

Examen à blanc mid-term 2020

Scientific Python (Numpy + Matplotlib) + Programmation Objet + Git

Basé sur l'examen SIE-BA3 2019

Contents

1	Températures moyennes à Aigle entre 1981 et 2011	1
1.1	Les fichiers attendus	1
1.2	Le code Python attendu	2
1.3	Bouts de code offerts	3
1.4	Attentes sur le contenu du code	3
1.5	Le résultat	4
1.6	Travail versionné avec Git	4
2	Soumission de votre copie	5
2.1	Voie normale - Github.com	5
2.2	Voie alternative - Moodle.epfl.ch	5
3	Barème sur 7.5 points	6

1 Températures moyennes à Aigle entre 1981 et 2011

1.1 Les fichiers attendus

Vous allez rendre un dossier nommé `blank_test_2020`, versionné avec Git, comprenant 4 fichiers :

1. `aigle_1980-12-31_2019-07-09.csv`
Ce fichier est à télécharger depuis ici :
https://enacit.epfl.ch/cours/docs/aigle_1980-12-31_2019-07-09.csv
Il contient les données brut qu'il faudra transformer.
2. `process_aigle_temperature.py`
Ce fichier contiendra le code *Python* que vous aurez écrit pour atteindre l'objectif décrit ci-dessous.
3. `aigle_temperature_means.png`
Image au format PNG, généré par votre code Python. L'exemple du

résultat attendu est montré ci-dessous.

4. `my_pledge.md`

Fichier dont le contenu vous est donné au chapitre Code versionné avec Git. Il décrit l'engagement que vous prenez en passant ce test.

1.2 Le code Python attendu

L'objectif du code que vous allez écrire est de :

- Lire les données mesurées par une station météo basée à Aigle depuis le fichier `'aigle_1980-12-31_2019-07-09.csv'`. Celui-ci contient toutes les températures moyenne par jour, entre le 1er janvier 1981 et le 9 juillet 2019.

Cette opération se fera avec la librairie `pandas` qui automatise pour nous une quantité non négligeable d'opérations. Comme nous n'avons pas eu le temps de travailler avec cette librairie en cours, je vous fournis ci-dessous toutes les instructions qui vous seront utiles pour cet exercice.

- Manipuler ces données à l'aide de `numpy` afin d'en extraire
 - la température moyenne de chaque mois des années *1981*, *1991*, *2001* et *2011*
 - la température moyenne sur l'année entière pour ces mêmes années
- Grapher à l'aide de `matplotlib` les températures moyennes obtenues pour chaque mois :
 - *1981* (`color='blue'`)
 - *1991* (`color='green'`)
 - *2001* (`color='orange'`)
 - *2011* (`color='red'`)
- Ajuster le rendu du graphique avec notamment
 - un ratio de 8x6 et 120dpi
 - un titre
 - un label pour chaque axe
 - une grille horizontale
 - une légende indiquant pour chaque couleur :
 - * l'année observée
 - * la températures moyenne globale de l'année

Notez que nous nous basons sur des vraies mesures. Dans ce cas précis, il y a des *trous* de mesures pour plusieurs jours. Pour effectuer la moyenne de ces valeurs sans avoir d'erreur, il vous faudra donc utiliser la fonction `np.nanmean()` qui ignore les valeurs manquantes.

1.3 Bouts de code offerts

Comme mentionné plus haut, nous n'avons pas travaillé avec **pandas** durant le semestre. Voici les bouts de code utilisant cette librairie qui vous seront utiles.

```
# Import de Pandas
import pandas as pd

# Lecture du fichier csv
# Stockage de son contenu dans un DataFrame `temperatures_aigle`
temperatures_aigle = pd.read_csv(
    'aigle_1980-12-31_2019-07-09.csv',
    sep=';', usecols=[1, 2], index_col=0,
    parse_dates=True, dtype={'t06200ds': np.float64},
    na_values='-')

# Exemple de code pour extraire toutes les températures
# entre le 1er janvier 1981 et le 31 décembre 1981 (compris)
# Cela crée un object Numpy ndarray avec les valeurs correspondantes
start_date = '1981-01-01'
end_date = '1981-12-31'
temperatures = temperatures_aigle[start_date : end_date].values
```

1.4 Attentes sur le contenu du code

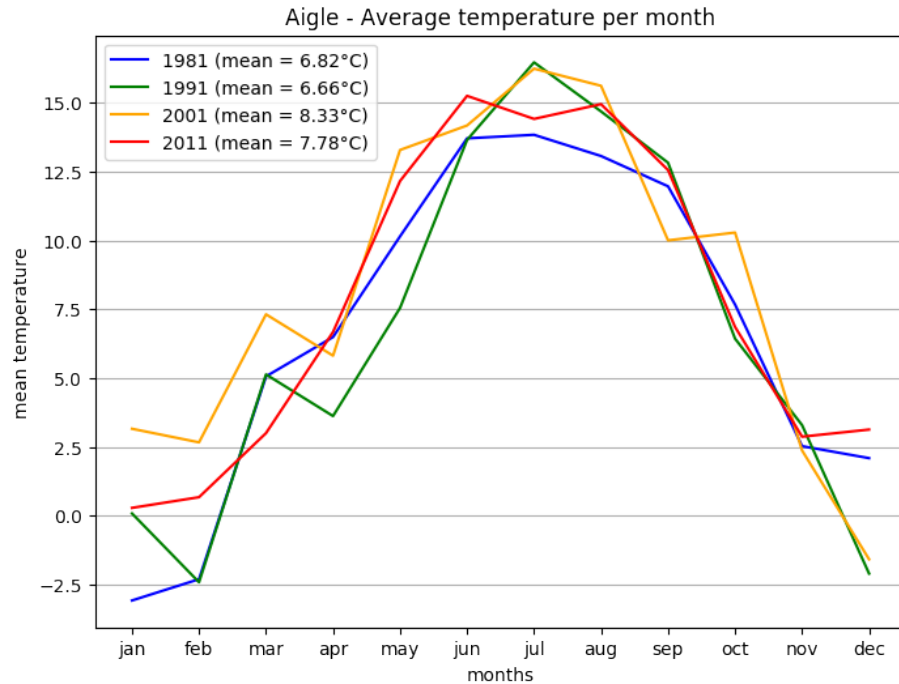
Il est attendu que votre code soit structuré en 6 portions distinctes et successives :

1. Documentation globale du code réalisé dans le fichier
2. Import des librairies nécessaires
3. Définition des **CONSTANTES** utiles pour l'ensemble du code (p. ex. la définition des mois, du nombre de jours par mois et les couleurs pour chaque année graphée)
4. Import des données provenant du fichier CSV
5. Définition d'une fonction **def extract_temperatures(...)**. Il est attendu de cette fonction qu'elle :
 1. soit documentée, tel que **help(extract_temperatures)** nous permette de connaître ce qu'elle fait (c-à-d ce qui est dit ici).
 2. prenne un argument obligatoire : **year** -> l'année pour laquelle on fait l'extraction
 3. prenne un second argument optionnel : **month** -> le mois pour lequel on fait l'extraction. Si aucun mois n'est fourni, alors l'extraction couvre l'année entière.Attention, tous les mois n'ont pas le même nombre de jours ... et les févriers des années bissextile (ce qui se résume aux années multiples de 4 pour l'échantillon sur lequel nous travaillons ici) ont 29 jours...!

6. Réalisation du travail logique et enregistrement du graphique à proprement dit.

1.5 Le résultat

Le graphique que vous devez finalement obtenir est le suivant :



Il vous est demandé de le reproduire à l'identique, avec un code le plus lisible possible (merci d'avance ;) .

1.6 Travail versionné avec Git

L'entier du travail devra être versionné avec Git et déposé sur un dépôt privé hébergé sur Github.

- Chaque commit devra être fait avec comme auteur: votre nom, prénom et email.
- Le dépôt Github, nommé `blank_test_2020`, devra être privé et vous devrez me donner les droits pour que je puisse récupérer votre travail. Pour cela, rendez-vous sur la page principale de votre dépôt sur github https://github.com/your_username/blank_test_2020 . Dans l'onglet (en haut à droite) *Settings* > volet (à gauche) *Manage access*, choisissez *[Invite a collaborator]*, cherchez l'utilisateur "sbancal", puis confirmez avec *[Add sbancal to this repository]*.

- Le 1er commit, intitulé “*Mon engagement*”, contiendra exclusivement le fichier 'my_pledge.md' dont le contenu sera le suivant :

Engagement

Je m'engage à passer cet examen sans faire appel à une personne externe, excepté les assistants et le professeur de ce cours.
Que ce soit oralement ou par tout autre moyen de communication.

votre Prénom Nom

- le 2ème commit, intitulé “*Ajout des mesures de température à Aigle*”, inclura le fichier 'aigle_1980-12-31_2019-07-09.csv' que vous aurez téléchargé.
- Chaque étape significative franchie dans le développement de votre solution sera accompagnée d'un *commit* avec un commentaire explicite.

2 Soumission de votre copie

Ce chapitre décrit le mode de soumission qui sera appliqué le jour de l'examen. Il ne s'applique pas réellement au contexte d'un examen à blanc.

2.1 Voie normale - Github.com

Note : voie uniquement possible pour ceux qui :

1. m'ont communiqué leur username Github par moodle
2. m'ont donné accès à leur dépôt durant l'examen selon la procédure décrite plus haut.

Je récupérerai votre travail à l'heure annoncée depuis votre dépôt Github.

Une fois le travail fini, n'oubliez pas de vérifier que tous les fichiers sont *commités* à la dernière version et que vous avez *pushé* tout ça sur Github! ; -)

2.2 Voie alternative - Moodle.epfl.ch

Si vous ne parvenez pas à versionner votre travail avec Git ou à l'envoyer sur Github, vous devrez préparer un Zip du dossier dédié à cet examen et l'envoyer dans l'espace sur Moodle prévu à cet effet.

3 Barème sur 7.5 points

Points	Sujet
1.0	Rendre une copie sur Moodle ou Github.com
2.0	Git
0.25	Dépôt Git initialisé
0.25	Auteur correctement configuré
0.25	2 premiers commits tel que demandés
0.5	Commits aux étapes significatives
0.5	Messages des commits compréhensibles et décrivant ce qui est fait
0.25	Dépôt sur Github.com correctement configuré
4.5	Python
0.25	Syntaxe OK
0.25	Le code s'exécute sans erreur
0.25	Définition des constantes utiles
0.25	Lecture des données avec pandas
	<i>Fonction <code>extract_temperatures</code></i>
0.25	Aide en ligne
0.25	Gestion de l'argument optionnel month
0.5	Gestion des années bissextiles
0.5	Retour des valeurs pour l'année choisie
0.5	Retour des valeurs pour le mois choisi
0.25	Calcul des moyennes par mois
0.25	Calcul des moyennes par année
	<i>Graphique</i>
0.25	Trace des 4 courbes
0.5	Mise en page de la figure, axes, quadrillage, titre
0.25	Affichage de la légende