Introduction

Normally, when the name "Drug" is heard, people generally relate it to narcotics. But, from the aspects of Pharmaceutical Sciences1, "Drugs" are wide variety of chemical substances which when consumed in any manner – inhaled, injected, smoked, orally-consumed, absorbed (via skin) etc. affects the living body and brings chemical consumptions inside the body. Most of the drugs are useful for the treatment of diseases found in Human body, and they are better known by the name "Medicine". Meanwhile, the other variant of Drugs are Recreational drugs. These drugs also cause chemical changes in the state of body, but rather their usage for treatment, these are used for recreation purposes by different age categories of humans. All of these drugs, more appropriately, the Narcotic drugs or recreational drugs leaves a devastating impact on the person who consumes it. These impacts remain in the body even after few days of consumption in human body. Since, they result to chemical changes, parameters and properties of blood, can be used to predict and determine when and what kind of narcotic one has consumed.

Drug use is a risk behavior that does not happen in isolation; it constitutes an important factor for increasing risk of poor health, along with earlier mortality and morbidity, and has significant consequences for society. Drug consumption and addiction constitutes a serious problem globally. It includes numerous risk factors, which are defined as any attribute, characteristic, or event in the life of an individual that increases the probability of drug consumption. A number of factors are correlated with initial drug use including psychological, social, individual, environmental, and economic factors. These factors are likewise associated with a number of personality traits. While legal drugs such as sugar, alcohol and tobacco are probably responsible for far more premature death than illegal recreational drugs, the social and personal consequences of recreational drug use can be highly problematic.

Psychologists have largely agreed that the personality traits of the Five Factor Model (FFM) are the most comprehensive and adaptable system for understanding human individual differences. The FFM comprises Neuroticism(N), Extraversion (E), Openness to Experience(O), Agreeableness(A), and Conscientiousness(C). A number of studies have illustrated that personality traits are associated to drug consumption. The importance of the relationship between high N and the presence of psychotic symptoms following cocaine-induced drug consumption. The personality traits of N,E, and C are highly correlated with hazardous health behaviors. A low score of C, and high score of E or high score of N correlate strongly with multiple risky health behaviors. Alcohol use to be associated with lower A and C, and higher E. They found also that lower A and C, and higher O are associated with marijuana. The relationship between low C and drug consumption is moderated by poverty; low C is a stronger risk factor for illicit drug usage among those with relatively higher socioeconomic status. They found that high N, and low A and C are associated with higher risk of drug use (including cocaine, crack, morphine, codeine, and heroin). It should be mentioned that high N is positively associated with many other addictions like Internet addiction, exercise addiction, compulsive buying, and study addiction. An individual's personality profile plays a role in becoming a drug user. The personality profiles for the users and non-users

of nicotine, cannabis, cocaine, and heroin are associated with a FFM of personality samples from different communities. They also highlight the links between the consumption of these drugs and low C. A positive correlation between N and O, and drug use, while, increasing scores for C and A decreases risk of drug use. Previous studies demonstrated that participants who use drugs including alcohol and nicotine have a strong positive correlation between A and C and a strong negative correlation for each of these factors with N. Three high-order personality traits are proposed as endophenotypes for substance use disorders: Positive Emotionality, Negative Emotionality, and Constraint. The statistical characteristics of groups of drug users and non-users have been studied by many authors. They found that the personality profile for the users and non users of tobacco, marijuana, cocaine, and heroin are associated with a higher score on N and a very low score for C. Sensation seeking is also higher for users of recreational drugs.

Dataset

The database was collected by Elaine Fehrman between March 2011 and March 2012. In January 2011, the research proposal was approved by the University of Leicester's Forensic Psychology Ethical Advisory Group, and subsequently received favorable opinion from the University of Leicester School of Psychology's Research Ethics Committee (PREC).

The study recruited 2051 participants over a 12-month recruitment period. Of these persons, 166 did not respond correctly to a validity check built into the middle of the scale, so were presumed to being inattentive to the questions being asked. Nine of these persons were found to also have endorsed using a fictitious recreational drug, and which was included precisely to identify respondents who over-claim, as have other studies of this kind. This led a useable sample of 1885 participants(male/female = 943/942).

The snowball sampling methodology recruited a primarily (93.7%) native English-speaking sample, with participants from the UK (1044; 55.4%), the USA (557; 29.5%), Canada (87; 4.6%), Australia (54; 2.9%), New Zealand (5; 0.3%) and Ireland (20; 1.1%). A total of 118 (6.3%) came from a diversity of other countries, none of whom individually met 1% of the sample or did not declare the country of location. Further optimizing anonymity, persons reported their age band, rather than their exact age; 18-24 years (643; 34.1%), 25-34 years (481; 25.5%), 35-44 years (356; 18.9%),45-54 years (294; 15.6%), 55-64 (93; 4.9%), and over 65 (18; 1%). This indicates that although the largest age cohort band were in the 18 to 24 range, some 40% of the cohort was 35 or above, which is an age range often missed in studies of this kind. The sample recruited was highly educated, with just under two thirds (59.5%) educated to, at least, degree or professional certificate level: 14.4% (271) reported holding a professional certificate or diploma, 25.5% (481) an undergraduate degree, 15% (284) a master's degree, and 4.7% (89) a doctorate. Approximately 26.8% (506) of the sample had received some college or university tuition although they did not hold any certificates; lastly, 13.6% (257) had left school at the age of 18 or younger. Participants were asked to indicate which racial category was broadly representative of their cultural

background. An overwhelming majority (91.2%; 1720) reported being white, (1.8%; 33) stated they were Black, and (1.4%; 26) Asian. The remainder of the sample (5.6%; 106) described themselves as 'Other' or 'Mixed' categories. This small number of persons belonging to specific non-white ethnicities precludes any analyses involving racial categories.

The dataset contains records for 1885 respondents. Each record has 12 attributes. These 12 attributes are:

1. ID	2. Age
3. Gender	4. Education
5. Country	6. Ethnicity
7. Nscore	8. Escore
9. Oscore	10. Ascore
11. Cscore	12. Impulsive
13. Sensation Seeing(SS)	

In addition, participants were questioned concerning their use of 18 legal and illegal drugs (**alcohol, amphetamines, amyl nitrite, benzodiazepine, cannabis, chocolate, cocaine, caffeine, crack, ecstasy, heroin, ketamine, legal highs, LSD, methadone, mushrooms, nicotine and volatile substance abuse and one fictitious drug (Semeron)** which was introduced to identify over-claimers. For each drug they have to select one of the answers: never used the drug, used it over a decade ago, or in the last decade, year, month, week, or day. Database contains 18 classification problems. Each of independent label variables contains seven classes: "Never Used", "Used over a Decade Ago", "Used in Last Decade", "Used in Last Year", "Used in Last Month", "Used in Last Week", and "Used in Last Day".

Personality measurements

In order to assess personality traits of the sample, the Revised NEO Five-Factor Inventory (NEO-FFI-R) questionnaire was employed. The NEO-FFI-R is a highly reliable measure of basic personality domains; internal consistencies are 0.84 (N); 0.78 (E); 0.78 (O); 0.77 (A), and 0.75 (C) Egan. The scale is a 60-item inventory comprised of five personality domains or factors. The NEO-FFI-R is a shortened version of the Revised NEO-Personality Inventory (NEO-PI-R). The five factors are: N, E, O, A, and C with 12 items per domain. The five traits can be summarized as:

1. Neuroticism (N) is a long-term tendency to experience negative emotions such as nervousness, tension, anxiety and depression;

2. Extraversion (E) is manifested in outgoing, warm, active, assertive, talkative, cheerful, and in search of stimulation characteristics;

3. Openness to experience (O) is a general appreciation for art, unusual ideas, and imaginative, creative, unconventional, and wide interests,

4. Agreeableness (A) is a dimension of interpersonal relations, characterized by altruism, trust, modesty, kindness, compassion and cooperativeness;

5. Conscientiousness (C) is a tendency to be organized and dependable, trong-willed, persistent, reliable, and efficient.

Predictions:

Since we have 18 types of drugs in the dataset and each drug is classified into the seven Classes. For each drug we have to select one of the answers. Each of independent label variables contains seven classes: "Never Used", "Used over a Decade Ago", "Used in Last Decade", "Used in Last Year", "Used in Last Month", "Used in Last Week", and "Used in Last Day".

For each drug we have to predict the output that when did the user use that drug. But first we need to train the modal that can predict the results for us.
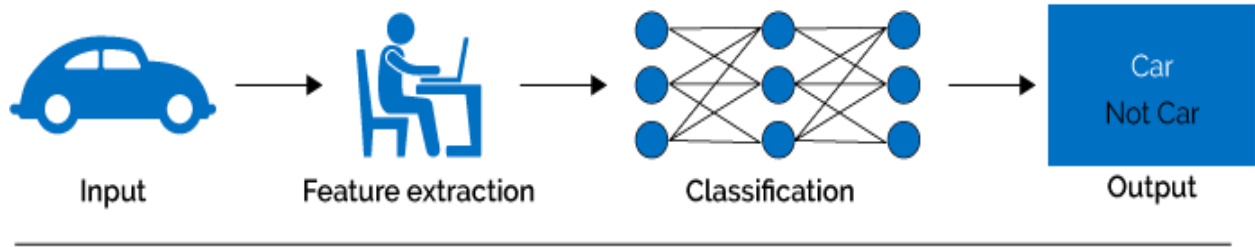
A number of classification methods were employed (Backpropagation, Perceptron, Decision tree, Support Vector Machine(SVM), k-nearest neighbors) and the most effective classifier was selected for each drug. The quality of classification was surprisingly high with sensitivity and specificity (evaluated by leave-one-out cross-validation) being greater than 70% for almost all classification tasks. The best results with sensitivity and specificity being greater than 75% were achieved for cannabis, crack, ecstasy, legal highs, LSD, and volatile substance abuse (VSA).
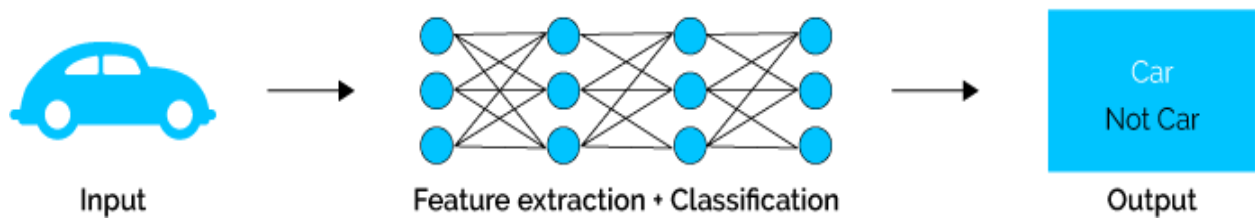
Algorithms:

What are artificial neural networks?

An artificial neuron network (ANN) is a computational model based on the structure and functions of biological neural networks. Information that flows through the network affects the structure of the ANN because a neural network changes - or learns, in a sense - based on that input and output. ANNs are considered nonlinear statistical data modeling tools where the complex relationships between inputs and outputs are modeled or patterns are found. ANN is also known as a neural network.

Machine Learning

Input → Feature extraction → Classification → Output (Car / Not Car)

Deep Learning

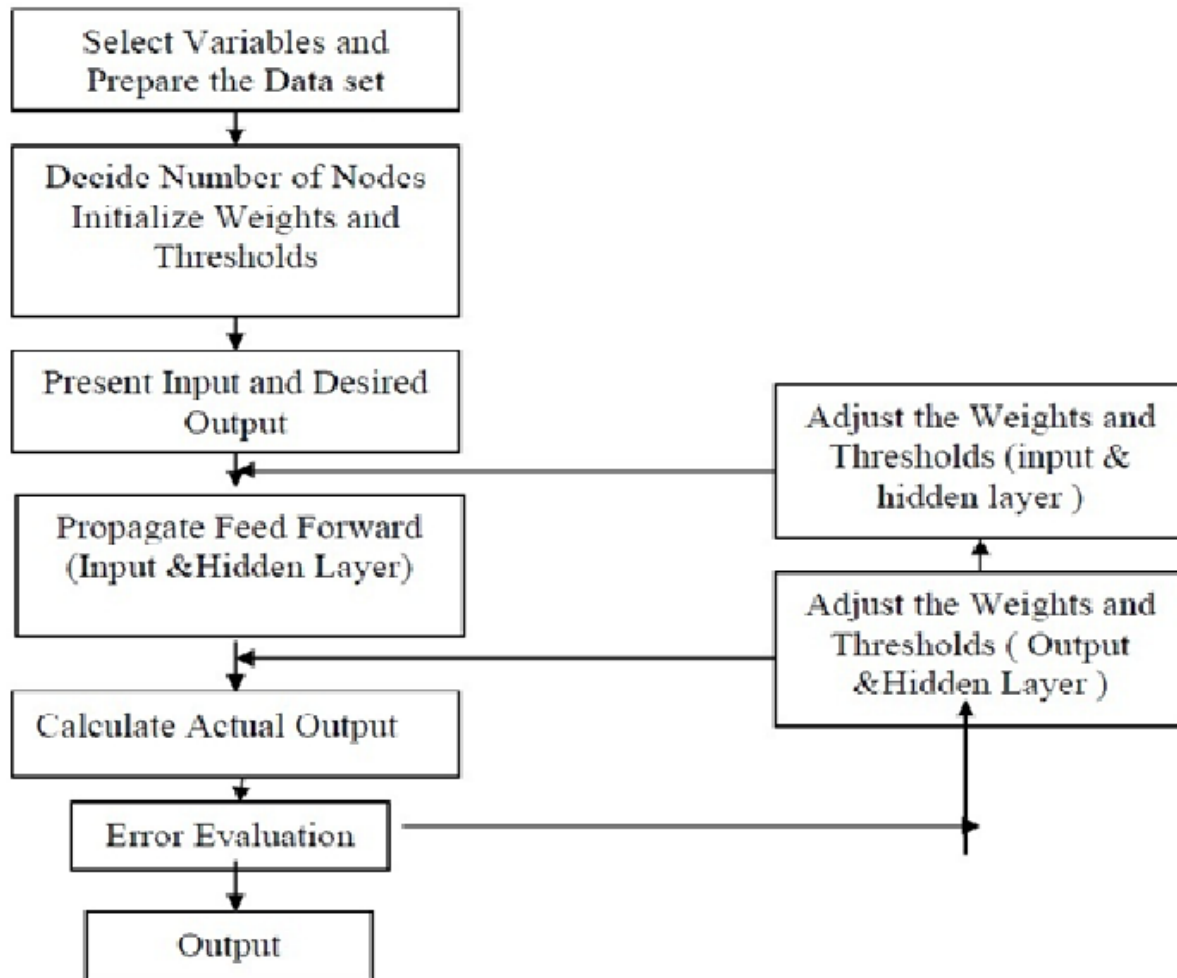Input → Feature extraction + Classification → Output (Car / Not Car)

A single neuron is known as a perceptron. It consists of a layer of inputs (corresponds to columns of a data frame). Each input has a weight which controls the magnitude of an input. The summation of the products of these input values and weights is fed to the activation function. Activation functions are really important for a Artificial Neural Network to learn and make sense of something really complicated and Non-linear complex functional mappings between the inputs and response variable.

They introduce non-linear properties to our Network. Their main purpose is to convert a input signal of a node in a A-NN to an output signal. That output signal now is used as a input in the next layer in the stack. Specifically, in A-NN we do the sum of products of inputs(X) and their corresponding Weights(W) and apply a Activation function f(x) to it to get the output of that layer and feed it as an input to the next layer.
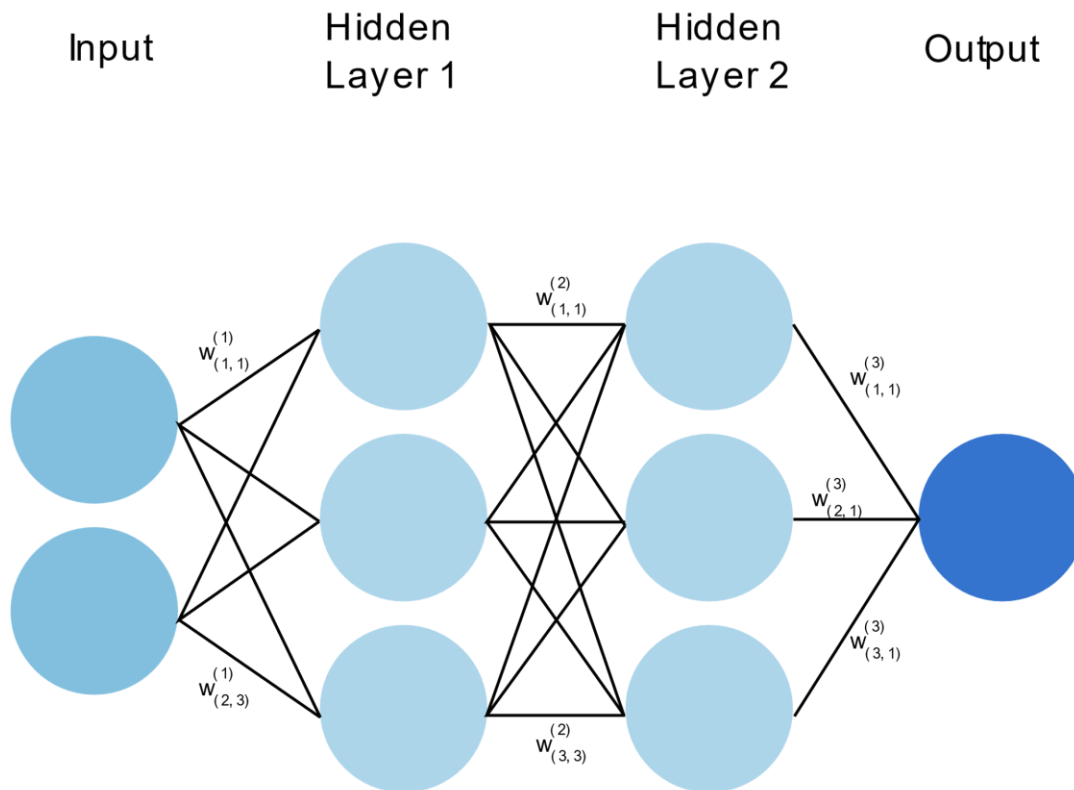
Inputs

Weights

x_1

x_2

x_3

x_n

w_1

w_2

w_3

w_n

Sum

Σ

z

Activation function

σ

a
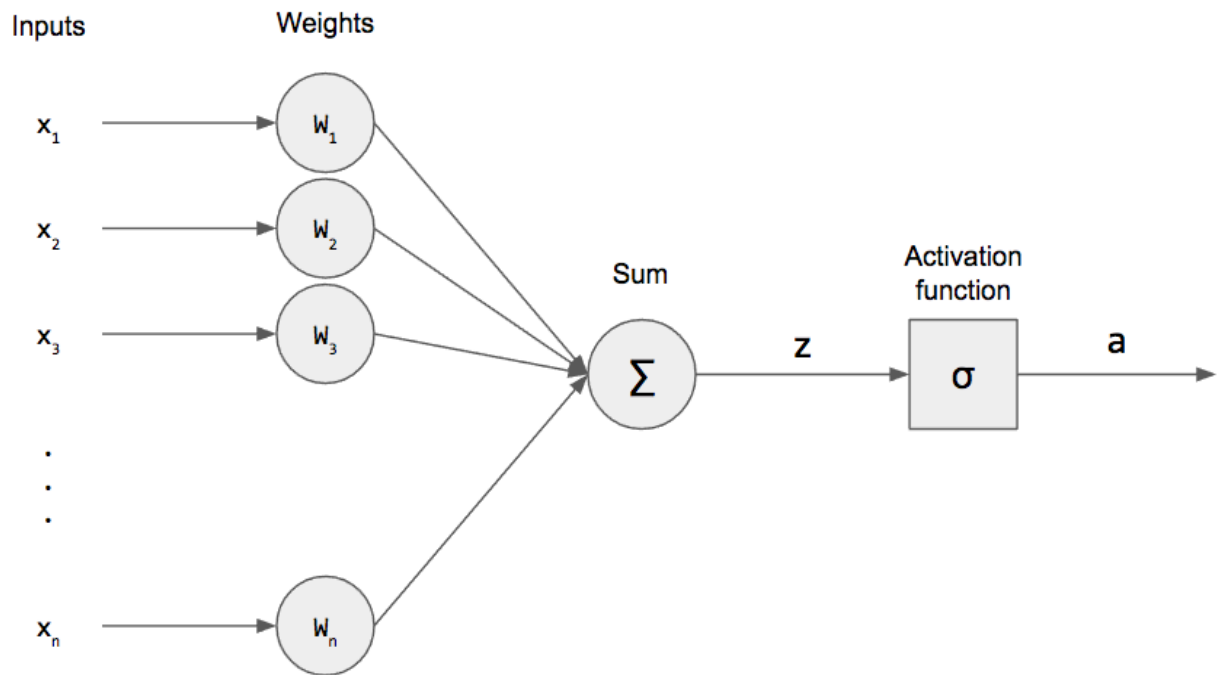
Concept of backpropagation - Backpropagation, short for "backward propagation of errors," is an algorithm for supervised learning of artificial neural networks using gradient descent. Given an artificial neural network and an error function, the method calculates the gradient of the error function with respect to the neural network's weights. It is a generalization of the delta rule for perceptron to multilayer feedforward neural networks.

```
┌─────────────────────────┐
│  Select Variables and   │
│   Prepare the Data set  │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│ Decide Number of Nodes  │
│  Initialize Weights and │
│        Thresholds       │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│ Present Input and Desired│         ┌──────────────────────────┐
│         Output          │         │  Adjust the Weights and  │
└─────────────────────────┘         │   Thresholds (input &    │
            │                       │      hidden layer )      │
            ▼                       └──────────────────────────┘
┌─────────────────────────┐                      ▲
│  Propagate Feed Forward │         ┌──────────────────────────┐
│  (Input &Hidden Layer)  │◄────────│  Adjust the Weights and  │
└─────────────────────────┘         │   Thresholds ( Output    │
            │                       │     &Hidden Layer )      │
            ▼                       └──────────────────────────┘
┌─────────────────────────┐                      ▲
│  Calculate Actual Output│                      │
└─────────────────────────┘                      │
            │                                     │
            ▼                                     │
┌─────────────────────────┐                      │
│     Error Evaluation    │──────────────────────┘
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│         Output          │
└─────────────────────────┘
```

With the democratization of deep learning and the introduction of open source tools like Tensor Flow or Keras, you can nowadays train a convolutional neural network to classify images of dogs and cats with little knowledge about Python. Unfortunately, these tools tend to abstract the hard part away from us, and we are then tempted to skip the understanding of the inner mechanics. In particular, not understanding backpropagation, the bread and butter of deep learning, would most probably lead you to badly design your networks. In a Medium article, Andrej Karpathy, now director of AI at Tesla, listed few reasons why you should understand backpropagation. Problems such as vanishing and exploding gradients, or dying relus are some of them. Backpropagation is not a very complicated algorithm, and with some knowledge about calculus especially the chain rules, it can be understood pretty quick.

Neural networks, like any other supervised learning algorithms, learn to map an input to an output based on some provided examples of (input, output) pairs, called the training set. In particular, neural networks perform this mapping by processing the input through a set of transformations. A neural network is composed of several layers, and each of these layers are made of units (also called neurons) as illustrated below:

In the picture above, the input is transformed first through the hidden layer 1, then the second one and finally an output is predicted. Each transformation is controlled by a set of weights (and biases). During training, to indeed learn something, the network needs to adjust these weights to minimize the error (also called the loss function) between the expected outputs and the ones it maps from the given inputs. Using gradient descent as an optimization algorithm, the weights are updated at each iteration as: $\boldsymbol{w(n+1) = w(n) - \frac{\epsilon \partial L}{\partial w}}$ where L is the loss function and $\epsilon\epsilon$ is the learning rate.

As we can see above, the gradient of the loss with respect to the weight is subtracted from the weight at each iteration. This is the so called gradient descent. The gradient can be interpreted as a measure of the contribution of the weight to the loss. Therefore, the larger is this gradient (in absolute value), the more the weight is updated during an iteration of gradient descent.
The minimization of the loss function task ends up being related to the evaluation of the gradients described above. We will review 3 proposals to perform this evaluation:

- Analytical calculation of the gradients.
- Approximation of the gradients as being: $\frac{\partial L}{\partial w} = \frac{L(w+\delta w) - L(w)}{\delta w}$, where $\delta w \to 0$.
- Backpropagation or reverse mode auto diff.

Let's discuss a little bit about how the input is transformed to produce the hidden layer representation. In neural network, a layer is obtained by performing two operations on the previous layer:

- First the previous layer is transformed via a linear operation: the value of the previous layer is multiplied by a weight, and a bias is added to it. It gives: $z = xw + b$, where x is the value of the previous layer unit, w and b are respectively the weight and the bias discussed above.
- Second, the previous linear operation is used as an input to the activation function of the unit. This activation is generally chosen to introduce non linearity in order to solve complex tasks. Here we will simply consider that this activation function is a sigmoid function: $\sigma(x) = \frac{1}{1 + \exp(-x)}$ As a consequence the value y of a layer can be written as: $y = \sigma(z) = \sigma(xw + b)$, with x, w and b defined as above.

Gradient Descent

Gradient descent is an optimization algorithm used to find the values of parameters (coefficients) of a function (f) that minimizes a cost function (cost).

Gradient descent is best used when the parameters cannot be calculated analytically (e.g. using linear algebra) and must be searched for by an optimization algorithm.

Gradient descent is a first-order iterative optimization algorithm for finding the minimum of a function. To find a local minimum of a function using gradient descent, one takes steps proportional to the negative of the gradient (or approximate gradient) of the function at the current point. To explain Gradient Descent, I'll use the classic mountaineering example. Suppose you are at the top of a mountain, and you must reach a lake which is at the lowest point of the mountain (a.k.a valley). A twist is that you are blind folded, and you have zero visibility to see where you are headed. So, what approach will you take to reach the lake? The best way is to check the ground near you and observe where the land tends to descend. This will give an idea in what direction you should take your first step. If you follow the descending path, it is very likely you would reach the lake.

K-Nearest Neighbours (kNN)

The basic concept of kNN is the class of an object is the class of the majority of its k nearest neighbours . This algorithm is very sensitive to distance definition. There are several commonly used variants of distance for kNN: Euclidean distance; Minkovsky distance; and distances calculated after some transformation of the input space.

In this study, we used three distances: the Euclidean distance, the Fisher's transformed distance and the adaptive distance. Moreover, we used a weighted voting procedure with weighting of neighbours by one of the standard kernel functions. The kNN algorithm is well known. The adaptive distance transformation algorithm is described in. kNN with Fisher's transformed distance is less known. The following parameters are used: k is the number of nearest neighbours, K is the kernel function, and K_f is the number of neighbours which are used for the distance transformation. To define the risk of drug consumption we have to do the following steps: 1. Find the K_f nearest neighbours of the test point.

2. Calculate the covariance matrix of K_f neighbours and Fisher's discriminant direction.

3. Find the k nearest neighbours of the test point using the distance along Fisher's discriminant direction among the K_f neighbours found earlier.

4. Define the maximal distance from the test point to k neighbours.

5. Calculate the membership for each class as a sum of the points' weights. The weight of a point is the ratio: the value of the kernel function K of distance from this point to the test point divided by the maximal distance defined at step 4.

6. Drug consumption risk is defined as the ratio of the positive class membership to the sum of memberships of all classes.

Algorithm: kNN with Fisher's transformed distance

1: Find the K_f nearest neighbors of the test point.

2: Calculate the covariance matrix of K_f neighbours and Fisher's discriminant direction.

3: Find the k nearest neighbours of the test point using the distance along Fisher's discriminant direction among the K_f neighbours found earlier.

4: Define the maximal distance from the test point to k neighbours.

5: Calculate the membership for each class as a sum of the points' weights. The weight of a point is the ratio: the value of the kernel function K of distance from this point to the test point divided by the maximal distance defined at step 4.

6: Drug consumption risk is defined as the ratio of the positive class membership to the sum of memberships of all classes.

The adaptive distance version implements the same algorithm but uses another transformation on step 2 and another distance on step 3. The Euclidean distance version simply defines K_f = k and omits steps 2 and 3 of algorithm. We tested 1,683 million versions of the kNN models per drug, which differ by

• The number of nearest neighbours, which varies between 1 and 30;

• The set of input features;

• One of the three distances: Euclidean distance, adaptive distance and Fisher's distance;
• The kernel function for adaptive distance transformation;

• The kernel functions for voting.

• Weight of class 'users' is varied between 0.01 and 5.0.

Decision Tree (DT)

The decision tree approach is a classifier that constructs a tree like structure, which can be used to choose between several courses of action. Binary decision trees are used in this study. A decision tree is comprised of nodes and leaves. Each node can have a child node. If a node has no child node, it is called a leaf or a terminal node. Any decision tree contains one root node, which has no parent node. Each non terminal node calculates its own Boolean expression (with the value 'true' or 'false'). According to the result of this calculation, the decision for a given sample would be delegated to the left child node ('true') or to the right child node ('false'). Each leaf (terminal node) has a label which shows how many samples of the training set belong to each class. The probability of each class is estimated as a ratio of the number of samples in this class to the total number of samples in the leaf. There are many methods for developing a decision tree. We use the methods based on information gain, Gini gain, and DKM gain. Let us consider one node and one binary input attribute which can take values 0 or 1. Let us use notation: N is the number of cases in the node, c is the number of categories of the target feature, n_{ij} is the number of i category cases with input attribute value j in the node, the number of i category cases in the node is

$ni. = ni0 + ni1$ the number of cases with the input attribute value j in the node is

$n.j = Pc\ i = 1\ nij$, where $nj = n1j, ..., ncj$ is the vector of frequencies with input attribute value j, and $n = n1., ..., nc.$ is the vector of frequencies with any input attribute value. To form a tree we select the base function for information criterion among

$$Entropy(n, M) = -\Sigma\left(\left(\frac{mi}{M}\right)log2\left(\frac{mi}{M}\right)\right)$$

$$Gini(m, M) = 1 - \Sigma\left(\frac{mi}{M}\right)^2$$

$$DKM(m, M) = 1 - \sum 2\sqrt{m_0 m_1 M^{-2}}$$

where m is the vector of frequencies and M is the sum of the elements of the vector m. The DKM can be applied to a binary target feature only. The value of the criterion is the gain of base function:

$$BG = Base(n, N) - \left(\frac{n0}{N}\right)(Base(n0, n0)) - \left(\frac{n1}{N}\right)(Base(n1, n1))$$

There are several approaches to use real valued inputs in decision trees. A commonly used approach is the binning of real valued attributes before forming the tree. In this study we implemented 'on the fly' binning: the best threshold is searched in each node for each real valued attribute and then this threshold is used to bin these feature in this node. The best threshold depends on the split criteria used (information gain, Gini gain, or DKM gain). Another possibility we employ is the use of Fisher's discriminant to define the best linear combination of the real valued features in each node. Pruning techniques are applied to improve the tree. The specified minimal number of instances in the tree's leaf is used as a criterion to stop node splitting. Each leaf of the tree cannot contain fewer than a specified number of instances. We tested 166 million decision tree models (per drug), which differ by:

• The three split criterion (information gain, Gini gain or DKM gain);

• The use of the real-valued features in the splitting criteria separately or in linear combination by Fisher's discriminant;

• The set of input features;

• The minimal number of instances in the leaf, which varied between 3 and 30.

• Weight of class 'users' is varied between 0.01 and 5.0

Support Vector Machine(SVM):

"Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N—the number of features) that distinctly classifies the data points. To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e. the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

Linear SVM:

We have the training dataset of n points of the form:

$$(X_1, y_1), \ldots\ldots\ldots, (x_n, y_n)$$

Where the y_i are either 1 or -1, each indicating the class to which the point xi is a p-dimensional real vector. We want to find the 'maximum-margin hyperplane' that divides the group of points xi for which $y_i = 1$ from the group of points for which $y_i = -1$, which is defined so that the distance between the hyperplane and the nearest point xi from either group is maximized.

Any hyperplane can be written as the set of points x satisfying

$$w.x - b = 0,$$

Where w is the normal vector to the hyperplane. This is much like Hesse Normal Form, except that w is not necessarily a unit vector. The parameter b/w determine the offset of the hyperplane from the origin along the normal vector w.

Nonlinear Classification:

The original maximum-margin hyperplane algorithm proposed by Vapnik in 1963 constructed a linear classifier. However, in 1992, Bernhard E. Boser, Isabelle M. Guyonand Vladimir N. Vapnik suggested a way to create nonlinear classifiers by applying the kernel trick to maximum-margin hyperplanes. The resulting algorithm is formally similar, except that every dot product is replaced by a nonlinear kernel function. This allows the algorithm to fit the maximum-margin hyperplane in a transformed feature space. The transformation may be nonlinear and the transformed space high dimensional; although the classifier is a hyperplane in the transformed feature space, it may be nonlinear in the original input space.

It is noteworthy that working in a higher-dimensional feature space increases the generalization error of support vector machines, although given enough samples the algorithm still performs well.

Computing The SVM Classifier:

Computing the (soft-margin) SVM classifier amounts to minimizing an expression of the form:

$$\left[\left(\frac{1}{n}\right)\sum_{i=1}^{N}\max\bigl(0, 1 - y_i\,(w.x_i - b)\bigr)\right] + \lambda(w)^2$$

We focus on the soft-margin classifier since, as noted above, choosing a sufficiently small value for λ yields the hard-margin classifier for linearly classifiable input data. The classical approach, which involves reducing to a quadratic programming problem, is detailed below. Then, more recent approaches such as sub-gradient descent and coordinate descent will be discussed.

Results:

The data set contains seven categories of drug uses: 'Never used', 'Used over a decade ago', 'Used in last decade', 'Used in last year', 'Used in last month', and 'Used in last week'. We form four problems based on four dichotomies of these classes (see 'Drug use' Section): the decade-, year-, month-, and week-based user/non-user separations. We identified the relationship between personality profiles (NEO-FFI-R) and drug consumption for the decade-, year-, month-, and week-based classification problems. We evaluated the risk of drug consumption for each individual according to their personality profiles. This evaluation was performed separately for each drug for the decade-based user/non-user separation. We also analyzed interrelations between the individual drug consumption risks for different drugs. Part of the results was presented in the preprint. In addition, in the 'Pleiades of drugs' Section we focus on use of correlation pleiades of drugs. We define three pleiades: heroin pleiade, ecstasy pleiade, and benzodiazepines pleiade with respect to the decade-, year-, month-, and week-based user/non-user separations.

Pleiades of Drugs

Consider correlations between drug usage for the year- and decade-based definitions. It can be seen from that the structure of these correlations for the year- and decade-based definitions of drug users is approximately the same. We found three groups of strongly correlated drugs, each containing several drugs which are pairwise strongly correlated. This means that drug consumption has a 'modular structure'. Let us consider a modular structure for the three modules we found:
• Crack, cocaine, methadone, and heroin;

The idea of merging correlated attributes into 'modules' is popular in biology. These modules are called the 'correlation Pleiades. The concept of correlation Pleiades was introduced in biostatistics in 1931. Correlation Pleiades were used in evolutionary physiology for the identification of the modular structure. Berg presented correlation data from three unspecialized and three specialized pollination species. According to Berg, correlation Pleiades are clusters of correlated traits. This

means that in the standard approach to clustering the pleads do not intersect. The classical clustering methods are referred to as 'hard' or 'crisp' clustering, meaning that each data object is assigned to only one cluster. This restriction is relaxed for fuzzy and probabilistic clustering. Such approaches are useful when the boundaries between clusters are not well separated. In our study, correlation Pleiades can be applied since the drugs can be grouped in clusters with highly correlated use:

• The Heroin Pleiad (heroinPl) includes crack, cocaine, methadone, and heroin;

• The Ecstasy Pleiad (ecstasyPl) includes amphetamines, cannabis, cocaine, ketamine, LSD, magic mushrooms, legal highs, and ecstasy;

• The Benzodiazepines Pleiad (benzoPl) includes methadone, amphetamines, and cocaine.

Discussion

These results are important as they examine the question of the relationship between drug use and personality comprehensively and engage the challenge of untangling 44 correlated personality traits (the FFM, impulsivity, and sensation seeking), and clusters of substance misuse (the correlation pleiades). The work acknowledged the breadth of a common behaviour which may be transient and leave no impact, or may significantly harm an individual. We examined drug use behaviour comprehensively in terms of the many kinds of substances that may be used (from the legal and anodyne, to the deeply harmful), as well as the possibility of behavioural over-claiming. We built into our study the wide temporality of the behaviour indicative of the chronicity of behaviour and trends and fashions (e.g. the greater use of LSD in the 1960s and 1970s, the rise of ecstasy in the 1980s, some persons being one-off experimenters with recreational drugs, and others using recreational substances on a daily basis). We defined substance use in terms of behaviour rather than legality, as legislation in the field is variable. Our data were gathered before ' legal highs' emerged as a health concern [96] so we did not differentiate, for example, synthetic cannabinoids and cathinone-based stimulants; these substances have been since widely made illegal. We were nevertheless able to accurately classify users of these substances (reciprocally, our data were gathered before cannabis decriminalisation in parts of North America, but again, we were able to accurately classify cannabis users). We included control participants who had never used these substances, those who had used them in the distant past, up to and including persons who had used the drug in the past day, avoiding the procrustean data-gathering and classifying methods which may occlude an accurate picture of drug use behaviour and risk [97]. Such rich data and the

complex methods used for analysis necessitated a large and substantial sample. The study was atheoretical regarding the morality of the behaviour, and did not medicalise or pathologise participants, optimising engagement by persons with heterogeneous drug-use histories. Our study used a rigorous range of data-mining methods beyond those typically used in studies examining the association of drug use and personality in the psychological and psychiatric literature, revealing that decision tree methods were most commonly effective for classifying drug users. We found that high N, low A, and low C are the most common personality correlates of drug use, these traits being sometimes seen in combination as an indication of higher-order stability and behavioural conformity, and, inverted, are associated with externalisation of distress [98–100]. Low stability is also a marker of negative urgency [47] whereby persons act rashly when distressed. Our research points to the importance of individuals acquiring emotional self-management skills anteceding distress as a means to reduce self-medicating drug-using behaviour, and the risk to health that injudicious or chronic drug use may cause.