UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación

TRABAJO FIN DE GRADO

BIG DATA ANALYTICS TO ASSESS PERSONALITY BASED ON VOICE ANALYSIS

Rodrigo Morales Ramírez
Tutor: Doroteo Torre Toledano

ENERO 2021

BIG DATA ANALYTICS TO ASSESS PERSONALITY BASED ON VOICE ANALYSIS

ANALÍTICA BIG DATA PARA ATRIBUCIÓN DE PERSONALIDAD A PARTIR DE ANÁLISIS DE VOZ

AUTHOR / AUTOR: Rodrigo Morales Ramírez
TUTOR: Doroteo Torre Toledano

< audias >

Audio, Data Intelligence and Speech

AUDIAS-UAM
Dpto. Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Enero / January 2021

Abstract

When humans speak, the produced series of acoustic signs do not encode only the linguistic message they wish to communicate, but also several other types of information about themselves and their states that show glimpses of their personalities and can be apprehended by judgers. As there is nowadays a trend to film job candidate's interviews, the aim of this Thesis is to explore possible correlations between speech features extracted from interviews and personality characteristics established by experts, and to try to predict in a candidate the Big Five personality traits: Conscientiousness, Agreeableness, Neuroticism, Openness to Experience and Extraversion. The features were extracted from a genuine database of 44 women video recordings acquired in 2020, and 78 in 2019 and before from a previous study.

Even though many significant correlations were found for each years' dataset, lots of them were proven to be inconsistent through both studies. Only extraversion, and openness in a more limited way, showed a good number of clear correlations. Essentially, extraversion has been found to be related to the variation in the slope of the pitch (usually at the end of sentences), which indicates that a more "singing" voice could be associated with a higher score. In addition, spectral entropy and roll-off measurements have also been found to indicate that larger changes in the spectrum (which may also be related to more "singing" voices) could be associated with greater extraversion too.

Regarding predictive modelling algorithms, aimed to estimate personality traits from the speech features obtained for the study, results were observed to be very limited in terms of accuracy and RMSE, and also through scatter plots for regression models and confusion matrixes for classification evaluation. Nevertheless, various results encourage to believe that there are some predicting capabilities, and extraversion and openness also ended up being the most predictable personality traits. Better outcomes were achieved when predictions were performed based on one specific feature instead of all of them or a reduced group, as it was the case for openness when estimated through linear and logistic regression based on time over 90% of the variation range of the deltas from the entropy of the spectrum module. Extraversion too, as it correlates well with features relating variation in F0 decreasing slope and variations in the spectrum. For the predictions, several machine learning algorithms have been used, such as linear regression, logistic regression and random forests.

Keywords

Personality assessment, voice analysis, personality estimation, big data, correlation analysis, pattern recognition, machine learning, psychology

Resumen

Cuando las personas hablan, la serie de señales acústicas que producen no codifican solo el mensaje lingüístico que quieren comunicar, sino también otros tipos de información sobre ellos mismos y sus estados, que descubren algo relacionado con sus personalidades y que puede ser percibido por los receptores. Hoy en día, se tiende a grabar las entrevistas de los candidatos a un puesto de trabajo, y el objetivo de este Trabajo de Fin de Grado fue investigar las posibles correlaciones entre rasgos del habla extraídos de las entrevistas y características de personalidad establecidas por expertos, e intentar predecir en una candidata los Cinco Grandes rasgos de personalidad: Conciencia, Amabilidad, Neuroticismo, Apertura a la Experiencia y Extraversión. Las características fueron extraídas de una base de datos de 44 grabaciones de video de mujeres adquiridas en 2020, y 78 de un estudio previo en 2019 y años anteriores.

Aunque se encontraron muchas correlaciones importantes para cada conjunto de datos por separado, se ha demostrado que muchas de ellas eran inconsistentes en ambos estudios. Solo la extraversión, y la apertura de manera más limitada, mostraron un buen número de correlaciones claras. En especial, se ha encontrado que la extraversión está relacionada con la variación de la pendiente del tono (que suele ocurrir al final de las frases), lo que indica que una voz más "cantarina" podría estar asociada con una extraversión más alta. Además, se ha podido comprobar que las mediciones de entropía espectral y de roll-off también señalan que los cambios más grandes en el espectro (que también pueden estar asociados con más voces "cantarinas") también pueden estar asociados con una mayor extraversión.

En cuanto a los algoritmos de modelos predictivos, que tratan de estimar los rasgos de personalidad a partir de las características del habla obtenidas para el estudio, se observó que los resultados eran muy limitados en cuanto a la exactitud y la RMSE, al igual que mediante gráficos de dispersión para los modelos de regresión y matrices de confusión para la evaluación de la clasificación. No obstante, hay resultados que animan a creer que existen algunas capacidades de predicción, y la extraversión y la apertura también acabaron siendo los rasgos de personalidad más predecibles. En concreto, se obtuvieron mejores resultados cuando las predicciones se realizaron respecto a una característica específica en lugar de todas o de un grupo reducido. Por ejemplo, la apertura se estimó mejor mediante regresión lineal y logística respecto al tiempo sobre el 90% del rango de variación de las de la entropía del módulo espectral. La extraversión también se correlaciona bien con características de la variación en la pendiente decreciente de la frecuencia fundamental y las variaciones en el espectro. Para las predicciones se han utilizado algoritmos de aprendizaje automático como la regresión lineal, la regresión logística y random forests.

Palabras clave

Evaluación de la personalidad, análisis de voz, estimación de la personalidad, big data, análisis de correlaciones, reconocimiento de patrones, aprendizaje automático, psicología

Agradecimientos

Cualquier excusa es siempre buena para agradecer y devolver el cariño que recibimos, y llegar hasta aquí no ha sido nada fácil. Me siento muy orgulloso de decir que en absoluto lo he conseguido solo, y es que a lo largo de este camino me han acompañado personas increíbles que siempre han estado ahí y a las que le debo todo.

En primer lugar, querría agradecer a Doroteo la oportunidad de participar en este proyecto y tu confianza en mí. Tu ayuda y dedicación han sido cruciales para que haya podido salir adelante y ha sido un verdadero placer trabajar contigo. No tengo nada más que palabras de agradecimiento por todo tu esfuerzo, y te deseo lo mejor.

A mis compañeros de aventura, también les querría agradecer las horas que han echado en ayudarme y los grandes momentos juntos. Especialmente a mi grupo de amigos que me llevo para siempre: Juan, Javier, Alberto, Jaime, Andrés y Sergio. A Andrés le querría mencionar en especial por la gran experiencia que vivimos juntos, siendo mi compañero de habitación y de aventura durante un Erasmus en el que, en los peores momentos, siempre supo animarme a seguir adelante y a disfrutar de una de las mejores experiencias de mi vida. Gracias, Andrés. También querría destacar en especial a mi compañero de peleas en los laboratorios y en la biblioteca, Sergio, eres una grandísima persona. Gracias por tu insistencia siempre hasta aclarar las cosas y por tu sinceridad. Por otro lado, me gustaría mencionar en especial a Iván, que me ayudó con un último empujón cuando las cosas se ponían feas y ha demostrado también ser un gran amigo, y a Javi, que además fue mi compañero de equipo, y también me ayudó mucho. Además de ellos, hay mucha gente más con la que he disfrutado durante estos años y que ha arrimado el hombro para que juntos lográsemos el objetivo, y les debo mucho. Muchísimas gracias a todos.

También a mis amigos de toda la vida, empezando por Villalovers, mis compañeros del colegio, del Erasmus, mis compañeros y entrenadores de equipos de baloncesto, del trabajo, campamentos… ojalá pudiera dedicaros unas palabras a cada uno, no estaría aquí sin vosotros. Querría destacar de todas formas a mis grandes amigos Pablo y en especial David, mi mejor amigo desde la infancia y que me ha acompañado en todas las etapas de mi vida. Has sido una referencia y, junto con Pablo, es un enorme privilegio poder contar con vosotros desde hace tantos años.

A mi familia, en especial a Gustavo por haber sido el mejor hermano que se pueda pedir. Todo lo que hemos vivido juntos me hace sentir muy afortunado de tenerte. Siempre has estado ahí y no creo que pueda devolverte nunca todo lo que me has dado. Estoy tan orgulloso de ti como agradecido, y te aseguro que es mucho decir ambas cosas. A mis padres, Antonio y Carmela por haberme brindado todas las oportunidades que me han traído hasta aquí y que han hecho de mí un hombre, del que espero que podáis sentiros orgullosos. También a todo el resto de mi familia, tanto por parte Morales como Ramírez, que habéis sido un apoyo fundamental y una gran fuente de inspiración. Mis tías, tíos, primas, primos y abuela, gracias por tanto. También a María, que me cuidó desde que apenas sabía hacer nada hasta ahora, que sigues velando por mí y eres parte de mi familia. Gracias por todo lo que has aguantado y hecho por mí. A toda la familia de mi novia, por haberme recibido, acogido y querido como uno más durante todos estos años, me habéis ayudado muchísimo y siempre os estaré agradecido.

También querría agradecer y recordar a los que hoy no podrán acompañarme. Mis abuelos Francisco y Mercedes, mi abuelo Antonio y mi abuela adoptiva Carmen. Siempre ha sido una gran motivación para mí haceros orgullosos allá donde estéis y hoy siento mucho vuestro cariño. Os tengo presentes en cada logro y paso de mi vida.

Por último a mi mayor apoyo, referencia e inspiración, mi compañera de viaje María. Gracias por iluminarme cuando no sabía qué hacer, apoyarme cuando no quería seguir y levantarme una y otra vez. Durante todos estos años hemos crecido juntos y siempre me has animado y empujado a soñar. Ahora sueño despierto teniendo una compañía tan bonita como la tuya y compartiendo éxitos contigo. Eres lo mejor que tengo, te lo debo todo. Gracias, María.

Como le decían a Rocky Balboa, el corazón es lo último que envejece y siempre os llevaré en él. Gracias, de corazón.

TABLE OF CONTENTS

FIGURE INDEX

TABLE INDEX

1 Introduction

1.1 Motivations

When humans speak, the produced series of acoustic signs do not encode only the linguistic message they wish to communicate, but also several other types of information about themselves and their states [1]. For centuries, people have been making estimations about others' personalities, even of people they did not know at all, trying to guess their inner characteristics based on a first impression, clothing or their way of moving or speaking. Surprisingly or not, studies have shown that the accuracy of these predictions has remarkable coincidence between personality ratings given to others and their own self-rating, especially from close acquaintances. Moreover, strangers' judgments agreed with each other and with subjects' self-judgments beyond a chance level [2]. Meta-analysis found an overall accuracy of the predictions of .39, even with zero-acquaintance [3].

A clear conclusion would be to consider that people arouse something related to their personalities that can be apprehended by judgers [4]. Different research programs have been looking for the keys that might be behind this phenomenon, some of them focusing on linguistic cues. Nowadays, the generalization of big data analysis, the computing power's increase and the development of a wide range of accessible technologies have created a great opportunity to promote automatic personality assessment systems. Furthermore, artificial personalities related to the robotics development have brought more interest to the identification of vocal expressive characteristics [5].

As there is nowadays a trend to film candidate's interviews, the motivation for this Bachelor Thesis is to find those clues connecting personality and speech by continuing with previous research studies and looking for possible correlations between speech features and personality characteristics with the aim of extracting information about candidate's personality from the recordings.

1.2 Objectives

Assuming that speech reveals our inner characteristics, as some personality theorists advocated in the past [6] [7], the main objective of this Bachelor Thesis will be mainly to analyse whether there is a relationship between the audio characteristics extracted from the interviews and the personality characteristics established by the experts. For this, a genuine database of 44 women video recordings acquired in 2020, and 78 in 2019 and before from a previous study [8], will be explored using current voice analysis technology and assessing personality variables.

Also, a second objective would be checking whether it is possible to use information extracted from the audio to estimate the Big Five Personality Traits: conscientiousness, agreeableness, neuroticism, openness to experience and extraversion; by using machine learning algorithms such as linear regression, logistic regression or random forests.

1.3 Realization phases

This Bachelor Thesis started in June 2020, when Doroteo assigned me the project after an interview carried out in May. My first step was to spend the summer studying Python for Data Science, and I did several courses in order to be prepared for the posterior development of this project.

In September, I began to study the State of Art of psychology and speech, and read different pieces of research related to this Thesis that could help focus on several speech features or personality traits, as well as specific technology used in this field.

During November, I worked on the scripts to read personality ratings and match them with each feature, looking for possible correlations for each year and exploring if these were consistent throughout both studies.

In December, I developed the machine learning algorithms to try to predict personality from the audio features, which showed to be the most correlated.

Finally, in January I obtained the results, studied them and redacted the rest of the Memory.

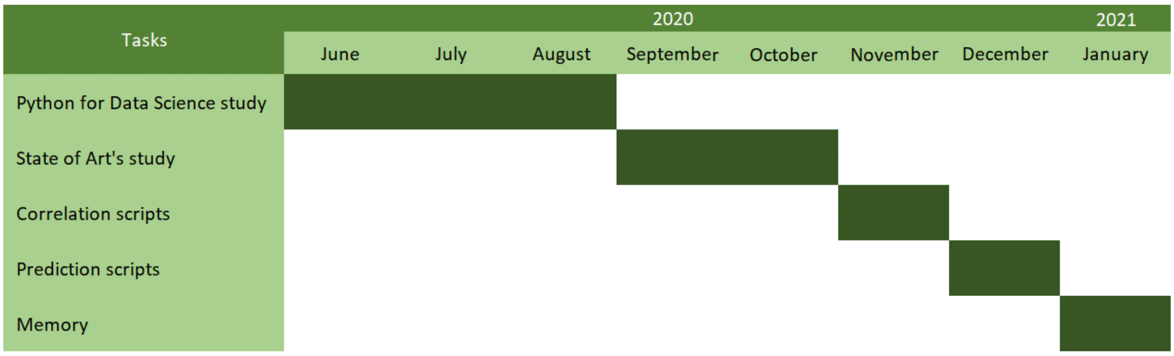

Figure 1: Gantt showing the different phases of the project

1.4 Document's structure

The present Memory consists of the following sections:

- **State of the Art:** Where a theorical background of both personality and speech will be presented, as well as the technologies used in the development of this project.
- **Recordings and audio features extraction:** This section will provide information about how data and audio features were obtained in both years.
- **Audio and personality correlations:** The results of the correlations search will be presented in this section by showing the top 3 correlations per personality trait of each year and the consistent ones through both datasets.

- **Personality predictions:** Estimations' results will be explained in this section according to three different approaches: Based on specific features, on all features, and on groups of features.

- **Conclusions and future work:** From the results obtained, conclusions extracted from this project will be detailed and analysed, as well as different alternatives that could be of interest for future work on this field.

- **Correlation's code:** Code used to calculate correlations is annexed.

- **Prediction's code:** Code used to try to estimate personality is also annexed.

2 State of the art

2.1 Introduction

In this section, the previous work and studies will be discussed and analysed, as well as the technologies used during this Bachelor Thesis, with the aim of giving a general background to the project and explaining the reasons behind the use of different tools and methodologies of work.

To begin with, the first analyses will be discussed, followed by new pieces of research based on the current technology innovations. This will explain the most relevant concepts and approaches that have been suggested in this topic. From linguistic to psychological and technological advances, today's investigation relies on them to reach further conclusions. Sections 2.3 to 2.5 of this section are based on the previous study [8]. However, this Thesis has extended the state of the art included in that study by incorporating a significant number of additional works and also by explaining in more detail the works mentioned in [8].

2.2 Personality and the Big Five

Personality is understood as the combination of psychological characteristics that make people different from each other, and this has brought interest within the field of psychology for decades [9]. The taxonomy of personality was firstly analysed by McDougall in 1932 when he separated it into five distinguishable factors: intellect, character, temperament, disposition and temper [10]. About ten years later, a more complex approach was made by Cattell, who proposed 16 primary factors and 8 second-order factors [11] [12] [13] [14]. However, researchers found the 5-factor model accounted for the data better, and they could not replicate Cattell's work successfully [15] [16]. For instance, Tupes and Christal found good support for five factors as they were reanalysing Cattell's correlations: surgency, emotional stability, agreeableness, dependability, and culture [17]. These, as well as McDougall's, are amongst the most accepted ones nowadays.

Tupes and Christal's five factor model was well supported by several subsequent studies [18] [19]. In 1964 Borgatta also found five stable factors across five methods of data gathering [20], while Norman's labels (extraversion, emotional stability, agreeableness, conscientiousness, and culture) are especially noteworthy as they are referred as "Norman's Big Five" or simply as the "Big Five" commonly in literature [21]. They are based on the hypothesis that language encodes the individual differences that are more relevant in society.

A conceptually similar model of personality would be the Five Factor Model described by Costa and McCrae in 1985 which maps personality traits onto five general factors: openness to experience, conscientiousness, extraversion, agreeableness, and neuroticism [22]. These were shown to be highly stable throughout life, especially over the age of thirty. Ten years later, they also concluded that there are some consistent sex differences, which is why it is going to be used for the present paper in which only women recordings will be studied [23]. The Five Factor Model is also known as "Big Five", as Norman's, and it is how it will be referred in this project.

Costa and McCrae also detailed each of their five factors. Openness to experience is described as characterized by attributes including independence of judgment, active imagination and preference for variety. High conscientiousness entails a sense of purposefulness and responsibility, and that this person is often very trustworthy. People with a high score on extraversion are usually sociable and assertive. The agreeableness scale will include at its top those who are trusting, accepting, and easily moved. And finally, neuroticism is considered to be the opposite of emotional stability. Low self-esteem, pessimism, and guilt are more intense in those who score high in this factor.

Neuroticism	Anxiety, angry, hostility, depression, self-consciousness, impulsiveness, vulnerability
Extraversion	Warmth, gregariousness, assertiveness, activity, excitement-seeking, positive emotions
Openness	Fantasy, aesthetics, feelings, actions, ideas, values
Agreeableness	Trust, straightforwardness, altruism, compliance, modesty, tendermindedness
Conscientiousness	Competence, order, dutifulness, achievement striving, self-discipline, deliberation

Table 1: Trait facets associated with the five domains of the Costa and McCrae five factor model of personality (Costa & McCrae, 1991)

2.3 Personality and speech

In the early 1930s, Gordon Allport, a psychologist considered to be one of the most representative founders of personality psychology, already thought that expressive behaviours were indicators of personality and noted that individuals' personality can be expressed by gestures, style of clothing, speech, posture and gait [6] [7]. Hadley Cantril and Allport were amongst the first scholars to launch a systematic analysis of the relationships between personality and speech, and both conducted ten experiments to determine if the voice could be a valid marker for personality. They concluded that judgers mostly succeeded in matching their ratings for twelve different personality features and their corresponding voice recordings, and that personality attributions were more accurate when based on a summary of features instead of single characteristics of speech.

Since then, the relation between speech and personality have been explored in different studies, which started off with inconclusive results. Concerning prosody, Smith et al. showed in 1975 that competence (conscientiousness) could be positively correlated to speech rate, which also has an inverted-U relationship with benevolence (agreeableness), suggesting a need for non-linear models [24]. Moreover, Klaus Sherer suggested in 1978 [25] an adaption of Brunswick's lens model from 1956, [26] a way of viewing the decisions people make and how they do it, in which he structured the different components involved in personality judgements from voice and speech. This approach considers that personality dispositions are expressed by means of objectively measured speech variables which act as "distal cues". These are perceived by the listener and represented as "proximal cues" and are key to the observers' attributions of personality. Scherer analysed four vocal aspects of the "externalization of language", including frequency, intensity and quality; fluency aspects of speech style, such as pauses and discontinuities; morphological and syntactical aspects; and conversational behaviour. Although he reviewed previous studies and felt sceptical to the results obtained in them, he found that an extraverted personality was often related to competence and dominance, associated with fundamental frequency and intensity.

Apple et al found in 1979 clear evidence that listeners consider pitch and speech rate when making personal attributions to speakers with an experiment in which they tried altering the recordings' pitch and speech rate. They concluded linking high-pitched voices to less truthful, less emphatic, less potent, and more nervous people, while slow-talking speakers were also evaluated as less truthful, but more potent and passive, and less fluent and persuasive [27].

Another review focused on three aspects (accuracy, externalization and attribution) and was carried out by Brown and Bradshaw in 1985 [28]. Accuracy, understood as the extent to which judges can identify personality characteristics from speakers' voice; Externalization, regarding the relationships between voice parameters such as frequency, intensity, etc. and personality characteristics; and Attribution, indicating how variations in speech features affect listeners' personality attributions. About the first aspect, authors realised that conclusive results could not be obtained from the subjective measures of personality used in the studies, as there were disagreements when judging objectively measurable characteristics such as age, sex or social class. In the case of externalization, Brown and Bradshaw pointed out the problems attached not only to measuring personality but also speech so clear conclusions can be drawn. Compared to this, the study of attribution has proven more productive and conclusive, as speech rate, pauses and temporal patterns have been useful to personality attributions.

These studies were reviewed in 1990 by Furham [29], who analysed speech variables and personality traits, particularly extraversion, which he thought to be related to voice intensity and quality. However, he also reflected on the lack of a theory-driven approach to the topic as he criticized methodological aspects of many of the studies.

2.4 New pieces of research

Nevertheless, the technology used in speech analysis has taken a huge leap, and universally accepted theories have supported instruments for personality measurement since. This has led to the development of several pieces of research afterwards. For example, extraversion has been found to be related to prosodic characteristics such as pitch and variation of frequency [30], suggesting also that ranking models are more accurate than multi-class classifiers for modelling personality. Moreover, the speech frequency and spectrum of different phonemes in Chinese (/sh/ and /i/) have shown significant correlation with fields of study like psychoticism, extraversion and neuroticism after being analysed by Praat voice software [31].

However, the artificial conditions in which language is produced in these studies is a clear limitation. For instance, Gocsál (2009) [1] based his research on spontaneous speech in order to analyse female listeners' personality judgements about male speakers, and found a correlation between temporal parameters (faster speech) and inferred openness and extraversion. Nevertheless, fundamental frequency parameters did not correlate with the inferred personality dimensions, in contrast with previous findings. In the perception of spontaneous speech, according to the author, other parameters might overshadow frequency, which shows its role clearer in isolated speech sounds or text readings.

2.5 Voice analysis for paralinguistic information extraction

From the technological perspective, there is currently a trend to go beyond text in speech technology [32], as voice analysis has shown it is capable of extracting both the linguistic message and different non-related content. Some studies have even shown that linguistic modelling is clearly outperformed by the acoustic modelling one [33] and that emotion assessments made by human judges can be automatically inferred from prosodic features extracted directly from the speech signal with an accuracy ranging from 65% to 80% [34].

For example, a speaker can be accurately identified by Automatic Speaker Recognition, a technology that has been developed since at least 1996 [35] [36] [37] and uses short-term spectral features known as Mel-Frequency Cepstral Coefficients (MFCC), which basically summarize the energy distribution of the audio in 20-40 ms. intervals in a psychoacoustically derived frequency scale (Mel scale). This technology is regularly evaluated in international competitive evaluations, called Speaker Recognition Evaluations (SRE) and organized by the US National Institute of Standards and Technology (NIST), and the results are then disseminated in conferences. The huge research effort involved is evident in the results of Sadjadi et al. (2017) [38], an evaluation in which 66 teams from 43 countries participated to submit 121 valid system outputs that produced scores. The difficulty of these evaluations has increased over the time, as the introduction of non-English conversational telephone speech data made SRE16 even more challenging due to domain/channel and language mismatches.

The most obvious individual physical trait that can be obtained from the voice of an adult may be the gender [39], as it is time invariant, phoneme independent, and speaker independent for a given gender and it is related to the average pitch except for children. In fact, gender detection has even been excluded from NIST Speaker Recognition Evaluations because of its little difficulty. Other physical characteristics that can be extracted are the height, as taller people tend to have a longer vocal tract, with an impact on the voice; and age, which also affects the vocal tract. Several studies have achieved accurate estimations with a mean absolute error of 5.02 cm in height and 6 years in age [40].

Another clear impact on the voice production system can be caused by medical conditions. There are conditions, such as polyps in the vocal cords, that alter the normal physiology of the human speech production system and can be traced by using phonation features such as jitter, shimmer and Harmonic-to-Noise Ratio (HNR), and biomechanical parameters of the vocal cords estimated from the voice. Normophonic samples have shown small unbalance indices, as opposed to pathologic ones, even though there could not be found a specific pattern of unbalance related to a given pathology [41]. Although results are still inconclusive, Sleep Apnea pathology may influence the voice of a patient, as it is sometimes associated to particular configurations of parts of the vocal tract [40].

As a matter of fact, many conditions affect the central nervous system and their trace on speech production can be detectable. Many non-automated diagnostic procedures involve the patient speaking in several tasks for Dementia, Alzheimer’s disease and Parkinson’s disease, as these have a prompt effect on voice. Detecting them in early stages can help modify their evolution through treatment. For instance, Skodda and Schlegel discovered in 2008 that Parkinson modifies speech rhythm with an augmented articulation rate and fewer pauses. Rusz et al. also found differences in articulation, phonation and prosody three years later [42]. Moreover, Satt et al., between 2013 and 2014, based the detection of very early Dementia on using duration features and measures of regularity in the durations, and achieved an Equal Error Rate (EER) below 20% using a set of 80 diagnosed individuals [43] [44]. In the case of Alzheimer's disease, while two of the most common signs are memory and cognitive impairment, literature suggests that language impairment is also a common sign that can be employed to support diagnosis and assessment of the disease's severity, given that speech and language production can provide information about the cognitive status of a person and other aspects related to brain damage [45]. In that line, Weiner et al. (2016) detected early stages of this disease using just ten features, including speech/silence duration and word duration, and achieved an F-score of 0.8 (out of 1) in a sample of 98 diagnosed individuals [46].

There is a common characteristic to these pieces of information extracted from speech, that is the absence of change over time or a very slow pace for it, as it had to do with a person's identity in the first case or their age, height and medical conditions in the second. However, voice analysis could detect internal speakers' states, such as fatigue, sleep deprivation and emotions, that change often and quickly. For example, Baykaner Huckvale et. al conducted an experiment in 2015 for training astronauts, predicting sleep deprivation time in minutes by using features such as MFCC, autocorrelation coefficients (ACC) and energy contours; and studied psychological performance to measure mental fatigue [47]. Their results had an error between 5% and 12% and correlations of .69 and above. Moreover, they showed that voice features and test scores are affected by both the total time spent awake and the time-of-day within each subject's circadian cycle, but they were poor predictors for the test results, while voice features could give good predictions of the psychophysiological test scores and sleep latency.

One of the paralinguistic information that has received more spotlight in the research community is emotion detection. Emotions need to be considered in spoken human-machine communications because of their importance in speech. The first public challenge in speech-based emotion recognition came in 2009 by the hands of Schuller, Steidl, and Batliner [48], and included a common benchmark with 5 emotional states (anger, emphatic, neutral, positive and rest) to be recognized. It suggested using some low-level descriptors (zero crossing rate, energy, pitch, HNR, MFCCs and their variations) and several functionals in order to transform these time-dependent features into a single feature vector per recording (mean, standard deviation, kurtosis, skewness, extremes and relative position, and offset, slope and MSE of a linear regression). In 2010 they proposed a Paralinguistic challenge to detect age, gender and level of interest using an extended set of features [49], and in the following year two new challenges [50]: detecting alcoholic and sleepy speech and the first Audio Visual Emotion Challenge (AVEC), in which audio, video or a combination of both could be used to recognize emotions.

The AVEC Challenge has continued yearly, and recent editions have focused on state-of-mind, detecting depression with AI, bipolar disorder and cross-cultural affect recognition [51][51]. The features proposed for the audio were the union of the features proposed in the two previous evaluations. The audio baseline feature set consisted of 1941 features, composed of 25 energy and spectral related low-level descriptors (LLD) x 42 functionals, 6 voicing related LLD x 32 functionals, 25 delta coefficients of the energy/spectral LLD x23 functionals, 6 delta coefficients of the voicing related LLD x 19 functionals, and 10 voiced/unvoiced durational features. This set is very complete and has been shown to have information about the emotional state of the speaker in previous research. In addition, it is easy to extract using an open-source software known as openSMILE [52]. For these reasons, this will be the set of features used in this study.

2.6 OpenSMILE: Audio Feature Extraction

As it has been mentioned before, openSMILE is the tool used to extract audio characteristics for this work. It is an open-source toolkit (Speech and Music Interpretation by Large-space Extraction) which can also be helpful for classification of speech and music signals. Moreover, it has also contributed as a real-time speech and emotion analysis component to automatic emotion recognition since 2008, when its history began in the scope of the SEMAINE project at the Technical University Munich by the hand of Florian Eyben, Martin Wöllmer and Björn Schuller, and it had the purpose of designing an automated virtual agent with affective and social skills.

This tool consists of a fast and efficient incremental cross-platform processing in real-time, which allows to extract up to 27k features with real-time factors of 0.08 and with high modularity and reusability of components and plugin support. Furthermore, different audio input/output formats are available as well as feature file formats for the results. It counts with great speech-related features as Signal energy, Loudness, Mel-/Bark-/Octave-spectra, MFCC, PLP-CC, Pitch, Voice quality (Jitter, Shimmer), Formants, LPC, Line Spectral Pairs (LSP), and Spectral Shape descriptors; statistical functionals as Means, Extremes, Moments, Segments, Samples, Peaks, Linear and quadratic regression, Percentiles, Durations, Onsets, DCT coefficients, Zero-crossings, and Modulation spectrum; and data processing as Mean-variance normalisation, Range normalisation, Delta-regression coefficients, Vector operations, and Moving average filters. It also counts with signal processing features and music-related features [53].

2.7 Pandas: Data analysis in Python

Pandas is one of the most famous Python libraries as it can be found in a wide variety of academic and commercial domains, and it has been used in this project to read csv and Excel files as well as to utilize *DataFrames* and indexing tools. Its first development started in 2008 at AQR Capital Management and, since 2009, Pandas has been an open-source tool. In 2012 the first edition of *Python for Data Analysis* was published and since 2015 it has become a NumFOCUS sponsored project, in an attempt to ensure their success as a world-class open-source project.

Pandas offers mainly reading and writing data tools for between in-memory data structures, different formats and a Dataframe object for data manipulation with integrated indexing. This Dataframe counts with intelligent data alignment and integrated handling of missing data, flexible reshaping and pivoting of data sets with the possibility of inserting and deleting columns and aggregating or transforming data, intelligent label-based slicing and subsetting of large data sets and much more. All of these functionalities have been used during the development of this project [54].

2.8 Scipy: Mathematics, science and engineering in Python

SciPy is an open-source scientific computing library of numerical routines for the Python programming language that provides fundamental building blocks for modelling and solving scientific problems. SciPy was built in 2001 on top of NumPy, which provides array data structures and related fast numerical routines, to add a collection of mathematical algorithms and convenience functions for manipulating and visualizing data including algorithms for optimization, integration, interpolation, eigenvalue problems, algebraic equations, differential equations and many other classes of problems [55] [56].

2.8.1 Pearson Correlation Coefficient

The Pearson correlation coefficient measures the linear relationship between two datasets. Its values vary between -1 and +1, implying an exact linear relationship, with 0 meaning no correlation. Positive correlations imply that as x increases, so does y, while for negative correlations as x increases, y decreases.

$$r = \frac{\sum(x - m_x)(y - m_y)}{\sqrt{\sum(x - m_x)^2 \sum(y - m_y)^2}}$$

Where m_x is the mean of the vector x and m_y is the mean of the vector y [57].

The p-value for Pearson's correlation coefficient roughly indicates the probability of an uncorrelated system producing datasets that have a Pearson correlation at least as extreme as the one computed from these datasets. This calculation relies on the assumption that each dataset is normally distributed.

$$p = 2 * dist.cdf(-abs(r))$$

Where *dist* is the exact distribution of *r* and *cdf* is Cumulative distribution function [58].

2.9 Scikit-learn: Machine Learning in Python

Scikit-learn is an open-source simple and efficient tool for predictive data analysis that was built on NumPy, SciPy and Matplotlib and integrates a wide range of machine learning algorithms for both supervised and unsupervised medium-scale problems. This package uses a general-purpose high-level language with especial emphasis on ease of use, performance, documentation, and API consistency. This project was started in 2007 as a Google Summer of Code project and was finally released in 2010. Since then, several releases have followed [59].

2.9.1 Linear Regression

Linear regression attempts to model the relationship between a target and one or more predictors by fitting a linear equation to observed data. In this case, the residual sum of squares between the observed targets in the dataset and the targets predicted by the linear approximation is minimized by fitting a linear model with coefficients [60].

2.9.2 Logistic Regression

Logistic regression is a linear model for classification rather than a regression model and is also known in literature as logit regression, maximum-entropy classification or the log-linear classifier. It is a mathematical modelling approach that can be used to describe the relationship of several independent variables to a dichotomous dependent variable. In this model, the probabilities describing the possible outcomes of a single trial are modelled using a logistic function [61] [62].

2.9.3 Random Forest

Random forests are a combination of tree predictors where each tree in the ensemble is built from a sample drawn with replacement from the training set. This algorithm uses the perturb-and-combine technique specifically designed for trees, which consists in a diverse set of classifiers that is created by introducing randomness in the classifier construction. This randomness aims to decrease the variance of the forest estimator as individual decision trees typically exhibit high variance and tend to overfit. The consequent decoupled prediction errors can help cancel out some of them, which, although bias may be increased, often improves the model. In these forests, the prediction of the ensemble is given as the averaged prediction of the individual classifiers. Nevertheless, the scikit-learn implementation does not let each classifier vote for a single class but combines them by averaging their probabilistic prediction [63].

2.10 Summary

In conclusion, personality has been a very common area of investigation, and the technological advances bring us closer to being able to find the connection between personality features and linguistic features. Therefore, previous pieces of research have been studied because of the interdisciplinary approach this Thesis stands on, considering the involvement of the fields of psychology and linguistics.

The background for this project includes the previous deduction of physical traits such as age, sex or height; and the early diagnose through speech characteristics of Sleep Apnea pathology, Dementia, Alzheimer's and Parkinson's disease; as well as the detection of emotion and mental health problems.

Amongst the variety of technological tools, OpenSMILE, Python's Pandas, SciPy and Scikit-learn have been selected as the most useful for this Thesis.

3 Recordings and audio features extraction

3.1 Introduction

In this section, data collection will be explained and detailed. In particular, how the audio was recorded each year and how it was treated to extract the audio features for the study.

3.2 Recordings

3.2.1 Recordings from 2019

The audio was recorded in an office of the Faculty of Psychology by Professor of Psychology Victor Rubio and Prof. David Aguado, with the microphone of a video camera that took the video. For this reason, its quality is not high. In addition, the camera also captured background noise, which was especially noticeable during class changes, when a significant number of students were talking outside the office.

The acquisition protocol included three long responses in which the subject spoke for approximately one minute and many short responses in which the subject simply said a number or a word. Only the long answers were considered, and they were manually segmented and then concatenated to form an audio with several minutes of the speaker's voice. These files were converted to PCM WAV format with 16bits/sample and 16 KHz sampling rate. It is on these files where the feature extraction with OpenSMILE was finally applied.

The number of subjects was one hundred psychology students, including 78 women that have been selected for this study, as it will only focus on the female gender.

3.2.2 Recordings from 2020

This time, data was also captured in the same environment, but a directional video camera microphone (Sennheiser MKE-400) was used pointed at the subject. This microphone includes noise and vibration reduction, so audio obtained in 2020 is of higher quality than the one obtained in 2019. In addition, the acquisition protocol only included several questions in which the subject spoke for a long interval of time (about 1 minute). In this case, the number of subjects was 44 women, and the questions were manually segmented and concatenated to generate a single audio. From this point on, the processing is identical to the previous one.

3.3 Audio features extraction

Extractions were performed using the audio feature extractor open source OpenSMILE, where the audio baseline feature set consisted of 1941 features, composed of 25 energy and spectral related low-level descriptors (LLD) x 42 functionals, 6 voicing related LLD x 32 functionals, 25 delta coefficients of the energy/spectral LLD x23 functionals, 6 delta coefficients of the voicing related LLD x 19 functionals, and 10 voiced/unvoiced durational features. These were used in the AVEC Challenge to recognize emotions.

Features' naming in OpenSMILE is done under a strict scheme that specifies a suffix appended to the field name of the previous level, and these two are separated by a '_'. The following example can form a clear idea of this scheme. Assuming you start with an input field 'pcm' and compute delta regression coefficients afterwards, the result is the name 'pcm_de'. Applying functionals (only the extreme values max and min) will then lead to two new fields: 'pcm_de_max' and 'pcm_de_min'. Although the complete processing chain could be deducted from the field name under the right conditions, this would lead to feature names being rather long and redundant, since most speech and music features are based on framing, windowing and spectral transformation. Therefore, the majority of components copy the input name instead of appending anything to it [64].

Low-level descriptor	**Description**
Loudness	It is a measure of subjective perception of sound pressure
Zero crossing rate	It is the number of times the amplitude crosses the zero value in a given interval
Psychoacoustic sharpness	It is how much a sound’s spectrum is in the high end
Harmonicity	It represents the degree of acoustic periodicity. An HNR of 0 dB means that there is equal energy in the harmonics and in the noise
MFCC 1-10	Mel Frequency Cepstral Coefficients are a representation of the short-term power spectrum of a signal in a psychoacoustic (Mel) scale
Kurtosis	It is a statistical measure that is used to describe the distribution of a signal, it measures extreme values in tail relative to a normal distribution. High kurtosis has heavy tails or outliers, and low kurtosis has light tails or lack of outliers
Skewness	It is a statistical measure that is used to describe symmetry. Skewness near zero means symmetric data
Jitter and Shimmer	Both represent the variations in vibration of the vocal cords. Jitter is the variability in frequency and shimmer is the variability in amplitude
Spectral Flux	It is a measure of how quickly the power spectrum of a signal is changing
Voicing (F0)	The frequency at which the vocal cords vibrate to produce sonorous sounds
PCM	Pulse Code Modulation refers to the digital audio signal as it is, so it groups together the characteristics that are obtained directly from the waveform
Audspec	It is a frequency band conversion. It consists in the reduction of the Fourier frequencies of a signal's power spectrum to a reduced number of frequency bands in an auditory frequency scale.

Table 2: LLD low-level descriptors

Functional	
Statistical functional (23):	Arithmetic mean (1), root quadratic mean (2), standard deviation (3), flatness (4), skewness (5), kurtosis (6), quartiles (7-9), inter-quartile ranges (10-12), 1% percentile (13), 99% percentile (14), percentile range 1%-99% (15), percentage of frames where the contour is above the minimum + 25% (16), 50% (17) and 90% (18) of the range, percentage of frames where contour is rising (19), maximum (20), mean (21), minimum (22) and standard deviation of segment length (23)
Regression functionals (4):	Linear regression slope (1) and corresponding error (2), quadratic regression coefficient (3) and error (4)
Local minima/ maxima related functionals (9):	Mean and standard deviation of rising and falling slopes (1-4), mean (5) and standard deviation (6) of inter maxima distances, amplitude mean of maxima (7), amplitude range of maxima (8) and minima (9)
Others (6):	Linear Prediction Coefficients (LPC) (1 to 5), Linear Prediction (LP) gain (6)

Table 3: Set of 42 functionals

3.4 Summary

In conclusion, audio was extracted similarly in 2019 and 2020. The most relevant changes that were applied so this part of the project was more oriented to the study’s objective, and they consisted of improving the recordings, including only questions that led to approximately one-minute-long answers, and limiting the subjects to women.

4 Audio and personality correlations

4.1 Introduction

In the present section, the results about audio and personality correlations obtained during the study will be presented, firstly by showing the top 3 correlations per personality trait for both years studied (2019 and 2020), followed by a deeper study looking for consistency in these correlations and lastly the conclusions obtained from these results.

The details of the audio features appearing in this chapter are available in *table 18* annexed at the end of the Memory to make this chapter more accessible.

4.2 Top 3 correlations per personality trait

In this section, the top 3 correlations per personality trait will be analysed and compared between both year's data with the aim of presenting the features showing higher correlation values. This will be measured with the Pearson Correlation Coefficient (PCC) and its associated p-value, which have been detailed previously.

4.2.1 Openness

As it can be seen in tables 4 and 5, the highest correlating feature belonged to the same LLD in both years: MFCC, which is a representation of the short-term power spectrum of a signal in a psychoacoustic (Mel) scale. However, in both years different MFCC coefficient appeared in the top three.

Although none of the complete features matched between the two studies, PCM related features, which come from the digital audio signal itself, were seen to appear in both top threes too, especially in the frequency band from 250 to 650 Hz.

Feature	PCC	p-value
mfcc_sma[3]_kurtosis	0.558636994	0.0000812
pcm_Mag_fband250-650_sma_lpc0	0.494224387	0.000650228
jitterLocal_sma_quartile1	0.494068417	0.000653195

Table 4: Top 3 correlations for openness in 2020

Moreover, in 2020 there could also be seen a remarkable correlation with the local (frame-to-frame) Jitter, that represents pitch period-length deviations.

SMA, which appears in many of the correlating features, stands for Smoothing Moving Average and is a smoothing of the temporal evolution of the feature that is not too relevant and will not be further commented.

Feature	PCC	p-value
mfcc_sma[10]_lpc3	-0.363065309	0.001087019
pcm_Mag_spectralEntropy_sma_de_upleveltime90	-0.354738485	0.001439058
pcm_Mag_fband250-650_sma_stddevFallingSlope	0.34708865	0.001850016

Table 5: Top 3 correlations for openness in 2019

4.2.2 Extraversion

In this case, no LLD coincided between both years' top threes. Nevertheless, notable correlations were found in 2020 for voicing related (F0final) features that measure the vibration of the vocal cords, as well as for MFCC, in particular when measuring the asymmetry of the distribution around its centroid (skewness).

Feature	PCC	p-value
F0final_sma_quartile3	0.460232638	0.00166998
F0final_sma_de_pctlrange0-1	0.456598379	0.00183688
mfcc_sma[9]_skewness	-0.452068721	0.002065459

Table 6: Top 3 correlations for extraversion in 2020

Meanwhile, in 2019 audspec (which is a frequency band conversion) related features showed significant correlation when calculating the normalized value of the hearing spectrum. Furthermore, how quickly the power spectrum of the signal changes (spectral flux) appeared to be correlated to extraversion.

Feature	PCC	p-value
audspec_ lengthL1norm_sma_iqr2-3	0.476805241	0.0000102
audspec_lengthL1norm_sma_meanRisingSlope	0.472660824	0.0000125
pcm_Mag_spectralFlux_sma_iqr2-3	0.460848311	0.0000218

Table 7: Top 3 correlations for extraversion in 2019

4.2.3 Neuroticism

MFCC related features, especially for the first coefficient in 2020, ended up being the ones showing higher correlations with neuroticism.

Feature	PCC	p-value
mfcc_sma[1]_percentile99.0	0.516909359	0.000327476
mfcc_sma[1]_rqmean	0.491031047	0.000713454
mfcc_sma[1]_quartile3	0.489711895	0.00074114

Table 8: Top 3 correlations for neuroticism in 2020

Nevertheless, measures coming from the digital audio signal itself calculated from the module of the spectrum (pcm_Mag) were present as well in the 2019's top three.

Feature	PCC	p-value
pcm_Mag_spectralRollOff75.0_sma_quartile1	-0.467758572	0.0000157
mfcc_sma[2]_meanFallingSlope	-0.434229666	0.0000713
pcm_Mag_psySharpness_sma_quartile1	-0.433531465	0.0000734

Table 9: Top 3 correlations for neuroticism in 2019

4.2.4 Conscientiousness

Conscientiousness did not present the exact same features in both studies' tops neither but, as it can be seen in tables 10 and 11, MFCC related features appeared in both, and in particular their deltas calculations, seeking variation.

It is also interesting that in 2020 all the features belonged to the seventh coefficient, while in 2019 the features showing higher correlation correspond to the 4 and 5th coefficients.

Feature	PCC	p-value
mfcc_sma_de[7]_kurtosis	-0.532351911	0.000199595
mfcc_sma_de[7]_upleveltime50	-0.512544013	0.000375081
mfcc_sma_de[7]_skewness	0.487807485	0.000782805

Table 10: Top 3 correlations for conscientiousness in 2020

Feature	PCC	p-value
mfcc_sma[5]_lpc4	0.442518211	0.0000498
mfcc_sma_de[4]_iqr1-2	0.393330987	0.000367402
mfcc_sma_de[4]_quartile2	0.361311704	0.001153898

Table 11: Top 3 correlations for conscientiousness in 2019

4.2.5 Agreeableness

Agreeableness was found to be correlated mainly to PCM features calculated from the module of the spectrum (pcm_Mag). However, zero crossing rate, and symmetry for the second MFCC are also represent in the top threes.

In 2020, both correlations were features related to the segment lengths obtained in a segmentation based on the symmetry of the spectrum.

Feature	PCC	p-value
mfcc_sma[2]_skewness	-0.4623554	0.00157884
pcm_Mag_spectralSkewness_sma_maxSegLen	0.453253194	0.002003382
pcm_Mag_spectralSkewness_sma_segLenStddev	0.453239121	0.00200411

Table 12: Top 3 correlations for agreeableness in 2020

Meanwhile, in 2019 two correlations were features associated to spectral roll-off points.

Feature	PCC	p-value
pcm_Mag_ spectralRollOff75.0_sma_linregc1	0.346714923	0.001872564
pcm_Mag_ spectralRollOff90.0_sma_linregc1	0.339721	0.002342767
pcm_zcr_sma_linregc1	0.327801597	0.003392511

Table 13: Top 3 correlations for agreeableness in 2019

4.3 Consistent correlations

As it can be observed analysing the results presented previously, no exact feature was repeated in neither top 3 from both years, so in order to look for consistent correlations a deeper exploration was required.

Firstly, a filter was applied to both years' results trying to narrow them just to consistent ones, but it was found to be too demanding as there was no feature showing significant consistent correlation with p-value below 0.01. The filter cut was then raised to p-values under 0.05 and the results are the following, excluding agreeableness that did not show any consistent correlations at all.

4.3.1 Openness

As it can be seen in table 14, there are four consistent correlations for openness. Two of them are associated with MFCC, and the other two, with the digital signal itself and the module of the spectrum.

The *minSegLen* and *maxSegLen* characteristics seem to be related to speech rate. As positive correlations appear with minimum and maximum segments, it seems that the correlation is with variations in the rate of speech, meaning less monotonous speech.

The iqr1-3 characteristic is the interquartile dispersion of the fifth MFCC, which measures variation in the distribution of voice energy in frequency, although in this case the correlation is negative.

Finally, the last characteristic measures the time when the variation (delta) of the spectral entropy is very high, which is related to whether the voice energy is very concentrated or distributed in frequency. It is a measure that also seems to be related to the previous ones and its correlation is also negative.

	2020		**2019**	
Correlation	**PCC**	**p-value**	**PCC**	**p-value**
mfcc_sma[1]_minSegLen	0.33061517	0.02838171	0.22381112	0.04886152
mfcc_sma[5]_iqr1-3	-0.3189114	0.03486176	-0.2293224	0.04342255
pcm_Mag_fband250-650_sma_maxSegLen	0.30557749	0.04367948	0.24472675	0.03081725
pcm_Mag_spectralEntropy_sma_de_upleveltime90	-0.2995934	0.0481877	-0.3547385	0.00143906

Table 14: Consistent openness correlations

4.3.2 Extraversion

When it comes to extraversion, we can see there are many more audio features that seem to show a good correlation to the personality trait ratings. In concrete, the clearest correlation is the one with the standard deviation (dispersion) of the falling slopes of the fundamental frequency or tone of the voice, as it is clear that having varied pitch falls (rather than similar ones) is related to extraversion. Furthermore, *logHNR_sma_quartile3* is related with the dispersion of the HNR, which is a measure of phonation also related to pitch like the one mentioned before; although not with dynamic characteristics but with high values of the logHNR.

In addition, *pcm_Mag_psySharpness_sma_quartile2* is related to the central values of sharpness, which leads to think that greater sharpness (higher frequencies) in the voice indicates greater extraversion.

The *pcm_Mag_spectralEntropy* related features are all measures of spectral entropy dispersion, which indicates whether the spectra have many peaks and valleys or remain constant. Since they are all positive correlations with dispersion, spectra that show greater variation between being flatter and having more peaks and valleys seem to indicate greater extraversions.

Spectral roll-off is the frequency below which 75% or 90% of the energy is. A 25% roll-off measurement with *delta* and *upleveltime50* means that it is determining the variations in the low frequency areas of the spectrum. Moreover, the 75% measurements are related with the dispersion of the variation again, which is a measure very related to that spectral entropy.

Lastly, the measure related to the Zero-Crossing Rate in the second quartile is associated to the average values of this measure, and it seems to indicate a tendency of higher extraversion at higher ZCR. However, this does not seem to make much sense, as the ZCR is mostly affected by phonation (using more voiced phonemes than unvoiced ones, for example), although it can also be affected by pitch (f0).

	2020		**2019**	
Correlation	**PCC**	**p-value**	**PCC**	**p-value**
F0final_sma_stddevFallingSlope	0.35138698	0.01934227	0.23922054	0.03491361
logHNR_sma_quartile3	-0.3739307	0.01240487	-0.26375	0.01963518
pcm_Mag_psySharpness_sma_quartile2	0.32077749	0.03375375	0.31244834	0.00535264
pcm_Mag_spectralEntropy_sma_de_iqr1-3	0.32811077	0.02967688	0.24963036	0.02751717
pcm_Mag_spectralEntropy_sma_de_iqr2-3	0.36942159	0.01359026	0.25971809	0.02165921
pcm_Mag_spectralEntropy_sma_de_quartile3	0.37831561	0.0113379	0.26084411	0.02107679
pcm_Mag_spectralEntropy_sma_minRangeRel	0.30699185	0.04266574	0.23245672	0.04055977

pcm_Mag_spectralRollOff25.0_sma_de_upleveltime50	-0.3228881	0.03253567	-0.250678	0.0268523
pcm_Mag_spectralRollOff75.0_sma_de_iqr1-2	0.2984903	0.04905845	0.23700278	0.03668758
pcm_Mag_spectralRollOff75.0_sma_de_iqr1-3	0.30749261	0.04231146	0.24245517	0.03245534
pcm_Mag_spectralRollOff75.0_sma_de_iqr2-3	0.31237433	0.03898168	0.24655904	0.02954714
pcm_Mag_spectralRollOff75.0_sma_de_quartile1	-0.2978036	0.04960692	-0.2386497	0.03536322
pcm_Mag_spectralRollOff75.0_sma_de_quartile3	0.31264076	0.03880631	0.24480196	0.03076422
pcm_Mag_spectralRollOff75.0_sma_quartile2	0.34024225	0.02383174	0.26044369	0.02128235
pcm_Mag_spectralRollOff90.0_sma_quartile1	0.30918629	0.0411309	0.31802586	0.0045476
pcm_Mag_spectralRollOff90.0_sma_quartile2	0.31024699	0.04040534	0.25857524	0.02226431
pcm_Mag_spectralRollOff90.0_sma_risetime	0.32607584	0.03076497	0.35522511	0.00141595
pcm_zcr_sma_quartile2	0.31708484	0.0359751	0.26308764	0.01995614

Table 15: Consistent extraversion correlations

4.3.3 Neuroticism

Audspec, which is a frequency band conversion, related to the hearing spectrum frame by frame normalized value, and in particular its 1% percentile and forth MFCC, and its 99% percentile, were the only features to show some consistent correlation with neuroticism ratings.

Extreme values (percentile 1 and percentile 99) can, on these features, show a positive association with neuroticism.

	2020		**2019**	
Correlation	**PCC**	**p-value**	**PCC**	**p-value**
audspec_lengthL1norm_sma_percentile1.0	-0.3473451	0.02088025	-0.2420035	0.03278961
mfcc_sma[4]_percentile99.0	0.32611092	0.03074594	0.27490932	0.0148562

Table 16: Consistent neuroticism correlations

4.3.4 Conscientiousness

Conscientiousness ended up being correlated mainly to MFCC's features. Regarding the third coefficient, the distribution of the signal, linear error computed as the difference of the linear approximation and linear error between contour and quadratic regression line; and related to the second coefficient, the time during which the signal is rising. Furthermore, regarding deltas from the second coefficient of the quartile 2 and from the fifth of the distribution of the signal, consistent correlations could be found as well.

Finally, harmonicity, calculated as the deltas from the module of the spectrum regarding the distribution of a signal, was also found to be correlated to this personality trait. Nevertheless, these features correlations could not be clearly associated to any conscientiousness characteristic in a theorical way, and would require a deeper study.

	2020		**2019**	
Correlation	**PCC**	**p-value**	**PCC**	**p-value**
mfcc_sma[2]_risetime	0.37827681	0.01134699	0.24798089	0.02859221
mfcc_sma[3]_kurtosis	-0.3776768	0.01148828	-0.240275	0.03409569
mfcc_sma[3]_linregerrA	0.3374141	0.02510025	0.28503252	0.01142573
mfcc_sma[3]_qregerrA	0.3294386	0.02898421	0.28520707	0.0113732
mfcc_sma_de[2]_quartile2	0.3735859	0.01249228	0.2569898	0.02312758
mfcc_sma_de[5]_kurtosis	-0.3602315	0.01630692	-0.2414308	0.03321761
pcm_Mag_harmonicity_sma_de_kurtosis	-0.3699899	0.01343574	-0.280786	0.01277029

Table 17: Consistent conscientiousness correlations

4.4 Conclusions

Even though in a previous study from 2019 lots of correlations were found [8], the new focus on women only and the addition of 2020's new dataset have proven many of those correlations to be inconsistent, as well as many others from the current work.

As it could be seen in each year's top 3, no exact feature was repeated in both datasets, and when the filters requirements were lowered, only extraversion showed a good number of clear correlations. This seems to be a good outcome, as this personality trait has previously been thought to be externalized through speech [29]. Essentially, extraversion is related to the variation in the slope of the pitch (which usually occurs at the end of sentences) and indicates that a more "singing" voice could be associated with a higher score.

In addition, spectral entropy and roll-off measurements also indicate that larger changes in the spectrum (which may also be related to more "singing" voices) could be associated with greater extraversion.

The remaining matching correlations with the other personality traits are very limited, and it may be necessary to corroborate them with an additional sample that could perhaps rule them out.

5 Personality predictions

5.1 Introduction

As the biggest challenge for this project is trying to estimate personality based on speech features, several machine learning algorithms were used for this purpose. Three different estimating methods that have been explained in detail earlier were applied: Linear Regression, Logistic Regression and Random Forests. The predictions took into consideration different perspectives and data sources trying to achieve the best results. In this section they will be presented and described as predictions based on specific features, on all features and on groups of features.

Data from 2019 was used to train the model, and 2020's to test it. Because personality from different years was evaluated in different ways, all the ratings and feature values were normalized in mean and variance. Furthermore, as the purpose was to analyse the capability of estimating people's personality in general, and not the exact value, results were split in three percentiles: percentile 25, percentile 75 and over percentile 75; corresponding to low scorer (0), medium scorer (1) and high scorer (2). In this way, we transform the regression problem into a 3-class classification problem for which we can compute the percentage of subjects correctly classified.

Plots have also been used to present the results in a clearer and more understandable way with a scatter plot, representing predictions values vs true values, and a confusion matrix for the 3-class classification problem. The predictions have been evaluated using two statistics: Accuracy, which is how many predictions have been correct for the percentile's 3-class classification problem, and Root Mean Squared Error (RMSE), a very common metric that informs about the actual size of an error produced by a regression model [65].

5.2 Predictions based on specific features

The first approach presented will be predictions based on specific features except agreeableness, for which it could not be found any feature showing a consistent correlation in the 2019 and 2020 data, as this leads to simpler models and allows to check the predictive power of individual features.

5.2.1 Openness

Openness was better predicted when using the feature *pcm_Mag_spectralEntropy_sma_de_upleveltime90*, which can be described as the time over 90% of the deltas from the variation range for the spectral entropy. Limited results were obtained with both linear regression and logistic regression, but in the first case they were slightly better: the predictor scored 61.4% with an RMSE value of 0.81 while the second scored 56.8% with a RMSE value of 0.99.

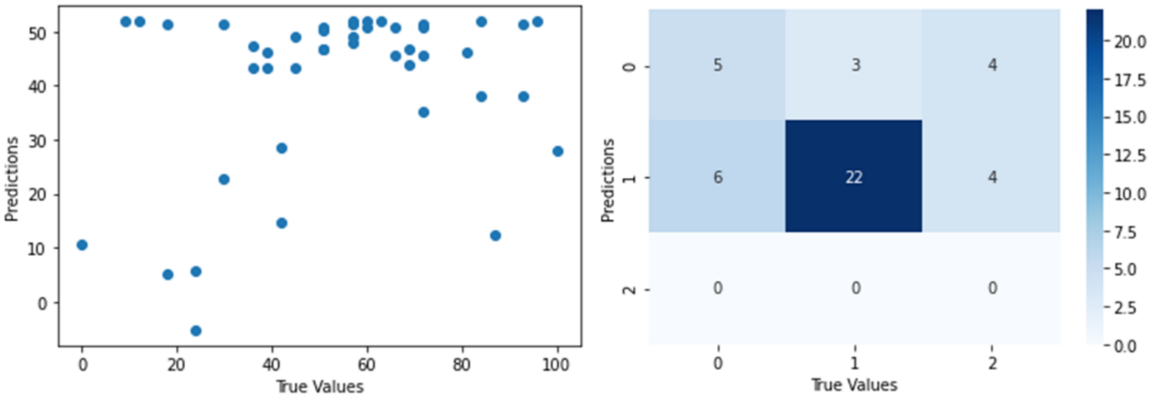

Figure 2: Linear regression scatter plot and confusion matrix for pcm_Mag_spectralEntropy_sma_de_upleveltime90 predicting openness

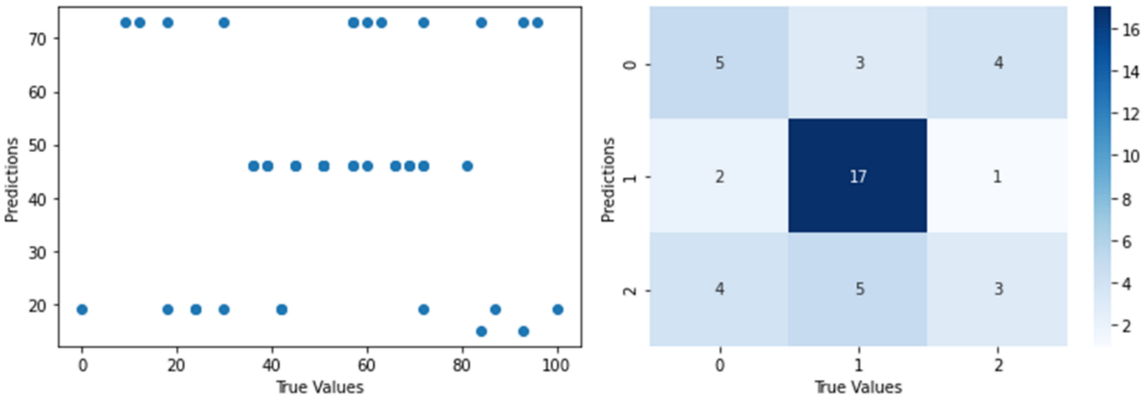

Figure 3: Logistic regression scatter plot and confusion matrix for pcm_Mag_spectralEntropy_sma_de_upleveltime90 predicting openness

Moreover, *mfcc_sma[5]_iqr1-3* showed similar results estimating openness too. This feature represents the 1-3 interquartile for the fifth MFCC and showed better results for the linear regression model as it can be seen in the scatter plot, with RMSE value of 0.75. The accuracy was of 56.8%.

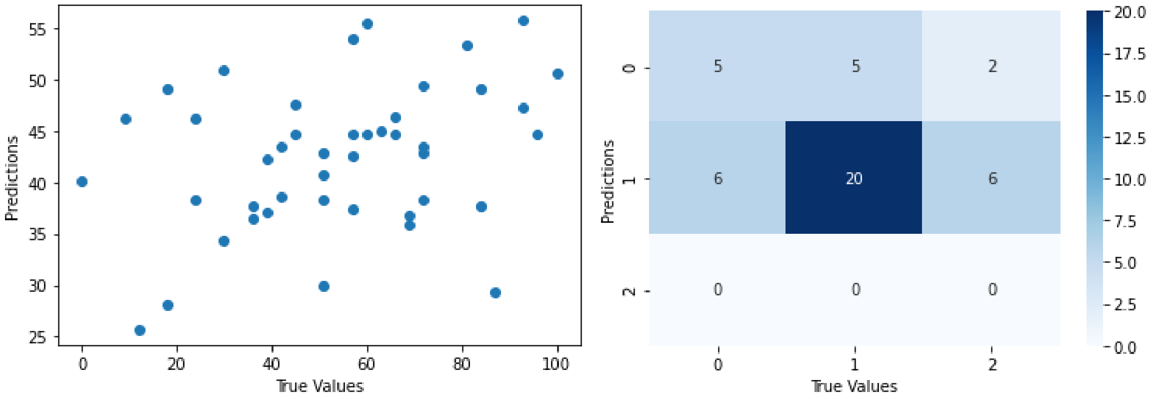

Figure 4: Linear regression scatter plot and confusion matrix for mfcc_sma[5]_iqr1-3 predicting openness

5.2.2 Extraversion

The best predictors turned out to be two features with the same prediction results: *F0final_sma_stddevFallingSlope* and *pcm_Mag_spectralRollOff75.0_sma_de_iqr1-2*. Both predicted extraversion with a 50% of accuracy and an RMSE value of 0.71.

The first one could be explained as the fundamental frequency's standard deviation of the falling slopes.

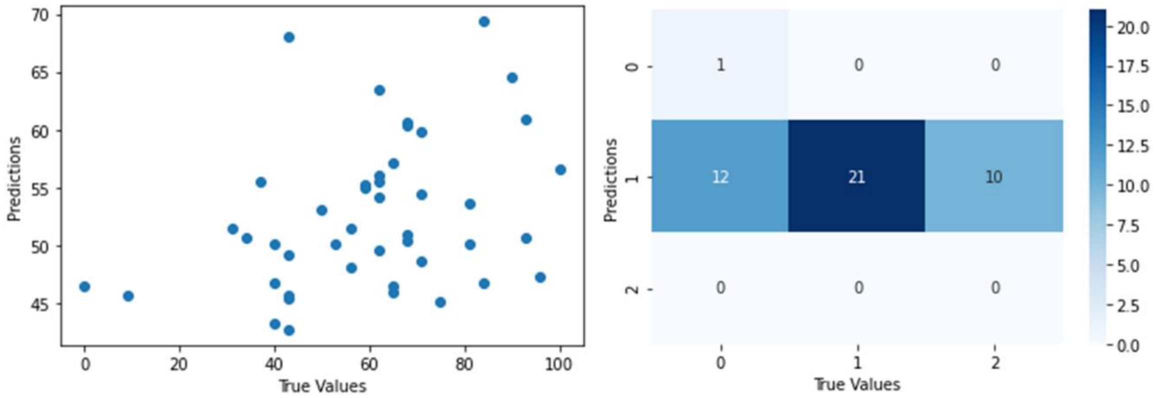

Figure 5: Linear regression scatter plot and confusion matrix for F0final_sma_stddevFallingSlope predicting extraversion

On the other hand, *pcm_Mag_spectralRollOff75.0_sma_de_iqr1-2* is a feature associated to the spectral entropy being calculated from the module of the spectrum. It represents the interquartile 1-2 of the rolloff.

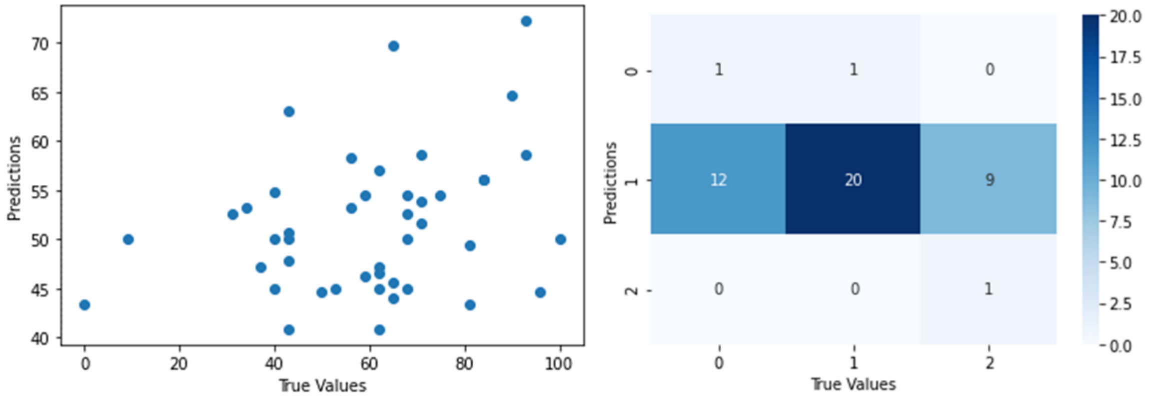

Figure 6: Linear regression scatter plot and confusion matrix for pcm_Mag_spectralRollOff75.0_sma_de_iqr1-2 predicting extraversion

These results were followed by two other features that also showed similar relationships with this personality trait: *pcm_Mag_spectralEntropy_sma_de_iqr1-3*, the equivalent of the one explained earlier for interquartile 1-3, obtaining an accuracy of 45.4% and RMSE value of 0.83 for the linear regression model; and *pcm_Mag_spectralRollOff90.0_sma_risetime*, which is also the same as the mentioned feature but for the 90% and the calculation of the time during which the signal is rising. This one scored 45.4% as well and had an RMSE value of 0.90.

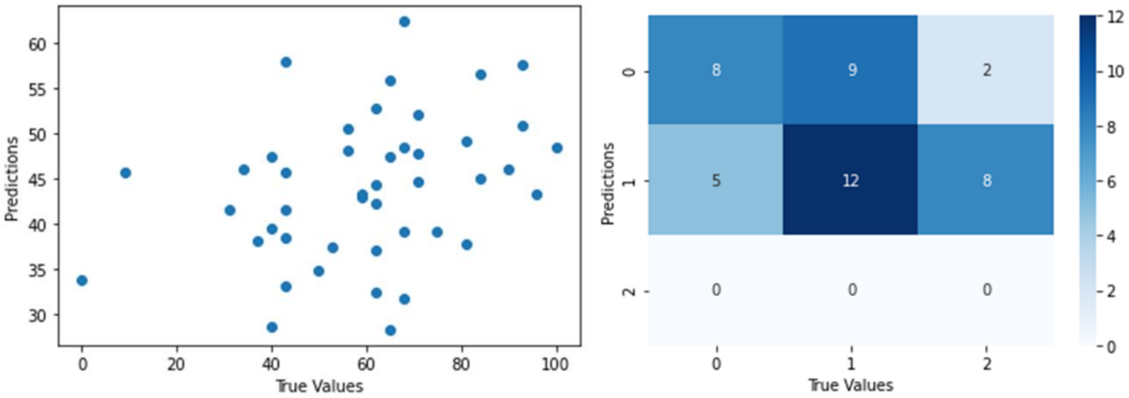

Figure 7: Linear regression scatter plot and confusion matrix for pcm_Mag_spectralEntropy_sma_de_iqr1-3 predicting extraversion

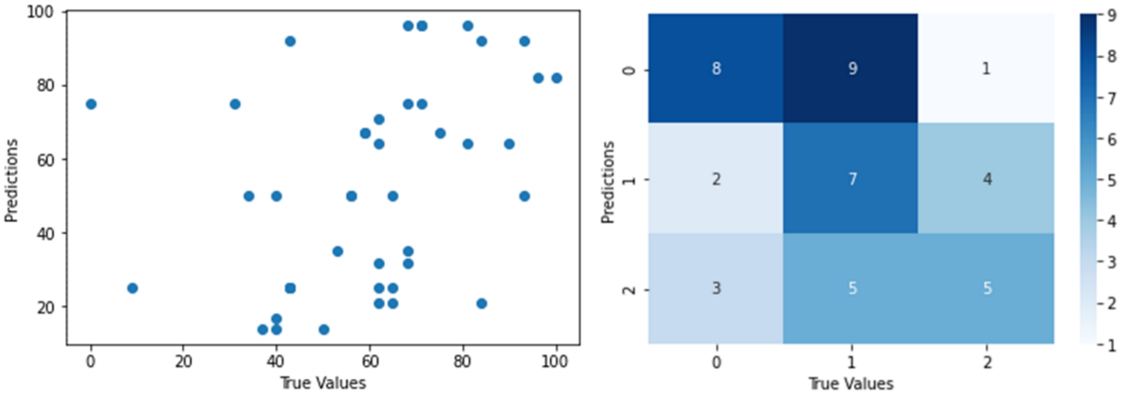

Figure 8: Random forest scatter plot and confusion matrix for Mag_spectralRollOff90.0_sma_risetime predicting extraversion

5.2.3 Neuroticism

The best results predicting neuroticism were found when estimating with the feature *audspec_lengthL1norm_sma_percentile1.0* using linear regression, with an accuracy of 52.3% and RMSE of 0.78. This feature represents a frequency band conversion, related to the hearing spectrum frame by frame normalized value regarding percentile 1.

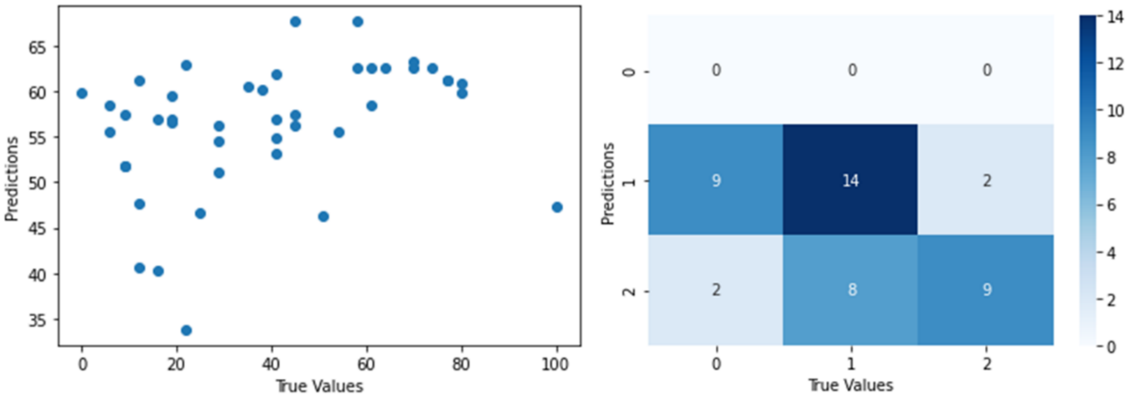

Figure 9: Linear regression scatter plot and confusion matrix for audspec_lengthL1norm_sma_percentile1.0 predicting neuroticism

The previous result was followed by the one provided for the linear regression model by the feature *mfcc_sma[4]_percentile99.0*, which is the fourth MFCC in the percentile 99 and scored 50% with a RMSE value of 0.8.

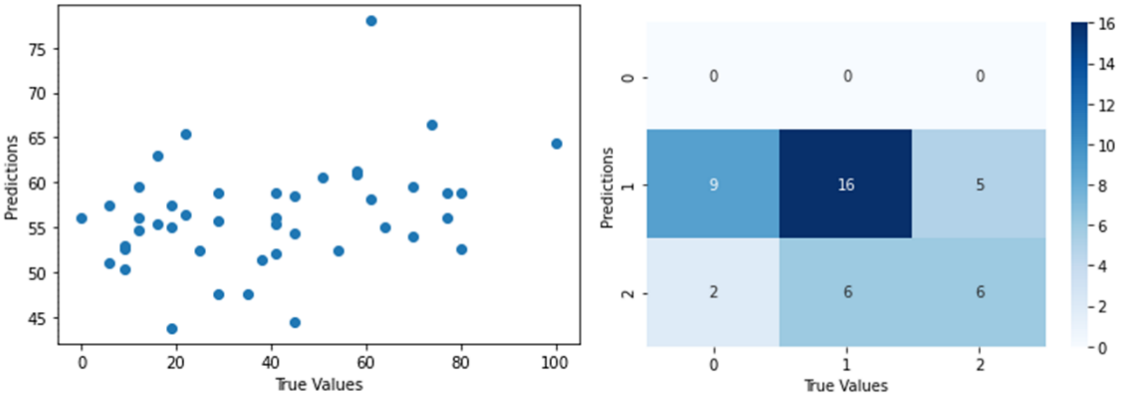

Figure 10: Linear regression scatter plot and confusion matrix for mfcc_sma[4]_percentile99.0 predicting neuroticism

5.2.4 Conscientiousness

In this case, the features giving better results were *pcm_Mag_harmonicity_sma_de_kurtosis* and *mfcc_sma[3]_kurtosis*, both with the linear regression model, with accuracies of 52.3% and 54.5%, and RMSE values of 0.74 and 0.72 respectively. The first one represents harmonicity being calculated as the deltas from the module of the spectrum regarding distribution of the signal, and the second as the third MFCC regarding the distribution of the signal. The last one also showed the highest correlation to openness in 2020's dataset.

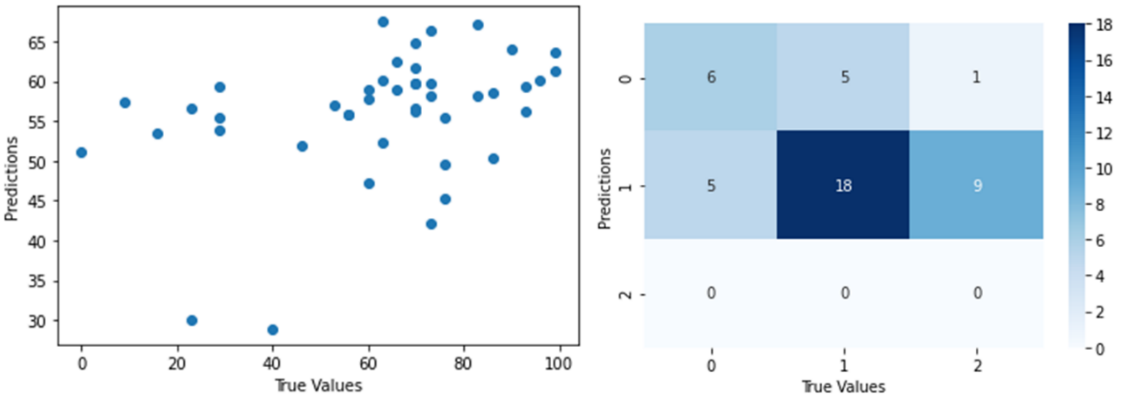

Figure 11: Linear regression scatter plot and confusion matrix for pcm_Mag_harmonicity_sma_de_kurtosis predicting conscientiousness

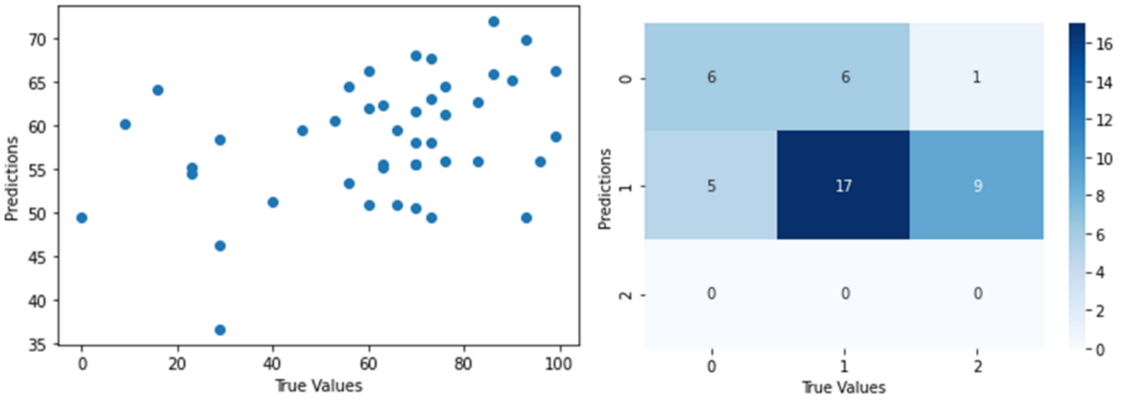

Figure 12: Linear regression scatter plot and confusion matrix for mfcc_sma[3]_kurtosis predicting conscientiousness

5.3 Predictions based on all features

Seeking the “strength of the group”, all features (1941 in total) were evaluated as predictors for the personality traits, and the results will be detailed next. However, these results ended up being worse than for specific features, so only those for openness and extraversion will be shown, as the other personality traits results were even worse.

5.3.1 Openness

The best result for predicting openness with all the features was found when applying the logistic regression model. The accuracy was of 34.1% with a RMSE of 1.06.

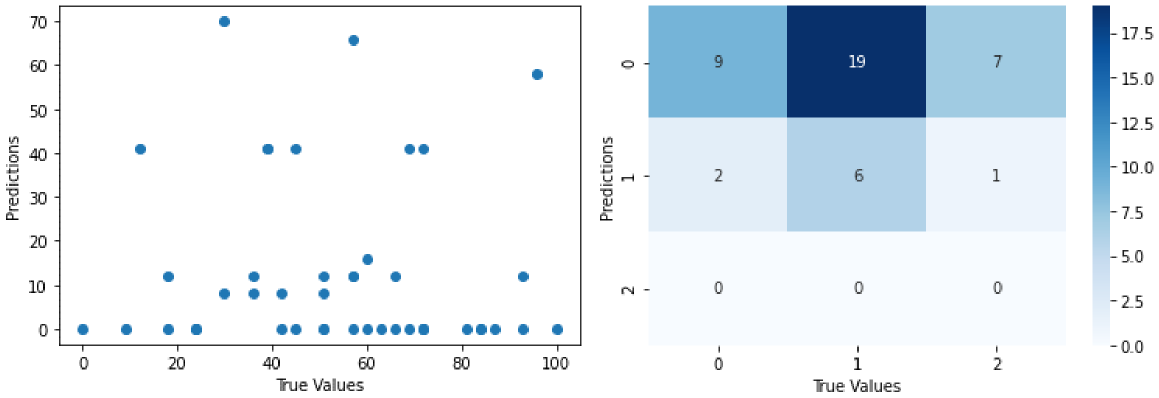

Figure 13: Logistic regression scatter plot and confusion matrix for all features predicting openness

5.3.2 Extraversion

Extraversion was found to be the personality trait that could be best estimated based on all features, especially with the linear regression model. The accuracy was 38.6% with a RMSE of 1.13.

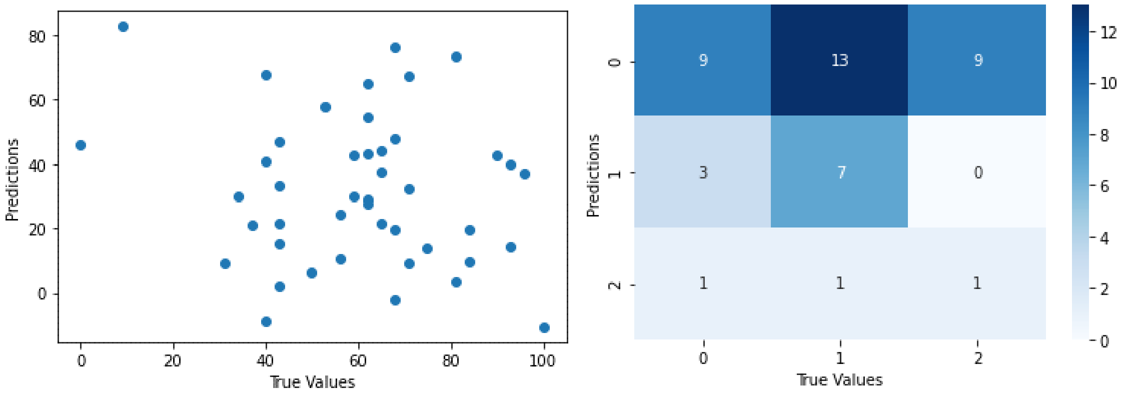

Figure 14: Linear regression scatter plot and confusion matrix for all features predicting extraversion

5.4 Predictions based on groups of features

Finally, this was the last approach to the prediction modelling algorithms, trying to estimate different traits based on the general feature or LLD with all the associated statistics and measures. Neuroticism and conscientiousness will not be presented as their results were even more limited.

5.4.1 Openness

This personality trait found its best predictor in the *jitterDDP*, which is the differential frame-to-frame variability in frequency, in other words, the jitter of the jitter. This group was formed by 51 features and results were relatively good for both logistic regression (accuracy of 40.9% and RMSE of 1.0) and random forest (accuracy of 43.2% and RMSE of 0.95), that had slightly better results.

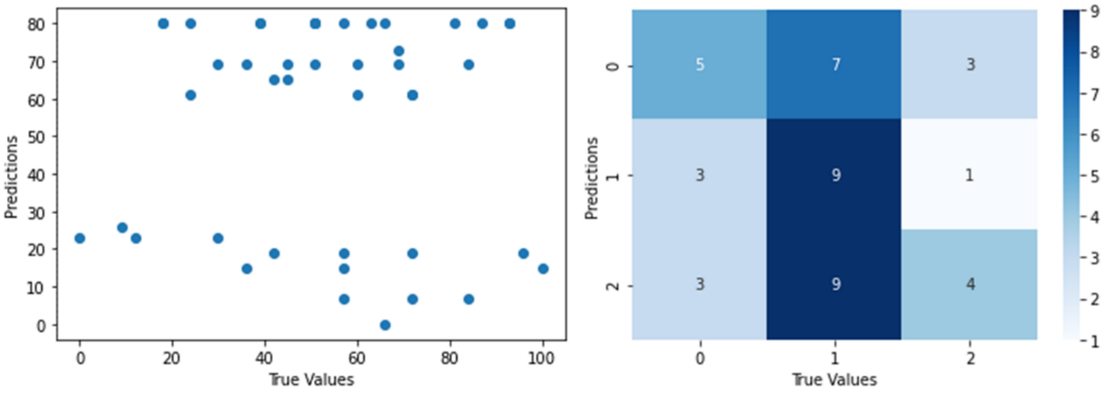

Figure 15: Logistic regression scatter plot and confusion matrix for jitterDDP predicting openness

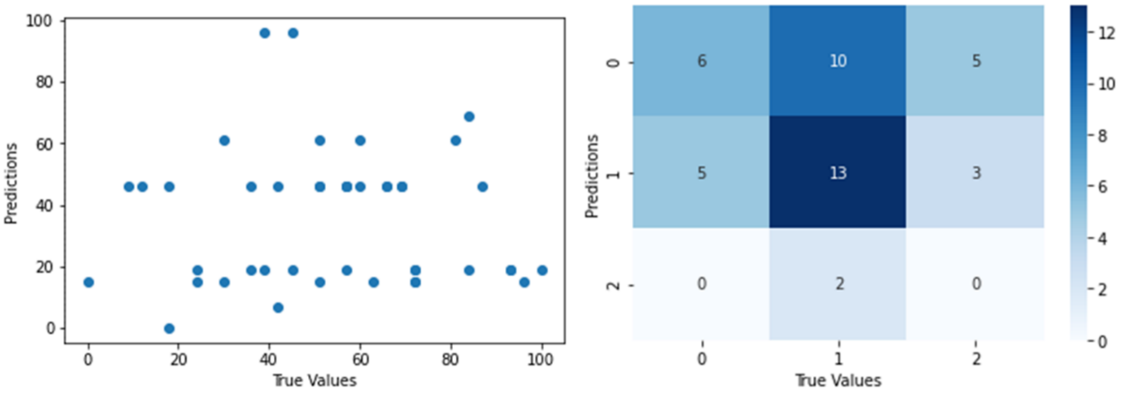

Figure 16: Random Forest scatter plot and confusion matrix for jitterDDP predicting openness

Nevertheless, *mfcc* related features (650 in total) also showed similar results for logistic regression (accuracy of 38.6% and RMSE of 1.01) and random forest (accuracy of 38.6% and RMSE of 0.97) models. Mel Frequency Cepstral Coefficients are a representation of the short-term power spectrum of a signal in a psychoacoustic (Mel) scale.

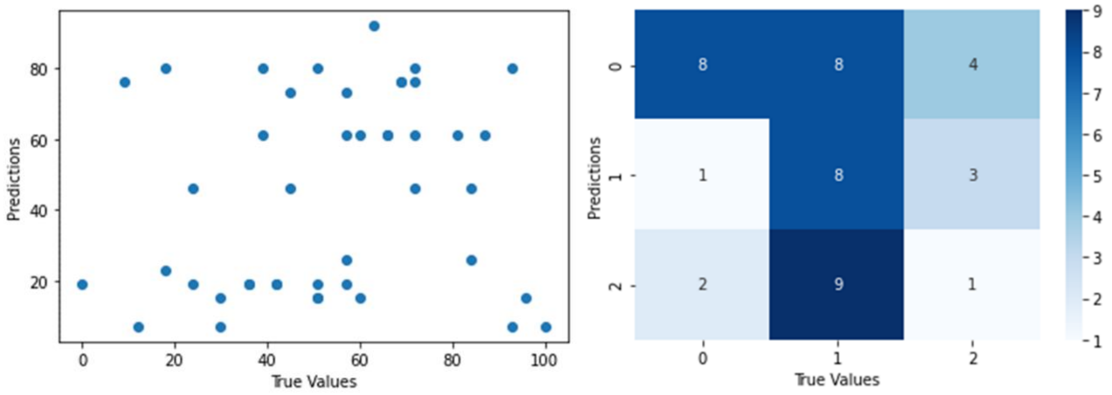

Figure 17: Logistic regression scatter plot and confusion matrix for mfcc predicting openness

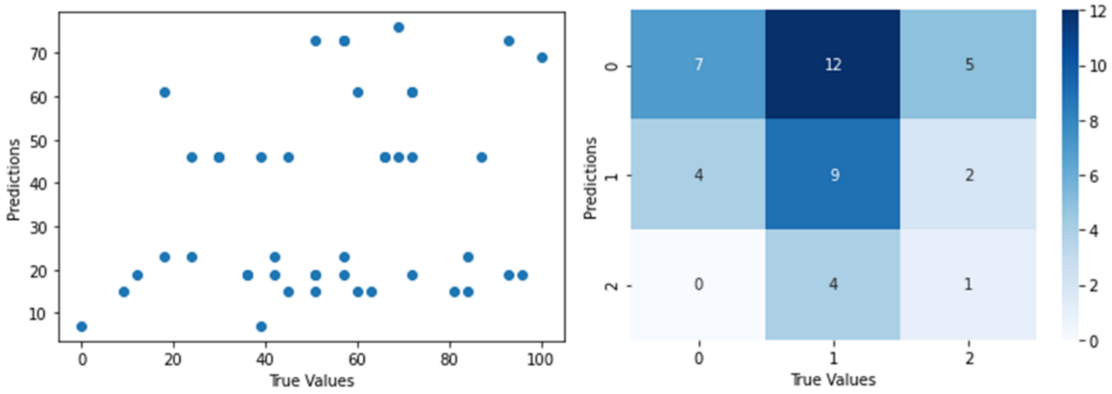

Figure 18: Random Forest scatter plot and confusion matrix for mfcc predicting openness

5.4.2 Extraversion

Extraversion turned out to be the best predictable trait based on an LLD. In concrete, for *F0final* (61 features in total) the accuracy was of 45.4% with a RMSE value of 1.08 with the logistic regression model. This feature represents the frequency at which the vocal cords vibrate to produce voiced sounds.

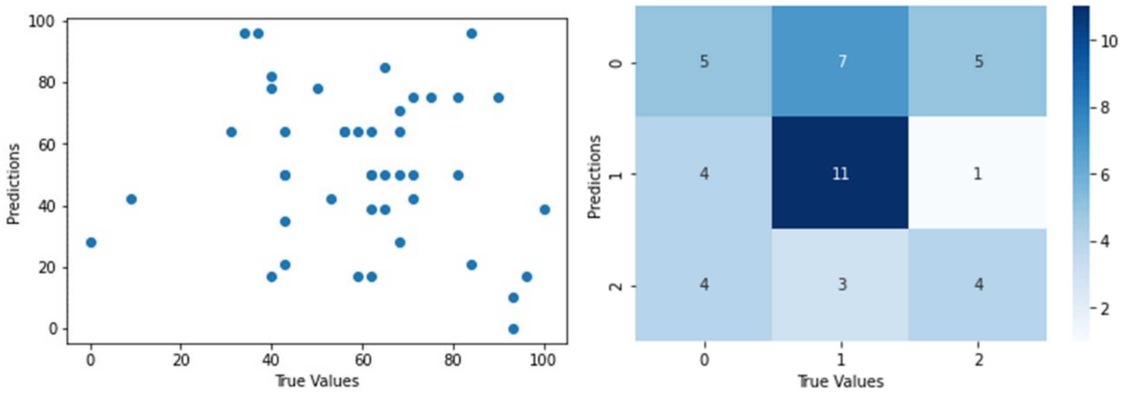

Figure 19: Logistic regression scatter plot and confusion matrix for F0final predicting extraversion

5.4.3 Agreeableness

Finally, group of features was the only way of obtaining reasonable results for the prediction of agreeableness. The LLD with better results in this case was *mfcc* with an accuracy of 38.6% and an RMSE value of 1.01 with the linear regression model, and 34.1% and 1.0 with logistic regression.

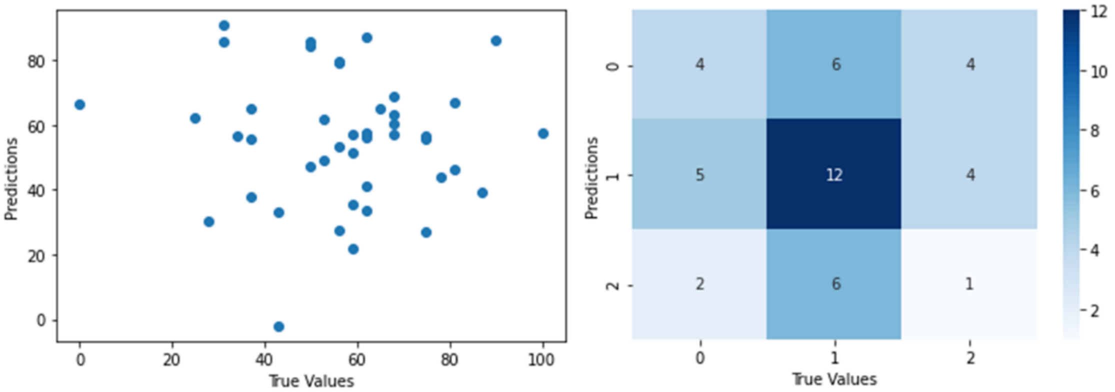

Figure 20: Linear regression scatter plot and confusion matrix for mfcc predicting agreeableness

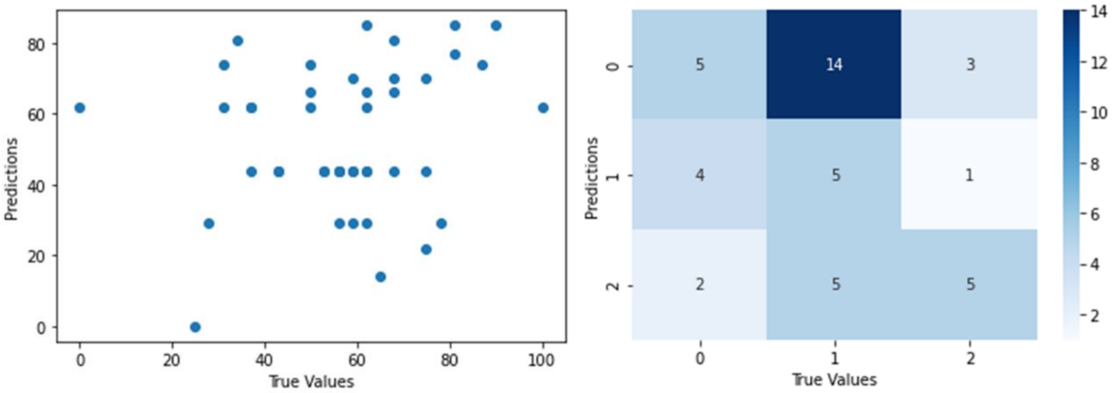

Figure 21: Logistic regression scatter plot and confusion matrix for mfcc predicting agreeableness

5.5 Conclusions

Even though predictions have been observed to be very limited, various results encourage to believe there are some predicting capabilities. For instance, better outcomes were observed when predictions were performed based on one specific feature, as in the case of *pcm_Mag_spectralEntropy_sma_de_upleveltime90*, to estimate openness through linear regression or logistic regression. Another trait that seems to be possible to predict is extraversion, which correlates well with features relating variation in F0 decreasing slope and variations in the spectrum. Outcomes were not good enough when based on all features or on groups of features neither, and only openness' and extraversion's were slightly better.

6 Conclusions and future work

6.1 Conclusions

The shifted focus of the study, from a dataset that included men and women to a new one with different information and restricted to women, lead to lots of the correlations that were found in 2019 being proven to be inconsistent, as well as many others from the present work. Only extraversion (and openness in a more limited way) showed better results. The remaining matching correlations were very limited.

However, this seems like a good result as this personality trait is thought to be externalized trough speech. Essentially, extraversion has been found related to the variation in the slope of the pitch (which usually occurs at the end of sentences) and indicates that a more "singing" voice could be associated with a higher score. In addition, spectral entropy and roll-off measurements have also been found to indicate that larger changes in the spectrum (which may also be related with more "singing" voices) could be associated with greater extraversion too.

Regarding predictive modelling algorithms, which aimed to estimate personality traits from the speech features obtained for the study, results were observed to be very limited in terms of accuracy and RMSE, and also through scatter plots for regression models and confusion matrixes for classification evaluation. Nevertheless, various results encourage to believe that there are some predicting capabilities, and extraversion and openness ended up being the most predictable personality traits.

Better outcomes were achieved when predictions were performed based on one specific feature instead of all of them or a reduced group, as it was the case for openness. For extraversion, an interesting conclusion is that correlation found with features relating variation in F0 decreasing slope and variations in the spectrum is consistent with previous research, and it also seems to make sense from a theoretical point of view.

6.2 Future work

Based on these conclusions, a future work suggestion could be finding a smaller set of features that could be explored in more detail focusing on specific personality traits, so it was not as hard to explore as a 1941 feature set. Moreover, exploring pauses structures and speech speed, which have not been treated in this project, could add more depth to this work. For this purpose, instead of using OpenSMILE as audio feature extractor, Praat (another open-source voice software) could be applied.

Furthermore, checking the correlations found in this project, going deeper into the explanations of the relationships discovered and trying to confirm and explain the correlations yet to understand would be the next steps for this study.

References

[1] Gocsál, Á. (2009). Female listeners' personality attributions to male speakers: The role of acoustic parameters of speech. Pollack Periodica, 4(3), 155-165.

[2] Funder, D. C., & Colvin, C. R. (1988). Friends and strangers: Acquaintanceship, agreement, and the accuracy of personality judgment. Journal of Personality and Social Psychology, 55(1), 149–158

[3] Ambady, N., & Rosenthal, R. (1992). Thin slices of expressive behaviour as predictors of interpersonal consequences: A meta-analysis. Psychological Bulletin, 111(2), 256–274.

[4] Goffman, E. (1979). Gender advertisements. New York: Harper & Rowe

[5] Aylett, M. P., Vinciarelli, A., & Wester, M. (2017). Speech synthesis for the generation of artificial personality. IEEE transactions on affective computing.

[6] Allport, G. W (1937). Personality: A psychological interpretation. New York: Holt.

[7] Allport, G. W., & Cantril, H. (1934). Judging personality from voice. The Journal of Social Psychology, 5(1), 37-55.

[8] Victor J. Rubio, David Aguado, María Pilar Fernandez-Gallego, Doroteo T. Toledano, Estrella Pulido and Gonzalo Martínez, (2019). Feasibility of Big Data analytics to assess personality based on movement expression and voice analysis. Unpublished UAM report

[9] Orozco, L.M. (2010). An Empirical Comparison between the NEO-FFI and the WPI and the Relationship between Self-Efficacy and Workplace Personality.

[10] McDougall W. (1932). Of the words character and personality. Character Personality, 1, 3-16.

[11] Cattell RB. (1943). The description of personality: basic traits resolved into clusters. Journal of Abnormal Social Psychology, 38, 476-506.

[12] Cattell RB. (1946). The description and measurement of personality. Yonkers, NY: World Book.

[13] Cattell RB. (1947). Confirmation and clarification of primary personality factors. Psychometrika, 12,197-220.

[14] Cattell RB. (1948). The primary personality factors in women compared with those in men. British Journal of Psychology, 1,114-130.

[15] Fiske DW. (1949). Consistency of the factorial structures of personality ratings from different sources. Journal of Abnormal Social Psychology, 44, 329-344.

[16] Tupes EC. (1957). Personality traits related to effectiveness of junior and senior Air Force officers (USAF Personnel Training Research, No. 57-125). Lackland Airforce Base, TX: Aeronautical Systems Division, Personnel Laboratory

[17] Tupes EC, Christal RE. (1961). Recurrent personality factors based on trait ratings (ASD-TR-61-97). Lackland Air Force Base, TX: Aeronautical Systems Division, Personnel Laboratory.

[18] Smith GM. (1967). Usefulness of peer ratings of personality in educational research. Educational and Psychological Measurement, 27, 967-984

[19] Hakel MD. (1974). Normative personality factors recovered from ratings of personality descriptors: The beholder's eye. Personnel Psychology, 27,409-421.

[20] Borgatta EE (1964). The structure of personality characteristics. Behavioural Science, 12, 8-17.

[21] Norman WT (1963). Toward an adequate taxonomy of personality attributes: Replicated factor structure in peer nomination personality ratings. Journal of Abnormal & Social Psychology, 66, 574-583.

[22] Costa, P. T., Jr., & McCrae, R. R. (1985). The NEO Personality Inventory manual. Odessa, FL: Psychological Assessment Resources

[23] Costa, P.T., Jr., & McCrae, R.R. (1997). Set like plaster? Evidence for the stability of adult personality. In T.F. Heatherton & J.L. Weinberger (Eds.). Can personality change (pp. 21-41). Washington DC: American Psychological Association.

[24] Smith, B. L., Brown, B. L., Strong, W. J., & Rencher, A. C. (1975). Effects of speech rate on personality perception. Language and Speech, 18, 145–152.

[25] Sherer, K.R. (1978). Personality markers in speech. In K.R. Scherer and H. Giles (Eds.), Social markers in speech (pp. 147-209). Cambridge, Ma: Cambridge University Press.

[26] Brunswick, E. (1956). Perception and the representative design of experiments. Berkeley: Univ. of California Press

[27] W. Apple, L. A. Streeter, and R. M. Krauss (1979). Effects of pitch and speech rate on personal attributions. Journal of Personality and Social Psychology, vol. 37, no. 5, pp. 715–727.

[28] Brown, B. & Bradshaw, J. (1985). Towards a social psychology of voice variations. In H. Giles and R. St Clair (Eds.), Recent advances in language communication and social psychology (pp. 144-181). London: Erlbaum.

[29] Furham, A. (1990). Language and personality. In H. Giles & W.P. Robinson (Eds.), Handbook of Language and Social Psychology. Chichester, UK: John Wiley.

[30] Mairesse, F., Walker, M. A., Mehl, M. R., & Moore, R. K. (2007). Using linguistic cues for the automatic recognition of personality in conversation and text. Journal of Artificial Intelligence Research, 30, 457-500.

[31] Hu, C., Wang, Q., Short, L. A., & Fu, G. (2012). Speech spectrum's correlation with speakers' Eysenck Personality Traits. PloS one, 7(3), e33906.

[32] Clifford Nass and Scott Brave. (2005). Wired for Speech: How Voice Activates and Advances the Human-Computer Relationship. The MIT Press.

[33] Tim Polzehl, Alexander Schmitt, Florian Metze, Michael Wagner, Anger recognition in speech using acoustic and linguistic cues. (2011). Speech Communication, Volume 53, Issues 9–10, Pages 1198-1209, ISSN 0167-6393

[34] Gelareh Mohammadi, Alessandro Vinciarelli, and Marcello Mortillaro. (2010). The voice of personality: mapping nonverbal vocal behavior into trait attributions. In Proceedings of the 2nd international workshop on Social signal processing (SSPW '10). Association for Computing Machinery, New York, NY, USA, 17–20.

[35] Martin, A. F., & Przybocki, M. A. (2001). The NIST speaker recognition evaluations: 1996-2001. In 2001: A Speaker Odyssey-The Speaker Recognition Workshop.

[36] Alvin, M. P., & Martin, A. (2004). NIST speaker recognition evaluation chronicles. In Proc. Odyssey 2004, The Speaker and Language Recognition Workshop.

[37] Przybocki, M. A., Martin, A. F., & Le, A. N. (2007). NIST speaker recognition evaluations utilizing the Mixer corpora—2004, 2005, 2006. IEEE Transactions on Audio, Speech, and Language Processing, 15(7), 1951-1959.

[38] Sadjadi, S. O., Kheyrkhah, T., Tong, A., Greenberg, C. S., Reynolds, D. A., Singer, E., Mason, L. & Hernandez-Cordero, J. (2017). The 2016 NIST Speaker Recognition Evaluation. In Interspeech (pp. 1353-1357).

[39] Wu, K., & Childers, D. G. (1991). Gender recognition from speech. Part I: Coarse analysis. The journal of the Acoustical society of America, 90(4), 1828-1840.

[40] Espinoza-Cuadros, F., Fernández-Pozo, R., Toledano, D. T., Alcázar-Ramírez, J. D., López-Gonzalo, E., & Hernández-Gómez, L. A. (2016). Reviewing the connection between speech and obstructive sleep apnea. Biomedical engineering online, 15(1), 20.

[41] Gómez, P., Martínez, R., Díaz, F., Lázaro, C., Álvarez, A., Rodellar, V., & Nieto, V. (2005). Voice pathology detection by vocal cord biomechanical parameter estimation. In International Conference on Nonlinear Analyses and Algorithms for Speech Processing (pp. 242-256). Springer, Berlin, Heidelberg.

[42] Skodda, S., & Schlegel, U. (2008). Speech rate and rhythm in Parkinson's disease. Movement disorders: official journal of the Movement Disorder Society, 23(7), 985-992.

[43] Satt, A., Sorin, A., Toledo-Ronen, O., Barkan, O., Kompatsiaris, I., Kokonozi, A., & Tsolaki, M. (2013). Evaluation of speech-based protocol for detection of early-stage dementia. In INTERSPEECH (pp. 1692-1696).

[44] Satt, A., Hoory, R., König, A., Aalten, P., & Robert, P. H. (2014). Speech-based automatic and robust detection of very early dementia. In Fifteenth Annual Conference of the International Speech Communication Association.

[45] Raghavendra Pappagari, Jaejin Cho, Laureano Moro-Velazquez, Najim Dehak (2020). Using state of the art speaker recognition and natural language processing technologies to detect Alzheimer's disease and assess its severity. Center for Language Speech Processing, Johns Hopkins University, Baltimore, MD, USA

[46] Weiner, J., Herff, C., & Schultz, T. (2016). Speech-Based Detection of Alzheimer's Disease in Conversational German. In INTERSPEECH (pp. 1938-1942).

[47] Baykaner, K. R., Huckvale, M., Whiteley, I., Andreeva, S., & Ryumin, O. (2015). Predicting fatigue and psychophysiological test performance from speech for safety-critical environments. Frontiers in bioengineering and biotechnology, 3, 124.

[48] Schuller, B., Steidl, S., & Batliner, A. (2009). The interspeech 2009 emotion challenge. In Tenth Annual Conference of the International Speech Communication Association.

[49] Schuller, B., Steidl, S., Batliner, A., Burkhardt, F., Devillers, L., Müller, C., & Narayanan, S. S. (2010). The INTERSPEECH 2010 paralinguistic challenge. In Eleventh Annual Conference of the International Speech Communication Association.

[50] Schuller, B., Valstar, M., Eyben, F., McKeown, G., Cowie, R., & Pantic, M. (2011). Avec 2011–the first international audio/visual emotion challenge. In International Conference on Affective Computing and Intelligent Interaction (pp. 415-424). Springer, Berlin, Heidelberg.

[51] Ringeval, F., Schuller, B., Valstar, M., Cowie, R., Kaya, H., Schmitt, M., Amiriparian, S., Cummins, N., Lalanne, D., Michaud, A., Çiftçi, E., Güleç, H., Salah, A. A., & Pantic, M. (2018). AVEC 2018 workshop and challenge: Bipolar disorder and cross-cultural affect recognition. In Proceedings of the 2018 on Audio/Visual Emotion Challenge and Workshop (pp. 3-13). ACM.

[52] Eyben, F., Wöllmer, M., & Schuller, B. (2010). Opensmile: the munich versatile and fast open-source audio feature extractor. In Proceedings of the 18th ACM international conference on Multimedia (pp. 1459-1462). ACM.

[53] Audeering. (2020). openSMILE. https://www.audeering.com/opensmile/

[54] Pandas. (2020). About pandas. https://pandas.pydata.org/about/

[55] The SciPy community. (2020). Introduction. https://docs.scipy.org/doc/scipy/reference/tutorial/general.html

[56] Virtanen, P., Gommers, R., Oliphant, T.E. et al. (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat Methods 17, 261–272

[57] The SciPy community. (2020). Scipy.stats.pearsonr. https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.pearsonr.html

[58] Scikit-learn. (2020). Machine learning in Python. https://scikit-learn.org/stable/index.html

[59] Scikit-learn. (2020). sklearn.linear_model.LogisticRegression. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

[60] Scikit-learn. (2020). Ordinary Least Squares. https://scikit-learn.org/stable/modules/linear_model.html#ordinary-least-squares

[61] Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M., & Klein, M. (2002). Logistic regression. New York: Springer-Verlag.

[62] Scikit-learn. (2020). Logistic Regression. https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

[63] Scikit-learn. (2020). Forests of randomized trees. https://scikit-learn.org/stable/modules/ensemble.html#forest

[64] audEERING. (2020). Reference section: Feature names. openSMILE. https://audeering.github.io/opensmile/reference.html#feature-names

[65] Willmott, C. J. (1981). On the validation of models. Physical geography, 2(2), 184-194.

Glossary

MFCC	Mel-Frequency Cepstral Coefficients
SRE	Speaker Recognition Evaluations
NIST	National Institute of Standards and Technology
HNR	Harmonic-to-Noise Ratio
EER	Equal Error Rate
AAC	Autocorrelation Coefficients
MSE	Mean Square Error
AVEC	Audio Visual Emotion Challenge
AI	Artificial Intelligence
LLD	Low Level Descriptors
PLP-CC	Perceptual Linear Predictive Cepstral Coefficients
LPC	Linear Predictive Coefficients
LSP	Line Spectral Pairs
DCT	Discrete Cosine Transformation
PCC	Pearson Correlation Coefficient
API	Application Programming Interface
F0	Fundamental Frequency
PCM	Pulse Code Modulation
LPC	Linear Prediction Coefficient
SMA	Smoothing Moving Average
RMSE	Root Mean Square Error
ZCR	Zero Cross Rate

Annexes

A Audio features detailed description

Feature	Explanation
mfcc_sma[3]_kurtosis	Peakedness of the spectrum of the third MFCC
pcm_Mag_fband250-65_sma_lpc0	Linear prediction coefficient zero of the spectrum module from the 250-650 Hz frequency band
jitterLocal_sma_quartile1	First quartile of the frame-to-frame pitch period deviations
mfcc_sma[10]_lpc3	Linear prediction coefficient three of the tenth MFCC
pcm_Mag_spectralEntropy_sma_de_ upleveltime90	Percentage of time over 90% of the range of variation of the deltas of the spectral entropy
pcm_Mag_fband250-65_sma_stddevFallingSlope	Standard deviation of the falling slopes of the spectrum module from the 250-650 Hz frequency band
F0final_sma_quartile3	Third quartile of the smoothed fundamental frequency contour
F0final_sma_de_pctlrange0-1	Outlier robust signal range 'max-min' represented by the range of the 1% and the 99% percentile from the smoothed fundamental frequency contour
mfcc_sma[9]_skewness	Symmetry measure of the ninth MFCC
audspec_ lengthL1norm_sma_iqr2-3	Interquartile 2-3 of the hearing spectrum frame-by-frame normalized value on a frequency band conversion
audspec_lengthL1norm_sma_ meanRisingSlope	Mean of the rising slope for the hearing spectrum frame-by-frame normalized value
pcm_Mag_spectralFlux_sma_iqr2-3	Interquartile 2-3 of how quickly the power spectrum of the signal changes when calculated from the module of the spectrum
mfcc_sma[1]_percentile99.0	Percentile 99 of the first MFCC

mfcc_sma[1]_rqmean	Root-quadratic mean of the first MFCC
mfcc_sma[1]_quartile3	Third quartile of the first MFCC
pcm_Mag_spectralRollOff75.0_sma_quartile1	First quartile of the 75% of the spectral roll-off points obtained from the spectrum module
mfcc_sma[2]_meanFallingSlope	Mean of the falling slope of the second MFCC
pcm_Mag_psySharpness_sma_quartile1	First quartile of the psychoacoustic sharpness obtained from the spectrum module
mfcc_sma_de[7]_kurtosis	Peakedness of the spectrum of the deltas from the seventh MFCC
mfcc_sma_de[7]_upleveltime50	Time over 50% of the variation range of the deltas from the seventh MFCC
mfcc_sma_de[7]_skewness	Symmetry measure of the deltas from the seventh MFCC
mfcc_sma[5]_lpc4	Linear prediction coefficient four of the fifth MFCC
mfcc_sma_de[4]_iqr1-2	Interquartile range 1-2 of the deltas of MFCC number 4.
mfcc_sma_de[4]_quartile2	Second quartile of the deltas of MFCC 4
mfcc_sma[2]_skewness	Symmetry measure of the second MFCC
pcm_Mag_spectralSkewness_sma_maxSegLen	Maximum of the segment lengths of the spectral symmetry measure from the spectrum module
pcm_Mag_spectralSkewness_sma_segLenStddev	Standard deviation of the segment lengths of the spectral symmetry measure from the spectrum module
pcm_Mag_ spectralRollOff75.0_sma_linregc1	Slope of a linear approximation of the contour for spectral roll-off points' 75% from the spectrum module
pcm_Mag_ spectralRollOff90.0_sma_linregc1	Slope of a linear approximation of the contour for spectral roll-off points' 90% from the spectrum module

pcm_zcr_sma_linregc1	Slope of a linear approximation of the contour for zero crossing rate
mfcc_sma[1]_minSegLen	Minimum segment length of the first MFCC
mfcc_sma[5]_iqr1-3	Interquartile 1-3 of the fifth MFCC
pcm_Mag_fband250-650_sma_maxSegLen	Maximum segment length of the spectrum module from the 250-650 Hz frequency band
pcm_Mag_spectralEntropy_sma_de_upleveltime90	Time over 90% of the variation range of the deltas of the spectral entropy from the spectrum module
F0final_sma_stddevFallingSlope	Standard deviation of the falling slope of the smoothed fundamental frequency contour
logHNR_sma_quartile3	Third quartile of the log of the harmonic-to-noise (HNR) ratio
pcm_Mag_psySharpness_sma_quartile2	Second quartile of the psychoacoustic sharpness obtained from the spectrum module
pcm_Mag_spectralEntropy_sma_de_iqr1-3	Interquartile 1-3 of the delta of the spectral entropy from the spectrum module
pcm_Mag_spectralEntropy_sma_de_iqr2-3	Interquartile 2-3 of the delta of the spectral entropy from the spectrum module
pcm_Mag_spectralEntropy_sma_de_quartile3	Third quartile of the delta of the spectral entropy from the spectrum module
pcm_Mag_spectralEntropy_sma_minRangeRel	Relative minimum range of the spectral entropy from the spectrum module
pcm_Mag_spectralRollOff25.0_sma_de_upleveltime50	Time over 50% of the variation range of deltas for spectral roll-off points' 25% of the spectrum module
pcm_Mag_spectralRollOff75.0_sma_de_iqr1-2	Interquartile 1-2 of the deltas for spectral roll-off points' 75% of the spectrum module
pcm_Mag_spectralRollOff75.0_sma_de_iqr1-3	Interquartile 1-3 of the deltas for spectral roll-off points' 75% of the spectrum module

pcm_Mag_spectralRollOff75.0_sma_de_iqr2-3	Interquartile 2-3 of the deltas for spectral roll-off points' 75% of the spectrum module
pcm_Mag_spectralRollOff75.0_sma_de_quartile1	First quartile of the deltas for spectral roll-off points' 75% of the spectrum module
pcm_Mag_spectralRollOff75.0_sma_de_quartile3	Third quartile of the deltas for spectral roll-off points' 75% of the spectrum module
pcm_Mag_spectralRollOff75.0_sma_quartile2	Second quartile of the deltas for spectral roll-off points' 75% of the spectrum module
pcm_Mag_spectralRollOff90.0_sma_quartile1	First quartile of the deltas for spectral roll-off points' 90% of the spectrum module
pcm_Mag_spectralRollOff90.0_sma_quartile2	Second quartile of the deltas for spectral roll-off points' 90% of the spectrum module
pcm_Mag_spectralRollOff90.0_sma_risetime	Rise time of the signal for spectral roll-off points' 90% of the spectrum module
pcm_zcr_sma_quartile2	Second quartile of the zero-crossing rate
audspec_lengthL1norm_sma_percentile1.0	First percentile of the hearing spectrum frame by frame normalized value on a frequency band conversion
mfcc_sma[4]_percentile99.0	Percentile 99 of the fourth MFCC
mfcc_sma[2]_risetime	Rising time of the second MFCC
mfcc_sma[3]_kurtosis	Peakedness of the spectrum of the third MFCC
mfcc_sma[3]_linregerrA	Linear error computed as the difference of the linear approximation and the actual contour of the third MFCC
mfcc_sma[3]_qregerrA	Linear error between contour and quadratic regression line of the third MFCC
mfcc_sma_de[2]_quartile2	Second quartile deltas of the second MFCC
mfcc_sma_de[5]_kurtosis	Peakedness of the spectrum of deltas of the fifth MFCC
pcm_Mag_harmonicity_sma_de_kurtosis	Peakedness of the deltas of the harmonicity computed from the module of the spectrum

Table 18: Audio features detailed description

B Correlation's Code

2019 Correlations

```
#IMPORTS
import pandas as pd
from scipy.stats.stats import pearsonr
import zipfile

#EXTRA FUNCTIONS
#Function to iterate through columns
def getcolumn(matrix, col):
    columna = []
    for row in matrix:
        columna.append(row[col])
    return columna

#PERSONALITY DATA
#Obtaining a list of the ids
dataunfiltered = pd.read_csv(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\ids_equivalence.csv')
data1 = dataunfiltered[dataunfiltered['genero']==1] #Women's ids
idslist = data1['id_rodrigo'].to_list()

#Obtaining personality data from experts
data2 = pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\personalidad_2019.xlsx')

list1_as_set = set(data1['id_rodrigo'])
intersection = list1_as_set.intersection(data2['id_rodrigo'])
ids = list(intersection)
data = data2[data2['id_rodrigo'].isin(ids)]

#PERSONALITY FEATURES
#Obtaining all audio features
featuresunfiltered =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\audio_2019.xlsx')
all_features =
featuresunfiltered[featuresunfiltered['id_rodrigo'].isin(ids)]
features = all_features
```

```
#VARIABLES

m_data = data.values
m_features = features.values

c_data = data.columns
c_features = features.columns

l_data = len(m_data[0])
l_features = len(m_features[0])

#CORRELATIONS

correlations = {}
i,j = 2,1

while i < l_data:
    col_data = getcolumn(m_data, i)
    while j < l_features:
        col_features = getcolumn(m_features, j)
        correlations[str(c_data[i]) + '__' + str(c_features[j])] =
pearsonr(col_data, col_features)

        j+=1
    i+=1
    j=1

result = pd.DataFrame.from_dict(correlations, orient='index')
result.columns = ['PCC', 'p-value']

potentialcorrelations =
result.sort_index()[result.sort_index()['p-value'].between(0,
0.05)]

#EXPORTATION

with zipfile.ZipFile('NEW-correlations2019.zip', 'w') as csv_zip:
    csv_zip.writestr("NEW-all-correlations2019.csv",
result.sort_index().to_csv())
    csv_zip.writestr("NEW-potential-correlations2019.csv",
potentialcorrelations.sort_index().to_csv())
```

2020 Correlations

```
#IMPORTS
import pandas as pd
import glob
from scipy.stats.stats import pearsonr
import zipfile

#EXTRA FUNCTIONS
#Function to iterate through columns
def getcolumn(matrix, col):
    columna = []
    for row in matrix:
        columna.append(row[col])
    return columna

#PERSONALITY FEATURES
path =
r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020PersonalityDet
ection'

all_files = glob.glob(path + "/*.csv")

li = []

for filename in all_files:

    df = pd.read_csv(filename, index_col=None, header=0)

    li.append(df)

all_features = pd.concat(li, axis=0, ignore_index=True)
features = all_features

#PERSONALITY DATA
data =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020
Personalidad\PuntuacionesPersonalidad.xlsx')

#VARIABLES

m_data = data.values
m_features = features.values

c_data = data.columns
c_features = features.columns

l_data = len(m_data[0])
l_features = len(m_features[0])
```

```
#CORRELATIONS

correlations = {}
i,j = 3,1

while i < l_data:
    col_data = getcolumn(m_data, i)
    while j < l_features:
        col_features = getcolumn(m_features, j)
        correlations[str(c_data[i]) + '__' + str(c_features[j])] =
pearsonr(col_data, col_features)

        j+=1
    i+=1
    j=1

result = pd.DataFrame.from_dict(correlations, orient='index')
result.columns = ['PCC', 'p-value']

potentialcorrelations =
result.sort_index()[result.sort_index()['p-value'].between(0,
0.05)]

#EXPORTATION

with zipfile.ZipFile('NEW-correlations.zip', 'w') as csv_zip:
    csv_zip.writestr("NEW-all-correlations.csv",
result.sort_index().to_csv())
    csv_zip.writestr("NEW-potential-correlations.csv",
potentialcorrelations.sort_index().to_csv())
```

Correlations comparation

```
#IMPORTS
import pandas as pd
import zipfile


#PERSONALITY DATA

data20 = pd.read_csv(r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\0.
BUENOS\Correlations\RESULTADOS\CONTRAST\NEW\NEW-potential-
correlations.csv')

data19 = pd.read_csv(r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\0.
BUENOS\Correlations\RESULTADOS\CONTRAST\NEW\NEW-potential-
correlations2019.csv')

list1_as_set = set(data20['correlation'])
intersection = list1_as_set.intersection(data19['correlation'])

features = list(intersection)
print(features)

data2020 = data20[(data20['correlation']).isin(features)]
data2019 = data19[(data19['correlation']).isin(features)]


with zipfile.ZipFile('last-year-COMPARATION.zip', 'w') as csv_zip:
    csv_zip.writestr("coincidences-2020.csv", data2020.to_csv())
    csv_zip.writestr("coincidences-2019.csv", data2019.to_csv())
```

C Prediction’s Code

Linear Regression for specific features

```
#IMPORTS
import pandas as pd
import glob
from sklearn.linear_model import LinearRegression
from sklearn import preprocessing
from matplotlib import pyplot as plt
import numpy as np
from sklearn.metrics import  confusion_matrix
from sklearn.metrics import mean_squared_error
from sklearn import metrics
import seaborn as sns

#PERSONALITY DATA 2019
#Obtaining a list of the ids
dataunfiltered_2019 = pd.read_csv(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\ids_equivalence.csv')
data1_2019 = dataunfiltered_2019[dataunfiltered_2019['genero']==1]
#Women's ids
idslist = data1_2019['id_rodrigo'].to_list()

#Obtaining personality data from experts
data2_2019 = pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\personalidad_2019.xlsx')
list1_as_set_2019 = set(data1_2019['id_rodrigo'])
intersection_2019 =
list1_as_set_2019.intersection(data2_2019['id_rodrigo'])
ids_2019 = list(intersection_2019)
all_data_2019 =
data2_2019[data2_2019['id_rodrigo'].isin(ids_2019)]

#PERSONALITY FEATURES 2019
#Obtaining all audio features
featuresunfiltered_2019 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\audio_2019.xlsx')
all_features_2019 =
featuresunfiltered_2019[featuresunfiltered_2019['id_rodrigo'].isin
(ids_2019)]
```

```
#PERSONALITY FEATURES 2020
path =
r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020PersonalityDet
ection'

all_files = glob.glob(path + "/*.csv")

li = []

for filename in all_files:
    df = pd.read_csv(filename, index_col=None, header=0)
    li.append(df)

all_features_2020 = pd.concat(li, axis=0, ignore_index=True)

#PERSONALITY DATA 2020
all_data_2020 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020
Personalidad\PuntuacionesPersonalidad.xlsx')

#FILTERING
principal_feature = 'mfcc_sma_de[7]_kurtosis'
personality_treat = 'RESP'

#2019
filter_col2_2019 = [col for col in all_data_2019 if
col.startswith('Exp1_'+personality_treat)]
data_2019 =
all_data_2019[filter_col2_2019].reset_index().drop('index',
axis=1)

filter_col1_2019 = [col for col in all_features_2019 if
col.startswith(principal_feature)]
features_2019 =
all_features_2019[filter_col1_2019].reset_index().drop('index',
axis=1)

#2020
filter_col1_2020 = [col for col in all_features_2020 if
col.startswith(principal_feature)]
features_2020 = all_features_2020[filter_col1_2020]

filter_col2_2020 = [col for col in all_data_2020 if
col.startswith('total'+personality_treat+'auto')]
data_2020 = all_data_2020[filter_col2_2020]
```

```
#2020 DATAFRAME CREATION
min_max_scaler = preprocessing.MinMaxScaler() #Normalization

df_2020 = pd.DataFrame(features_2020)
df_2020['total'+personality_treat+'auto'] = data_2020

#Normalization
df_2020_normalized = min_max_scaler.fit_transform(df_2020.values)

df_2020 = pd.DataFrame(df_2020_normalized)

#Dropping the last column to be the independent variable
df_2020 = df_2020.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2020 = df_2020.drop('personality', axis=1) * 100
y_2020 = df_2020.personality * 100

#2019 DATAFRAME CREATION

df_2019 = pd.DataFrame(features_2019)
df_2019['Exp1_'+personality_treat] = data_2019

#Normalization
df_2019_normalized = min_max_scaler.fit_transform(df_2019.values)

df_2019 = pd.DataFrame(df_2019_normalized)

#Dropping the last column to be the independent variable
df_2019 = df_2019.rename(columns={len(df_2019.columns)-1:
'personality'})

x_2019 = df_2019.drop('personality', axis=1) * 100
y_2019 = df_2019.personality * 100

#SPLIT INTO TRAINING MODEL
#2019 for training and 2020 for testing
x_train = x_2019.astype('int')
x_test = x_2020.astype('int')
y_train = y_2019.astype('int')
y_test = y_2020.astype('int')

#LINEAR REGRESSION

lr_Model = LinearRegression()
lr_Model.fit(x_train, y_train)

y_pred = lr_Model.predict(x_test)
```

```
#PERCENTIL DECLARATION

low_scorers_real = 0
medium_scorers_real = 0
high_scorers_real = 0

low_scorers_pred = 0
medium_scorers_pred = 0
high_scorers_pred = 0

percentil25 = np.percentile(y_test,25)
percentil75 = np.percentile(y_test,75)

y_len = len(y_test)


#PREDICTIONS' PERCENTILES CALCULATION
i1 = 0

d1 = [None] * y_len

while i1<y_len:
    if y_pred[i1] <= percentil25:
     d1[i1] = 0
    elif y_pred[i1] <= percentil75:
     d1[i1] = 1
    else:
     d1[i1] = 2
    i1+=1

#TEST'S PERCENTILES CALCULATION
i2 = 0

d2 = [None] * y_len

while i2<y_len:
    if y_test[i2] <= percentil25:
        d2[i2] = 0
    elif y_test[i2] <= percentil75:
        d2[i2] = 1
    else:
        d2[i2] = 2
    i2+=1
```

```
#STATISTICS CALCULATIONS
cf_matrix = confusion_matrix(d1, d2)

score = metrics.accuracy_score(d2, d1)
print(score)

rms = mean_squared_error(d2, d1, squared=False)
print(rms)

#PLOTS
plt.scatter(y_test, y_pred)
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()


sns.heatmap(cf_matrix, annot=True, cmap='Blues')
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()
```

Logistic Regression for specific features

```
#IMPORTS
import pandas as pd
import glob
from sklearn.linear_model import LogisticRegression
from sklearn import preprocessing
from matplotlib import pyplot as plt
import numpy as np
from sklearn.metrics import  confusion_matrix
from sklearn.metrics import mean_squared_error
from sklearn import metrics
import seaborn as sns

#PERSONALITY DATA 2019
#Obtaining a list of the ids
dataunfiltered_2019 = pd.read_csv(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\ids_equivalence.csv')
data1_2019 = dataunfiltered_2019[dataunfiltered_2019['genero']==1]
#Women's ids
idslist = data1_2019['id_rodrigo'].to_list()

#Obtaining personality data from experts
data2_2019 = pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\personalidad_2019.xlsx')
list1_as_set_2019 = set(data1_2019['id_rodrigo'])
intersection_2019 =
list1_as_set_2019.intersection(data2_2019['id_rodrigo'])
ids_2019 = list(intersection_2019)
all_data_2019 =
data2_2019[data2_2019['id_rodrigo'].isin(ids_2019)]

#PERSONALITY FEATURES 2019
#Obtaining all audio features
featuresunfiltered_2019 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\audio_2019.xlsx')
all_features_2019 =
featuresunfiltered_2019[featuresunfiltered_2019['id_rodrigo'].isin
(ids_2019)]
```

```
#PERSONALITY FEATURES 2020
path =
r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020PersonalityDet
ection'

all_files = glob.glob(path + "/*.csv")

li = []

for filename in all_files:
    df = pd.read_csv(filename, index_col=None, header=0)
    li.append(df)

all_features_2020 = pd.concat(li, axis=0, ignore_index=True)

#PERSONALITY DATA 2020
all_data_2020 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020
Personalidad\PuntuacionesPersonalidad.xlsx')

#FILTERING
principal_feature = 'mfcc_sma_de[7]_kurtosis'
personality_treat = 'RESP'

#2019
filter_col2_2019 = [col for col in all_data_2019 if
col.startswith('Exp1_'+personality_treat)]
data_2019 =
all_data_2019[filter_col2_2019].reset_index().drop('index',
axis=1)

filter_col1_2019 = [col for col in all_features_2019 if
col.startswith(principal_feature)]
features_2019 =
all_features_2019[filter_col1_2019].reset_index().drop('index',
axis=1)

#2020
filter_col1_2020 = [col for col in all_features_2020 if
col.startswith(principal_feature)]
features_2020 = all_features_2020[filter_col1_2020]

filter_col2_2020 = [col for col in all_data_2020 if
col.startswith('total'+personality_treat+'auto')]
data_2020 = all_data_2020[filter_col2_2020]
```

```
#2020 DATAFRAME CREATION
min_max_scaler = preprocessing.MinMaxScaler() #Normalization

df_2020 = pd.DataFrame(features_2020)
df_2020['total'+personality_treat+'auto'] = data_2020

#Normalization
df_2020_normalized = min_max_scaler.fit_transform(df_2020.values)

df_2020 = pd.DataFrame(df_2020_normalized)

#Dropping the last column to be the independent variable
df_2020 = df_2020.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2020 = df_2020.drop('personality', axis=1) * 100
y_2020 = df_2020.personality * 100

#2019 DATAFRAME CREATION

df_2019 = pd.DataFrame(features_2019)
df_2019['Exp1_'+personality_treat] = data_2019

#Normalization
df_2019_normalized = min_max_scaler.fit_transform(df_2019.values)

df_2019 = pd.DataFrame(df_2019_normalized)

#Dropping the last column to be the independent variable
df_2019 = df_2019.rename(columns={len(df_2019.columns)-1:
'personality'})

x_2019 = df_2019.drop('personality', axis=1) * 100
y_2019 = df_2019.personality * 100

#SPLIT INTO TRAINING MODEL
#2019 for training and 2020 for testing
x_train = x_2019.astype('int')
x_test = x_2020.astype('int')
y_train = y_2019.astype('int')
y_test = y_2020.astype('int')

#LOGISTIC REGRESSION

logisticRegr = LogisticRegression(max_iter=10000000)
logisticRegr.fit(x_train, y_train)

y_pred = logisticRegr.predict(x_test)
```

```
#PERCENTIL DECLARATION

low_scorers_real = 0
medium_scorers_real = 0
high_scorers_real = 0

low_scorers_pred = 0
medium_scorers_pred = 0
high_scorers_pred = 0

percentil25 = np.percentile(y_test,25)
percentil75 = np.percentile(y_test,75)

y_len = len(y_test)

#PREDICTIONS' PERCENTILES CALCULATION
i1 = 0

d1 = [None] * y_len

while i1<y_len:
    if y_pred[i1] <= percentil25:
     d1[i1] = 0
    elif y_pred[i1] <= percentil75:
     d1[i1] = 1
    else:
     d1[i1] = 2
    i1+=1

#TEST'S PERCENTILES CALCULATION
i2 = 0

d2 = [None] * y_len

while i2<y_len:
    if y_test[i2] <= percentil25:
        d2[i2] = 0
    elif y_test[i2] <= percentil75:
        d2[i2] = 1
    else:
        d2[i2] = 2
    i2+=1
```

```
#STATISTICS CALCULATIONS
cf_matrix = confusion_matrix(d1, d2)

score = metrics.accuracy_score(d2, d1)
print(score)

rms = mean_squared_error(d2, d1, squared=False)
print(rms)

#PLOTS
plt.scatter(y_test, y_pred)
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()


sns.heatmap(cf_matrix, annot=True, cmap='Blues')
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()
```

Random Forest for specific features

```
#IMPORTS
import pandas as pd
import glob
from sklearn.ensemble import RandomForestClassifier
from sklearn import preprocessing
from matplotlib import pyplot as plt
import numpy as np
from sklearn.metrics import  confusion_matrix
from sklearn.metrics import mean_squared_error
from sklearn import metrics
import seaborn as sns

#PERSONALITY DATA 2019
#Obtaining a list of the ids
dataunfiltered_2019 = pd.read_csv(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\ids_equivalence.csv')
data1_2019 = dataunfiltered_2019[dataunfiltered_2019['genero']==1]
#Women's ids
idslist = data1_2019['id_rodrigo'].to_list()

#Obtaining personality data from experts
data2_2019 = pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\personalidad_2019.xlsx')
list1_as_set_2019 = set(data1_2019['id_rodrigo'])
intersection_2019 =
list1_as_set_2019.intersection(data2_2019['id_rodrigo'])
ids_2019 = list(intersection_2019)
all_data_2019 =
data2_2019[data2_2019['id_rodrigo'].isin(ids_2019)]

#PERSONALITY FEATURES 2019
#Obtaining all audio features
featuresunfiltered_2019 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\audio_2019.xlsx')
all_features_2019 =
featuresunfiltered_2019[featuresunfiltered_2019['id_rodrigo'].isin
(ids_2019)]
```

```
#PERSONALITY FEATURES 2020
path =
r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020PersonalityDet
ection'

all_files = glob.glob(path + "/*.csv")

li = []

for filename in all_files:
    df = pd.read_csv(filename, index_col=None, header=0)
    li.append(df)

all_features_2020 = pd.concat(li, axis=0, ignore_index=True)

#PERSONALITY DATA 2020
all_data_2020 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020
Personalidad\PuntuacionesPersonalidad.xlsx')

#FILTERING
principal_feature = 'mfcc_sma_de[7]_kurtosis'
personality_treat = 'RESP'

#2019
filter_col2_2019 = [col for col in all_data_2019 if
col.startswith('Exp1_'+personality_treat)]
data_2019 =
all_data_2019[filter_col2_2019].reset_index().drop('index',
axis=1)

filter_col1_2019 = [col for col in all_features_2019 if
col.startswith(principal_feature)]
features_2019 =
all_features_2019[filter_col1_2019].reset_index().drop('index',
axis=1)

#2020
filter_col1_2020 = [col for col in all_features_2020 if
col.startswith(principal_feature)]
features_2020 = all_features_2020[filter_col1_2020]

filter_col2_2020 = [col for col in all_data_2020 if
col.startswith('total'+personality_treat+'auto')]
data_2020 = all_data_2020[filter_col2_2020]
```

```
#2020 DATAFRAME CREATION
min_max_scaler = preprocessing.MinMaxScaler() #Normalization

df_2020 = pd.DataFrame(features_2020)
df_2020['total'+personality_treat+'auto'] = data_2020

#Normalization
df_2020_normalized = min_max_scaler.fit_transform(df_2020.values)

df_2020 = pd.DataFrame(df_2020_normalized)

#Dropping the last column to be the independent variable
df_2020 = df_2020.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2020 = df_2020.drop('personality', axis=1) * 100
y_2020 = df_2020.personality * 100

#2019 DATAFRAME CREATION

df_2019 = pd.DataFrame(features_2019)
df_2019['Exp1_'+personality_treat] = data_2019

#Normalization
df_2019_normalized = min_max_scaler.fit_transform(df_2019.values)

df_2019 = pd.DataFrame(df_2019_normalized)

#Dropping the last column to be the independent variable
df_2019 = df_2019.rename(columns={len(df_2019.columns)-1:
'personality'})

x_2019 = df_2019.drop('personality', axis=1) * 100
y_2019 = df_2019.personality * 100

#SPLIT INTO TRAINING MODEL
#2019 for training and 2020 for testing
x_train = x_2019.astype('int')
x_test = x_2020.astype('int')
y_train = y_2019.astype('int')
y_test = y_2020.astype('int')

#RANDOM FOREST

rF_Model = RandomForestClassifier()
rF_Model.fit(x_train, y_train)

y_pred = rF_Model.predict(x_test)
```

```
#PERCENTIL DECLARATION

low_scorers_real = 0
medium_scorers_real = 0
high_scorers_real = 0

low_scorers_pred = 0
medium_scorers_pred = 0
high_scorers_pred = 0

percentil25 = np.percentile(y_test,25)
percentil75 = np.percentile(y_test,75)

y_len = len(y_test)

#PREDICTIONS' PERCENTILES CALCULATION
i1 = 0

d1 = [None] * y_len

while i1<y_len:
    if y_pred[i1] <= percentil25:
     d1[i1] = 0
    elif y_pred[i1] <= percentil75:
     d1[i1] = 1
    else:
     d1[i1] = 2
    i1+=1

#TEST'S PERCENTILES CALCULATION
i2 = 0

d2 = [None] * y_len

while i2<y_len:
    if y_test[i2] <= percentil25:
        d2[i2] = 0
    elif y_test[i2] <= percentil75:
        d2[i2] = 1
    else:
        d2[i2] = 2
    i2+=1
```

```
#STATISTICS CALCULATIONS
cf_matrix = confusion_matrix(d1, d2)

score = metrics.accuracy_score(d2, d1)
print(score)

rms = mean_squared_error(d2, d1, squared=False)
print(rms)

#PLOTS
plt.scatter(y_test, y_pred)
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()


sns.heatmap(cf_matrix, annot=True, cmap='Blues')
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()
```

Linear Regression for all features

```
#IMPORTS
import pandas as pd
import glob
from sklearn.linear_model import LinearRegression
from sklearn import preprocessing
from matplotlib import pyplot as plt
import numpy as np
from sklearn.metrics import  confusion_matrix
from sklearn.metrics import mean_squared_error
from sklearn import metrics
import seaborn as sns

#PERSONALITY DATA 2019
#Obtaining a list of the ids
dataunfiltered_2019 = pd.read_csv(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\ids_equivalence.csv')
data1_2019 = dataunfiltered_2019[dataunfiltered_2019['genero']==1]
#Women's ids
idslist = data1_2019['id_rodrigo'].to_list()

#Obtaining personality data from experts
data2_2019 = pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\personalidad_2019.xlsx')
list1_as_set_2019 = set(data1_2019['id_rodrigo'])
intersection_2019 =
list1_as_set_2019.intersection(data2_2019['id_rodrigo'])
ids_2019 = list(intersection_2019)
all_data_2019 =
data2_2019[data2_2019['id_rodrigo'].isin(ids_2019)]

#PERSONALITY FEATURES 2019
#Obtaining all audio features
featuresunfiltered_2019 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\audio_2019.xlsx')
all_features_2019 =
featuresunfiltered_2019[featuresunfiltered_2019['id_rodrigo'].isin
(ids_2019)]
```

```
#PERSONALITY FEATURES 2020
path =
r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020PersonalityDet
ection'

all_files = glob.glob(path + "/*.csv")

li = []

for filename in all_files:
    df = pd.read_csv(filename, index_col=None, header=0)
    li.append(df)

all_features_2020 = pd.concat(li, axis=0, ignore_index=True)

#PERSONALITY DATA 2020
all_data_2020 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020
Personalidad\PuntuacionesPersonalidad.xlsx')

#FILTERING
personality_treat = 'RESP'

#2019
filter_col2_2019 = [col for col in all_data_2019 if
col.startswith('Exp1_'+personality_treat)]
data_2019 =
all_data_2019[filter_col2_2019].reset_index().drop('index',
axis=1)
features_2019 = all_features_2019

#2020
features_2020 = all_features_2020

filter_col2_2020 = [col for col in all_data_2020 if
col.startswith('total'+personality_treat+'auto')]
data_2020 = all_data_2020[filter_col2_2020]

#2020 DATAFRAME CREATION
min_max_scaler = preprocessing.MinMaxScaler() #Normalization

df_2020 = pd.DataFrame(features_2020)
df_2020['total'+personality_treat+'auto'] = data_2020
df_2020 = df_2020.drop('name', axis=1)

#Normalization
df_2020_normalized = min_max_scaler.fit_transform(df_2020.values)

df_2020 = pd.DataFrame(df_2020_normalized)
```

```
#Dropping the last column to be the independent variable
df_2020 = df_2020.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2020 = df_2020.drop('personality', axis=1) * 100
y_2020 = df_2020.personality * 100

#2019 DATAFRAME CREATION

df_2019 = pd.DataFrame(features_2019)
df_2019['Exp1_'+personality_treat] = data_2019
df_2019 = df_2019.drop('name', axis=1)
df_2019 = df_2019.drop('id_rodrigo', axis=1)

#Normalization
df_2019_normalized = min_max_scaler.fit_transform(df_2019.values)

df_2019 = pd.DataFrame(df_2019_normalized)

#Dropping the last column to be the independent variable
df_2019 = df_2019.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2019 = df_2019.drop('personality', axis=1) * 100
y_2019 = df_2019.personality * 100

#SPLIT INTO TRAINING MODEL

x_train = x_2019.fillna(0).astype('int')
x_test = x_2020.fillna(0).astype('int')
y_train = y_2019.fillna(0).astype('int')
y_test = y_2020.fillna(0).astype('int')

#LINEAR REGRESSION

lr_Model = LinearRegression()
lr_Model.fit(x_train, y_train)

y_pred = lr_Model.predict(x_test)
```

```
#PERCENTIL DECLARATION

low_scorers_real = 0
medium_scorers_real = 0
high_scorers_real = 0

low_scorers_pred = 0
medium_scorers_pred = 0
high_scorers_pred = 0

percentil25 = np.percentile(y_test,25)
percentil75 = np.percentile(y_test,75)

y_len = len(y_test)

#PREDICTIONS' PERCENTILES CALCULATION
i1 = 0

d1 = [None] * y_len

while i1<y_len:
    if y_pred[i1] <= percentil25:
     d1[i1] = 0
    elif y_pred[i1] <= percentil75:
     d1[i1] = 1
    else:
     d1[i1] = 2
    i1+=1

#TEST' PERCENTILES CALCULATION
i2 = 0

d2 = [None] * y_len

while i2<y_len:
    if y_test[i2] <= percentil25:
        d2[i2] = 0
    elif y_test[i2] <= percentil75:
        d2[i2] = 1
    else:
        d2[i2] = 2
    i2+=1

#STATISTICS CALCULATIONS
cf_matrix = confusion_matrix(d1, d2)

score = metrics.accuracy_score(d2, d1)
print(score)
```

```
rms = mean_squared_error(d2, d1, squared=False)
print(rms)

#PLOTS
plt.scatter(y_test, y_pred)
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()


sns.heatmap(cf_matrix, annot=True, cmap='Blues')
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()
```

Logistic Regression for all features

```
#IMPORTS
import pandas as pd
import glob
from sklearn.linear_model import LogisticRegression
from sklearn import preprocessing
from matplotlib import pyplot as plt
import numpy as np
from sklearn.metrics import  confusion_matrix
from sklearn.metrics import mean_squared_error
from sklearn import metrics
import seaborn as sns

#PERSONALITY DATA 2019
#Obtaining a list of the ids
dataunfiltered_2019 = pd.read_csv(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\ids_equivalence.csv')
data1_2019 = dataunfiltered_2019[dataunfiltered_2019['genero']==1]
#Women's ids
idslist = data1_2019['id_rodrigo'].to_list()

#Obtaining personality data from experts
data2_2019 = pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\personalidad_2019.xlsx')
list1_as_set_2019 = set(data1_2019['id_rodrigo'])
intersection_2019 =
list1_as_set_2019.intersection(data2_2019['id_rodrigo'])
ids_2019 = list(intersection_2019)
all_data_2019 =
data2_2019[data2_2019['id_rodrigo'].isin(ids_2019)]

#PERSONALITY FEATURES 2019
#Obtaining all audio features
featuresunfiltered_2019 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\audio_2019.xlsx')
all_features_2019 =
featuresunfiltered_2019[featuresunfiltered_2019['id_rodrigo'].isin
(ids_2019)]
```

```
#PERSONALITY FEATURES 2020
path =
r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020PersonalityDet
ection'

all_files = glob.glob(path + "/*.csv")

li = []

for filename in all_files:
    df = pd.read_csv(filename, index_col=None, header=0)
    li.append(df)

all_features_2020 = pd.concat(li, axis=0, ignore_index=True)

#PERSONALITY DATA 2020
all_data_2020 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020
Personalidad\PuntuacionesPersonalidad.xlsx')

#FILTERING
personality_treat = 'RESP'

#2019
filter_col2_2019 = [col for col in all_data_2019 if
col.startswith('Exp1_'+personality_treat)]
data_2019 =
all_data_2019[filter_col2_2019].reset_index().drop('index',
axis=1)
features_2019 = all_features_2019

#2020
features_2020 = all_features_2020

filter_col2_2020 = [col for col in all_data_2020 if
col.startswith('total'+personality_treat+'auto')]
data_2020 = all_data_2020[filter_col2_2020]

#2020 DATAFRAME CREATION
min_max_scaler = preprocessing.MinMaxScaler() #Normalization

df_2020 = pd.DataFrame(features_2020)
df_2020['total'+personality_treat+'auto'] = data_2020
df_2020 = df_2020.drop('name', axis=1)

#Normalization
df_2020_normalized = min_max_scaler.fit_transform(df_2020.values)

df_2020 = pd.DataFrame(df_2020_normalized)
```

```
#Dropping the last column to be the independent variable
df_2020 = df_2020.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2020 = df_2020.drop('personality', axis=1) * 100
y_2020 = df_2020.personality * 100

#2019 DATAFRAME CREATION

df_2019 = pd.DataFrame(features_2019)
df_2019['Exp1_'+personality_treat] = data_2019
df_2019 = df_2019.drop('name', axis=1)
df_2019 = df_2019.drop('id_rodrigo', axis=1)

#Normalization
df_2020_normalized = min_max_scaler.fit_transform(df_2020.values)

df_2020 = pd.DataFrame(df_2020_normalized)

#Dropping the last column to be the independent variable
df_2020 = df_2020.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2020 = df_2020.drop('personality', axis=1) * 100
y_2020 = df_2020.personality * 100

#2019 DATAFRAME CREATION

df_2019 = pd.DataFrame(features_2019)
df_2019['Exp1_CORD'] = data_2019
df_2019 = df_2019.drop('name', axis=1)
df_2019 = df_2019.drop('id_rodrigo', axis=1)

#Normalization
df_2019_normalized = min_max_scaler.fit_transform(df_2019.values)

df_2019 = pd.DataFrame(df_2019_normalized)

#Dropping the last column to be the independent variable
df_2019 = df_2019.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2019 = df_2019.drop('personality', axis=1) * 100
y_2019 = df_2019.personality * 100
```

```
#SPLIT INTO TRAINING MODEL

x_train = x_2019.fillna(0).astype('int')
x_test = x_2020.fillna(0).astype('int')
y_train = y_2019.fillna(0).astype('int')
y_test = y_2020.fillna(0).astype('int')

#LOGISTIC REGRESSION

logisticRegr = LogisticRegression(max_iter=10000000)
logisticRegr.fit(x_train, y_train)

y_pred = logisticRegr.predict(x_test)

#PERCENTIL DECLARATION

low_scorers_real = 0
medium_scorers_real = 0
high_scorers_real = 0

low_scorers_pred = 0
medium_scorers_pred = 0
high_scorers_pred = 0

percentil25 = np.percentile(y_test,25)
percentil75 = np.percentile(y_test,75)

y_len = len(y_test)

#PREDICTIONS' PERCENTILES CALCULATION
i1 = 0

d1 = [None] * y_len

while i1<y_len:
    if y_pred[i1] <= percentil25:
     d1[i1] = 0
    elif y_pred[i1] <= percentil75:
     d1[i1] = 1
    else:
     d1[i1] = 2
    i1+=1
```

```
#TEST' PERCENTILES CALCULATION
i2 = 0

d2 = [None] * y_len

while i2<y_len:
    if y_test[i2] <= percentil25:
        d2[i2] = 0
    elif y_test[i2] <= percentil75:
        d2[i2] = 1
    else:
        d2[i2] = 2
    i2+=1

#STATISTICS CALCULATIONS
cf_matrix = confusion_matrix(d1, d2)

score = metrics.accuracy_score(d2, d1)
print(score)

rms = mean_squared_error(d2, d1, squared=False)
print(rms)

#PLOTS
plt.scatter(y_test, y_pred)
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()


sns.heatmap(cf_matrix, annot=True, cmap='Blues')
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()
```

Random Forest for all features

```
#IMPORTS
import pandas as pd
import glob
from sklearn.ensemble import RandomForestClassifier
from sklearn import preprocessing
from matplotlib import pyplot as plt
import numpy as np
from sklearn.metrics import  confusion_matrix
from sklearn.metrics import mean_squared_error
from sklearn import metrics
import seaborn as sns

#PERSONALITY DATA 2019
#Obtaining a list of the ids
dataunfiltered_2019 = pd.read_csv(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\ids_equivalence.csv')
data1_2019 = dataunfiltered_2019[dataunfiltered_2019['genero']==1]
#Women's ids
idslist = data1_2019['id_rodrigo'].to_list()

#Obtaining personality data from experts
data2_2019 = pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\personalidad_2019.xlsx')
list1_as_set_2019 = set(data1_2019['id_rodrigo'])
intersection_2019 =
list1_as_set_2019.intersection(data2_2019['id_rodrigo'])
ids_2019 = list(intersection_2019)
all_data_2019 =
data2_2019[data2_2019['id_rodrigo'].isin(ids_2019)]

#PERSONALITY FEATURES 2019
#Obtaining all audio features
featuresunfiltered_2019 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\AÑO
PASADO\datos_def\audio_2019.xlsx')
all_features_2019 =
featuresunfiltered_2019[featuresunfiltered_2019['id_rodrigo'].isin
(ids_2019)]
```

```
#PERSONALITY FEATURES 2020
path =
r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020PersonalityDet
ection'

all_files = glob.glob(path + "/*.csv")

li = []

for filename in all_files:
    df = pd.read_csv(filename, index_col=None, header=0)
    li.append(df)

all_features_2020 = pd.concat(li, axis=0, ignore_index=True)

#PERSONALITY DATA 2020
all_data_2020 =
pd.read_excel(r'C:\Users\rmrsg\Desktop\TFG\SCRIPTS\NuevosDatos2020
Personalidad\PuntuacionesPersonalidad.xlsx')

#FILTERING
personality_treat = 'RESP'

#2019
filter_col2_2019 = [col for col in all_data_2019 if
col.startswith('Exp1_'+personality_treat)]
data_2019 =
all_data_2019[filter_col2_2019].reset_index().drop('index',
axis=1)
features_2019 = all_features_2019

#2020
features_2020 = all_features_2020

filter_col2_2020 = [col for col in all_data_2020 if
col.startswith('total'+personality_treat+'auto')]
data_2020 = all_data_2020[filter_col2_2020]

#2020 DATAFRAME CREATION
min_max_scaler = preprocessing.MinMaxScaler() #Normalization

df_2020 = pd.DataFrame(features_2020)
df_2020['total'+personality_treat+'auto'] = data_2020
df_2020 = df_2020.drop('name', axis=1)

#Normalization
df_2020_normalized = min_max_scaler.fit_transform(df_2020.values)

df_2020 = pd.DataFrame(df_2020_normalized)
```

```
#Dropping the last column to be the independent variable
df_2020 = df_2020.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2020 = df_2020.drop('personality', axis=1) * 100
y_2020 = df_2020.personality * 100

#2019 DATAFRAME CREATION

df_2019 = pd.DataFrame(features_2019)
df_2019['Exp1_'+personality_treat] = data_2019
df_2019 = df_2019.drop('name', axis=1)
df_2019 = df_2019.drop('id_rodrigo', axis=1)
#Normalization
df_2020_normalized = min_max_scaler.fit_transform(df_2020.values)

df_2020 = pd.DataFrame(df_2020_normalized)

#Dropping the last column to be the independent variable
df_2020 = df_2020.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2020 = df_2020.drop('personality', axis=1) * 100
y_2020 = df_2020.personality * 100

#2019 DATAFRAME CREATION

df_2019 = pd.DataFrame(features_2019)
df_2019['Exp1_CORD'] = data_2019
df_2019 = df_2019.drop('name', axis=1)
df_2019 = df_2019.drop('id_rodrigo', axis=1)

#Normalization
df_2019_normalized = min_max_scaler.fit_transform(df_2019.values)

df_2019 = pd.DataFrame(df_2019_normalized)

#Dropping the last column to be the independent variable
df_2019 = df_2019.rename(columns={len(df_2020.columns)-1:
'personality'})

x_2019 = df_2019.drop('personality', axis=1) * 100
y_2019 = df_2019.personality * 100
```

```
#SPLIT INTO TRAINING MODEL

x_train = x_2019.fillna(0).astype('int')
x_test = x_2020.fillna(0).astype('int')
y_train = y_2019.fillna(0).astype('int')
y_test = y_2020.fillna(0).astype('int')

#RANDOM FOREST

rF_Model = RandomForestClassifier()
rF_Model.fit(x_train, y_train)

y_pred = rF_Model.predict(x_test)

#PERCENTIL DECLARATION

low_scorers_real = 0
medium_scorers_real = 0
high_scorers_real = 0

low_scorers_pred = 0
medium_scorers_pred = 0
high_scorers_pred = 0

percentil25 = np.percentile(y_test,25)
percentil75 = np.percentile(y_test,75)

y_len = len(y_test)

#PREDICTIONS' PERCENTILES CALCULATION
i1 = 0

d1 = [None] * y_len

while i1<y_len:
    if y_pred[i1] <= percentil25:
     d1[i1] = 0
    elif y_pred[i1] <= percentil75:
     d1[i1] = 1
    else:
     d1[i1] = 2
    i1+=1
```

```
#TEST' PERCENTILES CALCULATION
i2 = 0

d2 = [None] * y_len

while i2<y_len:
    if y_test[i2] <= percentil25:
        d2[i2] = 0
    elif y_test[i2] <= percentil75:
        d2[i2] = 1
    else:
        d2[i2] = 2
    i2+=1

#STATISTICS CALCULATIONS
cf_matrix = confusion_matrix(d1, d2)

score = metrics.accuracy_score(d2, d1)
print(score)

rms = mean_squared_error(d2, d1, squared=False)
print(rms)

#PLOTS
plt.scatter(y_test, y_pred)
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()


sns.heatmap(cf_matrix, annot=True, cmap='Blues')
plt.xlabel('True Values')
plt.ylabel('Predictions')
plt.show()
```