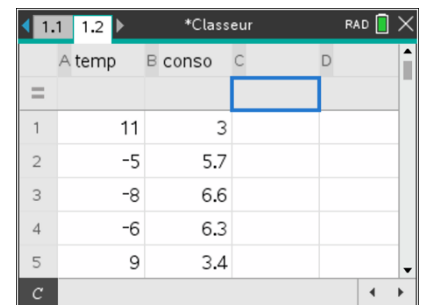
Pour établir cette prévision, vous disposez de dix relevés de consommations réalisés à 19h en fonction de la température et présentés dans le tableau ci-dessous.

T°C	11	-5	-8	-6	9	14	4	-1	-12	3
MW	3	5.7	6.6	6.3	3.4	2.7	4	5.1	7.1	4.6

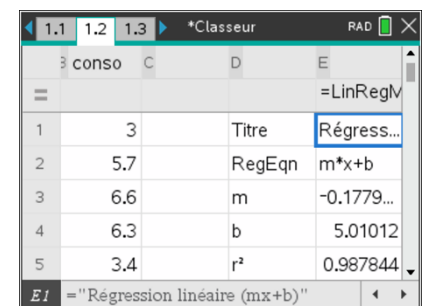


- Les données sont entrées dans les listes de la calculatrice, la température dans **temp** et la consommation en MW dans **conso**.
- Pour cela, ouvrir une application **tableur & listes**, puis entrer les données.

Remarque : Il est également possible d'utiliser l'application **Données & Statistiques**, mais aussi de définir les listes à partir de l'application **Calculs**.



- Effectuer une régression linéaire sous la forme  $mx+b$  (**menu** puis **4 Statistiques**).





# 10 Minutes de Code

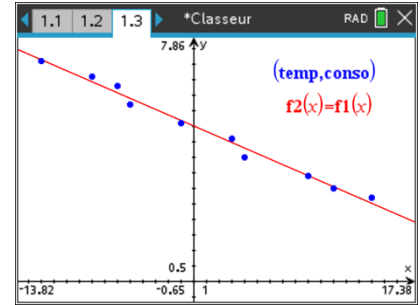
TI - NSPIRE™ CX II & TI - PYTHON

## UNITE 5 : COMPETENCE 2

### NOTES DU PROFESSEUR

La fonction permettant de prévoir la consommation en fonction de la température est donc :  $C = -0.18 \times t + 5.01$ .

Représenter dans l'application **Graphiques**, le nuage de points ainsi que la droite de régression.



### Utilisation des résultats de la modélisation dans un script Python.

- Commencer un nouveau script et le nommer U5SB2.
- Incorporer le menu **TI System** et **TI Plotlib**.
- Créer deux listes **temperature** et **consommation** vides.

```

1.1 1.2 1.3 *Classeur RAD
*U5SB2.py 3/6
from ti_system import *
import ti_plotlib as plt
temperature=[]
consommation=[]

```

Rappeler le contenu des listes **temp** et **conso** dans leurs noms respectifs. L'instruction **var=recall\_list(« nom »)** est accessible dans le menu du module **TI System**(voir Unité 5 Compétence 1).

```

1.1 1.2 1.3 *Classeur RAD
*U5SB2.py 7/8
from ti_system import *
import ti_plotlib as plt
temperature=[]
consommation=[]
temperature=recall_list("temp")
consommation=recall_list("conso")

```

Tester votre script et demander l'affichage des différentes variables ainsi créées en appuyant sur la touche **[var]**, puis en choisissant vos variables.

```

1.1 1.2 1.3 *Classeur RAD
Shell Python 5/5
>>>temperature
[11, -5, -8, -6, 9, 14, 4, -1, -12, 3]
>>>consommation
[3, 5.7, 6.6, 6.3, 3.4, 2.7, 4, 5.1, 7.1, 4.6]
>>>|

```





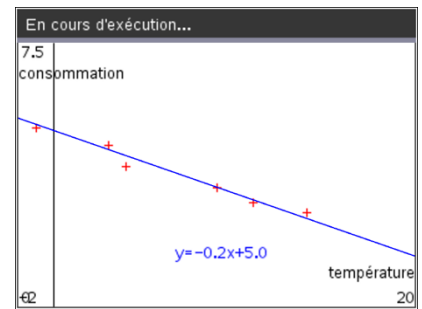
Pour compléter cette leçon et réinvestir les compétences acquises lors de l'unité 4, vous pouvez effectuer la représentation graphique de vos mesures ainsi que du modèle de régression calculé.

```

1.1 1.2 1.3 *Classeur RAD 6/18
U5SB2.py
consommation=recall_list("conso")
# Représentation graphique
plt.cla()
plt.window(-2,20,0,7.5)
plt.labels("température","consommation",12,2)
plt.axes("on")
plt.color(255,0,0)
plt.scatter(temperature,consommation,"+")
plt.color(0,0,255)
plt.lin_reg(temperature,consommation,"center")
plt.show_plot()

```

**Conseil à l'enseignant :** l'instruction `lin_reg(xliste, yliste, « aff », row)` comporte un paramètre supplémentaire `row` donnant la possibilité de placer sur une autre ligne l'équation de régression. Par défaut, celle-ci est placée à la ligne 11.



**Prolongement : prévoir la consommation pour une température donnée.**

Vous allez récupérer dans deux variables **a** et **b** les coefficients de l'équation afin de les utiliser dans une fonction vous permettant de réaliser ainsi une extrapolation ou une interpolation de la consommation électrique, lorsque la température est connue.

Définir une fonction **prev(t)** qui retournera la consommation prévue pour une température **t**.

Exécuter votre script et effectuer quelques tests pour différentes températures.

On rappelle que le modèle représente une prévision de consommation électrique à 19h02 en fonction de la température.

Vous pouvez ainsi déterminer les limites de validité de votre modèle.

- L'instruction **recall(value , « name »)** permet de rappeler au sein d'un script Python une variable définie ou déclarée au sein d'une application.
- Attention : les variables statistiques sont précédées du nom **stat n°.nom**. Pour les retrouver, appuyer sur la touche `var` lorsque l'application est active.

```

1.1 1.2 1.3 *Classeur RAD 19/23
*U5SB2.py
plt.color(0,0,255)
plt.lin_reg(temperature,consommation,"center")
plt.show_plot()
# Prévision
def prev(t):
    m=recall_value("stat1.m")
    b=recall_value("stat1.b")
    return round(m*t+b,1)

```





## 10 Minutes de Code

TI - NSPIRE™ CX II & TI - PYTHON

Exemple :

Prévoir la consommation électrique en MW pour une température de  $-10^{\circ}\text{C}$  puis de  $25^{\circ}\text{C}$ .

## UNITE 5 : COMPETENCE 2

### NOTES DU PROFESSEUR

```
1.1 1.2 1.3 *Classeur RAD 5/5
Shell Python
>>>prev(-10)
6.8
>>>prev(25)
0.6
>>>
```

Ce document est mis à disposition sous licence Creative Commons  
<http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>

