Initiation à la programmation

ENSAE
première année

Xavier Dupré

September 10, 2013

Intervenants pour l'année 2013

Xavier Dupré	xavier.dupre-AT-gmail.com, xavier.dupre-AT-ensae.fr
Emmanuel Guérin	emmanuel-AT-guerin.fr.eu.org
Olivier Levitt	olivier.levitt-AT-insee.fr
Alexis Dondon	alexis.dondon-AT-insee.fr
Arnaud De Myttenaere	Arnaud.De.Myttenaere-AT-ensae.fr
Julien Cazorla	julien.cazorla-AT-ensae.fr

Savoir programmer : pourquoi ?

- Impossible de travailler sans ordinateur.
- On ne traite plus les données manuellement (en trop grand nombre).
- Les traitements qu'on leur applique sont de plus en plus complexes.
- Illustrer une idée statistique abstraite avec un graphique simple est un processus souvent créatif et non répétitif.

Déroulement et évaluation

Premier semestre (obligatoire)

- 13 séances de travaux pratiques
- 2 QCM notés sur 5 points
- 1 QCM noté sur 10 points
- 1 séance de TD notée

Second semestre (facultatif)

- un projet informatique
- 4 suivis entre février et mai
- 1 programme, 1 rapport (fin mai)
- 1 soutenance orale (début juin)

Objectifs

- implémenter un modèle statistique
- connaître quelques techniques fréquemment utilisées
- connaître quelques algorithmes couramment utilisés

Objectifs

- travailler de façon autonome
- travailler à plusieurs
- développer une démarche scientifique et rigoureuse

Python comme choix de langage

- Le langage est open source et donc gratuit.
- Il fonctionne sur toutes les OS (Windows, Linux, Mac).
- Il dispose de nombreuses extensions.
- Il permet de nombreux usages (calcul scientifique, programmation web, jeux)
- Sa syntaxe est l'une des plus simples.

Resources spécifiques à ce cours

- **Mon site :** http://www.xavierdupre.fr/
- **La liste des TD et cette présentation :** http://www.xavierdupre.fr/site2013/enseignements/index.html
- **Support de cours :** même page (disponible à la bibliothèque)
- **Résumé de la syntaxe Python :** http://www.xavierdupre.fr/site2013/documents/python/resume_utile.pdf
- **Divers outils :** http://www.xavierdupre.fr/site2013/index_code.html

Liens

- **Le langage :** http://www.python.org/
- **La liste des packages publics :** https://pypi.python.org/pypi
- **Le site du zéro :** http://www.siteduzero.com/informatique/tutoriels/apprenez-a-programmer-en-python
- **Exécution illustrée d'un programme :** http://www.pythontutor.com/
- **Un livre :** http://inforef.be/swi/python.htm
- **Autres liens :** http://www.xavierdupre.fr/blog/2013-03-30_nojs.html

Datamining avec Python

- **Orange** : http://orange.biolab.si/, faire du datamining sans programmer
- **Python for Data Analysis** : http://shop.oreilly.com/product/0636920023784.do, ce livre illustre ce qu'on peut facilement faire avec Python pour manipuler des données sans trop programmer.
- **Building Machine Learning Systems with Python** : http://www.packtpub.com/building-machine-learning-systems-with-python/book, état de l'art des techniques les plus utilisées en machine learning aujourd'hui.
- **liste d'extensions scientifiques pour Windows** : http://www.lfd.uci.edu/~gohlke/pythonlibs/

Extensions conseillées (datamining)

- **cvxopt** : http://cvxopt.org/
- **ipython** : http://ipython.org/
- **matplotlib** : http://matplotlib.org/
- **numpy** : http://www.numpy.org/
- **pandas** : http://pandas.pydata.org/
- **pytables** : http://www.pytables.org/moin
- **scikit-learn** : http://scikit-learn.org/stable/
- **scipy** : http://www.scipy.org/

See http://www.xavierdupre.fr/blog/2013-08-10_nojs.html.

A l'ENSAE

- Le monde Python évolue vite.
- Tous les modules ne sont pas installés à l'ENSAE.
- Pour les projets, il faudra parfois installer ce dont vous avez besoin sur votre ordinateur.

- Pour certains projets, je mettrai à disposition du code sous la forme d'une extension.
- **pyensae** : http://www.xavierdupre.fr/app/pyensae/helpsphinx/index.html
- **pyhome3** : http://www.xavierdupre.fr/app/pyhome3/helpsphinx/index.html

Questions ?

Premier semestre : première partie

Etant donné que les élèves de l'ENSAE ne suivent pas tous le même cursus, cette première partie est facultative excepté la séance 6.

- Séance 1 : programmer à l'ENSAE, premiers pas.
- Séance 2 : variables, boucles, tests, listes
- Séance 3 : fonctions, dictionnaires
- Séance 4 : modules, fichiers, expressions régulières (+ QCM sur 5 points optionnel)
- Séance 5 : classes (méthodes, attributs, constructeur)
- Séance 6 : classes (héritage, opérateurs) (+ QCM 10 points obligatoire)

Premier semestre : seconde partie

Cette partie n'est **pas** facultative.

- Séance 7 : calcul scientifique (pandas, numpy)
- Séance 8 : introduction au SQL
- Séance 9 : outils de visualisation (graph XY, graph 3D, animation, graph (networkx), d3.js)
- Séance 10 : programmation dynamique
- Séance 11 : Arbre, Trie (+ QCM sur 5 points)
- Séance 12 : Optimisation linéaire et quadratique avec ou sans contrainte
- Séance 13 : TD noté

Second semestre : projet

- Janvier : choix du projet (informatique ou économique), puis choix du sujet

Pour le projet informatique :

- Février - Mai : 4 suivis espacés d'environ un mois.
- Fin Mai : rendu du projet (rapport + programme)
- Début juin : soutenance orale

Quelques exemples de projets :
http://www.xavierdupre.fr/site2013/enseignements/index_projets.html

Questions ?

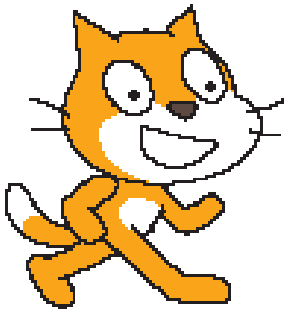
Les difficultés de l'apprentissage

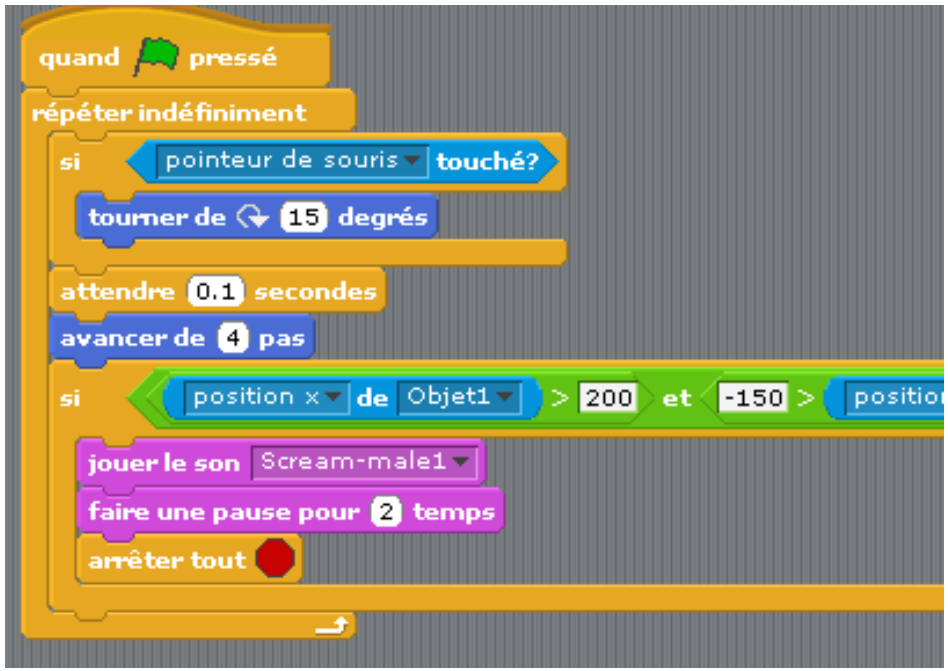
- La programmation est abstraite. Il n'existe pas de façon évidente de représenter un algorithme ou un raisonnement.
- Les programmes sont un long empilement de choses simples. L'ensemble est difficile à résumer.
- Il existe beaucoup de bonnes pratiques qu'on ne comprend vraiment et qu'on ne retient qu'après en avoir utilisé une mauvaise.
- On peut résumer simplement la finalité d'un programme informatique, rarement le moyen d'y arriver.
- Les techniques évoluent vite, on fait souvent les mêmes choses mais jamais avec les mêmes outils d'une année sur l'autre.

Peu de concepts, beaucoup de pratique.

Scratch

Scratch est un outil visuel pour réalisation de petites animations à l'aide de la programmation : http://scratch.mit.edu/.

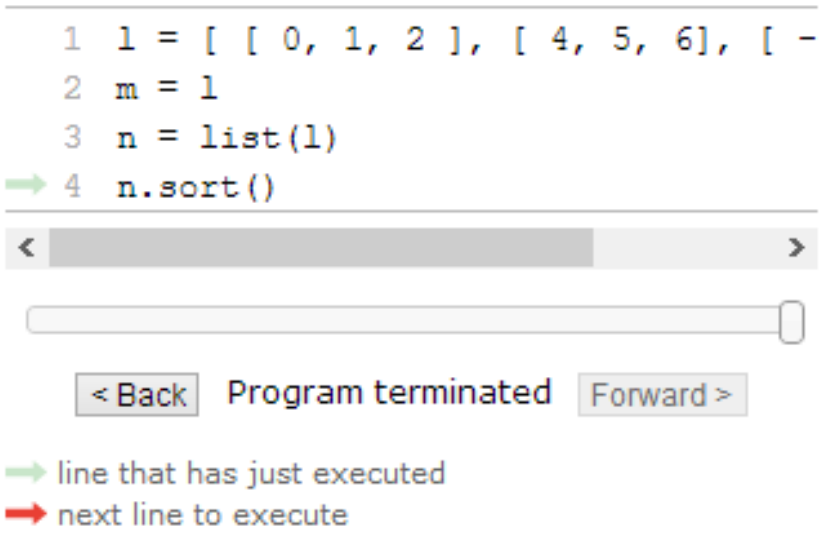


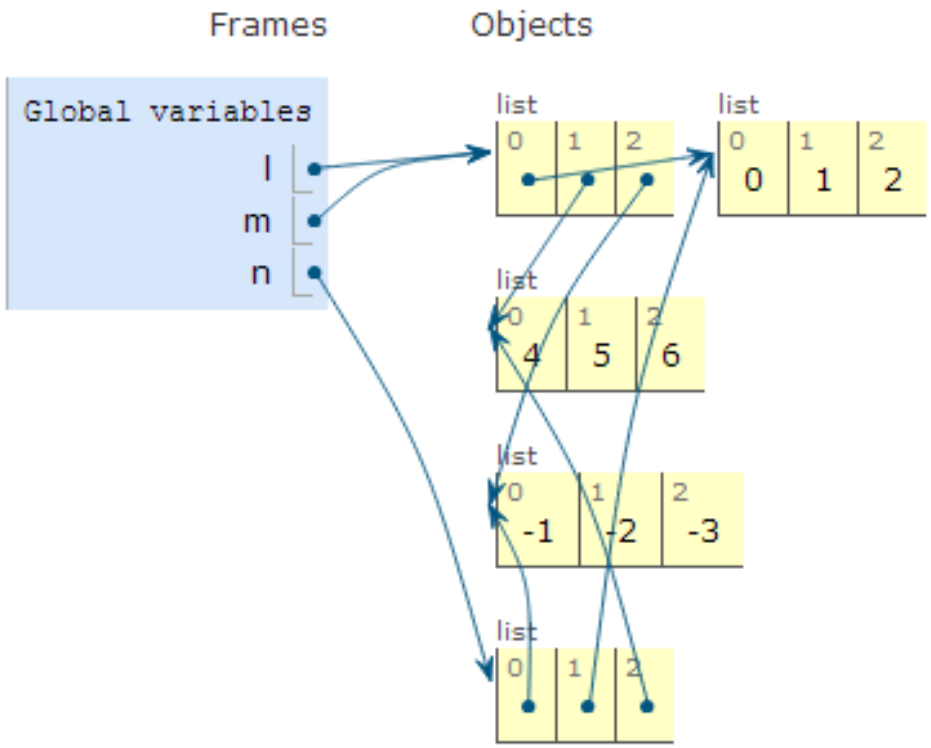


Scratch est pratique lorsqu'il s'agit de réaliser une animation, moins pour le calcul.

PythonTutor

PythonTutor permet de représenter visuellement ce qu'un court programme fait : http://www.pythontutor.com/.





Un autre exemple est disponible ici : http://www.xavierdupre.fr/blog/2013-04-05_nojs.html.

Tendances

- Tous les appareils électriques sont munis de capteurs.
- Votre médecin fera bientôt un premier diagnostique à partir des données récoltées par votre smartphone.
- La société se numérise de plus en plus, dans tous les domaines.

Mais :

- Plus de données difficiles à traiter sans outils complexes (informatiques, mathématiques).
- Plus de données difficiles à comprendre et visualiser.
- Phénomène *Big Data*
- Kaggle http://www.kaggle.com/

Besoins

- Besoin accru de personnes comprenant les statistiques et l'informatique (*Data Scientist*).
- Besoin accru de personnes capables d'analyer des processus de plus en plus complexes.

Où :
L'*informatique* ne signifie plus comprendre la machine mais plus maîtriser un ensemble d'outils dont le lien avec la ou les machines est de plus en plus lointain.

Big Data

- On ne sait pas encore vraiment traiter des données sans programmer.
- Que peut-on faire en utilisant les données que vous produisez en tant qu'individu (Facebook, Vélib) ?
- Que veut dire la notion d'anonymat aujourd'hui ?

Questions ?

Quelle version de Python ?

Il existe deux versions de Python actuellement utilisées :

- **Python 2.7** :
  - Toutes les extensions sont disponibles.
  - La fonction `print` ne prend pas de parenthèses.
  - La division `1 /0` retourne 0.
  - La gestion des accents dans les chaînes de caractères peut être parfois problématiques (encoding).
- **Python 3.3** :
  - La plupart des extensions sont disponibles. Les plus importantes le sont à quelques exceptions près.
  - La fonction `print` prend des parenthèses (c'est une fonction).
  - La division `1 /0` retourne `0.5`, `1//0` retourne 0.
  - La gestion des accents dans les chaînes de caractères est plus cohérente.

Les exemples du cours s'appuieront sur la version **Python 3.3**.

Windows

1. Aller sur le site officiel de Python : http://www.python.org/, télécharger la version *Python 3.3.2 Windows x86 MSI Installer*
2. Pour les extensions, aller sur cette page : http://www.lfd.uci.edu/~gohlke/pythonlibs/ ou aller sur les sites officiels de ces extensions.

Editeurs de texte recommandés :

1. **Notepad++** : http://notepad-plus-plus.org/fr/, léger, multifonction
2. **SciTe** : http://www.scintilla.org/SciTE.html, très léger, multifonction
3. **PyScripter** : https://code.google.com/p/pyscripter/, permet de débugger

Mac

1. Aller sur le site officiel de Python : http://www.python.org/, télécharger la version *Python 3.3.2 Mac OS X 32-bit i386/PPC Installer*

Editeurs de texte recommandés :

1. **SciTe** : http://www.scintilla.org/SciTE-OSX.html, très léger, multifonction mais payant
2. **TextWrangler** : http://www.barebones.com/products/textwrangler/

Linux

1. Aller sur le site officiel de Python : http://www.python.org/, télécharger la version *Python 3.3.2 xzipped source tarball*
2. Ecrire en ligne de commande :
   - `tar -zxvf /<votre chemin>/Python-3.3.2.tgz`
   - `cd /<votre chemin>/Python-3.3.2`
   - `./configure`
   - `make` ou `sudo make install` selon l'endroit où vous l'installez
3. Pour vérifier que cela a marché, écrire : `./python`

Editeurs de texte recommandés :

1. **SciTe** : http://www.scintilla.org/SciTE.html, très léger, multifonction
2. **XEmacs** : http://www.xemacs.org/, pour geek

Environnements intégrés

Un environnement intégré combine de nombreuses fonctionnalités : un éditeur, l'affichage de texte et de graphiques, un débuggeur...
Leur apprentissage est plus long mais souvent rentable pour un usage intense.

1. **Spyder** : http://www.scintilla.org/SciTE.html, seulement sur Windows en Python 2.7
2. **IEP** : https://code.google.com/p/iep/
3. **Eclipse + PyDev** : http://www.eclipse.org/, http://pydev.org/
4. **Visual Studio + Python Tools** : http://www.microsoft.com/france/visual-studio/essayez/express.aspx + http://pytools.codeplex.com/, permet de débugger

Extensions recommandées

Indispensables :

- **numpy** : http://www.numpy.org/, calcul matriciel
- **matploblib** : http://matplotlib.org/, graphiques

Autres :

- **ipython** : https://pypi.python.org/pypi/ipython, entre Python et MatLab
- **pandas** : http://pandas.pydata.org/, manipulation de données
- **pygame** : http://www.pygame.org/news.html, pour faire des jeux
- **SciPy** : http://www.scipy.org/, pour faire des jeux
- **PyTables** : http://www.pytables.org/moin, tables, base de données, calcul matriciel
- **virtualenv** : https://pypi.python.org/pypi/virtualenv, pour installer d'autres packages python et les retirer facilement
- **networkx** : http://networkx.github.io/, dessin de graphes (sociaux, ...)
- autres suggestions : http://www.xavierdupre.fr/blog/xd_blog_key_references.html

Outils collaboratifs

Travailler à plusieurs peut devenir un cauchemar. Synchroniser des programmes n'est jamais facile. Voici trois outils qui peuvent aider :

- **TortoiseSVN** : http://tortoisesvn.net/, pour versionner ses propres programmes
- **GitHub** : https://github.com/, pour versionner des fichiers sur le *cloud*
- **DropBox** : https://www.dropbox.com/ (il existe aussi SkyDrive sur Windows), répertoire sur le *cloud*, synchronisé avec un répertoire local

Cloud veut dire ici qu'une compagnie tierce stocke vos fichiers sur ses machines. Ce n'est pas autorisé dans un environnement privé.

Tests unitaires

Lorsqu'on travaille à plusieurs, on altère souvent une fonction : elle ne fait plus ce qu'elle est censée faire. Un *test unitaire* permet de vérifier cela.
http://docs.python.org/3.3/library/unittest.html
J'ai créé un modèle de programme qui propose une façon simple de :

- d'écrire des tests unitaires et de les exécuter
- de générer la documentation
- de créer un setup d'installation

http://www.xavierdupre.fr/site2013/index_code.html ou https://github.com/sdpython/python_project_template/

Questions ?

Pour toute question que vous pourriez avoir, le premier réflexe est d'utiliser un moteur de recherche (google, bing, yahoo, duckduckgo, baidoo, yandex, ...) :

python + <votre question>

Heure suivante : premiers pas à l'ENSAE.