

ALAIN BADIOU
Le Monde
13 août 2015

Je suis absolument contre cette thèse du bonheur toujours rêvé auquel on n'accède jamais. C'est faux, le bonheur est absolument possible, mais pas dans la forme d'une satisfaction conservatrice. Il est possible sous la condition des risques pris dans des rencontres et des décisions, lesquelles sont proposées, en définitive, à un moment ou à un autre, à toute vie humaine.

L'eudémonisme, le principe selon lequel le bonheur serait le but de la vie humaine, est une notion qui a traversé les âges depuis la Grèce Antique jusqu'à nos jours sans jamais réussir à proposer une solution assez concrète, précise et applicable à l'un de ses problèmes principaux : comment atteindre un bonheur parfait ? Parmi toutes les tentatives liées au topos que représente la quête du bonheur, certaines relativement récentes ont mis en œuvre une mathématisation visionnaire de ce dernier ; on pense notamment à l'approche utilitariste de l'hédonisme développée par Jeremy Bentham¹, ou encore à l'apport des travaux de Daniel Kahneman² à l'économie du bonheur et à la psychologie positive.

Aujourd'hui, la conjoncture est différente : les avancées technologiques sont arrivées à un stade inimaginable auparavant³, et le progrès rend ainsi accessible les réponses au questionnaire séculaire énoncé plus haut. Dans cet article, nous proposons une solution pratique, implémentable par chacun, à la question suivante : existe-t-il une équation du bonheur ? Est-il possible de le quantifier, de le prévoir, de le maximiser ? En partant de l'affirmative comme assumption de départ, que nous nommerons l'hypothèse eudémonique pure et parfaite⁴ (ou l'Hypothèse, pour la distinguer de toute autre), nous avons mis au point un protocole expérimental permettant de déterminer l'Équation du bonheur ; le procédé permettant de modéliser la journée parfaite, soit la journée qui procurerait le plus de bonheur. Nos recherches concernant l'élaboration de cette formule ont démarré le

6 janvier 2018, marquant ainsi la date de début d'une collecte de données personnelles. L'Hypothèse supposait de suivre sur le long terme, avec le plus de précisions possibles – aussi bien qualitativement que quantitativement – et avec la rigueur la plus appuyée qu'il est envisageable de maintenir, la vie d'un homme (à savoir moi, auteur de ce texte). Cette étude de cas, dont la phase d'observation a été conduite du 6 janvier 2018 au 22 février 2019, a permis la création d'un corpus éclectique et disparate de plus de 64 000 nombres, couvrant chaque aspect de mes journées, en allant du plus intime au plus mondain, du plus anecdotique au plus structurant. Cet agrégat de détails, reconstituant le récit de ma vie sur plus d'une année, a servi comme fondement sur laquelle la méthode présentée dans cet article a été établie.

Ladite méthode se compose en trois étapes principales. Dans une première partie, nous détaillerons les modalités de la récolte des données, les paramètres sur lesquels nous nous sommes focalisés et la méthodologie mise en place pour l'occasion. La seconde partie développera les points traitant de l'analyse et de la compréhension du jeu de données présenté dans la première partie. Cette partie précise l'usage qui a été fait des dernières technologies d'apprentissage automatique, dont le but était de réaliser une analyse prédictive révélant les intrications à l'œuvre dans le jeu de données pour permettre la conception de l'Équation du bonheur. La partie finale dressera le bilan des résultats obtenus, puis présentera une interprétation de ceux-ci en mettant l'accent sur ses modes d'utilisation possibles.

Récolte des données

LA LISTE DES PARAMÈTRES

Notre jeu de données est composé de paramètres très divers : 179 paramètres ont été définis puis groupés dans 18 catégories différentes. Cette taxonomie et son exhaustivité ont été établies en isolant les évènements les plus déterminants de mes journées, prenant ainsi le portrait numérique de chacune des 413 journées. Les 179 paramètres et leurs descriptions longent les bas de page, et les 18 catégories auxquelles ils appartiennent sont les suivantes :

- (1) Notes de la journée
- (2) Social (3) Météo
- (4) Métabolisme (5) Alimentation
- (6) Nombres d'occurrences des émotions (7) Nouvelles
- (8) Communications (9) Sommeil
- (10) Moyenne des intensités des émotions (11) Métadonnées
- (12) Intensités des émotions
- (13) Loisirs (14) Géographie
- (15) Transports
- (16) Intimité (17) Travail
- (18) Calendrier

Des informations complémentaires sont nécessaires à la compréhension de certains paramètres. En effet, la plupart des intitulés sont préfixés de lettres répondant au code d'identification suivant : on a n pour notes, moy pour moyenne, t pour durée, h pour l'heure de l'évènement en minutes à partir de minuit, nb pour nombre et enfin b pour booléen.

MÉTHODOLOGIE DE LA RÉCOLTE

Bien qu'atteindre l'exhaustivité dans la translittération de la vie réelle dans un format numérique soit une tâche presque impossible⁵, c'est bien l'objectif que nous nous étions fixé pour quantifier les jours qui passaient⁶. Il fallait, pour que cela soit satisfaisant, que chaque action effectuée puisse trouver sa place parmi nos paramètres. Ainsi, nous avons fait en sorte que chaque instant de la vie puisse nourrir ces paramètres, puis que ces intrants (ou *inputs*) soient les plus divers et les plus précis possibles. La constitution de cette liste de paramètres s'est établie dans le temps – les premiers jours étaient d'abord lacunaires – et s'est ensuite complexifiée au fur et à mesure de l'expérience. Le modèle est devenu complet autour du jour 200.

On peut classer ces paramètres en trois grandes familles de données : les données personnelles relevées manuellement, les données personnelles relevées par des outils tiers, et les données exogènes :

Manuelles : Ces données sont celles qui demandent le plus d'abnégation. Elles doivent être relevées le jour même ou ne le seront jamais. (*n_journee** (notes de la journée) ou *n_conversations*** (social)). Il y en a 114.

Automatiques : Ce sont des informations calculées à partir des autres données (*moy_trois_jours**** (notes de la journée)) ou relevées par des applications. Il y en a 55.

(1) **n_journee* = "La note que j'ai attribuée à cette journée" ; *n_matinee* = "La note que j'ai attribuée à cette matinée" ; *n_apres_midi* = "La note que j'ai attribuée à cet après-midi" ; *n_veille* = "La note de la veille" ; ****moy_trois_jours* = "La moyenne des notes des trois jours qui précèdent cette journée" ; *moy_semaine* = "La moyenne des notes de la semaine qui précède cette journée" ; (2) *nb_bonjour* = "Le nombre de personnes à qui j'ai dit bonjour" ; ***nb_conversations* = "Le nombre de personnes avec qui j'ai eu une discussion" ;

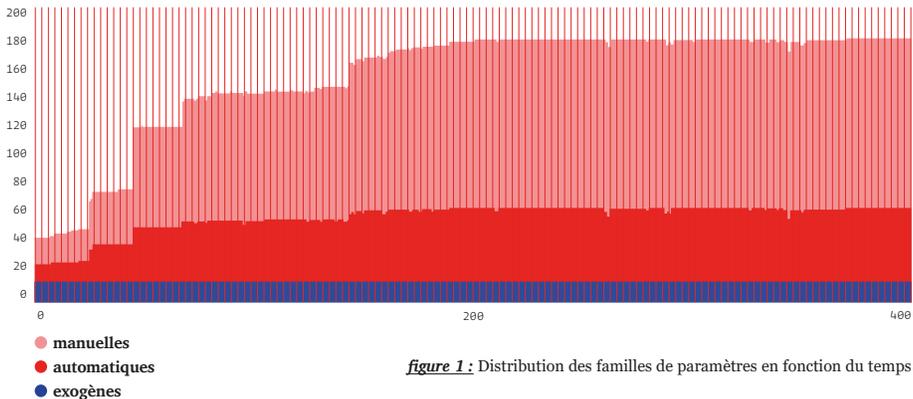
Récolte des données

Exogènes : Les données exogènes n'ont pas été relevées lors des 413 jours de l'expérience. Elles sont arrivées par la suite après avoir été extraites de bases de données publiques (h_lever_soleil**** (météo)). Il y en a 10.

La *figure 1* montre l'évolution du nombre de paramètres relevés au cours du temps. Les paramètres exogènes agissent comme une base sur laquelle sont venues s'ajouter les données personnelles. La partie supérieure gauche du graphique est ce qui constitue l'extrême majorité des données manquantes. Le taux de remplissage de la base de données est de 86,15 %, c'est-à-dire qu'environ un septième des données sont manquantes.

Les paramètres personnels (manuels et automatiques) ont été inscrits dans des carnets que je portais tous les jours sur moi.

Les nombres étaient ensuite retranscrits dans un fichier ods alors que les pages des carnets se remplissaient. Je remplissais chaque soir à vingt-deux heures le carnet de son apport quotidien. L'inscription des valeurs se faisait alors en deux temps : les valeurs « simples » comme le nombre de fois où je suis allé aux toilettes, et les valeurs « complexes » comme les émotions ou les nouvelles advenues durant la journée. Pour ces dernières, il était nécessaire de revivre la journée en profondeur, se forcer à retracer chaque instant et combler tous les sauts temporels qui existent dans la reconstitution mentale. Le cerveau s'adapte rapidement à cet exercice ; les personnes saluées viennent s'installer de plus en plus naturellement dans la mémoire à court terme. Le soir venu, tant que celles-ci n'ont pas été toutes remémorées puis reportées dans le carnet, la sensation d'avoir oublié quelque chose ne disparaissait pas.



nb_rencontres = "Le nombre de personnes que j'ai rencontrées" ; n_rires = "La note attribuée à l'intensité et au nombre de fois où j'ai ri" ; nb_even_soc = "Le nombre d'événements sociaux auxquels j'ai pris part" ; moy_even_soc = "La note moyenne des événements sociaux auxquels j'ai pris part" ; t_even_soc = "Le temps que j'ai passé à participer à des événements sociaux" ; b_even_soir = "Un booléen correspondant à la participation à un événement social le soir" ; (3) ****h_lever_soleil = "L'heure à laquelle le soleil s'est levé" ;

Analyse

FONCTIONNEMENT D'UN PSYCHOLOGUE

La première étape de l'analyse des données avait pour but de comprendre comment les 178 paramètres influençaient mon appréciation finale – ou mon taux de satisfaction – de la journée. Dans notre jeu de donnée, ce taux de satisfaction est matérialisé par le paramètre `n_journee`, soit la note comprise entre 0 et 20 reflétant au plus proche mon humeur de la journée. Nous associons cette phase de développement au dressage d'un profil psychologique, soit à la manière que nous avons de réagir à chacun des événements se produisant. Pour ce faire, un premier algorithme a été mis en place: il prenait en entrée les 178 paramètres qui composaient mes journées, et fournissait en sortie une prédiction de `n_journee` se voulant la plus proche possible de la vraie valeur associée à mon taux de satisfaction de la journée.

Établir un profil psychologique n'est pas chose aisée. Il ne faut non pas prendre chaque paramètre un à un et lui attribuer un coefficient fixe pour le calcul de ma satisfaction personnelle, mais plutôt regarder comment chacun des paramètres évolue en fonction des autres, comprendre ce qu'implique chaque combinaison possible de paramètres. Illustrons ce point par un exemple basique: imaginons que le paramètre `nb_toilettes*` (métabolisme) ait une valeur anormalement élevée. À première vue, rien ne permet d'affirmer qu'il y aurait une incidence positive ou négative sur le calcul de `n_journee`. Cependant, associé

à l'interprétation de paramètres tels que `n_maux**` (métabolisme) ou `n_fatigue***` (métabolisme) nous pouvons commencer à reconstruire ce qu'aurait pu être le scénario de cette journée. Des questions de plus en plus complexes peuvent trouver une réponse dans l'étude de certaines combinaisons de paramètres, et ces réponses, étudiées elles aussi conjointement, sont ce qui vont résulter sur un profil psychologique.

Étant incapables de formuler cette myriade de questions artisanalement, il nous fallait un programme qui le fasse de manière autonome, et dont la structure reflète clairement l'hypothèse faite sur la complexité des profils psychologiques. Il fallait que le profil puisse s'adapter à chaque nouvelle situation qui lui est présentée (dans ce cas, une nouvelle situation correspond à une nouvelle journée) et qu'il n'ait au départ aucun *a priori* sur comment nous pourrions réagir aux variations de chacun des paramètres. En d'autres termes, tous les paramètres doivent partir avec la même pondération initiale, et c'est par empirisme que des discriminations s'effectuent. Par exemple, si les premières journées montrent que `n_journee` est supérieur avec des combinaisons de paramètres où `n_temperature` (météo) est grand, ce dernier aura de prime abord une grande influence positive sur la note finale de la journée dans le modèle – ou profil psychologique. Par la suite, si `n_temperature` se révèle être élevé dans des combinaisons résultant sur une note finale faible,

`h_coucher_soleil` = "L'heure à laquelle le soleil s'est couché" ; `n_meteo` = "La note attribuée à la météo" ; `***n_temperature` = "La note attribuée à la température extérieure" ; `t_jour` = "La durée du jour" ; `t_dehors` = "Le temps passé dehors" ; `t_dehors_soleil` = "Le temps passé dehors alors qu'il faisait beau" ; (4) `***n_fatigue_matin` = "La note attribuée à mon état de fatigue durant la matinée" ; `n_fatigue_soir` = "La note attribuée à mon état de fatigue durant l'après-midi" ; `moy_fatigue` = "La note moyenne de l'état de fatigue matinal et de l'état de

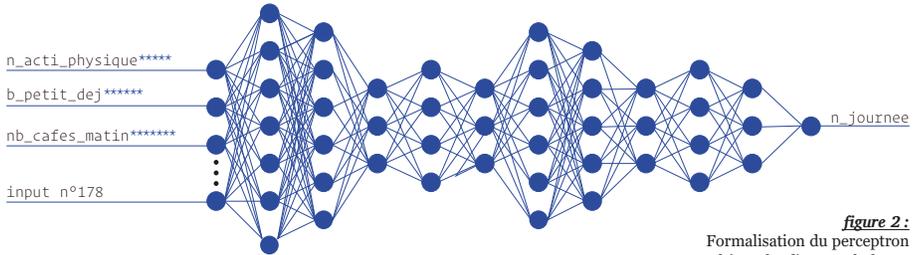


figure 2 :
Formalisation du perceptron multicouche d'un psychologue

alors le modèle s'adaptera pour refléter cette réalité tout en respectant ce qui avait déjà été observé.

Mise en pratique

Un outil statistique s'est imposé face aux autres pour remplir cette mission: un réseau de neurones avec la même structure que celle énoncée plus haut (178 *in*, 1 *out*). Il existe de nombreuses classes de réseaux de neurones, et celui qui correspond le mieux à notre problématique était un perceptron multicouche (*figure 2*) avec une fonction d'activation linéaire. La bibliothèque Python utilisée pour implémenter cet algorithme fut celle du projet scikit-learn (la classe MLPRegressor⁸ de la bibliothèque sklearn.neural_network). Cet algorithme dispose d'une vingtaine d'attributs à définir, nous n'entrerons pas dans les détails de chacun de ceux-ci dans l'article, mais pour connaître la manière la plus efficace d'étudier notre jeu de données, nous devions tester un grand nombre de différents modèles d'intelligence artificielle – les attributs d'un modèle sont représentés par un Psychologue P – et comparer

les résultats obtenus. Pour cela, nous avons programmé un générateur de modèles, éprouvant chacun des réseaux de neurones sur notre jeu de données avec des valeurs d'attributs différentes et établissant le score d'efficacité $\epsilon(P)$ de chacun des modèles. Les critères permettant de calculer ce score sont les suivants: l'erreur maximale e_{max} faite par le modèle lors d'une prédiction et l'erreur moyenne e_{μ} du modèle sur toutes les prédictions. Le score satisfait alors l'équation suivante :

$$\epsilon(P) = e_{\mu} + e_{max}$$

Plus la valeur du score $\epsilon(P)$ est faible, plus le modèle a été précis dans ses prédictions. Attribuer le même coefficient à e_{μ} et e_{max} permet d'accorder une plus grande importance à l'erreur maximale qu'à l'erreur moyenne dans le calcul du score. Les modèles mis en avant sont alors ceux qui ont été les plus réguliers dans leurs performances. Cette manière heuristique de discriminer les « bons » modèles des « mauvais » est encore très simpliste mais a permis d'obtenir des résultats exploitables.

fatigue postméridien" ; *****n_acti_physique = "La note attribuée à l'intensité globale des activités physiques" ; n_poids_porte = "La note attribuée au poids total des choses que j'ai portées" ; **n_maux = "La note attribuée à l'intensité des maux physiques" ; *nb_toilettes = "Le nombre de fois où je suis allé aux toilettes" ; (5) *****b_petit_dej = "Un booléen correspondant au fait de manger un petit-déjeuner" ; *****nb_cafes_matin = "Le nombre de cafés bus le matin" ; nb_cafes_aprem = "Le nombre de cafés bus l'après-midi" ;

Analyse

D'autres fonctions de score plus performantes pourront être envisagées dans le futur.

Une fois la méthode pour déterminer le score d'un Psychologue établie, il nous restait encore à calculer les valeurs de e_μ et de e_{max} . Résumons maintenant les étapes de la création du Psychologue et du calcul de $\epsilon(P)$:

- (1) Le générateur de modèles propose une combinaison d'attributs cohérents pour définir les caractéristiques du perceptron multicouche, le Psychologue.
- (2) Le Psychologue s'exerce sur un jeu de données d'entraînement. Un jeu d'entraînement est un échantillon de nos données (93 % du jeu de données initial) composé par des journées complètes, à savoir les paramètres relevés durant ces journées et les notes finales de celles-ci. Lorsque la première journée du jeu lui est présentée, le Psychologue cherche à trouver une combinaison d'opérations simples à appliquer sur les 178 paramètres pour obtenir en sortie la note de la journée. Avec la seconde journée, le Psychologue garde le même objectif – trouver une combinaison d'opérations – mais en essayant de faire correspondre cette combinaison le plus précisément possible aussi bien à la première journée qu'à la seconde. Cette opération est répétée autant de fois qu'il y a de journées dans le jeu d'entraînement.

- (3) Une fois le Psychologue entraîné, nous évaluons ses capacités sur un jeu de données de test. Le jeu de données de test se compose de toutes les journées du jeu complet qui n'étaient pas dans le jeu de données d'entraînement (soit les 7% restants du jeu de données initial). Nous donnons au Psychologue les 178 paramètres qui ont constitué une journée en se préservant de l'informer de la note finale, et lui demandons de prédire quelle serait la valeur de $n_journee$, puis nous comparons la valeur prédite $v_{prédiction}$ à la valeur réelle $v_{réelle}$ que nous avons attribuée à cette journée le jour où elle a été vécue. On obtient alors l'erreur absolue e telle que:

$$e = |v_{réelle} - v_{prédiction}|$$

Nous répétons cette opération autant de fois qu'il y a de journées dans le jeu de données de test. Une fois toutes les journées vérifiées et les erreurs calculées, on obtient la valeur e_μ et de e_{max} .

`nb_total_cafes = "Le nombre de cafés bus" ; t_dejeuner = "La durée de la pause du midi" ;
nb_sandwichs = "Le nombre de sandwichs mangés" ; nb_plats_cuisi = "Le nombre de plats faits
maison mangés" ; n_quantite_bouffe = "La note attribuée à la quantité de nourriture mangée" ;
n_qualite_bouffe = "La note attribuée à la qualité de la nourriture mangée" ; n_eau = "La note
de la quantité d'eau bue" ; nb_alcool = "Le nombre de verres d'alcool bus" ; couts_bouffe =
"Le prix de la nourriture achetée" ; (6) nb_joie = "Le nombre de fois où j'ai ressenti de la`

Bilan 1

Soit n le nombre de prédictions faites par les Psychologues et E l'ensemble des valeurs prises par e , on a donc :

$$e_{\mu} = \frac{1}{n} \sum_{i=1}^n e_i$$

$$\forall e_i \in E, i \in \mathbb{N}, i < n, \\ e_{max} < e_i \Rightarrow e_{max} = e_i$$

- (4) Nous calculons le score du Psychologue via la fonction de score. Le Psychologue au score le plus proche de zéro celui qui a été le plus performant. Nous pouvons imaginer un modèle avec lequel $\epsilon(P) = 0$, impliquant donc des valeurs nulles pour e_{μ} et e_{max} . Ce modèle serait le modèle parfait, il ne se serait jamais trompé sur aucune prédiction; plus qu'un psychologue extérieur, il serait un *alter ego*.

Au total, près de quatre cents (381) perceptrons multicouches ont été imaginés par le générateur de modèles, entraînés sur le jeu de données d'entraînement puis évalués. Le meilleur des Psychologues a obtenu un score $\epsilon(P) = 1,77$ avec $e_{\mu} = 0,52$ et $e_{max} = 1,25$. Ces nombres peuvent s'interpréter de la manière suivante : pour chaque journée où nous avons demandé au meilleur Psychologue d'estimer la note finale de la journée, ce Psychologue s'est en moyenne trompé d'environ 0,5 points sur 20, et s'est trompé au maximum de 1,25 point sur 20. Seulement 9 des Psychologues ont obtenu un score inférieur à 2 et 127 ont reçu un score compris entre 2 et 3. Le meilleur Psychologue est donc capable d'en moyenne, deviner avec une erreur de 0,52 points la valeur que j'ai attribuée à `n_journee`.

221 Psychologues ont été conservés sur les 381 générés après que les valeurs aberrantes et les scores trop élevés aient été retirés du jeu. Nous reviendrons sur ces 221 modèles d'intelligence artificielle une fois explicité le fonctionnement du Faussaire et du Détective.

joie" ; nb_surprise = "Le nombre de fois où j'ai ressenti de la surprise" ; nb_tristesse = "Le nombre de fois où j'ai été triste" ; nb_colere = "Le nombre de fois où j'ai été en colère" ; nb_peur = "Le nombre de fois où j'ai eu peur" ; nb_degout = "Le nombre de fois où j'ai ressenti du dégoût" ; nb_mepriis = "Le nombre de fois où j'ai méprisé quelqu'un ou quelque chose" ; nb_con = "Le nombre de fois où je me suis senti con" ; nb_soulagement = "Le nombre de fois où j'ai ressenti du soulagement" ; nb_admiration = "Le nombre de fois où j'ai

Analyse

FONCTIONNEMENT DU FAUSSAIRE ET DU DÉTECTIVE

Étant compris entre 0 et 20, on peut conjecturer que n_{journee} prendrait pour valeur 20 si jamais nous venions de vivre la journée parfaite. Cependant, la meilleure journée vécue, durant toute la durée de la récolte des données, a obtenu la note de 18 sur 20. Comment donc faire pour modéliser la combinaison parfaite de paramètres qui, si suivie scrupuleusement, m'amènerait à vivre la journée la plus heureuse possible? En parallèle des Psychologues, il fallait imaginer une méthode générant des journées qui pourraient potentiellement advenir, puis trouver parmi celles-ci la journée parfaite – celle qui obtiendrait, selon les Psychologues, un 20 sur 20.

C'est à partir de cette situation qu'ont été développés le Faussaire et le Détective. Ces deux personnages sont, comme les Psychologues, des modèles d'intelligence artificielle. Leur mission est cependant différente: leur singularité se situe majoritairement dans leur rôle plus que dans leur structure. En effet, nous pourrions dire que les Psychologues, le Faussaire et le Détective appartiennent à la même espèce, à la même famille d'outils statistiques.

(a) Le processus commence avec la phase d'apprentissage du Faussaire. Au début de son entraînement, celui-ci ne sait pas à quoi ressemble une journée si ce n'est qu'elle se compose de 178 paramètres. Sa mission est de proposer au Détective

de fausses journées jusqu'à ce qu'il réussisse à le tromper. Le tout premier lot de fausses journées se constitue d'une série de signaux aléatoires de 178 paramètres. Cette série est présentée au Détective à côté d'une série témoin assemblée à partir de vraies journées. Sans savoir quelles journées sont les bonnes, le Détective essaye de répondre à la question suivante: «Entre ces deux journées, laquelle a réellement existé?». Il donne ensuite sa réponse au Faussaire qui adapte sa prochaine proposition pour essayer d'augmenter ses chances de tromper le Détective. Si le Détective a donné la bonne réponse, alors le Faussaire tente une approche différente. S'il s'est trompé, alors le Faussaire mémorise son succès avec cette combinaison tout en explorant de nouvelles pistes.

(b) Vient ensuite le Détective. Il doit apprendre comment différencier les journées du Faussaire de celles réellement vécues. Son expérience est initialement nulle; il est encore un apprenti Détective. Pour qu'il s'améliore, le Faussaire lui présente des journées qu'il a fabriquées, puis le Détective doit comparer celles-ci à des journées ayant existé. Les résultats de ses performances lui sont communiqués à la fin de son cycle de travail. Ainsi, en se remettant en question sans cesse et

ressenti de l'admiration" ; $nb_{\text{pitié}}$ = "Le nombre de fois où j'ai eu pitié de quelqu'un ou de quelque chose" ; $nb_{\text{déception}}$ = "Le nombre de fois où j'ai été déçu" ; $nb_{\text{fierté}}$ = "Le nombre de fois où j'ai été fier de moi" ; nb_{jalousie} = "Le nombre de fois où j'ai été jaloux de quelqu'un" ; $nb_{\text{incompréhension}}$ = "Le nombre de fois où j'ai ressenti de l'incompréhension" ; nb_{honte} = "Le nombre de fois où j'ai eu honte de moi" ; $nb_{\text{agacement}}$ = "Le nombre de fois où j'ai été agacé par quelqu'un ou quelque chose" ; $nb_{\text{total_émotions}}$ = "La somme de toutes

- en répétant l'expérience suffisamment de fois, le Détective comprend comment une journée se construit et ce qui fait d'elle une journée cohérente avec un mode de vie. Pendant ce temps-là, le Faussaire ne cherche pas à s'améliorer.
- (c) Ces deux phases d'apprentissage sont complémentaires et doivent se faire tour à tour. En effet, si le Détective devient trop bon trop vite, alors le Faussaire n'arrivera jamais à le tromper et donc ne pourra pas s'améliorer lui-même. Au contraire, si le Faussaire devient trop vite meilleur que le Détective, alors le Détective sera toujours trompé et ne deviendra jamais assez bon pour que les deux algorithmes puissent chacun devenir des experts et se nourrir l'un l'autre. Les phases d'apprentissage du Faussaire et du Détective alternent régulièrement pour que les deux protagonistes puissent progresser au même rythme et que le cas de figure où l'un aurait distancé l'autre ne se réalise pas⁹.
- (d) Après des milliers de cycles comme celui décrit dans les étapes (a) et (b), les deux modèles ont atteint leurs potentiels maximaux. Malgré le fait que le Détective sache parfaitement faire la différence entre une vraie et une fausse journée, le Faussaire est devenu si bon dans son imitation que les imperfections n'existent plus. Le Faussaire est devenu un Maître Faussaire, et même le meilleur des experts ne peut différencier son travail d'un original.
- (e) Maintenant que le Faussaire est devenu un artisan d'exception, nous pouvons lui passer commande de ce pour quoi nous l'avons en premier lieu créé.

les fois où j'ai ressenti une des dix-sept émotions mesurées" ; (7) nb_bonnes_nouv = "Le nombre de bonnes nouvelles" ; n_meilleure_nouv = "La note de la meilleure bonne nouvelle" ; moy_bonnes_nouv = "La moyenne des notes des bonnes nouvelles" ; nb_mauv_nouv = "Le nombre de mauvaises nouvelles" ; n_pire_nouv = "La note de la pire mauvaise nouvelle" ; moy_mauv_nouv = "La moyenne des notes des mauvaises nouvelles" ; n_karma = "La note attribuée aux choses exogènes et inattendues" ; (8) nb_mails_envoi = "Le nombre d'emails envoyés" ; nb_mails_recus = "Le nombre d'emails reçus (hors spam)" ; nb_sms_envoi = "Le nombre de sms envoyés" ; nb_sms_recus = "Le nombre de sms reçus" ; nb_engag_sms = "Le nombre de conversations sms que d'autres ont engagées avec moi" ; nb_conv_sms = "Le nombre de conversations par sms" ; nb_messenger_recus = "Le nombre de messages reçus sur l'application Messenger" ; nb_messenger_envoi = "Le nombre de messages envoyés sur l'application Messenger" ; nb_interlocuteurs = "Le nombre d'interlocuteurs différents sur l'application Messenger" ; nb_telephone = "Le nombre d'appels téléphoniques" ; t_telephone = "La durée totale de tous les appels téléphoniques" ; t_skype = "La durée passée sur Skype" ; (9) h_lever = "L'heure à laquelle je me suis levé" ; h_coucher = "L'heure à laquelle je me suis couché" ; t_sommeil = "La durée de mon sommeil" ; delta_sommeil = "La différence de durée entre mon sommeil et huit heures de sommeil" ; moy_delta_sommeil_3d = "La différence moyenne de durée entre mon sommeil et huit heures de sommeil sur trois jours" ; moy_delta_sommeil_7d = "La différence moyenne de durée entre mon sommeil et huit heures de sommeil sur une semaine" ; moy_lever = "La moyenne des heures du lever sur

Analyse

Mise en pratique

Le Détective et le Faussaire sont deux instances très proches de deux réseaux neuronaux. Une fois paramétrée, leur structure algorithmique est mise en place. Cet agencement précis – les deux réseaux neuronaux encapsulés avec cette alternance de phases d'apprentissage – porte un nom : il s'agit de réseaux antagonistes génératifs. Dans cet article, et comme dans la littérature relative à cette technologie en général, nous nous référerons aux réseaux antagonistes génératifs par leur acronyme anglo-saxon : les GAN (pour *generative adversarial networks*). Régulièrement utilisés pour générer des images, ils sont ici adaptés pour générer des journées. Dans notre cas de figure, nous avons choisi d'utiliser la version la plus classique des GAN, celle développée en 2014 par Ian Goodfellow et son équipe¹⁰. La manière la plus simple d'implémenter l'algorithme de Goodfellow pour l'appliquer à notre problème était de représenter mes journées sous la forme d'images¹¹. En effet, les GAN ayant été développés pour l'étude des matrices de pixels, la plupart des structures de code en Python ont été pensées à cet effet : on les utilise principalement pour générer de fausses images. L'autre avantage de ce passage au sensible est de pouvoir visualiser les fausses journées en parallèle des vraies ; faites entièrement de nombres, il devient compliqué de rapidement obtenir une vision d'ensemble de ce à quoi « ressemblent » les journées, le problème est transformé une fois que les données sont représentées sous forme d'image.

La *figure 3* est une représentation utilisant l'échelle de couleurs *bwr* sur la journée du 6 janvier 2018. L'échelle *bwr*, pour *blue white red*, passe du bleu le plus saturé pour les valeurs les plus minimales au rouge le plus saturé pour les valeurs maximales, en passant par le blanc. Cette représentation n'inclut pas tous les paramètres qui composent cette journée, mais seulement 150 d'entre eux (10 colonnes et 15 lignes) car certains des paramètres (surtout des paramètres dont les valeurs sont calculées à partir d'autres paramètres) n'ont pas à être utilisés pour cette étape du projet. Pour pouvoir être représentée en nuances *bwr*, chaque donnée a vu sa valeur être normalisée entre 0 et 1.

```
trois jours" ; moy_coucher = "La moyenne des heures du coucher sur trois jours" ; b_maison_soir = "Un booléen correspondant au fait de m'être couché chez moi" ; b_maison_matin = "Un booléen correspondant au fait de m'être levé chez moi" ; t_eveil = "La durée que j'ai passé éveillé" ; n_nuit = "La note attribuée à la qualité de mon sommeil" ; b_reve = "Un booléen correspondant au fait d'avoir fait un rêve" ; t_lit = "La durée que j'ai passé dans un lit la nuit passée" ; n_sleepcycle = "Le pourcentage de qualité de mon sommeil selon l'application
```

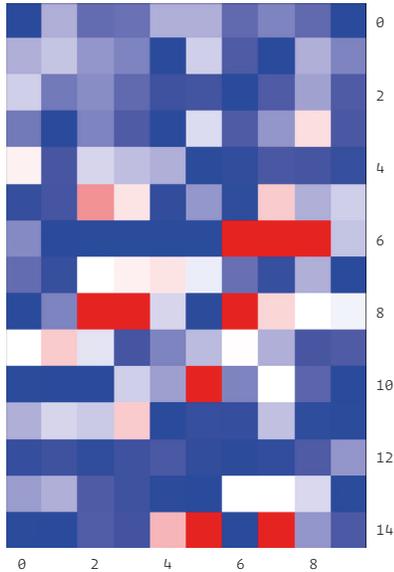


figure 3 :
Représentation graphique
du 6 janvier 2018

Bilan 2

Après avoir entraîné les GAN, nous avons demandé au Maître Faussaire de nous fournir 50 000 journées, représentées sous la forme de 50 000 images de 10 par 15 pixels. Ces images sont ensuite cassées en une ligne de 150 pixels dont nous ne gardons que la valeur de l'intensité. Chaque valeur est ensuite dénormalisée pour retrouver une valeur cohérente avec le jeu de données d'origine. Les 29 valeurs restantes sont calculées de manière endogène. Si l'Hypothèse est juste, alors il est probable que, en principe, la journée parfaite soit contenue dans les 50 000 faux obtenus à la fin du processus.

Sleepcycle" ; **(10)** moy_joye = "L'intensité moyenne des fois où j'ai ressenti de la joie" ; moy_surprise = "L'intensité moyenne des fois où j'ai ressenti de la surprise" ; moy_tristesse = "L'intensité moyenne des fois où j'ai été en colère" ; moy_colere = "L'intensité moyenne des fois où j'ai eu peur" ; moy_peur = "L'intensité moyenne des fois où j'ai ressenti du dégoût" ; moy_degout = "L'intensité moyenne des fois où j'ai méprisé quelqu'un ou quelque chose" ; moy_mepris = "L'intensité moyenne des fois où je me suis senti con" ; moy_con = "L'intensité moyenne des fois où j'ai été en colère" ; moy_soulagement = "L'intensité moyenne des fois où j'ai ressenti du soulagement" ; moy_admiration = "L'intensité moyenne des fois où j'ai ressenti de l'admiration" ; moy_pitie = "L'intensité moyenne des fois où j'ai eu pitié de quelqu'un ou de quelque chose" ; moy_deception = "L'intensité moyenne des fois où j'ai été déçu" ; moy_fierte = "L'intensité moyenne des fois où j'ai été fier de moi" ; moy_jalousie = "L'intensité moyenne des fois où j'ai été jaloux de quelqu'un" ; moy_incomprehension = "L'intensité moyenne des fois où j'ai ressenti de l'incompréhension" ; moy_honte = "L'intensité moyenne des fois où j'ai eu honte de moi" ; moy_agacement = "L'intensité moyenne des fois où j'ai été agacé par quelqu'un ou quelque chose" ; **(11)** numero_journee = "Le jour 1 marque le début de la collecte des données et le jour 413 en marque la fin" ; t_carnet = "Le temps passé à remplir le carnet des données de la journée" ; n_fiabilite = "La note attribuée à la fiabilité de la récolte des données" ; nb_donnees = "Le nombre de données

La journée parfaite

RÉPARTITION DES RÉSULTATS

Une fois les calculs du Faussaire terminés, nous nous sommes retrouvés avec ces 50 000 fausses journées à analyser. À notre disposition sont les 221 Psychologues précédemment entraînés, et prêts à estimer la valeur du paramètre `n_journee` pour chacune des journées à examiner. Chacun des 221 Psychologues a ainsi évalué toutes les fausses journées du Maître Faussaire puis s'est efforcé à se représenter ce à quoi pouvait ressembler la journée dont les caractéristiques défilaient sous ses yeux, avec sa propre expertise et sa propre manière de concevoir l'esprit. Chaque fois qu'un Psychologue avait fini son évaluation des 50 000 journées,

nous lui demandions de nous fournir celle qu'il pensait être la meilleure journée ainsi que sa note.

Ainsi, il résulta de cette étape 221 couples constitués d'une note et d'un numéro de journée. Dans cette table de résultats, certains numéros étaient redondants : les 14 meilleures journées ayant obtenu la note de 20 sont représentées par le diagramme de Sankey ci-dessous. Il se lit de la manière suivante : à gauche sont les journées choisies par les Psychologues et à droite est la note obtenue. Plus le nombre de 20 attribués à la journée est grand, plus l'épaisseur des lignes qui lient la note aux numéros des journées est importante.

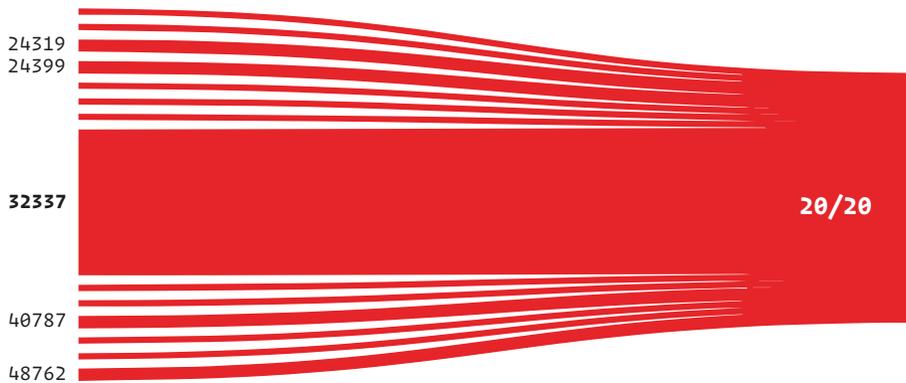


figure 4 :
Diagramme de Sankey
de la distribution des 20 obtenus

relevées" ; (12) `i_joye` = "L'intensité sommée de chaque fois où j'ai ressenti de la joie" ; `i_surprise` = "L'intensité sommée de chaque fois où j'ai ressenti de la surprise" ; `i_tristesse` = "L'intensité sommée de chaque fois où j'ai été triste" ; `i_colere` = "L'intensité sommée de chaque fois où j'ai été en colère" ; `i_peur` = "L'intensité sommée de chaque fois où j'ai été eu peur" ; `i_degout` = "L'intensité sommée de chaque fois où j'ai été ressenti du dégoût" ; `i_mepris` = "L'intensité sommée de chaque fois où j'ai méprisé quelqu'un ou quelque

EXISTENCE ÉNIGMATIQUE

Une journée ressort particulièrement face aux autres ; il s'agit de la journée 32337. Le fait que cette journée ait réussi à décrocher autant de 20 – 23 en tout, soit dix à vingt fois d'occurrences plus que les autres – ne peut vouloir dire qu'une seule chose : les Psychologues, bien que très différents, s'accordent à dire que certaines combinaisons de paramètres bien précises surclassent les autres.

LES BOÎTES À MOUSTACHE

Une dernière étape fut nécessaire pour extraire l'Équation du bonheur des résultats produits par les 221 Psychologues. Après avoir pu regrouper toutes les journées qui ont reçu un 20, il suffisait de les interpréter, et cette ultime étape allait passer par la représentation. Les résultats des Psychologues étant parfois assez changeants d'un psychologue à l'autre, nous pouvons considérer que la combinaison de ces multiples estimateurs indépendants permet de réduire la variance dans les résultats (cette technique est appelée le *bagging*¹²). En d'autres termes, nous supposons qu'un individu, dans notre cas présument expert, n'est pas capable de donner des résultats aussi valables que ceux d'un ensemble d'experts¹³.

Les diagrammes en boîte à moustache, présentés en une frise à la fin de cet article, se lisent de gauche à droite et de haut en bas. On retrouve à gauche les paramètres les plus importants et à droite ceux les moins importants. Les boîtes à moustaches se lisent par couples, avec en bleu la distribution réelle des valeurs d'un paramètre, et en rouge la distribution des valeurs appartenant aux journées ayant été générées par le Maître Faussaire et auxquelles au moins l'un des Psychologues aurait attribué un 20 (elles sont au nombre de 10). Les boîtes à moustaches sont des plus classiques avec à leurs extrémités les extrema de chacune des distributions, le minimum étant en bas et le maximum en haut. Les extrémités des rectangles correspondent respectivement aux premiers et troisièmes quartiles. La barre horizontale circonscrite dans le rectangle réfère à la valeur médiane de la distribution. Le segment relié par les deux cercles, pour finir, représente la moyenne. Si nous nous fions aux résultats des Psychologues, alors seul 0,02% des journées générées a réussi à décrocher un 20.

Cela représente environ une journée sur 5000, soit en moyenne une journée tous les 13,7 ans. En supposant que le Maître Faussaire ait réussi un travail fidèle dans sa reproduction de mon mode de vie, alors il semblerait que les journées parfaites soit atteignables, mais qu'elles ne seront rencontrées que très rarement.

chose" ; i_con = "L'intensité sommée de chaque fois où je me suis senti con" ; i_soulagement = "L'intensité sommée de chaque fois où j'ai ressenti du soulagement" ; i_admiration = "L'intensité sommée de chaque fois où j'ai ressenti de l'admiration" ; i_pitié = "L'intensité sommée de chaque fois où j'ai eu pitié de quelqu'un ou de quelque chose" ; i_deception = "L'intensité sommée de chaque fois où j'ai été déçu" ; i_fierté = "L'intensité sommée de chaque fois où j'ai été fier de moi" ; i_jalousie = "L'intensité sommée de chaque fois où j'ai

La journée parfaite

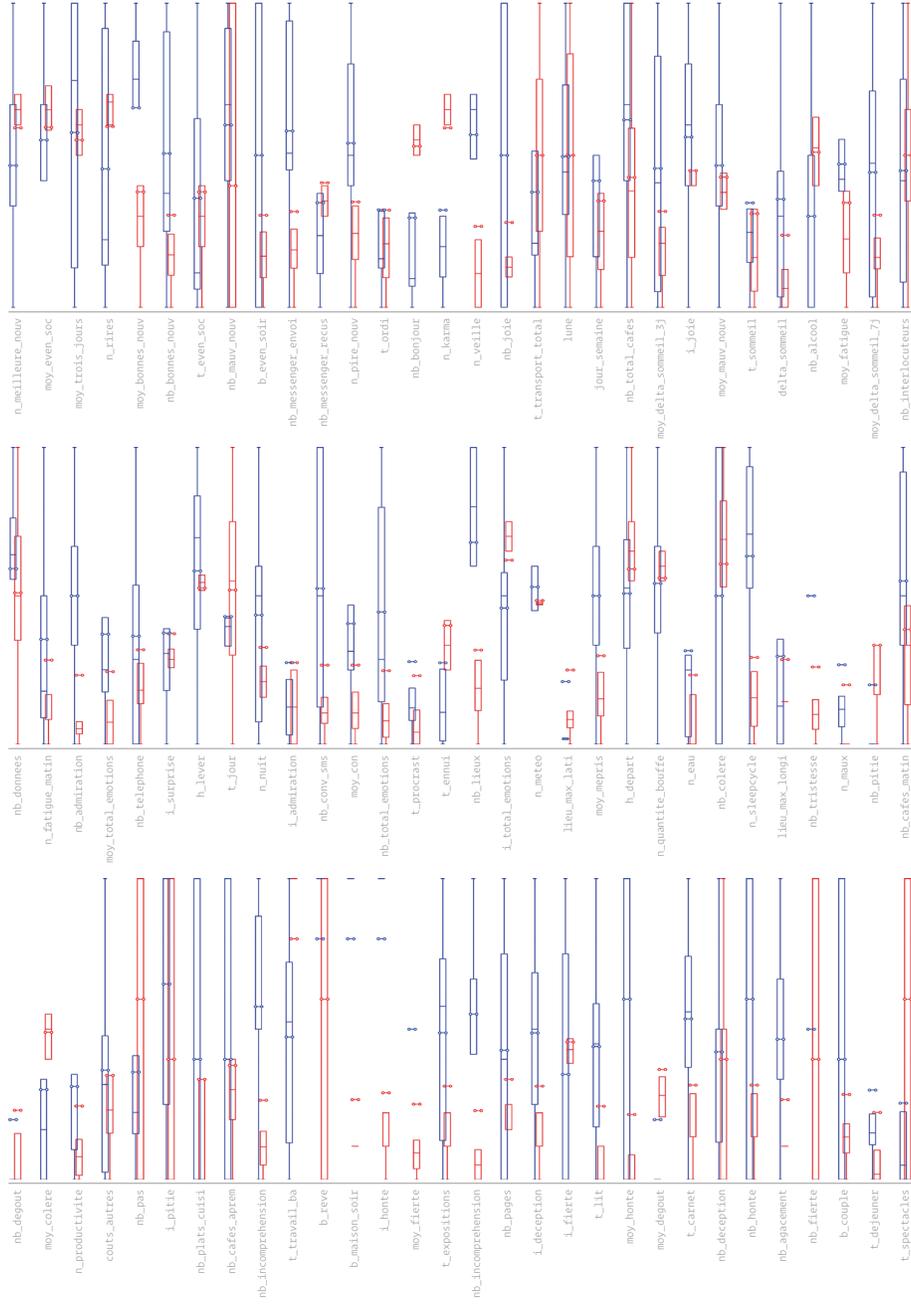
Qu'en est-il alors de l'Hypothèse? Est-il envisageable d'essayer de forcer le destin? De tenter d'approcher une de ces journées artificiellement et de forcer le bonheur à se réaliser? Il faudrait pour cela prédire la météo, calculer en amont le retard de sommeil à avoir, aller dans des lieux prédéfinis à l'avance et choisir les personnes à qui parler ou dire bonjour. Il faudrait se restreindre sur ce que nous ingurgitons, sur le temps que nous passons à faire X par rapport à Y ... En bref, respecter, contrôler et surtout obéir aux 178 paramètres comme s'ils étaient les composants de la liste d'ingrédients exhaustive et inaltérable d'une réaction chimique stochastique et folle. Est-ce possible? Peut-être. Est-ce que la journée parfaite en résulterait? Certainement pas. Imposer une ascèse aussi rigoureuse n'engrangerait rien de positif. La solution est-elle d'attendre que les étoiles s'alignent (Lune* (calendrier) étant, nous le rappelons, un des paramètres à satisfaire)? La réponse la plus plausible semble être oui. La seule certitude à extraire ici est que, si je ne change rien à mes habitudes actuelles, une journée sur 5000 en moyenne résultera sur la journée parfaite.

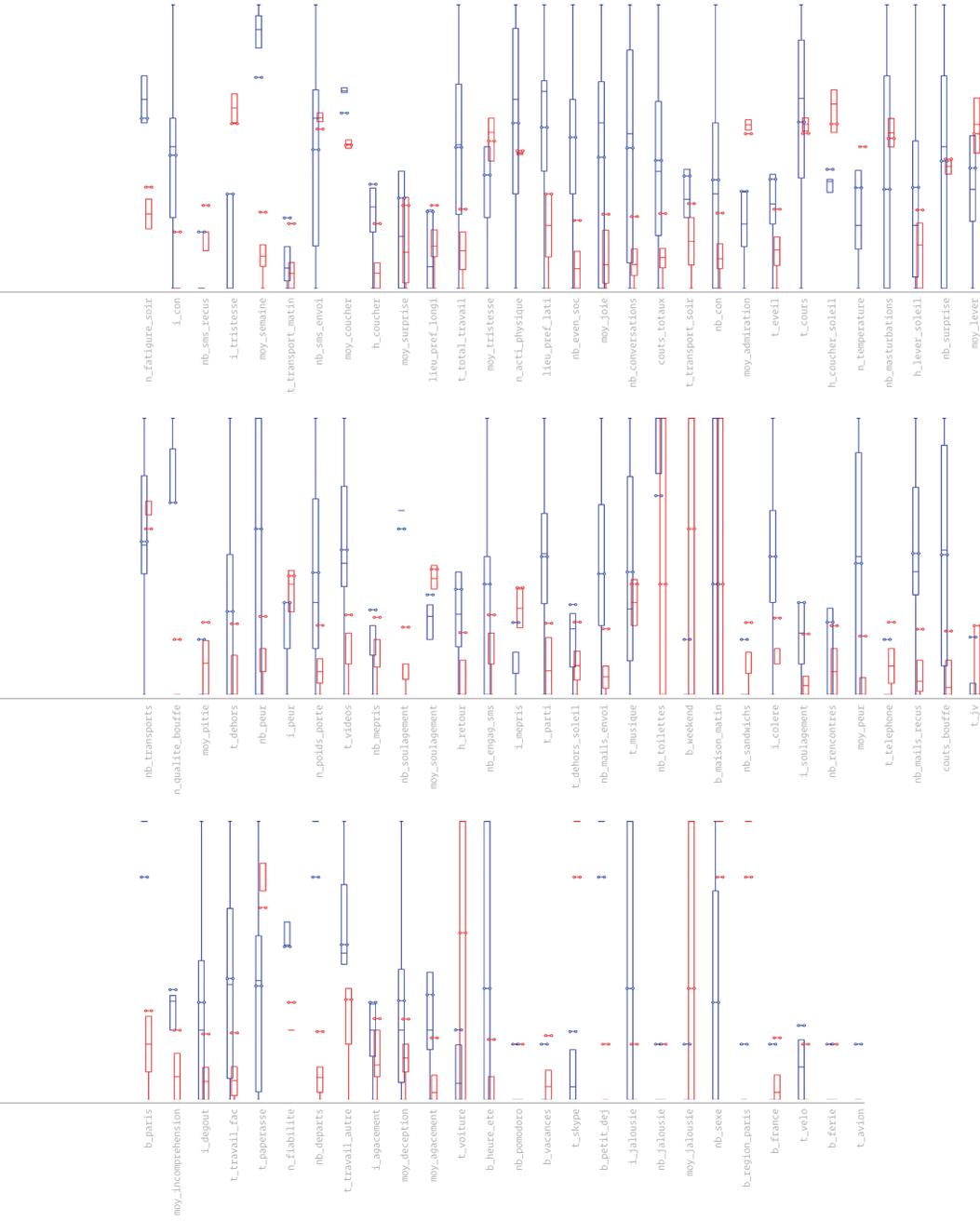
Il est possible que changer mon comportement d'une manière réfléchie et pesée pourrait augmenter ces chances, mais les combinatoires sont si grandes et les contingences si nombreuses qu'il nous est très difficile de savoir quoi modifier, et comment. La meilleure stratégie serait alors de miser sur la sérendipité et donc de vivre de manière totalement désintéressée de l'objectif que nous nous étions originellement fixé. La seule bonne décision à prendre serait peut-être de remonter au-delà de l'hédonisme utilitariste, en passant ces questions à travers le tamis d'une autre vision du monde, comme celle d'Al-Fârâbî¹⁴, ou bien plus loin encore, comme dans la pensée grecque antique telle que chez Épictète¹⁵, pour ainsi s'abandonner à l'ataraxie et à l'acatalepsie d'une vie plus simple. Voilà sûrement l'une des seules voies envisageables afin d'éviter l'écueil qui transformerait la recherche du bonheur en un amuse-bouche pour nihilistes¹⁶. Finalement, cette voie permet la réaffirmation de la noblesse de cette quête, tout comme, nous l'espérons, sa légitimité.

été jaloux de quelqu'un"; i_incomprehension = "L'intensité sommée de chaque fois où j'ai ressenti de l'incompréhension"; i_honte = "L'intensité sommée de chaque fois où j'ai eu honte de moi"; i_agacement = "L'intensité sommée de chaque fois où j'ai été agacé par quelqu'un ou quelque chose"; i_total_emotions = "La somme des intensités des dix-sept émotions mesurées"; (13) t_spectacles = "Le temps passé à être spectateur d'un spectacle"; t_expositions = "Le temps passé à visiter des expositions"; t_ordi = "Le temps passé sur l'ordinateur"; t_jv = "Le temps passé à jouer à des jeux vidéos"; nb_pages = "Le nombre de pages de livres lus"; t_musique = "Le temps passé à écouter de la musique"; t_videos = "Le temps passé à regarder des vidéos"; t_ennui = "Le temps passé à s'ennuyer"; t_procrast = "Le temps passé à procrastiner"; couts_autres = "La somme des prix des items achetés durant la journée (sauf nourriture)"; (14) lieu_max_lati = "La latitude du lieu où j'ai passé le plus de temps"; lieu_max_longi = "La longitude du lieu où j'ai passé le plus de temps"; lieu_pref_lati = "La latitude du lieu dans lequel j'ai préféré être"; lieu_pref_longi = "La longitude du lieu dans lequel j'ai préféré être"; b_region_paris = "Un booléen correspondant à ma présence en région parisienne"; b_paris = "Un booléen correspondant à ma présence à Paris"; b_france = "Un booléen correspondant à ma présence en France"; nb_lieux = "Le nombre de lieux auxquels je suis allé"; (15) h_depart = "L'heure à laquelle je suis parti de chez moi pour la première fois"; h_retour = "L'heure à laquelle je suis rentré chez moi pour la dernière fois"; t_parti = "La durée entre le premier départ et le dernier retour"; nb_departs = "Le nombre de fois

ទិន្នន័យ ទំនាស់ ៤៧

où je suis sorti de chez moi" ; t_transport_matin = "Le temps passé dans les transports en commun le matin" ; t_transport_soir = "Le temps passé dans les transports en commun l'après-midi" ; t_voiture = "Le temps passé dans une voiture" ; t_velo = "Le temps passé sur un vélo" ; t_avion = "Le temps passé dans un avion" ; t_transport_total = "Le temps total passé dans des moyens de transport" ; nb_transports = "Le nombre de fois où je suis rentré dans un moyen de transport" ; nb_pas = "Le nombre de milliers de pas" ; **(16)** nb_masturbations = "Le nombre de fois où je me suis masturbé" ; nb_sexe = "Le nombre de rapports sexuels que j'ai eu" ; b_couple = "Un booléen correspondant au fait d'être en couple." ; **(17)** t_cours = "Le temps passé à assister à des cours" ; t_travail_ba = "Le temps passé à travailler sur mes projets des Beaux-Arts" ; t_travail_fac = "Le temps passé à travailler sur mes cours à l'université" ; t_paperasse = "Le temps passé à remplir de la paperasse" ; t_travail_autre = "Le temps passé à travailler sur des projets n'ayant pas de lien avec les possibilités précédentes" ; t_total_travail = "Le temps total que j'ai passé à travailler" ; nb_pomodoro = "Le nombre d'itérations de la technique de productivité Pomodoro" ; n_productivite = "La note attribuée à ma productivité personnelle (au sens économique du terme)" ; **(18)** jour_semaine = "Les jours de la semaine ; du lundi (0) au dimanche (6)" ; b_weekend = "Un booléen correspondant au weekend" ; b_ferie = "Un booléen correspondant aux jours fériés" ; b_vacances = "Un booléen correspondant aux vacances scolaires" ; b_heure_ete = "Un booléen correspondant à l'heure d'été" ; *lune = "Les phases de la lune en démarrant à la nouvelle lune.





Notes

- [1] Bentham, J. (2000). *An Introduction to the Principles of Morals and Legislation*. Batoche Books. (Texte original publié en 1789).
- [2] Kahneman, D. (1999). *Well-Being: the Foundations of Hedonic Psychology*. Russell Sage Foundation.
- [3] Voir «epochalism». Morozov, E. (2013). *To Save Everything, Click Here*. PublicAffairs. p. 35.
- [4] Vigezzi, M. (2014). La mécanique des illusions dans la pensée économique. Dans Braitto, A. et Citton, Y. (dirs.), *Technologies de l'enchantement : pour une histoire multidisciplinaire de l'illusion*. UGA Éditions. p. 298.
- [5] *Ibid.*, pp. 301-304.
- [6] Voir « Démon de Laplace ». Laplace, P.-S. (1814). *Essai philosophiques sur les probabilités*. Bachelier. p. 2.
- [7] Connes, A. et al. (2018). Chapitre 35 : Le diable est dans la machine. Dans *Le Spectre d'Atacama*. Odile Jacob.
- [8] Pedregosa, F. et al. (2011). Scikit-learn: Machine Learning in Python. *JMLR*, (12), pp. 2825-2830.
- [9] Voir « mode collapse ». Thanh-Tung, H. et al. (2018). On Catastrophic Forgetting and Mode Collapse in GANS. arxiv 1807.04015.
- [10] Goodfellow, I. et al. (2014). Generative Adversarial Networks. arxiv 1406.2661.
- [11] Garima, N. (2019). MNIST_GAN, dépôt GitHub personnel. Consulté le 30 janvier 2020 sur : https://github.com/Garima13a/MNIST_GAN.
- [12] Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2), pp. 123-140.
- [13] Surowiecki, J. (2004). *The Wisdom of Crowds*. Anchor Books. p. 30.
- [14] Al-Fârâbî. (2005). *Kitâb tahsîl al-sa'âda* [De l'obtention du bonheur]. (Trad. N. Lévy et O. Sedeyn). Éditions Allia. (Texte original publié au IX^e siècle).
- [15] Épictète. (1875). *Ἐγχειρίδιον* [Manuel d'Épictète]. (Trad. J.-M. Guyau). Librairie Ch. Delagrave. (Texte original publié vers 125).
- [16] Ligotti, T. (2010). *The Conspiracy Against the Human Race: A Contrivance of Horror*. Hippocampus Press. p. 66.