Computational Intelligence
Measuring Parkinson's disease progression

Aglaia Elli Galata - aglaia.elli.galata@est.fib.upc.edu
Albert Rial Farràs - albert.rial@est.fib.upc.edu
Daniel Ordoñez Apraez - daniel.ordonez@est.fib.upc.edu
Karen Lliguin - karen.yadira.lliguin@est.fib.upc.edu

***Abstract*—This documents studies the performance of various regressor models on the prediction of progression of the Parkinson's disease's total and motor UPDRS metrics. The studied models are Random Forest Regressor (RFR), Support Vector Regressor (SVR), Multi Layer Perceptron networks (MLP), Gradient Boosting (GBR) and Adaptive Neuro-fuzzy Inference System (ANFIS). These models where tested on a variety of configurations, involving the use of Principal Component Analysis (PCA) and Recursive Feature Elimination (RFE) for dimensionality reduction, and the use of clustering and ensembling techniques.**

***Keywords*—** UPDRS, total and motor UPDRS, Parkinson Diseases diagnosis, PCA, SOM, Fuzzy C-Means, SVR, ANFIS, MLP, Gradient Boost, Random Forest, Ensemble

I. Introduction

Parkinson's disease (PD) is a long-term degenerative disorder of the central nervous system that mainly affects the motor system [1]. The cause of this disease is still unknown and there is no available cure. Hence, all the available treatments aim to improve the symptoms. PD affects the dopaminergic and non-dopaminergic areas of the brain [2] and is mainly seen in 1% of individuals older than 60years [3]. Considering the aging population, the number of people affected by PD is expected to double in the next twenty years [4]. PD is the second most common neurodegenerative disorder following Alzheimer's disease.

PD symptoms consist of rigidity, bradykinesia, resting tremor, and impaired postural instability. Other risk factors for PD are age, race (ethnicity), heredity, gender, and exposure to environment/toxins. Additionally, PD occurs more in males than in females. According to the epidemiological study conducted by Stephen [5], the rate of men affected by PD was 91% higher than the women.

Unified Parkinson's Disease Rating Scale (UPDRS) has been one of the most commonly used tools for a comprehensive assessment of PD [6]. Severity and progression of PD symptoms as well as symptom fluctuations are typically evaluated using *UPDRS*. *Motor-UPDRS* and *Total-UPDRS* are two important clinical scales of PD. Since PD patients exhibit characteristic vocal features, voice recordings are a useful and noninvasive tool for diagnosis. Thus, speech tests can be used for monitoring PD, since vocal impairment constitutes a common symptom and early indicator. Using an at-home recording device, such as one developed by *Intel* for PD telemonitoring, it can conveniently allow PD patients' health to be monitored remotely.

A. Parkinson Tele-monitoring dataset

In this document we will use the Parkinson Tele-monitoring Data-Set, available in the UCI Machine Learning repository, created by *Athanasios Tsanas* and *Max Little* in collaboration with 10 medical centers in the US and Intel Corporation, who developed the telemonitoring device to record the speech signals. [7] This dataset is composed of a range of biomedical voice measurements from 42 people with early-stage Parkinson's disease recruited to a six-month trial for telemonitoring the progression of their symptoms. The recordings were automatically captured in the patient's homes. The dataset contains a total of 5875 data instances with 26 numerical features that are referencing speech attributes.

Unified Parkinson's Disease Rating Scale (UPDRS) is the most widely standardized scale, which reflects the presence and severity of the symptoms of the disease. The original UPDRS scale consists of four segments: (i) assesses mentation, behavior, and mood problems, (ii) assesses patient's activities of daily living, (iii) covers motor examination and (iv) covers treatment complications. For this reason the attributes *Total-UPDRS* and *Motor-UPDRS* are suitable output variables. The ranges of Total-UPDRS and Motor-UPDRS are 0–176 (0 indicating healthy and 176 indicating total disability) and 0–108 (with 0 indicating healthy state and 108 indicating severe motor impairment), respectively.

The features present in the PD telemonitoring dataset have proven successful for the diagnosis of PD, and the prediction of its progression through the UPDRS scales. The improvement of the predictive accuracy of the PD progression has been an important and eye-catching topic and the PD telemonitoring dataset has been a key part of this research goal. [8][9]

This documents presents an approach to compare the performance of different classical machine learning and data prepossessing techniques. Additionally, we study the performance of the previously mentioned ensemble techniques on multiple models and try to conclude over their applicability to this newly proposed architecture. More specifically we try to elucidate the effects of the use of dimensionality reduction, clustering plus ensembling, and regression models of different natures applied to the PD telemonitoring dataset.

Document structure

The document is organized as follows: Section II presents related work. Section III provides the research methodology

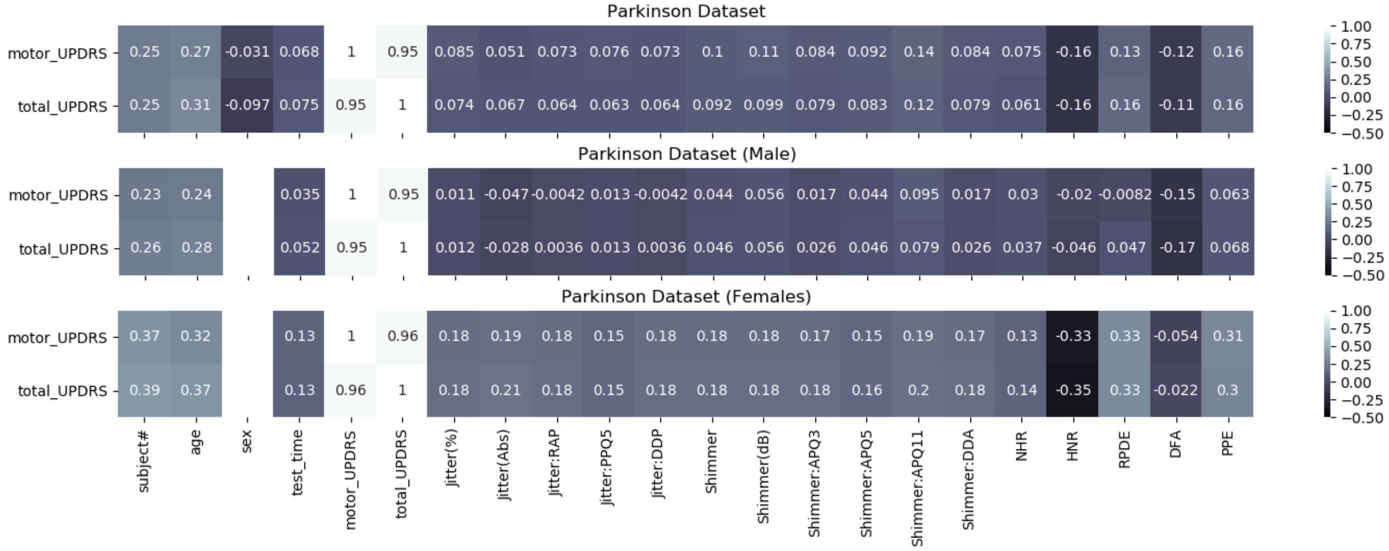


Figure 1: Parkinson Telemonitoring Dataset features correlation w.r.t Total and Motor UPDRS for all subjects, only males, and only female subjects

along with all experiments performed with the proposed models. Section IV presents the results and finally, conclusions and a description on future work is provided in the Section V.

II. Related Work

Several known approaches use non-invasive speech recordings to draw conclusions on Parkinson's Disease. The most distinguished applications with these features refer to the diagnosis of PD and the prediction of its progression.

- Diagnosing PD:
  The diagnose of the disease entails the classification of a patient as healthy or non-healthy. For this endeavor, *R.Das* showed in his study that Neural Network approaches outperform other classifiers models by obtaining an accuracy equal to 92.9%. [10] Then, in 2013 *Zuo et al* developed a diagnosis system, *PSO–FKNN*, based on *Particle Swarm Optimization* (PSO) enhanced with *Fuzzy k-Nearest Neighbor*, which achieved 97.47% accuracy. Several other approaches have been developed, [11] [12][13] but the performance of the models on this task has been already established, and currently, the literature is in search of other sources of data to address similar of more complex tasks involving the PD.

- Predicting the stage/scale of the PD in a subject:
  The other application area of the non-invasive speech recordings features is the prediction of the progression/stage of the PD. To address this regression task, several approaches have been developed for the exploitation of the dataset's underlying information.
  First of all, *Tsanas* approached the problem with some classical linear techniques: least squares *(LS)*, iteratively re-weighted least squares (IRLS), and Lasso.[7] . He compared these models with a classification and regression tree (CART) algorithm, which outperformed all the previous linear experiments since CART achieved the smallest prediction error.
  Moreover *Tsanas* explored a nonlinear random forests *(RFs)* algorithm with exhaustive signal prepossessing over the data. Particularly, he performed regression analysis, employing a least absolute shrinkage and selection operator *(Lasso)*, to achieve the final feature selection. From this study, the best motor and total UDPRS obtained through the RF algorithm.[14]
  Furthermore, *Eskidere* investigated the performance of Support Vector Machines (SVM), General Regression Neural Networks (GRNN), Multilayer Perceptron Neural Networks (MLPNN) and Least Square Support Vector Machine (LS-SVM) on the PD telemonitoring dataset.[15] The results demonstrated that LS-SVM achieved better prediction accuracy compared to the other methods.
  Finally, a state-of-the-art research that proposed by *Nilashi et. al.* used supervised and unsupervised learning, building an ensembling prediction model. [16][17] His study on 2018 excelled comparing to all the previous approaches. In his work, the dataset is clusterized into a predefined number of groups through Expectation maximization (EM) algorithm or Self Organizing Maps (SOM). Sequentially, each of the clusters reduces the dimensions of the original dataset through a Factor Analysis method. Hence, several parallel regressor models (Support Vector Regression (SVR) and Adaptive neuro-fuzzy inference system (ANFIS)) are employed to obtain a better prediction of the total and motor UPDRS, by ensembling learning.

III. Experimentation Setup

This section presents the experimentation framework that was designed and implemented for the project. In subsection III-B, we reason about the features nature, statistical properties and relevance. Next, on subsection III-C, the employed regression models and the assumptions taken for their implementation are explained. Then, in subsection III-D we analyze the clustering methods that were used and in subsection III-A we describe the different experiments that were performed for each one of the models.

As previously mentioned, this document will compare the performance of several regression models, being applied either directly to the original PD telemonitoring dataset, or to a projected version of it upon a low-dimensional vector space (by the use of Principal Component Analysis (PCA)) or in an ensemble architecture which engages a new unsupervised learning layer. Furthermore, the performance of the Recursive Feature Elimination (RFE) method for several algorithms is also tested and compared. Each one of these experiments is described below.

A. Experiments

All the experiments described in this section use 5-fold cross validation, and the Mean Absolute Error (MAE) performance metric. Additionally all the features were scaled in order to have a normal distribution centered at 0 (remove mean value) and a standard deviation equal to 1 (divide by feature variance).

1) **Models with entire PD dataset and PCA projections**
   These experiments aim to test the performance of *GBR*, *MLP*, *SVR*, and *RFR* algorithms when applied directly on the PD telemonitoring dataset, or a projected version of it obtained through PCA. For this reason, a cross-validated hyper-parameter search was performed for each of the regression models, having as input either the original dataset (without any dimensionality reduction) or a projected low-dimensional version of it, which was obtained through PCA (several numbers of Principal Components retained were tested for each model).

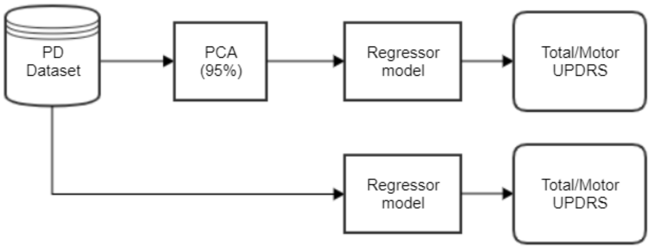


Figure 2: Experiment No. 1 structure. Regressor models with original dataset or PCA projected versions of it

   The best hyper-parameters obtained for each number of components and each model are used for the subsequent experiments and their comparison. Details on the parameters' grid search can be found on section III-C.

2) **Recursive Feature Elimination (RFE)**
   This feature selection method builds a model (in this case a regressor one) with the entire set of available features, and compute their individual relevance inside the model. Then, the least important one is removed, and the model re-build, in order to evaluate its performance without the filtered features.
   In this document, RFE was applied to the *GBR* and *RFR* models aiming the extraction of an optimal subset of features that allows the models to improve their prediction accuracy for the UPDRS metrics on all subjects, and on males and females apart.

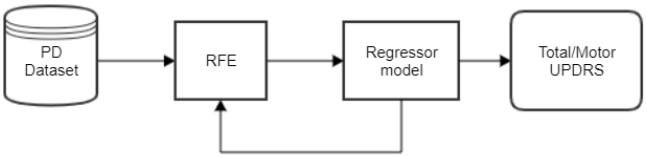


Figure 3: Experiment No. 2 structure. Regressor models with subset of features from original dataset obtained through recursive feature elimination

3) **Ensembling Architecture Experiment**
   In this experiment, the 3 clustering algorithms were performed upon the original dataset. We used the PCA reduction technique to extract the projected dataset features of each cluster. This technique tries to mimic the proposed architecture in [17] and [16], where the original dataset is clustered to a defined number of groups with low with-in variance. Subsequently, these groups are used to obtain a variate and meaningful set of dataset projections to different low-dimensional spaces by projecting the entire dataset with the sets of most meaningful Principal Components of each cluster (the ones that achieve a cumulative explain 95% of the variance). Thus, if a clustering algorithm provides n output clusters, the original dataset will be projected to n different dimensional spaces, and then fed to n regressor models, whose output will be agglomerated by an ensembling technique. We chose to average the final results from each cluster in order to obtain the final score. An example of the graphical representation of this architecture is depicted in Figure 4

B. Feature Selection

Figure 5 presents a summary of the statistical characteristics of all of the features of the dataset. All features follow a normal distribution and have zero missing values and minor to none outliers that justify the use of filtering techniques [7] [12] [17]. Having this in mind we perform feature scaling using a mapping of the original ranges to a range between 0 and 1.

Additional to the features shown in Figure 5, each data instance holds information on the time when the recording was done, the subject ID, gender and age.

The task of predicting the progression of the PD symptoms can be addressed in two ways

1) Time series regression: By assuming a time dependency between all of the recordings of a single subject, a regressor model can be developed to predict the future values of Motor and Total UPDRS given the subject history.
2) Time independent regression: This approach ignores the dependency between time and subject and assumes all recordings to be independent features estimators of the Motor and Total UPDRS values at recording time.

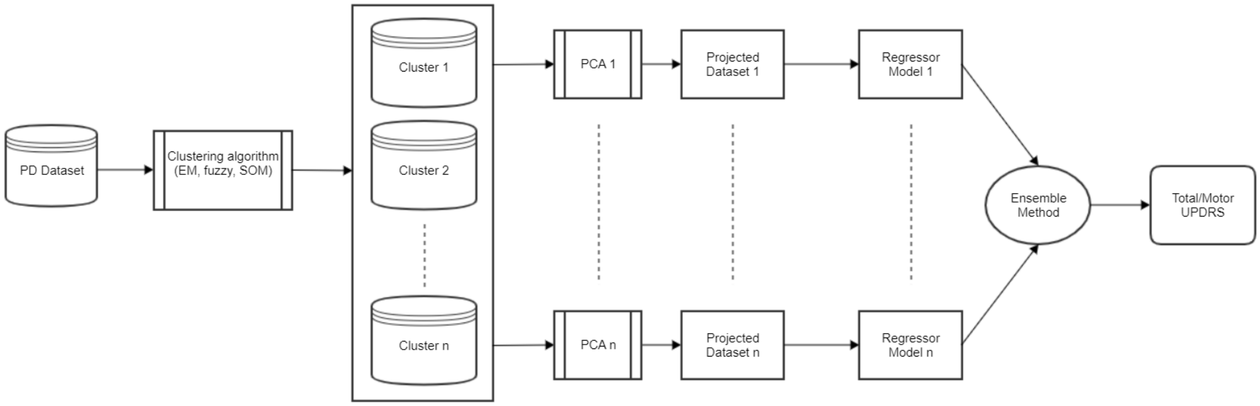
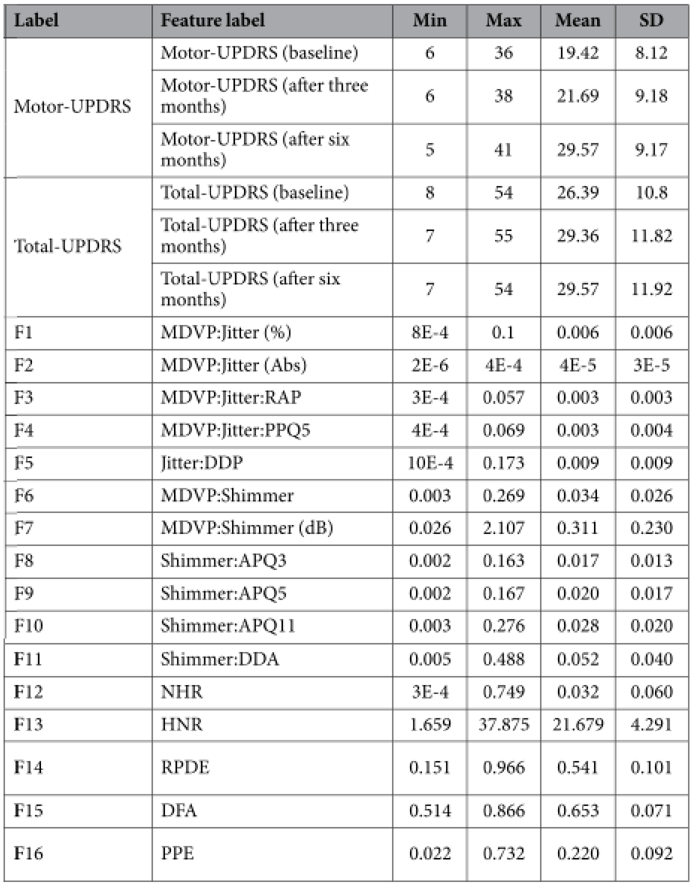


Figure 4: Experiment No. 3 structure. Dataset clustering, factor analysis, regression and ensembling.

In this document we present approaches to time-independent regression models, considering that in medical terms, such a model can prove more versatile and valuable in diagnosing different individuals at different stages of the disease. [7] For this reason, the subject ID number and time of recording will be ignored during testing of the proposed models, in light of their capability to produce overfitting to specific subjects.



Label	Feature label	Min	Max	Mean	SD
Motor-UPDRS	Motor-UPDRS (baseline)	6	36	19.42	8.12
	Motor-UPDRS (after three months)	6	38	21.69	9.18
	Motor-UPDRS (after six months)	5	41	29.57	9.17
Total-UPDRS	Total-UPDRS (baseline)	8	54	26.39	10.8
	Total-UPDRS (after three months)	7	55	29.36	11.82
	Total-UPDRS (after six months)	7	54	29.57	11.92
F1	MDVP:Jitter (%)	8E-4	0.1	0.006	0.006
F2	MDVP:Jitter (Abs)	2E-6	4E-4	4E-5	3E-5
F3	MDVP:Jitter:RAP	3E-4	0.057	0.003	0.003
F4	MDVP:Jitter:PPQ5	4E-4	0.069	0.003	0.004
F5	Jitter:DDP	10E-4	0.173	0.009	0.009
F6	MDVP:Shimmer	0.003	0.269	0.034	0.026
F7	MDVP:Shimmer (dB)	0.026	2.107	0.311	0.230
F8	Shimmer:APQ3	0.002	0.163	0.017	0.013
F9	Shimmer:APQ5	0.002	0.167	0.020	0.017
F10	Shimmer:APQ11	0.003	0.276	0.028	0.020
F11	Shimmer:DDA	0.005	0.488	0.052	0.040
F12	NHR	3E-4	0.749	0.032	0.060
F13	HNR	1.659	37.875	21.679	4.291
F14	RPDE	0.151	0.966	0.541	0.101
F15	DFA	0.514	0.866	0.653	0.071
F16	PPE	0.022	0.732	0.220	0.092

Figure 5: Dataset features statistical characteristics. Table taken from [16]

C. Regression Models

In this document, we tested the performance of different regressor models using a wide range of data pre-processing techniques. The selected models are:

1) **Gradient Boosting Regressor** (GBR): Also known as stochastic gradient regression trees, is an algorithm proposed in the early 2000s that constructs additive regression models by sequentially fitting a simple parameterized function (base learner) to current "pseudo"-residuals by least squares at each iteration. The pseudo-residuals are the gradient of the loss functional being minimized, with respect to the model values at each training data point evaluated at the current step. At each iteration, a subsample of the training data is drawn at random (without replacement) from the full training data set. This randomly selected subsample is then used in place of the full sample to fit the base learner and compute the model update for the current iteration. This randomized approach also increases robustness against overcapacity/overfitting of the base learner. [18] This algorithm has two main hyperparameters, the learning rate (lr), which defines the amount of influence each new tree node has on the overall prediction value, and the maximal depth (md) of the developed trees. Both parameters were selected through a 5-fold cross-validated grid search that iterated over the following ranges: $lr \in [1e^{-4}, ..., 5e^{-3}]$, $md \in [8, 4, 10, 15]$..

2) **Random Forest Regressor** (RFR): This algorithm proposed in 2001 uses a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. The generalization error for forests converges a.s. to a limit as the number of trees in the forest becomes large. [19] In a random forest, each node is split using the best among a subset of predictors randomly chosen at that node. This somewhat counterintuitive strategy turns out to perform very well compared to many other classifiers, including discriminant analysis, support vector machines, and neural networks and is robust against overfitting. [19] In addition, it is very user-friendly in

the sense that it has only two parameters (the number of variables in the random subset at each node and the number of trees in the forest), and is usually not very sensitive to their values. [20] For the application at hand, we made an exploration search for one of the hyper-parameters, which is the number of trees in the forest, also known as the number of estimators. We tested the performance of the model on different input data conditions with a number of estimators from 150 to 500.

3) **Support Vector Regressor** (SVR): The SV algorithm is a nonlinear generalization of the Generalized Portrait algorithm developed in Russia in the sixties. In SVR instead of minimizing the observed training error, the algorithm attempts to minimize the generalization error bound to achieve generalized performance. The idea of SVR is based on the computation of a linear regression function in a high dimensional feature space where the input data are mapped via a nonlinear function, also known as kernel function.[21] For the current application, we chose to use a *Radial Basis Function* kernel, following the results from [17]. Additionally, in the following experiments the SVR model was tuned by a hyper-parameter exploration of parameters C (regularization term) and γ (the rbf kernel coefficient) in the following ranges: $C \in [0.01, ..., 1000]$, $\gamma \in [8, 4, 10, 15]$.

4) **Multi Layer Perceptron** (MLP): Neural Networks architectures have proven optimal models in recent years for classification and regression tasks. These are composed of layers of fully connected units or neurons that use an aggregation function and a non-linear activation function to produce a non-linear mapping. Once there are sufficient neurons in the architecture the capability to extract deeply embedded abstract features from the dataset becomes a key point in learning very complex non-linear mappings.
In this document we perform an architecture search by trying different number of layers, number of hidden units in each layer learning rate and activation functions using the Adam optimizer for back-propagation.[22] At the architecture selected for the task at hand was a 4 hidden layer network, with $500, 400, 300, 200$ hidden units respectively. The output layer consists of 2 units, without any activation, since we want to predict both motor-UDPRS and total-UDPRS. The activation function in the hidden layers is set to *sigmoid*, whereas the loss function is the *mean square error* (MSE), since we are working on a regression problem. We also performed an extensive parameter exploration for the learning rate, which was finally set to 0.0005. Additionally, we also tried different techniques like Dropout and Early Stopping in order to avoid overfitting and increase the overall performance.

5) **Adaptive Neuro Fuzzy Inference System** (ANFIS): ANFIS is featured by a hybrid learning algorithm which consists of automatic tuning of *Sugeno-type* inference system and generation of outputs of a weighted linear combination of the consequents. The hybrid learning algorithm consists of two stages, i.e., *feedforward pass* to identify the consequent parameters by the *Least-squares Estimator* and the backward pass to update the premise parameters by the error backpropagation algorithm. We considered 3 *Gaussian* membership functions for the fuzzification aim, where the initialization is done uniformly in the scale of the input. The execution time of *ANFIS* is exponential augmented with respect to the number of features. So, the number of rules was multiple times bigger and the training procedure considering the entire dataset was not feasible based with our resources. This computational restriction forbid us to execute this model with the optimal number of features. So, in contrast to the previous algorithms, the final number of principal components that was retained on ANFIS model were equal to 4. In addition to the dimentionality reduction, the number of epochs was also set to 10 for computational efficiency.

D. Clustering Models

The clustering algorithms used in the ensemble architectures for each models are:

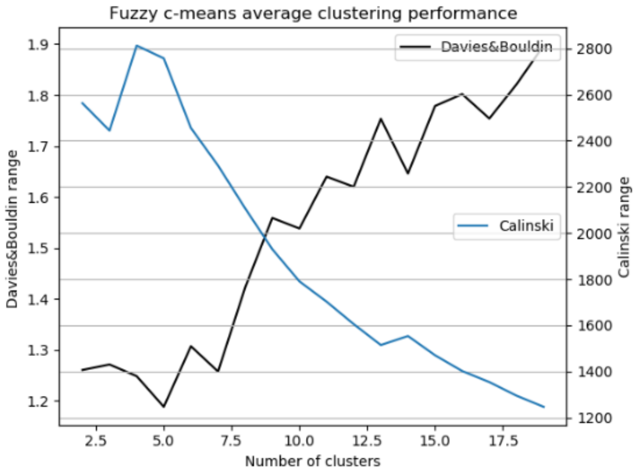


Figure 6: Davies & Bouldin and Calinski-harabasz index values on the clustering outcome of the Fuzzy-C-means algorithm applied to the PD telemononitoring dataset.

1) **Fuzzy c-Means**: This clustering algorithm generates fuzzy partitions and prototypes for any set of numerical data, instead of crisp partitions as in traditional algorithms like k-means. These partitions are useful for corroborating known substructures or suggesting substructure in unexplored data. The clustering criterion used to aggregate subsets is a generalized least-squares objective function. [23] For this application we choose the Euclidean as norm function in the objective function. In order to determine the optimal number of clusters that the algorithm generates over the PD telemonitoring dataset we use the external indexes: Davies & Bouldin and Calinski-harabasz, whose values over a different number of final clusters is depicted on Figure 6. Having in mind that by maximizing the Calinski-Harabasz and minimizing the Davies & Bouldin index the clustering will tend to have spread clusters centroids with small inner-

cluster variance, the optimal number of clusters selected for the PD telemonitoring dataset is 5.

2) **Self Organizing Maps** (SOM): *Kohonen* self-organizing feature maps are a technique designed to represent all points in a multi-dimensional source space by points in a target space, such that distance and proximity relations are preserved as much as possible.[24] The mapping is learned by a two-layer neural network that takes as input a vector belonging to the input space, and in turn produces low-dimensional, discretized representations. According to the Nilashi research [25], where different sizes of SOM were used for clustering the PD dataset, the best SOM size according to the quality of the map was equal to 9. So, this hyperparameter was used in our experiments.

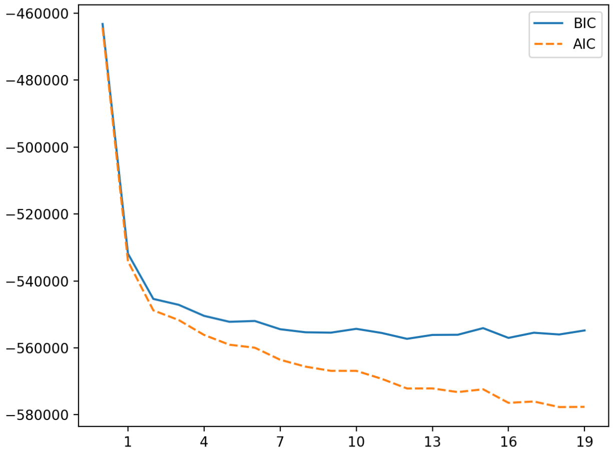


Figure 7: BIC and AIC assessment on dataset

3) **Expectation Maximization** (EM): This method, which proposed by *Dempster et al.* [26] tries to maximize the overall probability or likelihood of the parameters in statistical models.[27][28] The EM clustering algorithm computes the probabilities of cluster memberships based on one or more probability distributions. The EM iteration alternates between performing an expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the E step. The Bayesian Information Criterion constitutes an assessment metric for the optimal number of clusters data. As depicted in figure 7, the best results according to the bic information received for 16 and then 13 clusters. In order to decrease the computational cost, the final number of clusters was set according to the second choice.

E. Dimension Reduction

Whenever Principal Component Analysis was used as a dimensionality reduction technique in the experiments, the number of Principal Components retained were the ones that achieved a cumulative explained variance of 95%.

IV. Results

In this section, we are presenting the results for the experiments described in the III-A.

A. Models with entire PD dataset and PCA projections

As explained in previous sections, this experiment performs a model parameter cross-validated search, having as input-data either the original dataset with scaled features or the outcome of the PCA algorithm (with different number of final Principal Components). The results for the SVR, GBR and RFR models with different input data are depicted in Figure 8. It is clear that the RFR and GBR models perform considerably better without PCA dimensionality reduction. Furthermore, in the case of SVR, the performance of the model remains stable after 15 retain Principal Components. Therefore, we presumed that the overall efficiency-performance ratio would be improved by avoiding the use of PCA.

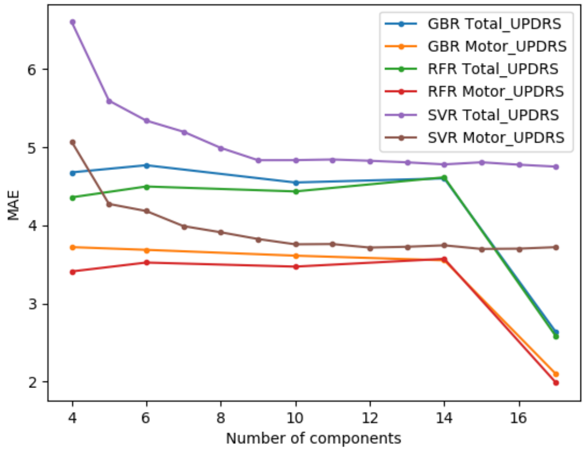


Figure 8: Results experiment No.1. Cross-validated best models testing average MAEs keeping different number of Principal Components while applying PCA. The last value to the right is the performance of the best models when not using any PCA.

Consequently, we tested the best models that got obtained by the scaled PD telemonitoring dataset without any dimensionality reduction. The cross-validated performance of the models for the entire dataset, and the gender-separated ones (only male and only female subjects) is presented on Figures 9 and 10 for total and motor UPDRS metrics respectively. It is worth noting that the best model was selected using the entire dataset as input, and then tested on the other two datasets; males and females. The results conclude that the GBR, RFR and SVR benefit form the gender partitioning of the dataset since it improves their prediction performance without jeopardizing the results with the opposite gender. Particularly, SVR is the only model that improves prediction for female subjects. Furthermore, MLP performance is greatly affected by gender partitioning, indicating that this model benefits from learning gender-independent features.

Lastly, for the ANFIS model, this exhaustive analysis was computationally infeasible. As analyzed in the III-D the execution

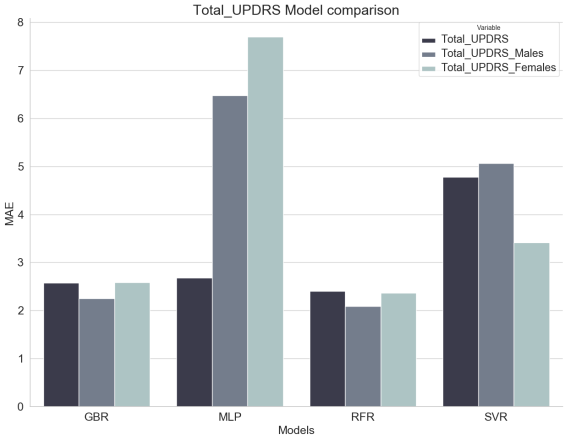


Figure 9: Results experiment No.1 Cross-validated best models testing average MAEs with the PD Telemonitoring scaled features validation average MEA for total_UDPRS

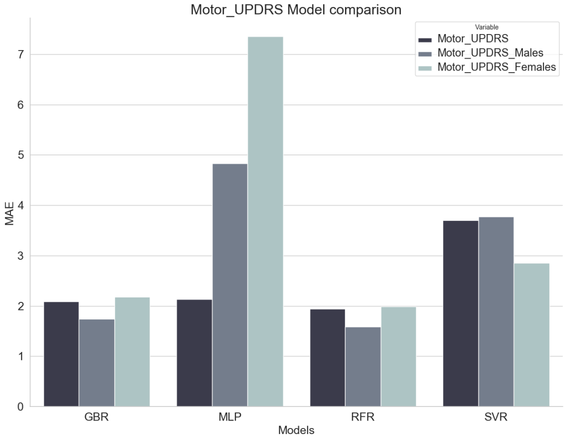


Figure 10: Results experiment No.1 Cross-validated best models testing average MAEs with the PD Telemonitoring scaled features validation average MEA for motor_UDPRS

training time is exponential augmented with respect to the number of features. The first experiment was executed with a maximum number of principal components equal to 4 and a maximum number of epochs equal to 10. The cross-validation average MAE value that obtained for total and motor UPDRS was 6.56684 and 4.88109 respectively.

B. Recursive Feature Elimination experiment

In this subsection we present the results of RFE applied to the *RFR* and *GBR* models using 5-fold cross-validation, for all subjects, only male, and only female. Figures 11, 12 show the cross-validated model performance with different number of retained features, for the aforementioned models.

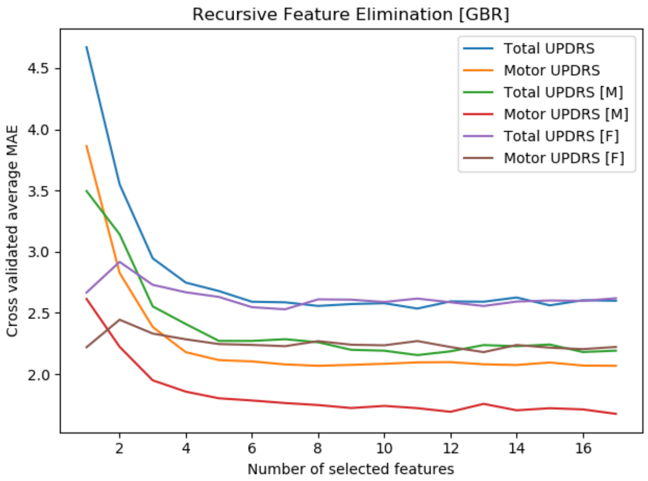


Figure 11: Results of recursive feature elimination for GBR

Furthermore, the results show that the separation by gender increases the prediction capacity of both motor and total UPDRS in the male subjects. However, in the case of female subjects the results are opposite. This behavior is present on both the regression models. This fact might confirm the hypothesis that these metrics behave differently across gender. Nevertheless, considering the small number of female subjects in the PD telemonitoring dataset, this behavior might also arise from the fact that there is not enough data for female subjects to accurately predict female total/motor-UPDRS.

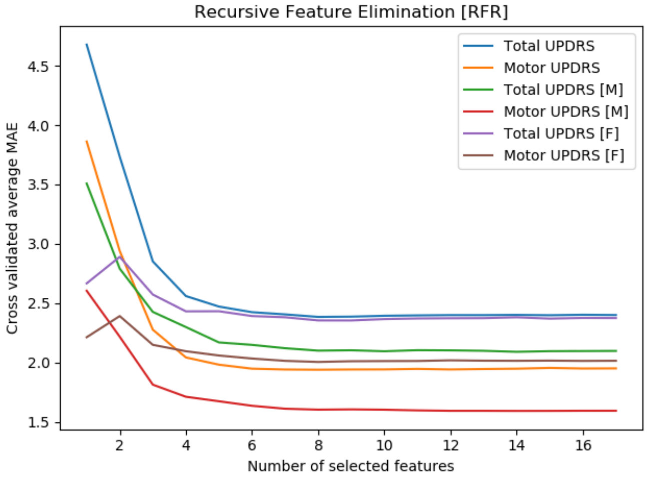
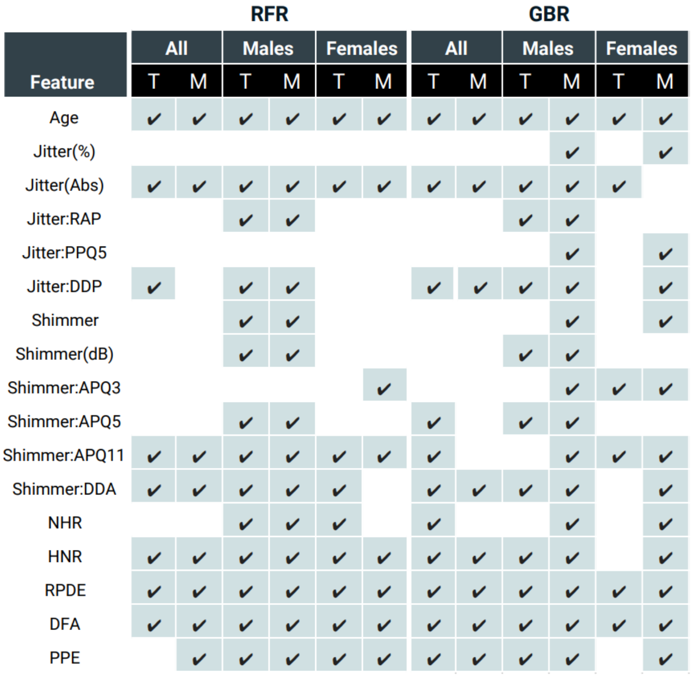


Figure 12: Results of recursive feature elimination for RFR

C. Ensembling Architecture experiment

This section presents the results of the architecture setup described in Figure 4 for all the regressor models. First, the best clustering algorithm for each of the regressor models is selected, this is done by comparing their cross-validated performance. The clustering comparison for the ANFIS, GBR, MLP, SVR and RFR models are depicted in Figures 15, 16, 17, 18 and 19 respectively, where the best clustering algorithm for each regressor model can be identified.



	RFR						GBR					
Feature	All		Males		Females		All		Males		Females	
	T	M	T	M	T	M	T	M	T	M	T	M
Age	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Jitter(%)										✓		✓
Jitter(Abs)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Jitter:RAP			✓	✓					✓	✓		
Jitter:PPQ5										✓		✓
Jitter:DDP	✓		✓	✓			✓	✓	✓	✓		✓
Shimmer			✓	✓						✓		✓
Shimmer(dB)			✓	✓					✓	✓		
Shimmer:APQ3						✓				✓	✓	✓
Shimmer:APQ5			✓	✓			✓		✓	✓		
Shimmer:APQ11	✓	✓	✓	✓	✓	✓	✓			✓	✓	✓
Shimmer:DDA	✓	✓	✓	✓	✓		✓	✓	✓	✓		✓
NHR			✓	✓	✓		✓			✓		✓
HNR	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
RPDE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
DFA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
PPE		✓	✓	✓	✓	✓	✓	✓	✓	✓		✓

Figure 13: Features selected through cross-validated RFE on GBR and RFR models. T stands for Total UPDRS and M for Motor UPDRS

Regarding the ANFIS model, this experiment was executed on a reduced dataset with less principal components than the optimal ones, due to the high computational cost. The best performance of the model achieved with EM as clustering algorithm, a behaviour that follows the results presented by [16].

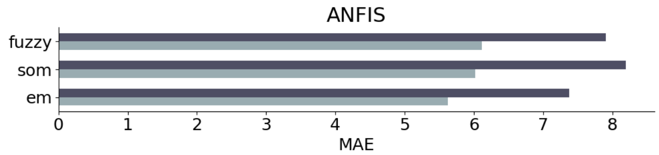


Figure 15: Total (dark) and Motor (light) UPDRS comparison of clustering algorithms for ANFIS model

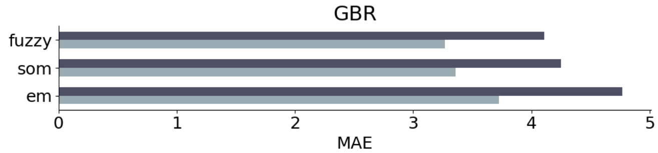


Figure 16: Total (dark) and Motor (light) UPDRS comparison of clustering algorithms for GBR model

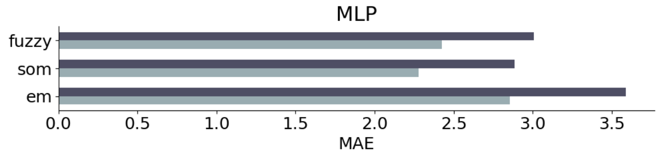


Figure 17: Total (dark) and Motor (light) UPDRS comparison of clustering algorithms for MLP model

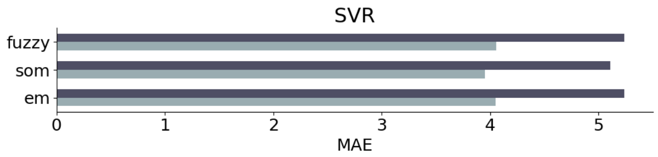


Figure 18: Total (dark) and Motor (light) UPDRS comparison of clustering algorithms for SVR model

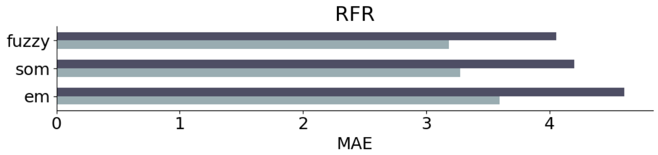


Figure 19: Total (dark) and Motor (light) UPDRS comparison of clustering algorithms for RFR model

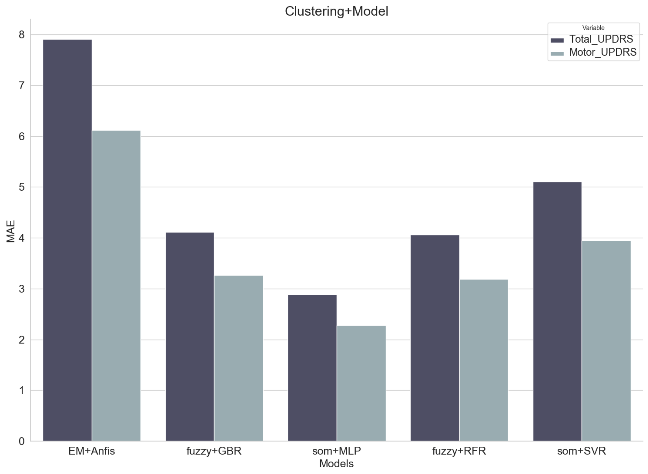


Figure 20: Experiment No.3 results. Best ensembling architectures performances per regression model

To summarize the results from experiment No. 3 the final comparison conducted between the optimal ensembling-configuration of each regression model is depicted on Figure 20. Although the performance of the ensembling architecture with the different models is close to the one obtained in experiment 1 and 2 by each of them, the results show a constant decrease in performance by the use of ensembling techniques (see Figure 14). This decline in performance, and the increase in the overall architecture complexity by integrating clustering and multiple parallel models seem to reduce the convenience of the use of such ensembling architectures.

D. Overall model comparison

Lastly, In figure 14 is presented a comparison of all of the implemented models and their varied architectures. The best model obtained was the Random Forest Regressor using the entire set of pre-processed features from the PD telemonitoring dataset for predicting motor UPDRS for all genders, and total UPDRS for male subjects. In the case of the total UPDRS prediction for all subjects of the dataset, or only female subjects, the Random Forest Regressor model using only a subset of the original dataset features proves to be the best option. (see Figure 13) To conclude, it is also observable that the MLP and GBR models perform in a similar fashion and should also be considered as possibilities for further research.

E. Dataset dimensionality reduction

As explained in experiment No. 1. on Section III, we tested the influence of applying Principal Component Analysis on the performance of the regressor models, while keeping a different amount of Principal Components (final feature dimensions). The results which are summarized in Table 14 and in Figure 8 show that the models tested (GBR, RFR and SVR) do not benefit from the use of this factor analysis technique.

F. Comparison with state-of-the-art

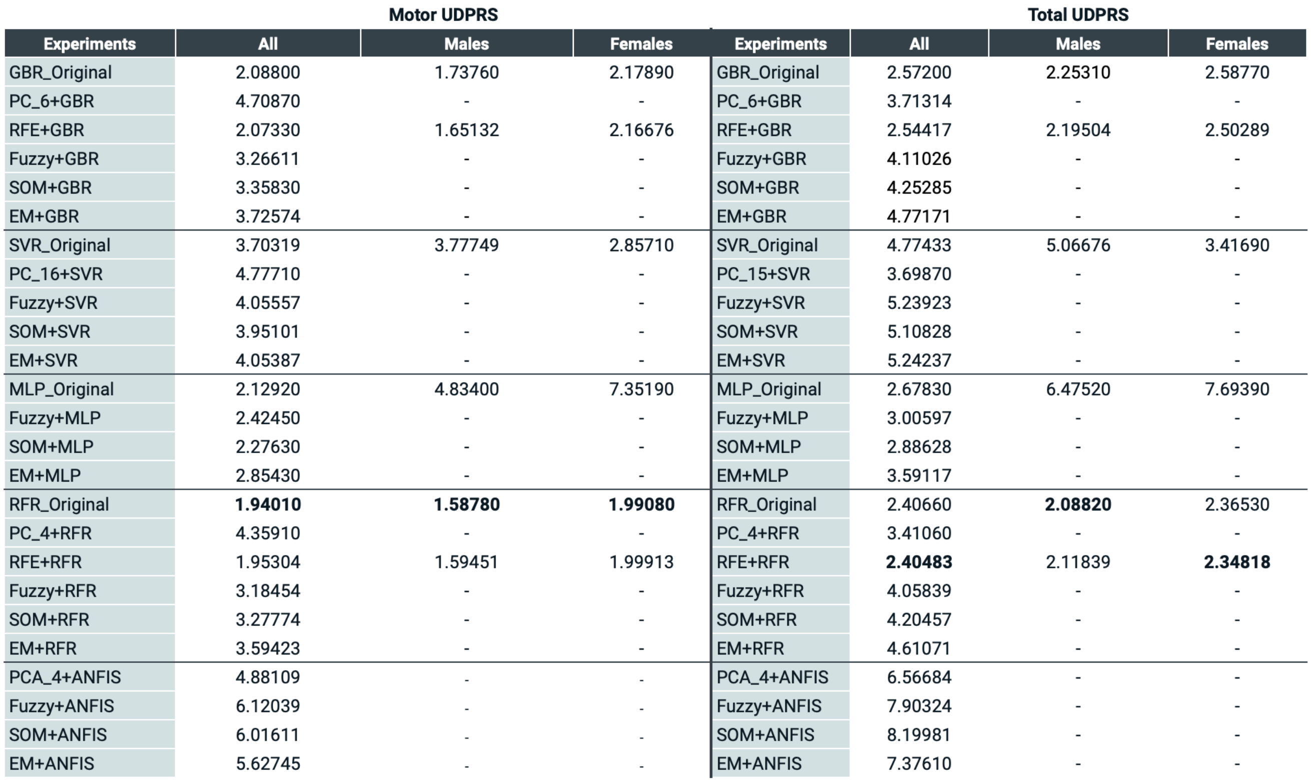
In order to evaluate the best model for PD UPDRS metrics prediction, we compared the obtained results with similar state-of-the-art studies: *Tsanas* proposed in 2010 several linear (Least squares (LS), iteratively re-weighted least squares *IRLS* and least absolute shrinkage and selection operator *Lasso*) and ensembling methods (Random Forest). [29] [14]. Furthermore, Eskidere [15] investigated the prediction of PD progression through neural networks and support vector machines. Additionally, the work from [16], which inspired us to apply the ensemble architecture presented in this document on experiment No.3 (which constitutes an extended implementation of the proposed architecture with additional non-linear models and clustering techniques) is also included.

It is clear that the non-linear models significantly outperform the linear ones. The LS, IRLS and Lasso models obtained the lowest score in both motor and total UPDRS. Additionally, although our Random Forest model achieve a highest score in the separate male dataset, their final MAE is slightly better on the female dataset. Moreover, the performance of our best model surpasses the neural network and support vector machines models proposed by Eskidere. Finally, regarding the work of [16] and [17], their models exceeds notably all the others. Regardless of the efforts invested in replicating the results of this work, we were not able to reach the proposed level of performance, even when using the same model hyper-parameters in the tested models. We believe these approaches are integrating into the algorithm features the subject ID and time of recording, which opens the possibility to overfitting predictions to individual subjects data history. As a final remark, it is worth noting that the *Tsanas* research avoid to use the subject ID and the time of recording values in their models and it relies only on non-invasive speech features.[14][30][29]

V. Conclusions

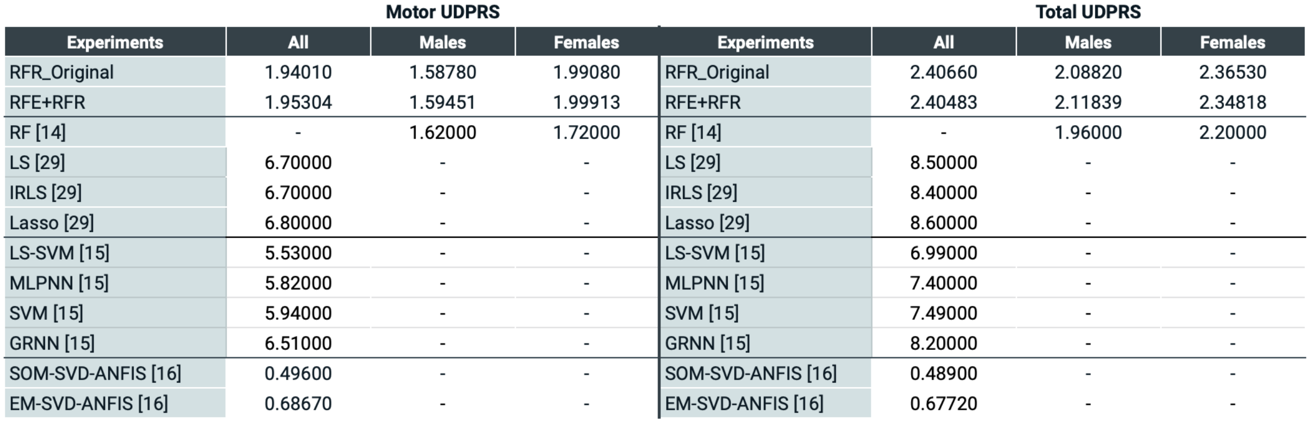
In this document, the performance of different regression models with different data pre-processing techniques and learning architectures were tested. Using combinations of dimensionality reduction, clustering, and ensembling techniques 25 different architecture configurations for 5 different regression models were tested.

Among all the models compared, Random Forest Regressor proved to be the most fitted one for this task, followed closely



	Motor UDPRS				Total UDPRS		
Experiments	**All**	**Males**	**Females**	**Experiments**	**All**	**Males**	**Females**
GBR_Original	2.08800	1.73760	2.17890	GBR_Original	2.57200	2.25310	2.58770
PC_6+GBR	4.70870	-	-	PC_6+GBR	3.71314	-	-
RFE+GBR	2.07330	1.65132	2.16676	RFE+GBR	2.54417	2.19504	2.50289
Fuzzy+GBR	3.26611	-	-	Fuzzy+GBR	4.11026	-	-
SOM+GBR	3.35830	-	-	SOM+GBR	4.25285	-	-
EM+GBR	3.72574	-	-	EM+GBR	4.77171	-	-
SVR_Original	3.70319	3.77749	2.85710	SVR_Original	4.77433	5.06676	3.41690
PC_16+SVR	4.77710	-	-	PC_15+SVR	3.69870	-	-
Fuzzy+SVR	4.05557	-	-	Fuzzy+SVR	5.23923	-	-
SOM+SVR	3.95101	-	-	SOM+SVR	5.10828	-	-
EM+SVR	4.05387	-	-	EM+SVR	5.24237	-	-
MLP_Original	2.12920	4.83400	7.35190	MLP_Original	2.67830	6.47520	7.69390
Fuzzy+MLP	2.42450	-	-	Fuzzy+MLP	3.00597	-	-
SOM+MLP	2.27630	-	-	SOM+MLP	2.88628	-	-
EM+MLP	2.85430	-	-	EM+MLP	3.59117	-	-
RFR_Original	**1.94010**	**1.58780**	**1.99080**	RFR_Original	2.40660	**2.08820**	2.36530
PC_4+RFR	4.35910	-	-	PC_4+RFR	3.41060	-	-
RFE+RFR	1.95304	1.59451	1.99913	RFE+RFR	**2.40483**	2.11839	**2.34818**
Fuzzy+RFR	3.18454	-	-	Fuzzy+RFR	4.05839	-	-
SOM+RFR	3.27774	-	-	SOM+RFR	4.20457	-	-
EM+RFR	3.59423	-	-	EM+RFR	4.61071	-	-
PCA_4+ANFIS	4.88109	-	-	PCA_4+ANFIS	6.56684	-	-
Fuzzy+ANFIS	6.12039	-	-	Fuzzy+ANFIS	7.90324	-	-
SOM+ANFIS	6.01611	-	-	SOM+ANFIS	8.19981	-	-
EM+ANFIS	5.62745	-	-	EM+ANFIS	7.37610	-	-

Figure 14: Comparison between all the models



Motor UDPRS				Total UDPRS			
Experiments	All	Males	Females	Experiments	All	Males	Females
RFR_Original	1.94010	1.58780	1.99080	RFR_Original	2.40660	2.08820	2.36530
RFE+RFR	1.95304	1.59451	1.99913	RFE+RFR	2.40483	2.11839	2.34818
RF [14]	-	1.62000	1.72000	RF [14]	-	1.96000	2.20000
LS [29]	6.70000	-	-	LS [29]	8.50000	-	-
IRLS [29]	6.70000	-	-	IRLS [29]	8.40000	-	-
Lasso [29]	6.80000	-	-	Lasso [29]	8.60000	-	-
LS-SVM [15]	5.53000	-	-	LS-SVM [15]	6.99000	-	-
MLPNN [15]	5.82000	-	-	MLPNN [15]	7.40000	-	-
SVM [15]	5.94000	-	-	SVM [15]	7.49000	-	-
GRNN [15]	6.51000	-	-	GRNN [15]	8.20000	-	-
SOM-SVD-ANFIS [16]	0.49600	-	-	SOM-SVD-ANFIS [16]	0.48900	-	-
EM-SVD-ANFIS [16]	0.68670	-	-	EM-SVD-ANFIS [16]	0.67720	-	-

Figure 21: Comparison with state-of-the-art models. RF: random Forest, LS: Least Squares, IRLS: Iteratively re-weighted Least Squares, lasso: Least Absolute Shrinkage and Selection Operator, SVD: Singular Value Decomposition, LS-SVM: Least Square Support Vector Machine, MLPNN: Multilayer Perceptron Neural Networks, GRNN: General Regression Neural Networks

by the Multi Layer Perceptron regression Neural Network and Gradient Boosting Regressor.

Moreover, the capacity for non-invasive speech recordings features to predict the state of the PD in the form of UPDRS metrics was reaffirmed. Additionally, by avoiding the use of the time feature (time of recording for each of the PD dataset instances), the regression model was forced to create a prediction based solely on current information, avoiding the need of a context or history of the patient, a characteristic that is highly desired for the application at hand. [14]

Finally, in this study we have experimented on the ensembling architecture proposed by [16][17] in order to increase the performance of single regressor models. Although this architecture was tested with different clustering algorithms and regression models, none of them outperformed the results of the traditional regression architecture. This tendency leads us to conclude that the models presented do not benefit from the use of ensembling techniques with parallel prediction sub-models. Nevertheless, this fact contradicts the results presented by these authors. In the case of the ANFIS model, the results of [16] could not be replicated due to restrictions on the computational power. For the rest of the models, we believe that performing a cross-validated hyper-parameter search for each of the sub-models assigned to each of the dataset clusters might increase the overall performance. However, this approach was restricted by the computational power required to perform it.

A. Future work

We believe that it is worthy to investigate the possible enhancements of the ensembling techniques that were used to agglomerate the predictions of the multiple sub-models in the architecture described on experiment No. 3. In this paper we averaged the predictions of all of the sub-models, as suggested by [16]. Nevertheless, more complex approaches can be tested (e.g. techniques based weighted average, boosting, bagging or stacking) using additional information provided by the clustering algorithm (e.g. likelihood for EM, or memberships for fuzzy C-means). This way the ensembling technique will learn to weight the sub-models predictions by their relevance to the specific data samples.

References

[1] A.L. Jurkiewicz K.F. Klagenberg J.M. Bassetto, B.S. Zeigelboim. *Neurotological findings in patients with Parkinson's disease*. J. Otorhinolaryngology 74, 2008.

[2] J. Van Hilten R. Dunnewold, C. Jacobi. *Quantitative assessment of Bradykinesia in patients with Parkinson's disease*. J. Neurosci. Methods 74, 1997.

[3] P.C. Marcogliese H.J. Bellen G. Lin, L. Wang. *Sphingolipids in the pathogenesis of Parkinson's disease and parkinsonism*. Trends Endocrinol. Metab, 2018.

[4] M. Munneke B.R. Bloem. *Revolutionising management of chronic disease: the ParkinsonNet approach*. BMJ 348, 2014.

[5] A.L. Bernstein R.D. Fross A. Leimpeter D.A. Bloch L.M. Nelson S.K. Van Den Eeden, C.M. Tanner. *Incidence of Parkinson's disease: variation by age, gender, and race/ethnicity*. Am. J. Epidem., 2003.

[6] W. Koller W. Poewe C. Sampaio O. Rascol, C. Goetz. *Treatment interventions for Parkinson's disease: an evidence based assessment*. Lancet 359 (9317), 2002.

[7] Athanasios Tsanas, Max A Little, Patrick E McSharry, and Lorraine O Ramig. Accurate telemonitoring of parkinson's disease progression by noninvasive speech tests. *IEEE transactions on Biomedical Engineering*, 57(4):884–893, 2009.

[8] Wrobel K. Porwik Froelich, W. *Diagnosis of parkinson's disease using speech samples and threshold-based classification*. Journal of Medical Imaging and Health Informatics 5, 2015.

[9] Ertas F. Hanilçi C Eskidere, Ö. *A comparison of regression methods for remote tracking of parkinson's disease progression*. Expert Systems with Applications 39, 2012.

[10] R. Das. *A comparison of multiple classification methods for diagnosis of Parkinson disease*. Expert Syst. Appl. 37, 2010.

[11] Kemal Polat. Classification of parkinson's disease using feature weighting method on the basis of fuzzy c-means clustering. *International Journal of Systems Science*, 43(4):597–609, 2012.

[12] Muthusamy Hariharan, Kemal Polat, and Ravindran Sindhu. A new hybrid intelligent system for accurate detection of parkinson's disease. *Computer methods and programs in biomedicine*, 113(3):904–913, 2014.

[13] S.C. Hu D.C. Li, C.W. Liu. *A fuzzy-based data transformation for feature extraction to increase classification performance with small medical data sets*. Artif. Intell. Med. 52, 2011.

[14] Athanasios Tsanas, Max A Little, Patrick E McSharry, and Lorraine O Ramig. Nonlinear speech analysis algorithms mapped to a standard metric achieve clinically useful quantification of average parkinson's disease symptom severity. *Journal of the royal society interface*, 8(59):842–855, 2010.

[15] C. Hanilçi Ö. Eskidere, F. Erta"s. *A comparison of regression methods for remote tracking of Parkinson's disease progression*. Expert Syst. Appl. 39, 2012.

[16] Mehrbakhsh Nilashi, Othman Ibrahim, and Ali Ahani. Accuracy improvement for predicting parkinson's disease progression. *Scientific reports*, 6:34181, 2016.

[17] Mehrbakhsh Nilashi, Othman Ibrahim, Hossein Ahmadi, Leila Shahmoradi, and Mohammadreza Farahmand. A hybrid intelligent system for the prediction of parkinson's disease progression using machine learning techniques. *Biocybernetics and Biomedical Engineering*, 38(1):1–15, 2018.

[18] Jerome H Friedman. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378, 2002.

[19] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[20] Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

[21] Debasish Basak, Srimanta Pal, and Dipak Chandra Patranabis. Support vector regression. *Neural Information Processing-Letters and Reviews*, 11(10):203–224, 2007.

[22] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[23] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. 1981.

[24] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. Wiley-Interscience, 2000.

[25] Sarminah Samad Hossein Ahmadi Leila Shahmoradi Elnaz Akbari Mehrbakhsh Nilashi, Othman Ibrahim. *An analytical method for measuring the Parkinson's disease progression: A case on a Parkinson's telemonitoring dataset*. 2019.

[26] D.B. Rubin A.P. Dempster, N.M. Laird. *Maximum likelihood from incomplete data via the EM algorithm*. J. royal stati. soci. Series B (methodological), 1977.

[27] D. Anderson M. Melcon M. Breteler D. Maraganore M.C. De Rijk, W. Rocca. *A population perspective on diagnostic criteria for Parkinson's disease*. Neurology 48 (5), 1977.

[28] N. Ithnin N.H. Sarmin M. Nilashi, O. bin Ibrahim. *A multi-criteria collaborative filtering recommender system for the tourism domain using Expectation Maximization (EM) and PCA-ANFIS*. Electron. Commer. Res. Appl. 14 (6), 2015.

[29] Little M. A. McSharry P. E. Ramig Tsanas, A. *L. O. 2010 Accurate telemonitoring of Parkinson's disease progression using non-invasive speech test*. s. IEEE Trans. Biomed. Eng. 57, 884 – 893. (doi:10.1109/TBME.2009. 2036000), 2010.

[30] Little M. A. McSharry P. E. Ramig Tsanas, A. *L. O. 2010 Enhanced classical dysphonia measures and sparse regression for telemonitoring of Parkinson's disease progression*. Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2010), Dallas, Texas, USA. pp. 594 – 597. IEEE Signal Processing Society, 2010.

APPENDIX A
CODE EXECUTION EXPLANATION

The proposed study is implemented in *Python* and its overall structure is illustrated below.

Under the main project folder you will find:

```
│ clustering_models
│  ├─em.py
│  └─som.py
├─dataset
├─experiments
│  ├─anfis_packages
│  ├─clustering
│  │  ├─em.py
│  │  ├─fuzzy_c_means.py
│  │  └─som.py
│  ├─hyperparameters_search
│  │  ├─search_gradient_boosting_regression.py
│  │  ├─search_multi_layer_perceptron.py
│  │  ├─search_random_forest_regression.py
│  │  └─search_support_vector_regression.py
│  ├─models_all_dataset
│  │  ├─gradient_boosting_regression.py
│  │  ├─multi_layer_perceptron.py
│  │  ├─random_forest_regression.py
│  │  └─support_vector_regression.py
│  ├─models_projected_dataset
│  │  ├─red_anfis_experiment.py
│  │  ├─red_gradient_boosting_regression.py
│  │  ├─red_multi_layer_perceptron.py
│  │  ├─red_random_forest_regression.py
│  │  └─red_support_vector_regression.py
│  ├─models_recursive_feature_elimination
│  │  ├─rfe_gradient_boosting_regression.py
│  │  └─rfe_random_forest_regression.py
│  └─reduction
│     └─pca.py
├─media--(images of the experiments)
├─results--(results of the experiments)
└─utils
   ├─comparison_best_model_cluster.py
   ├─dataset_loader.py
   ├─models_all_dataset_plot.py
   ├─models_diff_pca_components_plot.py
   └─models_projected_dataset_plot.py
```

The scripts that were executed in order to run our experiments are located under the *experiments* folder:

- hyperparameters_search: search of the hyperparameters for each model.
- models_all_dataset: regression using different models and the original dataset.
- models_projected_dataset: regression using different ensembled models and the projected datasets with Clustering + PCA.
- models_recursive_feature_elimination: regression using GBR and RFR models, with recursive feature elimination (RFE).
- reduction: inside there is the script that projects (reduces) the dataset using PCA and the different clustering algorithms.

The results of the aforementioned scripts are saved under the folder *results*. In the folder *utils*, are located the corresponding scripts for plot generation with respect to the previous results. The plots are saved in *media*. Additionally, the required external libraries are: *numpy, sklearn, skfuzzy, fuzzy-c-means, pandas, tensorflow, seaborn* and *matplotlib*.