## Journal de suivi

Contenu

[Journal de suivi 1](#_Toc498539968)

[Jour 1 (01/11/17) - Durée travail env. 8h : 1](#_Toc498539969)

[Jour 2 (02/11/17) – Durée travail env. 13h: 1](#_Toc498539970)

[Jour 3 (03/11/17) – Durée travail env. 13h: 2](#_Toc498539971)

[Jour 4 (04/11/17) – Durée travail env. 10h: 2](#_Toc498539972)

[Jour 5 (05/11/17) – Durée travail env. 2h: 3](#_Toc498539973)

[Jour 6 (06/11/17) – Durée travail env. 4h: 3](#_Toc498539974)

[Jour 7 (07/11/17) – Durée travail env. 3-4h: 3](#_Toc498539975)

[Jour 8 (08/11/17) – Durée travail < 2h: 3](#_Toc498539976)

[Jour 9 (09/11/17) – Durée travail 4h: 3](#_Toc498539977)

[Jour 10 (10/11/17) – Durée travail 6h: 4](#_Toc498539978)

[Jour 11 (11/11/17) – Durée travail 6h: 4](#_Toc498539979)

[Jour 12 (12/11/17) – Durée travail 8h: 4](#_Toc498539980)

[Jour 13 (13/11/17) – Durée travail 3h: 4](#_Toc498539981)

[Jour 14 (14/11/17) – Durée travail 2h: 5](#_Toc498539982)

[Jour 15 (15/11/17) – Durée travail 3h: 5](#_Toc498539983)

[Temps de travail : 86h 5](#_Toc498539984)

### Jour 1 (01/11/17) - Durée travail env. 8h :

1er jour de formation sur le cours de Data Analyst. Celui-ci commence par un premier cours pour apprendre à apprendre. N’ayant jamais « arrêté » d’apprendre de manière autodidacte, celui-ci s’est plutôt bien passé. Un peu de temps pour mettre à jour mes CVs (j’ai 2 versions différentes), une petite demi-heure pour la mise à jour de mon profil [Linkedin](https://www.linkedin.com/in/nicolas-m-b5a66174/) incluant le début de la formation mais aussi l’ajout des 19 certificats déjà faits… En effet, j’ai commencé les cours depuis pas mal de temps sur OpenClassrooms mais j’ai décidé de faire les Quiz/Activités que récemment en ayant pris goût à l’apprentissage de la Data Science. Le passage en Premium m’a donc permis d’avoir accès au Certificats.

J’ai aussi rapidement mis à jour mon [site personnel](http://www.nicolasmine.com/) (sur lequel j’ai mis mon CV ainsi que différents scripts fait avec des Notebooks, des news et divers liens concernant l’apprentissage du langage Python ou du Machine Learning) afin de mentionner le début de la formation sur celui-ci aussi !

Finalement, sachant bien pourquoi je me lançais dans cette formation, les objectifs d’apprentissages ont aussi été simples à mettre en place. Je passe donc au projet n°1 et là, 1ère petite frustration… En effet, je commence après le Projet 1 celui-ci qui, comme je m’en doutais un peu, s’oriente sur le nettoyage de données. J’avais déjà eu affaire à du traitement de dataset pas clean mais généralement c’est très léger (ajouter/supprimer des lignes sans valeurs, Encoder/OneHotEncoder les Labels) mais là je me rends compte que c’est un tout autre chantier… Un dataset de 800mo, 320 000 lignes et 162 colonnes. Le tout sans trop savoir ce que l’on cherche. Bref, il faut prendre le temps de comprendre chaque donnée uni variée et de les nettoyer.

### Jour 2 (02/11/17) – Durée travail env. 13h:

Etant en congé 2 jours, je me réveille tôt pour avancer (tout en sachant que la tâche sera longue et fastidieuse). Le data cleaning avance par à-coup. Notamment lors de colonne complexe à traiter (par exemple les traces, additifs, qui sont des concaténations de données). Cependant cela avance. A la fin du 1er nettoyage il reste 84 features mais je décide de commencer en parallèle l’exploration en espérant découvrir d’autres possibilités de nettoyage. Celle-ci-se solde par une réduction importante de dimensions ainsi que d'autres possibilités de Nettoyages de Données (notamment pour les données /100g. Certaines ont des valeurs négatives, d'autres supérieur à 100 ou même extrêmement grande pour ce type de données, par exemple 30g de vitamines).

Commençant à voir un nettoyage correct ainsi que certains patterns arriver, je commence aussi à bloquer sur la réduction de dimensions ainsi que l'objectif final de l'exercice un peu flou. Je décide de passer au projet 3 en attendant de discuter des blocages avec mon mentor quand je l'aurais en rendez-vous.

Le Projet n°3 me semble aussi assez sombre à première vue car on doit effectuer une Clustering mais sans avoir de labels particuliers. Du coup j'ai du mal à voir comment j'évaluerais par la suite les modèles. Mais bon il faut commencer par le commencement, place au nettoyage du dataset.

Le dataset étant beaucoup plus petit, celui-ci va aussi beaucoup plus vite (connaissant aussi certaines commandes pandas grâce au précédent nettoyage). Afin de réduire la taille du notebook, j'opte pour un générateur qui me permet de parcourir chaque colonne et d'effectuer la même analyse sur chaque feature. La majorité sont très simples, il faut juste remplir les valeurs manquantes par des données logique (dans certains cas 0, la moyenne ou couleur dans le cas de la couleur du film). Certaines sont plus complexe à traiter, c'est le cas du type de film, des mots clé et des acteurs. Je décide d'abandonner les mots clé ayant les types de film et cela ferait un peu doublons. La fatigue commençant à se faire sentir, je décide de laisser faire le traitement de ces colonnes le lendemain.

### Jour 3 (03/11/17) – Durée travail env. 13h:

Pour les 2 colonnes à problème, je décide dans un premier temps d'utiliser le LabelEncoder. L'objectif étant d'avoir "un premier jet" pour tester les modèles. Je l'applique donc sur chaque colonne de type texte et teste le Kmeans, PCA et TSNE. Peu importe le nombre de cluster, le résultat du kmeans ne me semble pas terrible visuellement. Si je regarde la liste des films pour chaque cluster, je trouve difficilement du sens au classement. Quant au PCA, si je réduis trop les dimensions, je n'explique qu'environ 30% de la variance, le résultat est donc d'autant plus mauvais. Je teste donc le TSNE. Je regarde par distance euclidienne les points les plus proches du film choisi et je commence à voir une corrélation (par exemple si je visualise un des Pirate des Caraïbes, il me propose de voir une partie des autres). Je décide donc d'améliorer mon nettoyage de données. Pour ce faire, je décide de compter les occurrences de chaque type de film ainsi que de chaque acteur et les encoder par leur nombre d'occurrence. Mon objectif étant d'avoir les acteurs qui jouent beaucoup assez proche. En effet le LabelEncoder pour donner le Label 1 à Johnny Depp et 55 à Orlando Bloom. De ce fait la distance entre ces 2 acteurs sera grand et donc illogique. Par contre tous l'acteur ayant joué dans 1 seul films, auront le même label et seront donc proche (cela pour permettre au fan de film peu connu d'avoir des choix de film peu connus aussi). J'applique la même méthode sur le type de film.

Là les résultats sont meilleurs, le PCA peut expliquer 60% de la variance en 10 dimensions, le K-means commence à classer les films par catégorie mais ses résultats ne sont pas fameux. Quant au TSNE, il réussit très bien. Je reste donc sur ce modèle, génère un dataset de production avec juste les données utilise et leurs positions afin de faire le site.

Ayant déjà codé des sites, j'ai juste à reprendre le cours de Flask sur OC et cela avance plutôt bien. J'ai fait une petite page d'accueil ou l'on peut entrer l'ID du film ou une liste déroulante avec tous les titres dans l'ordre, lors de la validation, celui-ci affiche 5 autres film similaire avec diverses info (date de sortie, acteurs, etc...). Je m'arrête donc là pour aujourd'hui.

Au passage, j'ai contacté ce jour mon mentor pour avoir un premier rendez-vous.

### Jour 4 (04/11/17) – Durée travail env. 10h:

Etant en attentes d'informations avec mon tuteur sur le projet n°2 et le projet n°3 étant démarré, je commence plus doucement le projet 4. Celui-ci s'appuie sur 2.8 go de données assez cleané avec pour objectif de mettre en place un modèle linéaire. Les données sur ce dataset sont propres, il ne manque pas de points, il n'y a pas outliers donc on passe assez rapidement à de l'exploration. Avant cela il faut juste se débarrasser apriori des features inutiles/ imprévisible. Pour certaines cela est évident à la description de la feature sur le site pour d'autres, cela a été décidé basé sur l'exploration. Au départ tout s'est fait sur un seul moins pour accélérer le processus et une fois fini, un script propre a été fait pour fusionner les dataset et faire le cleaning d'un coup. A cause de bug sur le dataset d'avril un peu de temps a été perdu lors de l'exploration mais globalement tout s'est bien passé. Le passage au modèle quant à lui pose beaucoup plus de soucis.

En effet, si on veut pouvoir prédire le retard, on a besoin des données prédictible (date, aéroport de départ, aéroport d'arrivé, compagnie, …) le problème c'est que le retard n'est aucunement linéaire avec ces features (il y a plus de retard en journée, plus le lundi et vendredi que le reste de la semaine, plus en été que le reste de l'année et variable en fonction des aéroports). Une solution connue pour ce problème est de faire un One Hot Encoder sur celle-ci. Bien que l'on ait 5m de points de données, on a peu de risque du fléau de la dimensionnalité mais le problème c'est que cela ne passe pas en mémoire. Je vais donc devoir en discuter avec mon mentor car je sèche.

Je profite donc de ce blocage pour commencer la veille thématique. Dans 1 premier temps, je cherche un sujet. Je pensais parler des Enpsulated networks ou du Synthetic Gradient mais il sera difficile de faire 3 exemples dessus. Ne voulant pas parcourir des sujets déjà vu dans les cours, je décide de partir sur les Recurrent Neural Network. Ceux-ci sont d'actualité avec Siri/Google Home et tous les outils de reconnaissance vocale donc je trouve que c'est un bon sujet. Cela commence par de la recherche sur les différentes cellules (Simple RNN, LSTM, GRU, QRNN). C'est là-dessus que ma journée va s'achever.

### Jour 5 (05/11/17) – Durée travail env. 2h:

Cette journée n'a pas été très productive. J'ai travaillé le matin (6h-14h) et ensuite j'ai dû changer mes pneus. Additionné à un début de maladie, j'ai juste passé quelques heures à commencer l'étude et la rédaction de la veille technologique.

### Jour 6 (06/11/17) – Durée travail env. 4h:

Malade, je ne peux donc pas aller au travail et essaye un peu de continuer les projets. Cependant, avec la tête prise ce n'est pas simple. Je continue donc la veille technologique que j'alterne avec quelques petites modifications sur les différents projets déjà commencés. Rien de fabuleux mais tout de même un peu d'avancement.

### Jour 7 (07/11/17) – Durée travail env. 3-4h:

Toujours malade (conjonctivite), je reste à la maison mais difficile de travailler. Je continue donc tranquillement la veille technologique en passant en revue le modèle du LSTM. Fatigué de lire, je décide de reprendre la compétition Kaggle sur la classification de race de chien que j'avais commencé car elle est utile pour le Projet 8. Je reprends donc la partie de préparation des données ou j'avais bloqué. En fois préparé, je me sers du VGG19 pre-trained dans Keras pour entrainer mon modèle. Je laisse le training 4-5h pour atteindre un résultat de 15% de classification correcte.

### Jour 8 (08/11/17) – Durée travail < 2h:

Ce jour, je me rends compte que mon pre-trained modèle n'a pas été sauvegardé malgré la mise en place de checkpoints si le résultat est meilleur. Mon training n'a donc servi à rien et s'est fait écrasé en fin de la 1ere Epoch… Je cherche donc comment réutiliser mon check point quand je relance le modèle mais sans succès pour l'instant.

### Jour 9 (09/11/17) – Durée travail 4h:

Pour ce 9eme jour, l'objectif était d'arriver à sauvegarder le modèle. Etant donné qu'il y a 5 min entre chaque epoch, je décide donc de le faire sur des scripts fait précédemment plus rapide à entrainer. Une fois le problème résolu, je l'ai implémenté sur le Projet 8 et lancé le training. En parallèle j'ai continué l'avancement du site pour la Classification de film. Le résultat visuel est maintenant correct et j'attends donc mon mentor pour discuter de quelques points (l'évaluation des modèles sans labels). Je lance donc l'entrainement du projet 8 pendant la nuit et cela clos ma journée.

### Jour 10 (10/11/17) – Durée travail 6h:

L'entrainement du Projet 8 ayant durée toute la nuit, j'étais content du résultat mais en arrêtant le process de training, la sauvegarde a été corrompue car j'ai interrompu le process pendant l'écriture de la sauvegarde suivante. J'ai donc modifié le code pour générer des sauvegardes différentes à chaque fois avec une incrémentation pour éviter ce souci par la suite.

Une fois cela fait, je repasse sur le projet 4 (retard des avions) et commence à faire des modèles. Vu la taille du dataset, c'est assez restreint (il n'y a que le SGD Regressor). Je lance un Grid Search exhaustif pour trouver les meilleurs paramètres et pendant le calcul, met aussi en place un petit NN pour faire aussi la régression Histoire de voir s'il performe vraiment mieux. Le résultat des 2 ne me plait pas plus que ça, je réfléchis donc à utiliser les ensembles pour faire le training mais je laisse ça pour le lendemain. En attendant, je commence aussi à préparer les données pour le site (le système de sauvegarde des modèles (scaler et modèle), les données nécessaire à l'UI, etc …)

### Jour 11 (11/11/17) – Durée travail 6h:

Un peu fatigué avec la maladie, j'ai décidé de me changer les idées et partir sur quelque chose de complètement nouveau avec le Projet 5. En suivant les cours j'ai mimé le principe du RBF en essayant d'inclure plus de paramètres (notamment la localité). Ce projet a bien commencé mais j'aimerais aller plus dans le détail par la suite. Cependant certaines techniques vues dans le cours vont m'être utile dans les précédents projets (notamment le Projet 2 avec le classement des aliments)

### Jour 12 (12/11/17) – Durée travail 8h:

Durant cette journée, l'objectif était d'avancer le projet 4 avec la prédiction du retard des avions. Le problème majeur sur ce dataset était le volume de donnés qui faisait crasher le notebook. En conséquence j'ai repris le cleaning sur un script appart et gérer 2 dataset (un d'exploration et un de prod). J'en ai profité pour faire des essais de "simplification de données" (par semaine / par mois et date du mois, agrégation par quart d'heure, …). Cela m'a permis de trouver une configuration permettant d'obtenir de meilleurs résultats (MAE réduite de 14min à 9). J'en ai aussi profité pour avancer le site pour pouvoir faire les premiers tests.

### Jour 13 (13/11/17) – Durée travail 3h:

L'objectif de cette journée est de reprendre le Projet 2 avec les connaissances vues lors du projet 5 (notamment le groupby pour transformer la dataframe). Malheureusement après 1h de recherche pour essayer d'afficher la moyenne de chaque colonne pour les bons et mauvais produits, impossible de trouver une façon. Je décide donc de faire un pause, essayer de déterminer le meilleur format de dataset pour le modèle de M/L concernant le retard des vols. D'un côté je décompose par semaines (de 0 à 52) et par jour (de 1 à 7) et de l'autre je garde par mois (de 1 à 12), jour du mois (de 1 à 31) et le jour de la semaine (de 1 à 7). Pour ce faire je génère les 2 datasets évalue chaque modèle dessus. Le modèle par semaine est légèrement plus précis je reste donc la dessus. Maintenant je nettoie un peu les code pour faire un script qui pilote les modèles et stocke les données dans un fichier texte.

### Jour 14 (14/11/17) – Durée travail 2h:

Comme commencé la veille, cette soirée n'a pas eu d'avancement incroyable. J'ai juste décomposer le code du projet 4 pour avoir 1 script pour alléger les dataset mensuelle, un autre pour les fusionner en 1 seul et générer le dataset d'exploration et de production. Un dernier script permet de scaler les données, préparer les matrices et les sauvegarder pour ne pas le refaire à chaque lancement de modèle. Ensuite je créer différents scripts pour chaque modèle qui écrivent leur résultat dans un fichier texte.

### Jour 15 (15/11/17) – Durée travail 3h:

Ayant tous les scripts prêts, j'ai appliqué aussi la méthode de la partie 4 avec l'affectation de note sur chaque aéroport basé sur les valeurs médianes. L'objectif étant d'avoir que 4 types d'aéroports au lieu de 331 aéroports distinct. L'intérêt était de pouvoir ensuite pourquoi ne pas utiliser un OHE sans avoir d'explosion de dimensions ayant déjà 4.5m de lignes. J'ai ensuite testé différentes façon de générer les features (Aéroport avec et sans OHE, gestion des dates par mois/jour ou par semaine) et j'ai appliqué" le GridSearch sur le SGDRegressor, Adaboost et un simple ANN. Comparé aux premiers modèles testés avec les données "brutes", une augmentation significative a eu lieu permettant de passer d'une MAE de 14 min à 10 min. Il sera malheureusement difficile de passer en dessous car chaque feature est désormais assez propre. C'est un point à discuter avec mon mentor lors d'une session de mentorat

### Jour 16 (16/11/17) – Durée travail 2h30:

Durant ce jour, j'ai repris le projet 3 avec pour objectif de prendre en compte quelques classes que j'avais passé pour l'instant (par exemple le rating). Les modèles ont ensuite été re-testé et donne des résultat légèrement meilleur (par exemple avec Spiderman 1, la 1ere prédiction est Spiderman 2 au lieu d'être en 2nd). Le kl\_divergent a aussi été réduit de 0.80 à 0.75. Cela a pris environ 1h30.

Une petite période a aussi été prise pour regarder le nouveau modèle de reconnaissance d'images (You Only Look Once - YOLO) au cas où le sujet sur les RNN ne soit pas valide comme Veille Technologique.

La seconde partie était la 1ère session de mentorat.

## Temps de travail : 89h