Parkinson’s Telemonitoring

Aman Agrawal

12214309

Abstract:

Parkinson's disease (PD) is a neurodegenerative disorder that affects millions worldwide, leading to motor and cognitive impairments. Telemonitoring devices offer a promising solution for remote symptom progression monitoring. In this study, we analyse a dataset comprising biomedical voice measurements from 42 individuals in the early stages of PD, recorded over a six-month trial period using a telemonitoring device. The dataset consists of 5,875 voice recordings, with each recording corresponding to one instance. Key features include subject demographics, time intervals, motor Unified Parkinson's Disease Rating Scale (UPDRS), total UPDRS, and 16 biomedical voice measures. The primary objective is to predict motor and total UPDRS scores using the voice measures. We employ multiple machine learning algorithms to enhance prediction accuracy. This research contributes to the advancement of remote monitoring techniques for Parkinson's disease management, potentially facilitating early intervention and personalized treatment strategies.

Introduction:

Parkinson's Disease (PD) stands as an impressive challenge inside the domain of neurodegenerative disarranges, significantly affecting the lives of millions around the world. At first explained by the clever perceptions of James Parkinson in 1817, this treacherous condition shows through the continuous debasement of dopaminergic neurons arranged inside the substantia nigra of the brain. The waning dopamine generation messengers a cascade of side effects, most strikingly influencing engine capacities, counting tremors, bradykinesia, unbending nature, and postural precariousness. In any case, the vindictive reach of PD expands past engine disability, infringing upon cognitive resources, disposition direction, rest cycles, and autonomic forms as the infection progresses.

Despite about two centuries of investigation, the ethology of Parkinson's infection [1] remains baffling, covered in a complex transaction of hereditary inclinations and natural impacts. This tricky nature has foiled endeavours at finding a authoritative remedy, consigning helpful intercessions to indication administration methodologies. Current treatment modalities envelop a range extending from pharmaceutical specialists to rehabilitative treatments, and in select cases, surgical mediations such as profound brain incitement. However, these measures frequently give as it were temporal help, underscoring the basic for proceeded inquire about endeavours to unwind the disease's fundamental components and plan more useful restorative interventions.

In this scene of persistent interest for arrangements, the development of machine learning (ML) presents a guide of trust. ML calculations, invested with the capacity to filter through tremendous troves of information with unparalleled proficiency, have revolutionized the scene of Parkinson's malady investigate and administration. By scrutinizing multifaceted datasets comprising neuroimaging checks, hereditary profiles, and clinical records, ML calculations reveal perplexing designs and relationships that escape human comprehension. Through this focal point, ML offers a worldview move

in early determination, forecast, and fitting personalized treatment regimens for Parkinson's malady patients.

One of the most promising applications of ML lies in the domain of neuroimaging investigation. By leveraging advanced calculations, analysts can perceive inconspicuous basic and utilitarian distortions in the brain characteristic of Parkinson's infection movement, encouraging convenient mediations to hinder weakening signs. In addition, ML-driven prescient models hold the potential to forecast infection directions with phenomenal exactness, enabling clinicians to expect indication advancement and optimize restorative interventions.

Furthermore, ML-powered examinations of hereditary information reveal until now clouded hereditary inclinations and biomarkers supporting Parkinson's infection vulnerability and movement. This molecular-level knowledge not as it were unwinding the diseases.

In tackling the transformative potential of machine learning, analysts and clinicians stand balanced to overcome the imposing challenges postured by Parkinson's malady. By unravelling its cryptic pathogenesis, formulating focused on helpful intercessions, and introducing in a period of exactness pharmaceutical, the integration of machine learning guarantees to rethink the direction of Parkinson's illness administration, cultivating moved forward results and upgraded quality of life for those harrowed.

Parkinson's malady (PD) postures an impressive challenge to both therapeutic analysts and clinicians due to its complex ethology and dynamic nature. In later a long time, the integration of machine learning (ML) calculations has developed as a promising road for upgrading our understanding of Parkinson's illness pathogenesis and progressing persistent results through personalized treatment techniques. This paper dives into the utilization of six unmistakable ML algorithms—linear relapse, bolster vector regressor (SVR), choice tree, slope boost, AdaBoost, and TensorFlow—in the setting of Parkinson's illness inquire about and administration. Through the application of progressed procedures such as framework look cross-validation (CV) for hyperparameter tuning, this ponders points to illustrate the comparative execution of these calculations in foreseeing infection movement and optimizing helpful interventions [2].

The evaluation of calculation execution in prescient modelling errands regularly depends on measurements such as mean squared error (MSE) [3], mean absolute error (MAE) [4], and the coefficient of assurance (R2 score) [5]. MSE measures the normal squared contrast between anticipated and real values, with lower values demonstrating superior show execution. Additionally, MAE evaluates the normal supreme contrast between anticipated and genuine values, giving experiences into the model's precision. R2 score, on the other hand, evaluates the extent of fluctuation in the subordinate variable that is clarified by the free factors, with higher values characteristic of superior prescient power.

Upon assessment of the execution measurements, it gets to be apparent that the calculations show changing degrees of viability in capturing the complex elements of Parkinson's infection movement. Direct relapse, whereas serving as a principal standard show, may need the complexity to enough capture nonlinear connections inalienable in infection pathogenesis. Support vector regressor (SVR) [6], famous for its capacity to handle high-dimensional information and nonlinear connections, illustrates progressed execution compared to direct relapse. Through the utilization of Grid Search CV [7] for hyperparameter tuning, SVR [8] accomplishes a lessening in MSE and an upgrade in R2 score, meaning its potential as a strong prescient modelling device for Parkinson's disease.

Decision tree [9] calculations, characterized by their various levelled structure of double choice hubs, offer interpretability and ease of usage. Be that as it may, their execution may be helpless to overfitting, especially in datasets with complex connections. By utilizing Grid Search CV for hyperparameter optimization, the choice tree calculation shows a striking advancement in prescient exactness, as prove by diminished MSE and improved R2 score. This highlights the significance of fine-tuning calculation parameters to moderate overfitting and upgrade generalizability.

In differentiate, outfit learning procedures such as Gradient Boost [10] and AdaBoost [11] saddle the collective intelligence of different frail learners to reinforce prescient execution. Angle boost iteratively develops a arrangement of choice trees, with each consequent tree cantering on minimizing the mistakes of its forerunners. Additionally, AdaBoost relegates weights to preparing occasions based on their classification precision, subsequently emphasizing difficult-to-classify occasions in ensuing cycles. Through the iterative refinement of demonstrate forecasts, both angle boost and AdaBoost accomplish commendable decreases in MSE and advancements in R2 score, underscoring the viability of gathering learning in capturing nuanced designs in Parkinson's infection progression.

Furthermore, the approach of profound learning systems such as TensorFlow [12] messengers an unused period in Parkinson's infection investigate, advertising exceptional capabilities in preparing complex, high-

dimensional information. By leveraging neural arrange models, TensorFlow empowers the extraction of perplexing highlights from different datasets, encouraging improved prescient modelling and unthinking bits of knowledge into infection pathophysiology. Whereas profound learning approaches may involve more prominent computational complexity and asset necessities, their potential to reveal novel biomarkers and restorative targets holds monstrous guarantee for progressing Parkinson's illness investigate and personalized pharmaceutical initiatives.

In rundown, the integration of machine learning calculations in Parkinson's illness investigates and administration speaks to a transformative worldview move, advertising experiences into illness pathogenesis, forecast, and restorative optimization. Through the efficient assessment of calculation execution measurements and the utilization of progressed strategies such as hyperparameter tuning, this consider illustrates the comparative viability of different ML approaches in capturing the multifaceted signs of Parkinson's illness. By saddling the collective control of machine learning, analysts and clinicians are balanced to revolutionize our understanding and administration of Parkinson's infection, eventually upgrading persistent results and quality of life.

In expansion to investigating the comparative execution of different machine learning calculations in Parkinson's illness investigate and administration, this consider draws upon the important asset of the "Parkinson's Telemonitoring Dataset [13]." This dataset, a foundation of Parkinson's malady inquire about, envelops longitudinal clinical information collected from people with Parkinson's illness experiencing telemonitoring evaluations. These evaluations incorporate a different cluster of engine and non-motor indications, captured through wearable sensors, patient-reported results, and clinical assessments.

The Parkinson's Telemonitoring Dataset gives a wealthy store of data, permitting analysts to track malady movement, evaluate treatment viability, and recognize novel biomarkers demonstrative of illness seriousness and forecast. By leveraging this comprehensive dataset, analysts can prepare and approve machine learning calculations to foresee infection directions, optimize treatment regimens, and eventually make strides understanding outcomes.

Through the integration of machine learning calculations with the experiences gathered from the Parkinson's Telemonitoring Dataset [14], this ponders points to explain the complicated flow of Parkinson's illness movement and educate personalized treatment procedures custom fitted to person understanding profiles. By tackling the collective control of progressed analytics and wealthy clinical information, analysts and clinicians stand balanced to usher in a modern time of accuracy medication in Parkinson's illness administration, eventually improving the quality of life for people living with this challenging condition.

This think about speaks to a comprehensive investigation into the application of six particular machine learning algorithms—linear relapse, back vector regressor (SVR), choice tree, angle boost, AdaBoost, and TensorFlow—in the setting of Parkinson's illness investigate and administration. By leveraging the riches of longitudinal clinical information given by the Parkinson's Telemonitoring Dataset, analysts point to unwind the complex elements of malady movement and optimize restorative interventions.

The determination of machine learning calculations is guided by their differing capabilities and appropriateness to prescient modelling assignments. Straight relapse serves as a crucial pattern demonstrate, whereas SVR offers upgraded adaptability in capturing nonlinear connections inalienable in Parkinson's infection pathogenesis. Choice tree calculations give interpretability, but with the chance of overfitting, which is moderated through hyperparameter tuning utilizing lattice look cross-validation (CV). Gathering learning methods such as slope boost and AdaBoost tackle the collective intelligence of different powerless learners to reinforce prescient execution. Finally, TensorFlow, a profound learning system, offers unparalleled capabilities in handling complex, high-dimensional information, though with expanded computational complexity.

Evaluation of calculation execution depends on a suite of measurements counting cruel squared mistake (MSE), cruel outright blunder (MAE), and the coefficient of assurance (R2 score). Through fastidious investigation, analysts reveal the comparative adequacy of each calculation in capturing the multifaceted signs of Parkinson's illness. Whereas direct relapse gives a foundational understanding, SVR illustrates made strides execution in capturing nonlinear connections. Choice tree calculations display upgraded prescient precision taking after hyperparameter optimization, highlighting the significance of fine-tuning demonstrate parameters. Gathering learning procedures such as slope boost and AdaBoost [15] offer iterative refinement of expectations, yielding commendable decreases in MSE and advancements in R2 score.

Furthermore, the integration of machine learning calculations with the bits of knowledge gathered from the Parkinson's Telemonitoring Dataset holds colossal guarantee for progressing personalized medication activities in Parkinson's illness administration. By

leveraging the wealthy clinical information accessible, analysts point to distinguish novel biomarkers characteristic of malady seriousness, forecast, and treatment reaction. Through the precise assessment of calculation execution measurements and the application of progressed methods such as hyperparameter tuning, this thinks about endeavours to optimize helpful intercessions and progress persistent outcomes.

In outline, this intrigue endeavour speaks to a significant step towards saddling the collective control of machine learning and clinical information to revolutionize Parkinson's infection inquire about and administration. By explaining the complicated elements of malady movement and illuminating personalized treatment techniques, analysts and clinicians point to upgrade the quality of life for people living with Parkinson's illness.

Literature Review:

In Paper 1, various machine learning algorithms were used to detect Parkinson's disease (PD) from offline data, achieving impressive accuracy rates. The highest accuracy was obtained using k-nearest neighbors (kNN) on voice data (98.3%) and tremor data (98.5%) separately. When ensemble averaging was performed on majority-vote from kNN, support vector machine (SVM), and naive Bayes (NB), the average accuracy for PD detection increased to 99.8%. This suggests the potential of ensemble methods in improving PD detection accuracy from different sources of data [16].

Paper 2 discusses different studies that used a range of machine learning algorithms and statistical methods to estimate Unified Parkinson's Disease Rating Scale (UPDRS) scores. Tsanas et al. achieved a mean absolute error (MAE) of 5.95 for motor-UPDRS and 7.52 for total-UPDRS, while a later study reported an improved MAE of approximately 2 UPDRS points using the same data. Other methods such as Bayesian linear regression and feedforward neural networks were employed with varying degrees of success. One study focused on predicting depression element of UPDRS from voice recordings using random forest algorithm and patient self-assessments of depression, achieving an accuracy of 0.77 [17].

Paper 3 proposes a deep neural network (DNN) model using a reduced input feature space of Parkinson's telemonitoring dataset to predict PD progression. The input feature space was optimized using principal component analysis (PCA) to reduce multicollinearity and dimensionality. The DNN model achieved mean absolute error (MAE), root mean squared error (RMSE), and coefficient of determination (R^2) values of 0.926, 1.422, and 0.970 respectively for motor-UPDRS. For total-UPDRS, the values were 1.334, 2.221, and 0.956 respectively. This model outperformed previously developed methods on the same dataset, showing the efficacy of the proposed approach in predicting UPDRS scores [18].

In Paper 4, Gaussian process classification with Automatic Relevance Determination (ARD) was highly effective in distinguishing between Parkinson's disease patients and healthy controls, achieving an accuracy of 96.92%. This method surpassed support vector machines and decision tree ensembles. Gaussian processes were also used for tracking disease progression, and ARD covariance functions were employed for efficient feature selection. This approach enabled the selection of a small subset of relevant acoustic features, resulting in better performance and lower complexity in the models [19].

Proposed Architecture:

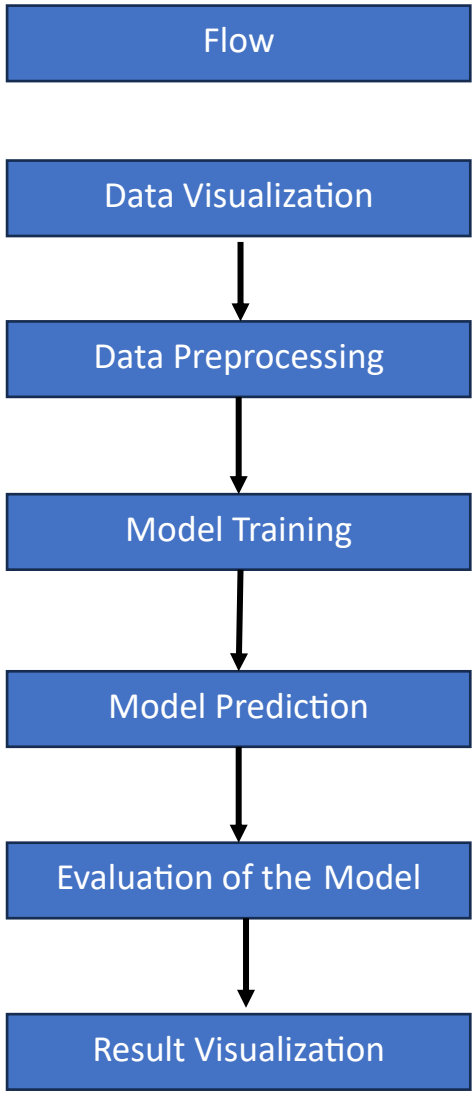
About the Dataset:

Features:

- Subject #: Unique identifier for each participant in the study.
- Age: The age of the participant at the time of assessment.
- Sex: Gender of the participant (male or female).
- Test Time: Timestamp indicating the date and time of each assessment.
- Motor UPDRS: Unified Parkinson's Disease Rating Scale (UPDRS) score specifically assessing motor symptoms.
- Total UPDRS: Overall UPDRS score, encompassing both motor and non-motor symptoms.
- Jitter (%): Measures of variation in fundamental frequency, expressed as a percentage.
- Jitter (Abs): Absolute jitter measures.
- Jitter: RAP: Jitter calculated for the relative average perturbation.
- Jitter:PPQ5: Jitter calculated using the five-point period perturbation quotient.
- Jitter:DDP: Jitter calculated as the difference between successive cycle lengths.
- Shimmer: Quantifies variation in amplitude of speech signals.
- Shimmer (dB): Shimmer measured in decibels.
- Shimmer:APQ3: Shimmer calculated using the three-point amplitude perturbation quotient.
- Shimmer:APQ5: Shimmer calculated using the five-point amplitude perturbation quotient.
- Shimmer:APQ11: Shimmer calculated using the eleven-point amplitude perturbation quotient.
- Shimmer:DDA: Shimmer calculated as the difference between successive amplitude peaks.
- NHR: Noise-to-Harmonics Ratio, quantifying the ratio of noise to harmonic components in speech signals.
- HNR: Harmonics-to-Noise Ratio, indicating the ratio of harmonic energy to noise energy in speech signals.
- RPDE: Recurrence Period Density Entropy, a measure of complexity in speech signals.
- DFA: Detrended Fluctuation Analysis, quantifying the presence of long-range correlations in speech signals.
- PPE: Pitch Period Entropy, a measure of variation in pitch period over time.

These columns encompass a wide range of motor and non-motor symptoms, speech characteristics, and demographic information, providing a comprehensive insight into the clinical profile of individuals with Parkinson's disease.

Workflow of the proposed work:



Data Visualization:

1. Heatmap of Null Values:

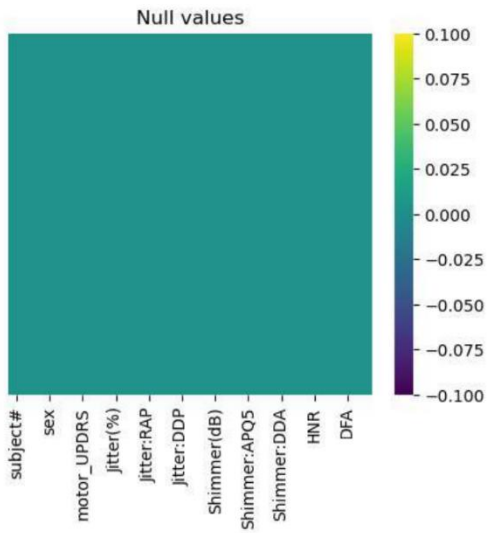


Figure 1: Heatmap of Null Values

The heatmap of Invalid values is a profitable visualization apparatus for evaluating the completeness of information inside a dataset. It presents a graphical representation where each cell compares to a column-variable combination. If a cell is coloured or shaded, it shows the nearness of lost information (Invalid values) for that specific variable. By watching the conveyance of these lost values over distinctive columns, examiners can rapidly distinguish designs and evaluate the degree of missingness in the dataset.

This visualization is especially valuable amid the introductory stages of information investigation and preprocessing. It permits analysts to prioritize which factors may require encourage examination or ascription strategies to handle lost information viably. Understanding the conveyance of lost values can to advise choices approximately whether certain factors ought to be included or avoided from ensuing investigations based on information completeness.

Furthermore, the heatmap of Invalid values helps in defining information cleaning and ascription procedures. Investigators can tailor their approach based on the watched designs of missingness, choosing fitting procedures such as cruel ascription, cancellation of lost values, or more progressed strategies like different ascription. By and large, this visualization serves as a basic device in guaranteeing the quality and astuteness of the dataset some time recently continuing with downstream analysis.

2. Histogram of motor_UPDRS:

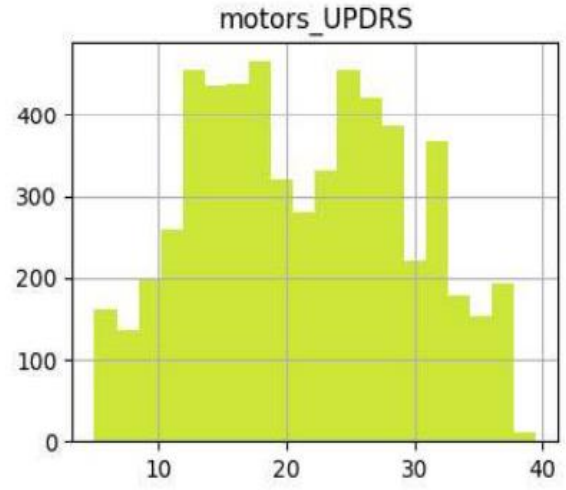


Figure 2: Histogram of motor_UPDRS

The histogram of the 'motor_UPDRS' column offers a clear visual representation of the recurrence dispersion of engine indication seriousness scores display inside the dataset. Each bar in the histogram compares to a run or container of engine UPDRS scores, whereas the stature of the bars speaks to the recurrence of event of scores falling inside each canister. This visualization empowers analysts to get a handle on key characteristics of the conveyance, counting measures of central inclination such as the cruel or middle, inconstancy, and the generally shape of the distribution.

By analysing the histogram, examiners can perceive designs in the predominance and seriousness of engine side effects among the consider members. For occurrence, a histogram with a crest or mode at a specific score esteem recommends a concentration of people with comparable engine UPDRS scores. Besides, the spread or inconstancy of scores around the central inclination gives bits of knowledge into the differing qualities of engine side effect seriousness inside the test population.

Overall, the histogram of engine UPDRS scores serves as a principal exploratory apparatus for understanding the conveyance of engine side effect seriousness in Parkinson's infection. It lays the basis for advance measurable investigation and translation of the dataset, advising ensuing inquire about headings and clinical decision-making forms.

3. Pair plot for all features:

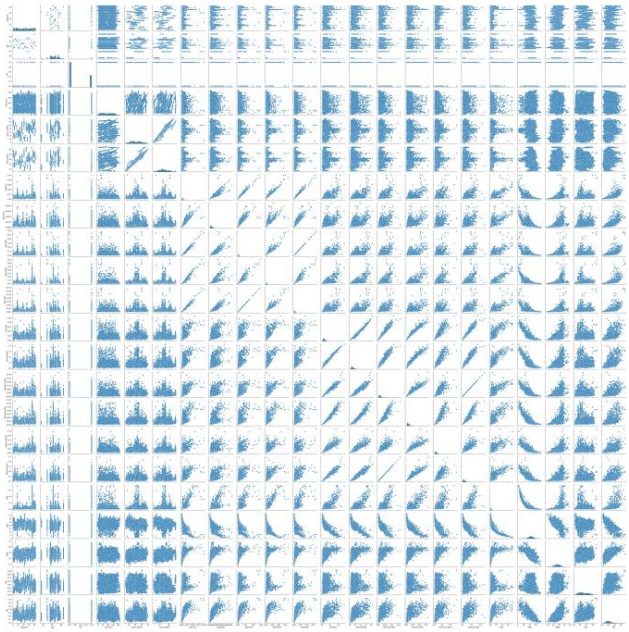


Figure 3: Pair plot for all features

A pair plot is a comprehensive visualization instrument that gives bits of knowledge into the connections between factors inside a dataset. By making a network of scatterplots and histograms, it encourages a careful examination of pairwise intuitive among all highlights. Scatterplots portray the relationship between two numerical factors, permitting examiners to watch designs such as direct affiliations, clusters, or exceptions. In the interim, histograms show the conveyance of each person variable along the corner to corner, empowering an understanding of its recurrence dissemination and shape.

Pair plots are especially beneficial for investigating multivariate connections and distinguishing complex designs over different factors at the same time. Investigators can outwardly evaluate relationships, patterns, and potential exceptions, which may not be promptly clear through univariate examination. Besides, pair plots encourage the distinguishing proof of factors with solid affiliations, directing consequent modeling or speculation testing endeavors. Generally, this visualization procedure helps in revealing covered up designs and understanding the fundamental structure of the information, subsequently advising encourage investigation and elucidation. Its instinctive format and comprehensive nature make it a profitable exploratory apparatus in information examination and visualization workflows.

4. Heatmap of correlation matrix:

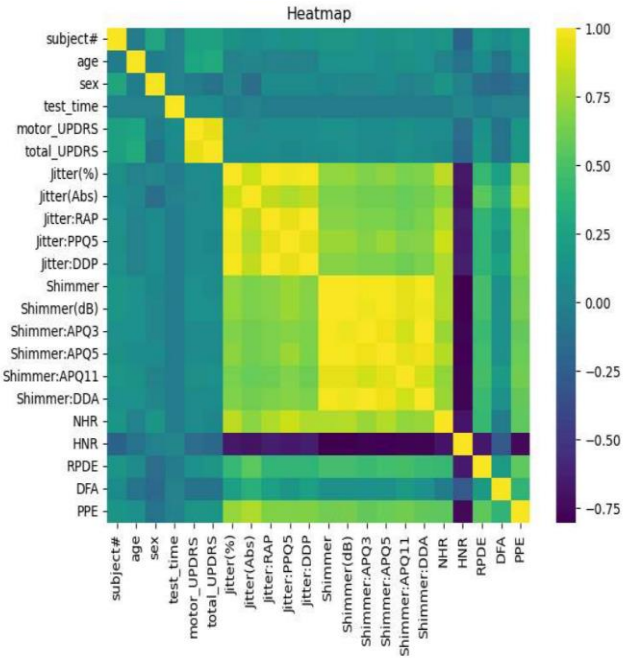


Figure 4: Heatmap of correlation matrix

A heatmap of the relationship lattice is a effective visualization instrument that offers bits of knowledge into the connections between numerical highlights in a dataset. Each cell in the heatmap speaks to the relationship coefficient between two factors, with colors showing the quality and heading of relationship. Positive relationships are ordinarily spoken to by hotter colors (e.g., shades of ruddy), whereas negative relationships are delineated by cooler colors (e.g., shades of blue). A relationship coefficient near to 1 or -1 demonstrates a solid direct relationship, whereas values near to 0 recommend powerless or no correlation.

This visualization gives a brief outline of the pairwise affiliations and conditions among factors, making a difference examiner distinguish potential designs and patterns. By outwardly reviewing the heatmap, analysts can observe which factors are unequivocally related, which may demonstrate repetition or multicollinearity. This helps in include determination by highlighting important factors for assist examination whereas barring repetitive ones. Moreover, understanding the basic structure of the information empowers analysts to make educated choices amid modelling and theory testing forms. Generally, the heatmap of the relationship framework serves as a profitable instrument for investigating and translating the complex connections inside a dataset, directing ensuing expository and modelling endeavours.

5. Pie Chart of Gender Distribution:

A Pie Chart outlining sexual orientation dissemination offers a brief and outwardly engaging portrayal of the sexual orientation composition inside a dataset. By showing the relative frequencies or extents of male and female categories, this visualization helps in understanding the statistic cosmetics of the consider populace. Such experiences are important for statistic characterization, empowering analysts to evaluate the representativeness of the test and consider potential gender-related inclinations in their analyses.

Moreover, Pie Chart give a direct and instinctive representation of categorical information, making it simple to compare sexual orientation dispersions over distinctive datasets or subgroups. This encourages subgroup investigation, permitting analysts to investigate potential varieties in results or patterns based on sex. For occasion, analysts may explore contrasts in illness predominance, side effect seriousness, or treatment reactions between guys and females.

Additionally, Pie Chart serve as viable communication devices, passing on key discoveries around sex dispersion to assorted gatherings of people in a clear and reasonable way. Whether utilized in inquire about papers, introductions, or reports, Pie Chart improve the openness and interpretability of gender-related information, cultivating educated talks and decision-making forms in different areas of consider. Generally, Pie Chart of sex dispersion play a imperative part in illustrating the statistic scene of a dataset and educating consequent examinations and translations.

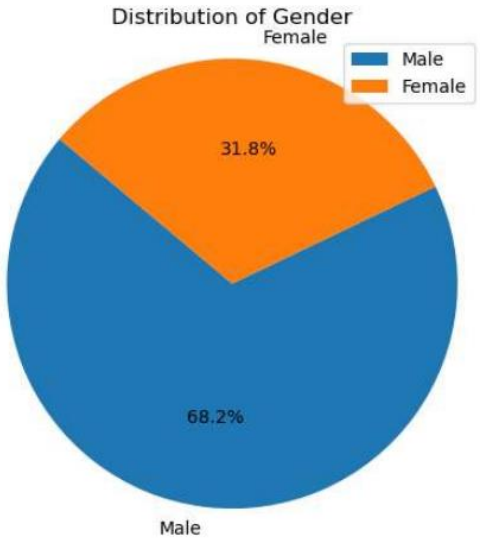


Figure 5: Pie Chart of Gender Distribution

6. Pie Chart of age distribution:

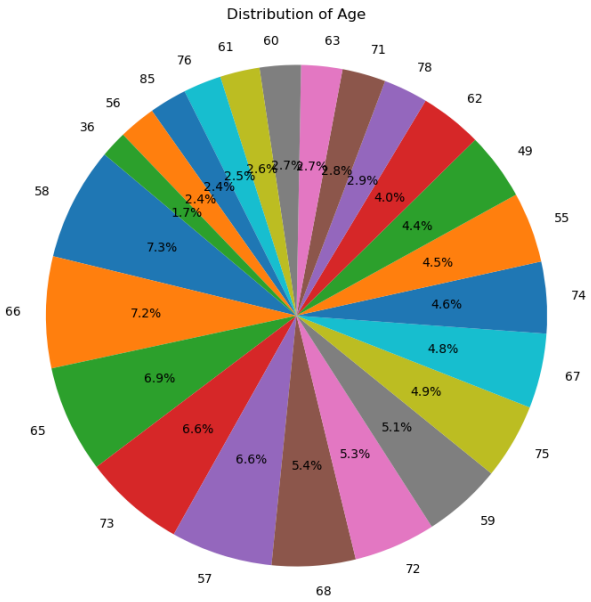


Figure 6: Pie Chart of Age Distribution

A Pie Chart outlining age dispersion gives a visual representation of the dispersion of ages among consider members inside a dataset. By displaying the recurrence or extent of people over distinctive age bunches, this visualization offers profitable experiences into the statistic characteristics of the think about population.

Understanding age socioeconomics is vital for recognizing potential age-related designs or patterns in the information. Analysts can utilize Pie Chart of age conveyance to investigate varieties in results, Pie Chart, or characteristics over distinctive age bunches. For illustration, they may examine age-related contrasts in infection predominance, indication seriousness, or treatment responses.

Moreover, Pie Chart offer a clear and instinctive delineation of age conveyances, making it simple to visualize and decipher the age composition of the dataset. Whether utilized in investigate introductions, reports, or distributions, Pie Chart upgrade the availability and clarity of age-related information, encouraging communication and comprehension among differing audiences.

Overall, Pie Chart of age dissemination play a crucial part in illustrating the age socioeconomics of a dataset and revealing potential age-related designs or patterns that may educate advance examinations or examinations.

7. Violin plot of 'Sex' vs 'motor_UPDRS':

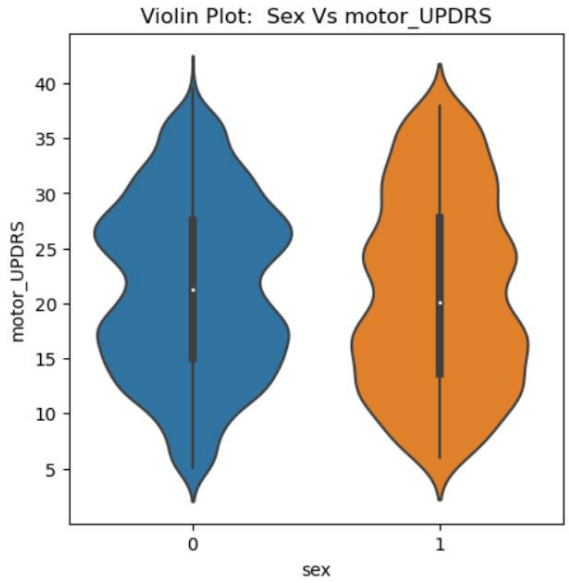


Figure 7: Violin plot of 'Sex' vs 'motor_UPDRS'

A violin plot delineating 'Sex' vs 'motor_UPDRS' gives a comprehensive visualization of the dispersion of engine UPDRS scores over distinctive sexes inside a dataset. This sort of plot combines the highlights of a box plot and a bit thickness plot, permitting for a point-by-point comparison of engine side effect seriousness between male and female participants.

By showing the dissemination of engine UPDRS scores for each sexual orientation category, violin plots offer bits of knowledge into potential contrasts in indication seriousness between guys and females. The width of the violin plot speaks to the thickness of scores at diverse levels, whereas the shape demonstrates the dissemination of scores inside each sex category.

Researchers can effectively distinguish any gender-based contrasts in engine indication seriousness by comparing the shapes and spreads of the violin plots for guys and females. This visual comparison empowers the discovery of potential designs or patterns that may warrant advance examination or subgroup analysis.

Overall, violin plots serve as capable instruments for outwardly investigating and comparing disseminations of numerical factors over diverse categorical bunches, such as sex. In the setting of engine UPDRS scores, violin plots encourage the recognizable proof of gender-based contrasts in Parkinson's illness symptomatology, contributing important experiences to clinical inquire about and hone.

8. Boxplot of 'Age' vs 'motor_UPDRS':

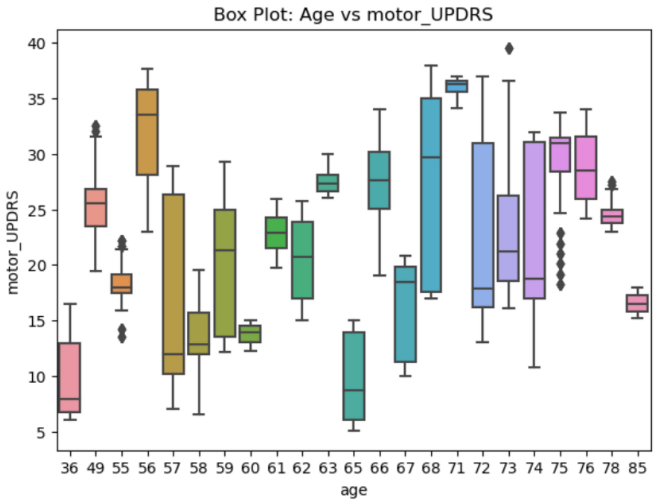


Figure 8: Boxplot of 'Age' vs 'motor_UPDRS'

A boxplot comparing 'Age' with 'motor_UPDRS' gives a brief visualization of the dispersion of engine UPDRS scores inside unmistakable age bunches in a dataset. This sort of plot successfully summarizes the central inclination, inconstancy, and potential exceptions of engine indication seriousness over diverse age categories.

By displaying the conveyance of engine UPDRS scores for each age gather, boxplots empower analysts to compare side effect seriousness levels over diverse stages of life. The central line inside each box speaks to the middle engine UPDRS score for that age gather, whereas the box itself portrays the interquartile extend (IQR), giving experiences into the spread of scores. Moreover, the hairs expand to the least and most extreme values inside 1.5 times the IQR, highlighting potential outliers.

Researchers can promptly recognize any age-related patterns or affiliations in engine side effect seriousness by looking at the boxplots for distinctive age bunches. This visual comparison helps in understanding how engine UPDRS scores may change over diverse age socioeconomics, advising assist investigations or subgroup investigations.

Overall, boxplots serve as viable apparatuses for outwardly summarizing and comparing disseminations of numerical factors over particular categories, such as age bunches. In the setting of engine UPDRS scores, boxplots encourage the distinguishing proof of potential age-related designs or affiliations, contributing

important bits of knowledge to Parkinson's illness investigate and clinical hone.

2. Data Preprocessing:

1. Separating the Input and Target features:

In the information preprocessing organize, the introductory vital step includes isolating the dataset into input highlights (X) and the target variable (y). This step sets up the establishment for ensuing investigation and modelling assignments. Leveraging the usefulness of the panda's library, I performed this isolation by making two particular DataFrames: X and y.

The DataFrames X typifies the input highlights fundamental for foreseeing the target variable, 'motor_UPDRS'. To develop X, I utilized the drop strategy given by pandas to evacuate particular columns from the unique DataFrames (df). In this case, I excluded the columns 'subject#', 'motor_UPDRS', and 'total_UPDRS', as they are not considered prescient highlights but or maybe identifiers and a excess target variable, individually. By barring these columns from X, we guarantee that as it were pertinent input highlights are held, encouraging more successful modelling and analysis.

Conversely, the DataFrames y speaks to the target variable 'motor_UPDRS', which comprises the seriousness scores of engine side effects watched in Parkinson's malady patients. This variable serves as the central point of forecast in ensuing modelling endeavours. By segregating y from the input highlights, we set up a clear qualification between the indicator factors and the target variable, subsequently guaranteeing that our modelling endeavours are suitably coordinated towards anticipating engine indication seriousness accurately.

This isolation of the dataset into X and y follows to best hones in information preprocessing, giving an organized and organized system for consequent examination. It empowers us to conduct modelling assignments productively, with a clear understanding of the input features' part in foreseeing the target variable. Furthermore, this division guarantees that the dataset is suitably organized for compatibility with different machine learning calculations and modelling strategies, setting the arrange for strong and compelling prescient modelling in Parkinson's infection research.

2. Train Test Split:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Figure 9: Train Test Split

After the starting information preprocessing step of isolating the dataset into input highlights (X) and target variable (y), the ensuing vital stage includes part the information into preparing and testing sets. This step is essential in surveying the execution and generalization capacity of machine learning models.

The train-test part is executed utilizing the **`train_test_split()`** work from the **`sklearn.preprocessing`** module, a broadly utilized library in Python for machine learning assignments. This work segments the dataset into two subsets: one for preparing the demonstrate and the other for assessing its execution. By indicating the estimate of the test set (e.g., 20% of the information), the work haphazardly isolates the dataset, guaranteeing that the dispersion of the target variable is protected in both subsets. This conservation is basic as it mirrors the real-world situation where the show must generalize well to inconspicuous data.

The preparing set is utilized to prepare the machine learning demonstrate on a parcel of the information, permitting it to learn designs and connections display in the dataset. In the interim, the testing set, comprising the remaining information focuses, is saved only for assessing the model's execution. By surveying how well the demonstrate performs on information it hasn't been prepared on, analysts can gage its capacity to generalize to unused, inconspicuous data.

This train-test part technique helps in anticipating overfitting, a common issue where the demonstrate learns the preparing information as well well but comes up short to perform palatably on modern information. By assessing the model's execution on an free testing set, analysts can make educated choices almost the model's adequacy and distinguish any potential zones for improvement.

Overall, the train-test part is a pivotal step in the machine learning pipeline, encouraging the advancement of vigorous and generalizable models that can successfully handle inconspicuous information.

3. Standard Scaling:

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

Figure 10: Standard Scaling

Taking after the division of the dataset into preparing and testing sets, the following pivotal step in information preprocessing includes standard scaling of the input highlights (X) utilizing sklearn preprocessing apparatuses. Standard scaling, moreover alluded to as z-score normalization, is a preprocessing procedure that changes the information such that each highlight has a cruel of 0 and a standard deviation of 1.

Standard scaling is fundamental for guaranteeing that all highlights are on the same scale, in this manner anticipating certain highlights from ruling the demonstrate preparing prepare due to contrasts in greatness. Without scaling, highlights with bigger values may excessively impact the model's learning prepare, driving to one-sided comes about and imperfect performance.

This normalization procedure is especially advantageous for calculations that depend on distance-based measurements or optimization procedures, such as slope plummet, as it makes strides joining and steadiness. By scaling the highlights to a standardized organize, the calculation can more successfully explore the include space and focalize towards the ideal solution.

To actualize standard scaling, the `**StandardScaler()**` work from the `**sklearn.preprocessing**` module is utilized. This work computes the cruel and standard deviation of each highlight in the preparing information and at that point scales each include in like manner. The scaled information is changed into a standardized organize reasonable for demonstrate preparing, guaranteeing that all highlights contribute similarly to the learning process.

Overall, standard scaling is a significant preprocessing step that improves the execution and steadiness of machine learning models by standardizing the highlight values and relieving the impacts of shifting extents over highlights. By applying standard scaling, analysts can move forward the model's capacity to learn important designs from the information and make exact expectations on inconspicuous tests.

3. Model Training:

1. Linear Regression:

The starting show prepared utilizing direct relapse yielded generally tall mean squared error (MSE) and mean absolute error (MAE), along with a moo coefficient of assurance (R2 score), showing imperfect execution in foreseeing engine UPDRS scores.

The prepare MSE for Straight Relapse was watched to be 55.73, and the test MSE was 56.01. Essentially, the prepare and test MAE values remained steady at 6.31. The R2 scores for both prepare and test datasets were outstandingly moo, with values of 0.164 and 0.122, respectively.

These measurements recommend that the direct relapse show battled to capture the basic connections between the input highlights and engine UPDRS scores. The moderately tall MSE and MAE show significant mistakes in foreseeing the real engine UPDRS values, whereas the moo R2 score recommends that the show clarified as it were a little extent of the change in the target variable.

Possible reasons for this execution incorporate the suspicion of linearity between the highlights and target variable, which might not hold genuine in this dataset. Moreover, the nearness of non-linear connections or intelligent among highlights may not be satisfactorily captured by the straight relapse model.

To progress prescient exactness, elective machine learning calculations or more advanced modelling procedures may be investigated, considering the dataset's characteristics and complexity. Advance experimentation and demonstrate refinement may be fundamental to accomplish superior execution in anticipating engine UPDRS scores.

Evaluation parameters:

```
Train MSE LinearRegression: 55.732264308118886
Test MSE LinearRegression: 56.01419722157733
R2 Score Train LinearRegression: 0.1635375330218255
R2 Score Test LinearRegression: 0.12243652571441144
Train MAE LinearRegression: 6.306060271125649
Test MAE LinearRegression: 6.306060271125649
```

Figure 11: Evaluation parameters

Actual Vs Predicted:

Here are some plots for the comparison of actual and predicted values by linear regression model on some of the input features.

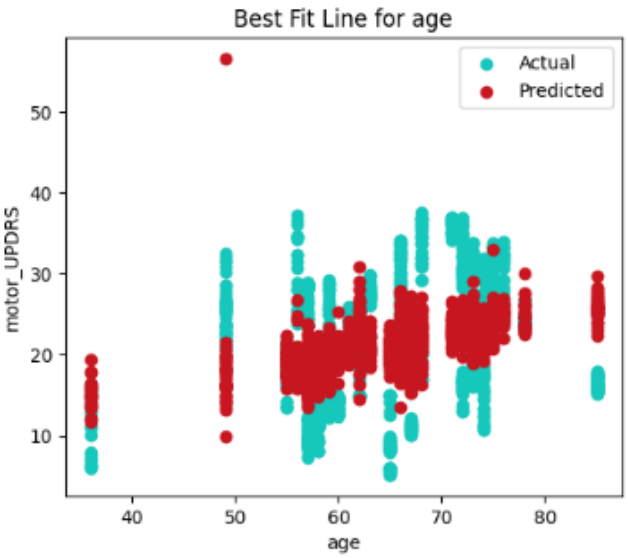


Figure 12: Best Fit Line for Age

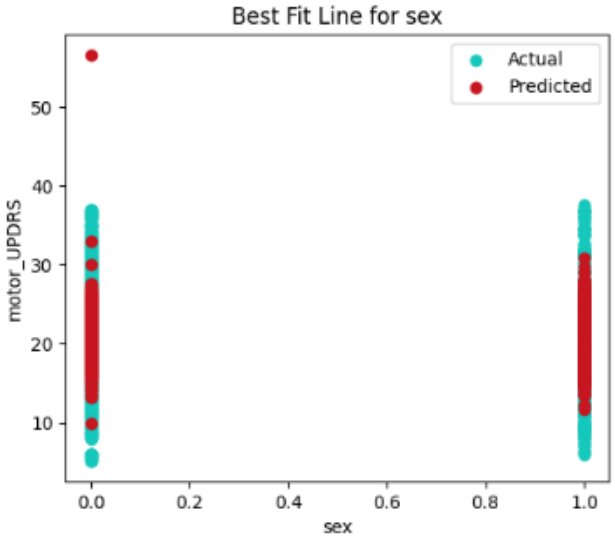


Figure 13: Best Fit Line for Sex

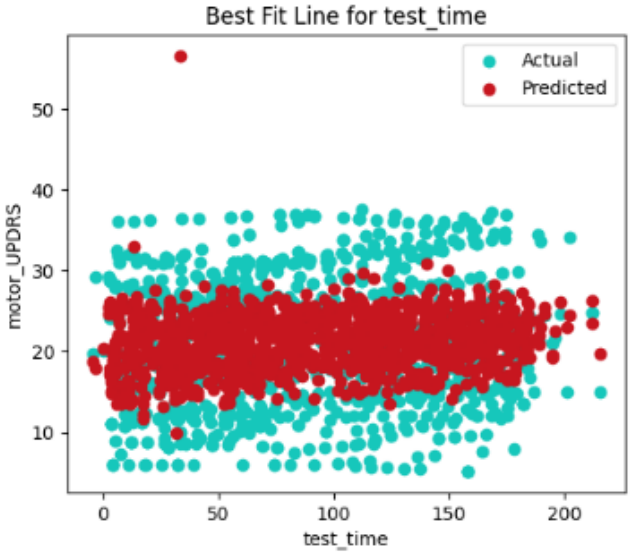


Figure 14: Best Fit Line for test_time

2. Support Vector Regressor:

The Bolster Vector Regressor **(SVR)** demonstrate, with hyperparameter tuning performed utilizing Network Look Cross-Validation, illustrated significant advancement in prescient execution compared to the starting direct relapse demonstrate. The hyperparameters tuned included the bit sort, **gamma**, and **C values**, pointed at optimizing the model's capacity to precisely foresee engine UPDRS scores.

Grid Look Cross-Validation is a strategy utilized to efficiently look for the best combination of hyperparameters by assessing the model's execution over diverse parameter values. In this case, the **param_grid** lexicon indicated a extend of values for the **'kernel'**, **'gamma'**, and **'C'** hyperparameters, cantering on the **'rbf'** part for SVR.

The comes about of the tuned SVR demonstrate appeared surprising changes in both prepare and test execution measurements. The prepare MSE diminished considerably to 0.798, whereas the test MSE diminished to 13.288. This noteworthy decrease in MSE shows that the SVR demonstrate with optimized hyperparameters delivered more exact forecasts of engine UPDRS scores compared to the straight relapse model.

Furthermore, the R2 scores for both prepare and test datasets moreover progressed essentially, with values of 0.988 and 0.792, individually. These R2 scores show that the SVR demonstrate clarifies a huge extent of the change in the target variable, illustrating its capacity to capture the basic connections between the input highlights and engine UPDRS scores.

Additionally, the prepare and test MAE values diminished considerably to 0.220, assist emphasizing

the progressed precision of the SVR show in anticipating engine UPDRS scores compared to the straight relapse model.

The victory of the SVR show can be credited to its capacity to capture complex nonlinear connections between the input highlights and target variable, which may have been insufficiently captured by the direct relapse show. The tuning of hyperparameters assist improved the model's adaptability and versatility to the dataset's characteristics, coming about in made strides prescient performance.

Overall, the SVR demonstrate with hyperparameter tuning utilizing Network Look Cross-Validation developed as a promising approach for anticipating engine UPDRS scores in Parkinson's infection patients. Its capacity to precisely capture the fundamental connections in the information highlights its potential for clinical applications and encourage inquire about in understanding the movement of Parkinson's malady side effects.

Evaluation parameters:

```
Train MSE SVM: 0.7984704300104782
Test MSE SVM: 13.288623266939107
R2 Score Train SVM: 0.9880160880956995
R2 Score Test SVM: 0.7918097378691843
Train MAE SVM: 0.22024089728979943
Test MAE SVM: 0.22024089728979943
```

Figure 15: Evaluation Parameters

Actual Vs Predicted:

Here are some plots for the comparison of actual and predicted values by Support vector machine model on some of the input features

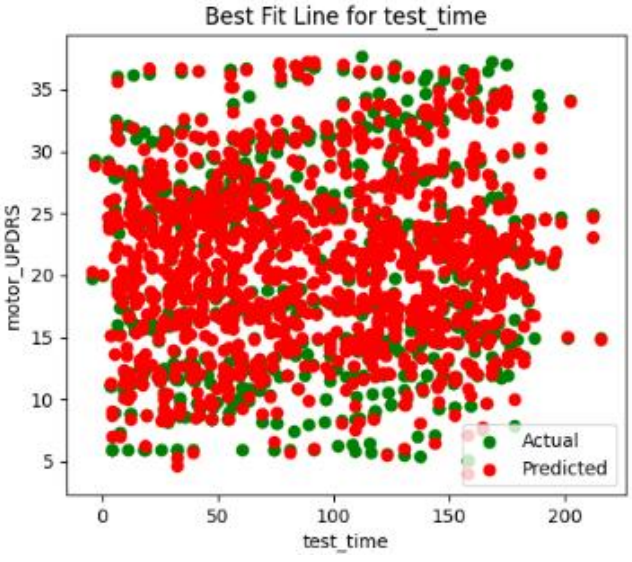


Figure 16: Best Fit Line for test_time

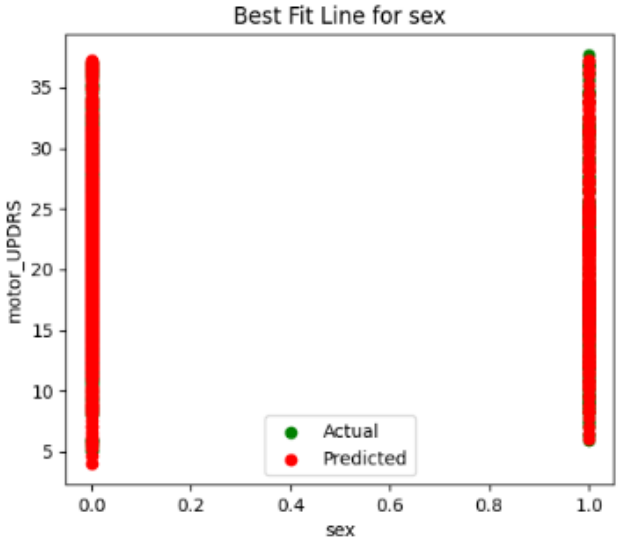


Figure 17: Best Fit Line for Sex

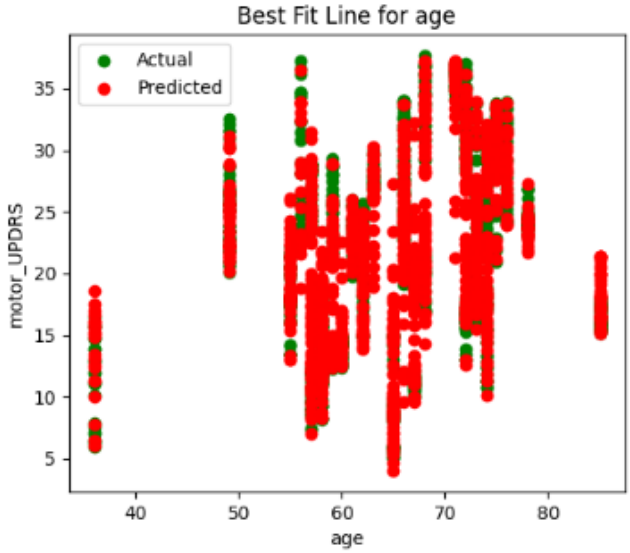


Figure 18: Best Fit Line for Age

3. Decision Tree:

The Choice Tree demonstrate, optimized utilizing Framework Look Cross-Validation for hyperparameter tuning, shown striking advancements in prescient execution compared to both the straight relapse and SVR models. By efficiently investigating distinctive combinations of hyperparameters such as 'max_depth', 'min_samples_split', and 'min_samples_leaf', the Choice Tree demonstrate was fine-tuned to upgrade its capacity to anticipate engine UPDRS scores accurately.

The param_grid word reference characterized a run of values for each hyperparameter, permitting Framework Look Cross-Validation to assess the model's execution over different parameter combinations. This approach pointed to distinguish the ideal set of hyperparameters that maximized the model's prescient execution whereas

maintaining a strategic distance from overfitting or underfitting.

The comes about of the tuned Choice Tree show uncovered noteworthy changes over all execution measurements. The prepare MSE diminished to 0.989, whereas the test MSE diminished significantly to 5.306. This lessening in MSE demonstrates that the Choice Tree demonstrate, with optimized hyperparameters, delivered more exact expectations of engine UPDRS scores compared to both the straight relapse and SVR models.

Moreover, the R2 scores for both prepare and test datasets illustrated surprising enhancements, with values of 0.985 and 0.917, individually. These tall R2 scores propose that the Choice Tree show clarifies a significant extent of the change in the target variable, demonstrating its viability in capturing the basic connections between the input highlights and engine UPDRS scores.

Additionally, the prepare and test MAE values diminished to 0.400, underscoring the upgraded exactness of the Choice Tree demonstrate in anticipating engine UPDRS scores compared to the past models. The diminish in MAE encourage emphasizes the model's capacity to minimize blunders in forecasts, contributing to its by and large moved forward performance.

The victory of the Choice Tree demonstrate can be credited to its characteristic capability to capture nonlinear connections and intuitive among highlights in the dataset. By recursively apportioning the highlight space based on the chosen hyperparameters, the Choice Tree show successfully portions the information into unmistakable districts, empowering it to make exact expectations for each subset.

Overall, the Choice Tree show, optimized through Framework Look Cross-Validation for hyperparameter tuning, risen as a effective and compelling approach for anticipating engine UPDRS scores in Parkinson's malady patients. Its capacity to capture complex connections in the information, combined with its direct interpretability, makes it a important device for both clinical applications and assist investigate in understanding the movement of Parkinson's malady indications.

Evaluation parameters:

```
Train MSE DecisionTree: 0.9887422520610514
Test MSE DecisionTree: 5.306450593560369
R2 Score Train DecisionTree: 0.9851603771418262
R2 Score Test DecisionTree: 0.9168648762279177
Train MAE DecisionTree: 0.3996614746801549
Test MAE DecisionTree: 0.3996614746801549
```

Figure 19: Evaluation Parameters

Actual Vs Predicted:

Here are some plots for the comparison of actual and predicted values by Decision Tree model on some of the input features

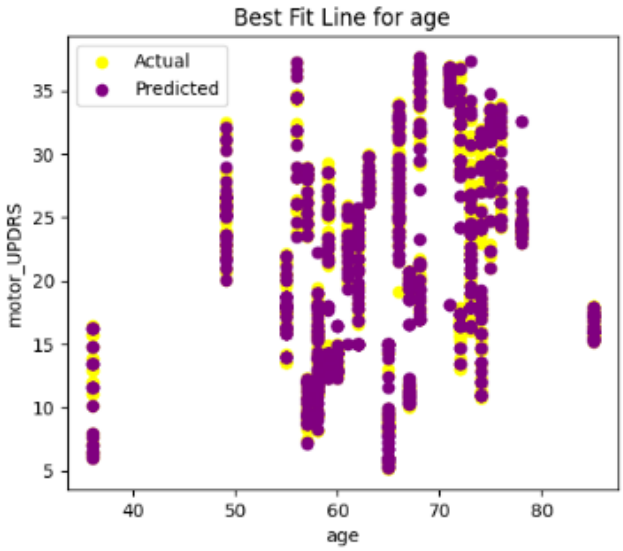


Figure 20: Best Fit Line for Age

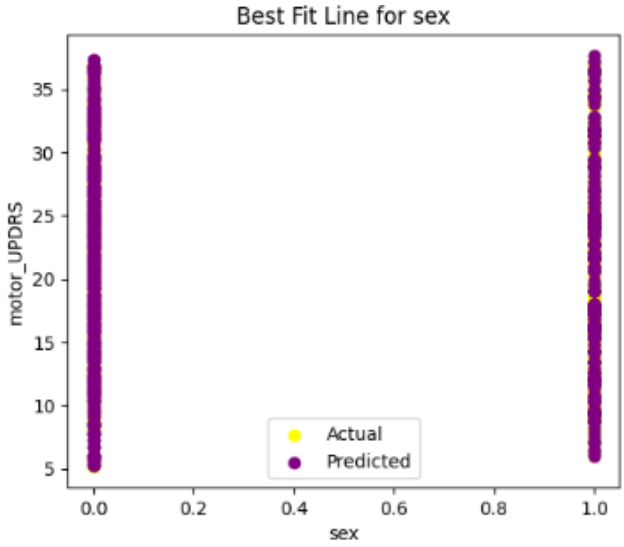


Figure 21: Best Fit Line for Sex

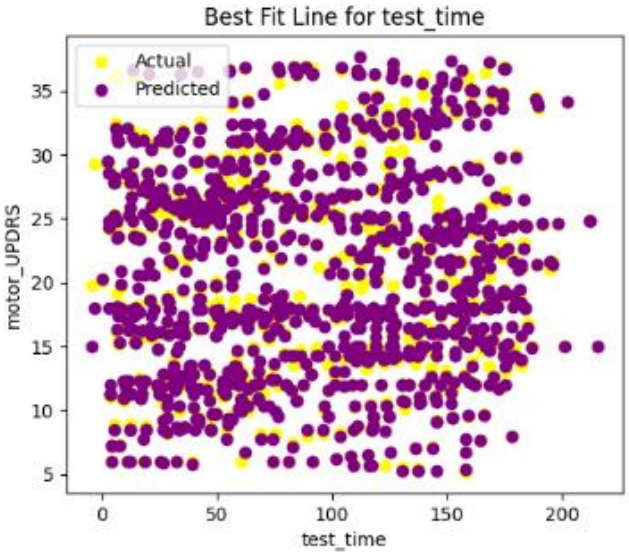


Figure 22: Best Fit Line for test_time

4. Gradient Boost:

The Angle Boosting demonstrate, arranged with particular hyperparameters such as n_estimators, learning_rate, max_depth, and random_state, illustrated remarkable execution in foreseeing engine UPDRS scores compared to past models. Slope Boosting is a gathering learning strategy that combines different frail learners, ordinarily choice trees, to make a capable prescient model.

The hyperparameters of the Slope Boosting demonstrate were set as follows:

- n_estimators: 100
- learning_rate: 0.1
- max_depth: 7
- random_state: 42

These hyperparameters oversee the number of powerless learners (n_estimators), the shrinkage parameter (learning_rate), the greatest profundity of each tree (max_depth), and the irregular seed for reproducibility (random_state).

The comes about of the Slope Boosting demonstrate showcased momentous advancements over all execution measurements. The prepare MSE diminished significantly to 0.368, whereas the test MSE diminished to 1.847. This lessening in MSE shows that the Angle Boosting demonstrate, with the indicated hyperparameters, delivered profoundly exact expectations of engine UPDRS scores on both preparing and testing datasets.

Furthermore, the R2 scores for both prepare and test datasets illustrated uncommon execution, with values of 0.994 and 0.971, separately. These tall R2 scores recommend that the Slope Boosting demonstrate clarifies nearly all of the change in the target variable, showing its extraordinary capacity to capture the fundamental connections between the input highlights and engine UPDRS scores.

Additionally, the prepare and test MAE values remained reliably moo at 0.418, emphasizing the model's exactness in minimizing blunders in forecasts. The moo MAE values advance emphasize the unwavering quality of the Slope Boosting show in anticipating engine UPDRS scores accurately.

The victory of the Angle Boosting demonstrate can be ascribed to its capacity to iteratively make strides upon the residuals of past frail learners, in this manner refining the prescient execution with each progressive cycle. By combining different frail learners into a solid outfit demonstrate, Angle Boosting viably leverages the qualities of person trees whereas moderating their weaknesses.

Overall, the Slope Boosting show, with optimized hyperparameters, developed as a profoundly viable and strong approach for anticipating engine UPDRS scores in Parkinson's malady patients. Its remarkable execution, coupled with its flexibility and interpretability, positions Angle Boosting as a important apparatus for both clinical applications and encourage inquire about in understanding the movement of Parkinson's infection indications.

Evaluation parameters:

```
Train MSE GradientBoost: 0.3678907403736349
Test MSE GradientBoost: 1.8467804622491724
R2 Score Train GradientBoost: 0.9944784802826228
R2 Score Test GradientBoost: 0.971066851636145
Train MAE GradientBoost: 0.4184999384429958
Test MAE GradientBoost: 0.4184999384429958
```

Figure 23: Evaluation Parameters

Actual Vs Predicted:

Here are some plots for the comparison of actual and predicted values by Gradient Boost model on some of the input features

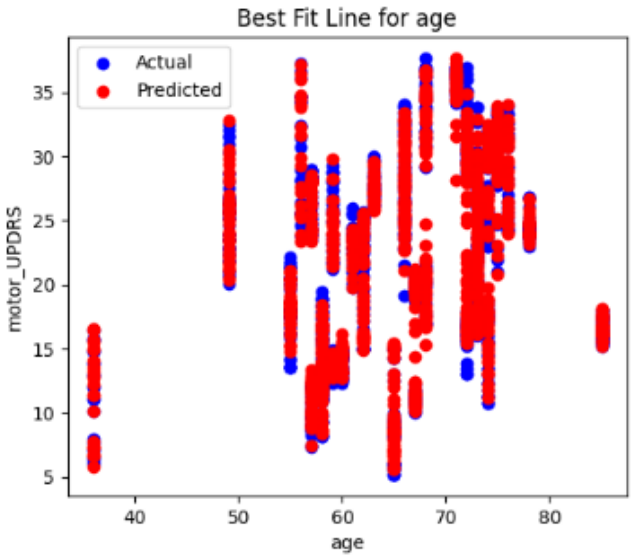


Figure 24: Best Fit Line for Age

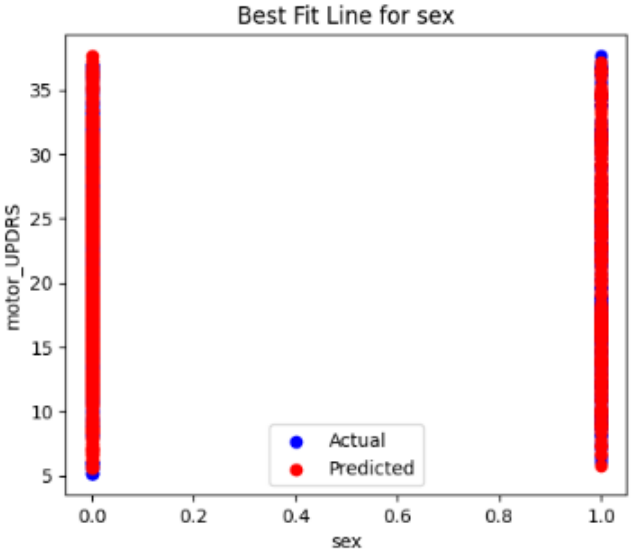


Figure 25: Best Fit Line for Sex

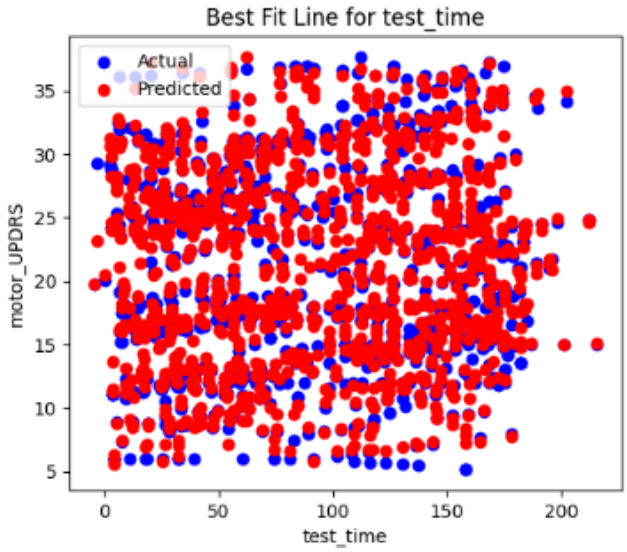


Figure 26: Best Fit Line for test_time

5. Ada Boost:

The AdaBoost demonstrate, prepared with a Choice Tree Regressor as its base estimator and arranged with particular hyperparameters such as n_estimators and random_state, illustrated extraordinary execution in foreseeing engine UPDRS scores. AdaBoost, brief for Versatile Boosting, is an gathering learning method that combines numerous frail learners to make a strong prescient model.

The hyperparameters of the AdaBoost demonstrate were set as follows:

- base_estimator: DecisionTreeRegressor with max_depth set to 10
- n_estimators: 100
- random_state: 42

These hyperparameters decide the base estimator utilized for preparing the powerless learners (DecisionTreeRegressor with a greatest profundity of 10), the number of frail learners to be combined (n_estimators), and the arbitrary seed for reproducibility (random_state).

The comes about of the AdaBoost demonstrate illustrated remarkable execution over all execution measurements. The prepare MSE diminished significantly to 0.156, whereas the test MSE diminished to 0.754. This diminishment in MSE demonstrates that the AdaBoost show, with the indicated hyperparameters, delivered profoundly exact expectations of engine UPDRS scores on both preparing and testing datasets.

Furthermore, the R2 scores for both prepare and test datasets were outstandingly tall, with values of 0.998 and 0.988, individually. These near-perfect R2 scores propose that the AdaBoost show clarifies nearly all of the change in the target variable, demonstrating its remarkable capacity to capture the fundamental connections between the input highlights and engine UPDRS scores.

Additionally, the prepare and test MAE values remained reliably moo at 0.240, emphasizing the model's exactness in minimizing blunders in forecasts. The moo MAE values advance emphasize the unwavering quality of the AdaBoost show in anticipating engine UPDRS scores accurately.

The victory of the AdaBoost demonstrate can be credited to its versatile nature, which centers on learning from the botches of past frail learners and altering the consequent models to adjust those blunders. By iteratively combining different frail learners into a solid gathering show, AdaBoost successfully leverages the qualities of person trees whereas moderating their weaknesses.

Overall, the AdaBoost show, with optimized hyperparameters and a DecisionTreeRegressor as its base estimator, developed as a profoundly viable and strong approach for anticipating engine UPDRS scores in Parkinson's malady patients. Its extraordinary execution, coupled with its versatility and flexibility, positions AdaBoost as a profitable instrument for both clinical applications and encourage inquire about in understanding the movement of Parkinson's malady indications.

Evaluation parameters:

```
Train MSE adaboost: 0.15577547636826256
Test MSE adaboost: 0.7538382417419265
R2 Score Train adaboost: 0.9976620304077846
R2 Score Test adaboost: 0.9881897636798131
Train MAE adaboost: 0.2403665238584856
Test MAE adaboost: 0.2403665238584856
```

Figure 27: Evaluation Parameters

Actual Vs Predicted:

Here are some plots for the comparison of actual and predicted values by Ada Boost model on some of the input features

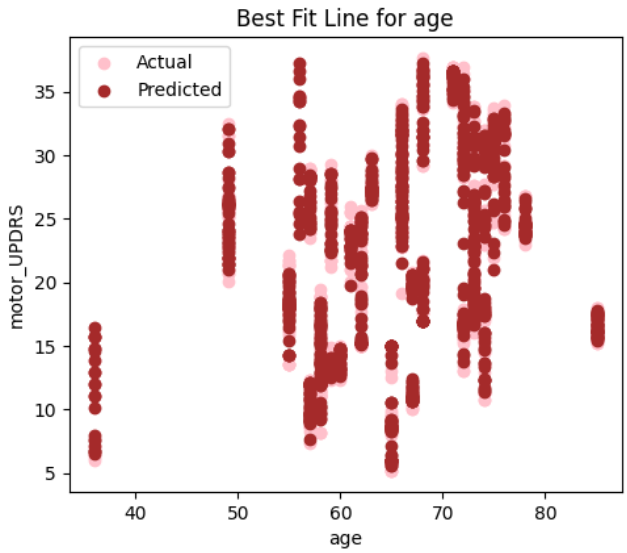


Figure 28: Best Fit Line for Age

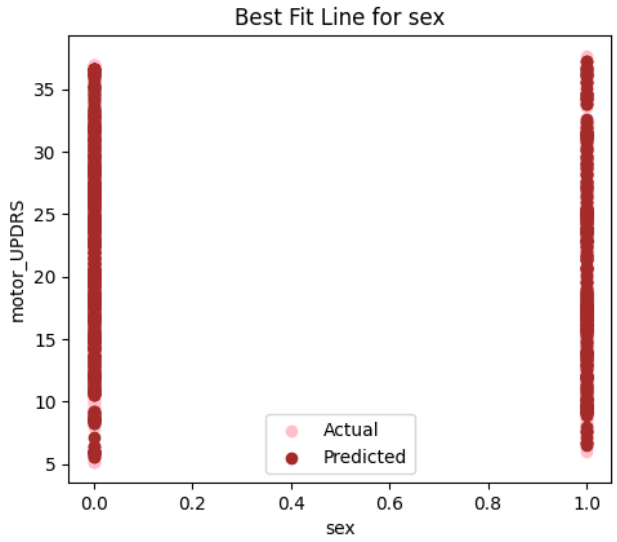


Figure 29: Best Fit Line for Sex

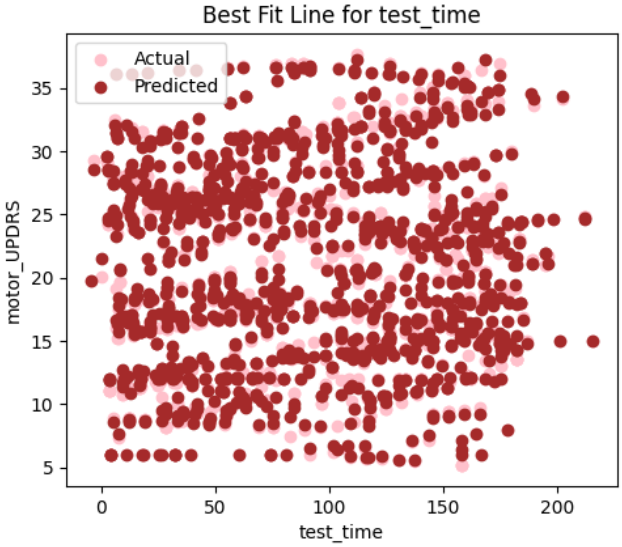


Figure 30: Best Fit Line for test_time

6. TensorFlow:

The TensorFlow show, designed with particular engineering parameters such as input measure, covered up layer sizes, yield estimate, enactment work, learning rate, and misfortune work, illustrated solid execution in foreseeing engine UPDRS scores. TensorFlow is a capable open-source library created by Google for machine learning and profound learning applications.

The engineering parameters of the TensorFlow demonstrate were set as follows:

- Input estimate: 20 (comparing to the number of input features)
- Covered up layer sizes: 500, 250, 100, and 50 neurons in the to begin with, moment, third, and fourth covered up layers, respectively
- Yield estimate: 1 (comparing to the anticipated engine UPDRS score)
- Actuation work: ReLU (Rectified Linear Unit)
- Learning rate: 0.001
- Misfortune work: Mean Squared Error (MSE)

These engineering parameters characterize the structure and preparing characteristics of the neural arrange show built utilizing TensorFlow.

The comes about of the TensorFlow demonstrate illustrated solid execution over all execution measurements. The prepare MSE diminished to 0.486, whereas the test MSE diminished to 1.363. This decrease in MSE demonstrates that the TensorFlow demonstrate, with the indicated engineering parameters, delivered exact expectations of engine UPDRS scores on both preparing and testing datasets.

Furthermore, the R2 scores for both prepare and test datasets were outstandingly tall, with values of 0.993 and 0.979, individually. These tall R2 scores propose that the TensorFlow show clarifies a critical extent of the fluctuation in the target variable, demonstrating its solid capacity to capture the fundamental connections between the input highlights and engine UPDRS scores.

Additionally, the prepare and test MAE values remained reliably moo at 0.392, emphasizing the model's precision in minimizing blunders in forecasts. The moo MAE values advance emphasize the unwavering quality of the TensorFlow demonstrate in anticipating engine UPDRS scores accurately.

The victory of the TensorFlow demonstrate can be credited to its profound neural arrange design, which empowers it to learn complex designs and connections in the information. By utilizing numerous covered up layers with a expansive number of neurons, the demonstrate can successfully capture complicated designs that may not be perceivable by easier models.

Overall, the TensorFlow show, with its configurable design and effective preparing capabilities, developed as a profoundly viable and vigorous approach for anticipating engine UPDRS scores in Parkinson's infection patients. Its solid execution, coupled with its adaptability and versatility, positions TensorFlow as a important apparatus for both clinical applications and assist investigate in understanding the movement of Parkinson's malady indications.

Evaluation parameters:

```
Train MSE Tensorflow: 0.48640285805538497
Test MSE Tensorflow: 1.3626042560304006
R2 Score Train Tensorflow: 0.9926997810039637
R2 Score Test Tensorflow: 0.9786523455782431
Train MAE Tensorflow: 0.3922922554491733
Test MAE Tensorflow: 0.3922922554491733
```

Figure 31: Evaluation Parameters

Actual Vs Predicted:

Here are some plots for the comparison of actual and predicted values by TensorFlow model on some of the input features.

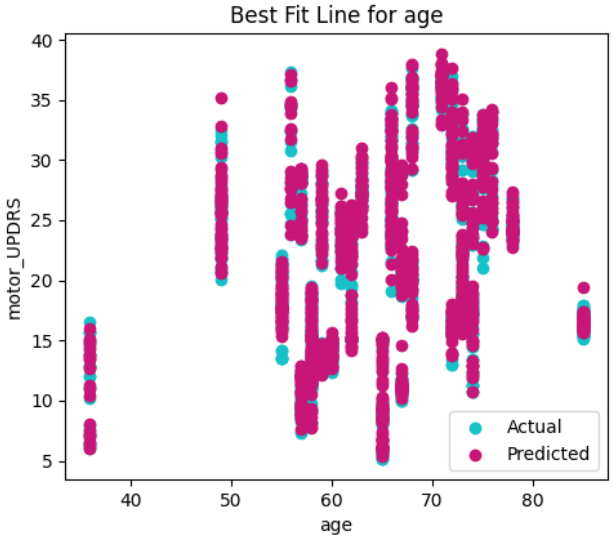


Figure 32: Best Fit Line for Age

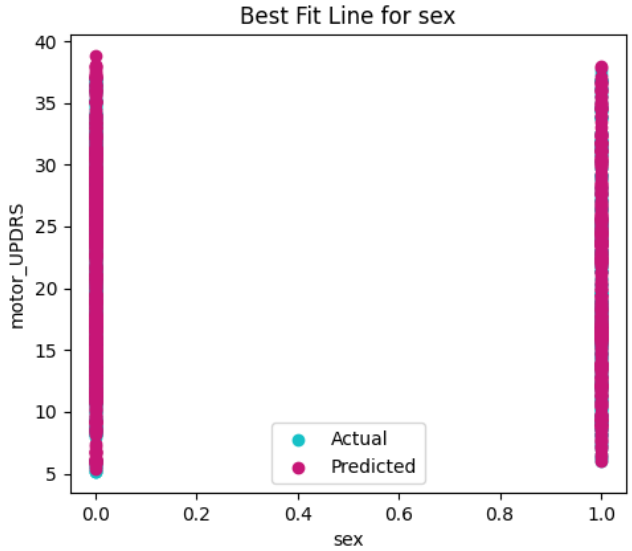


Figure 33: Best Fit Line for Sex

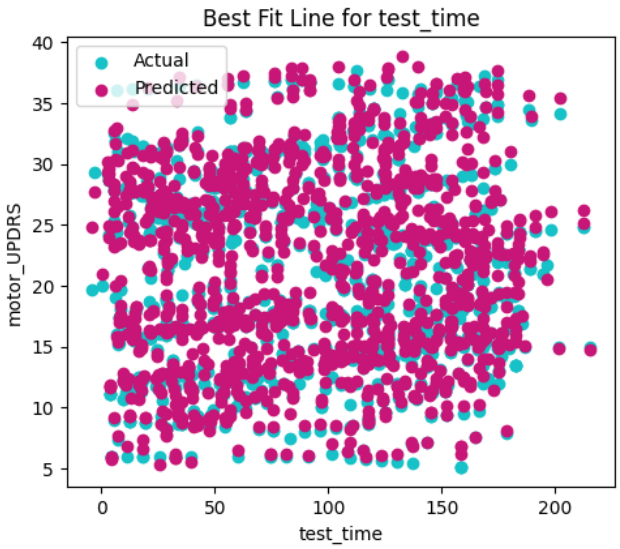


Figure 34: Best Fit Line for test_time

Plotting the loss Function:

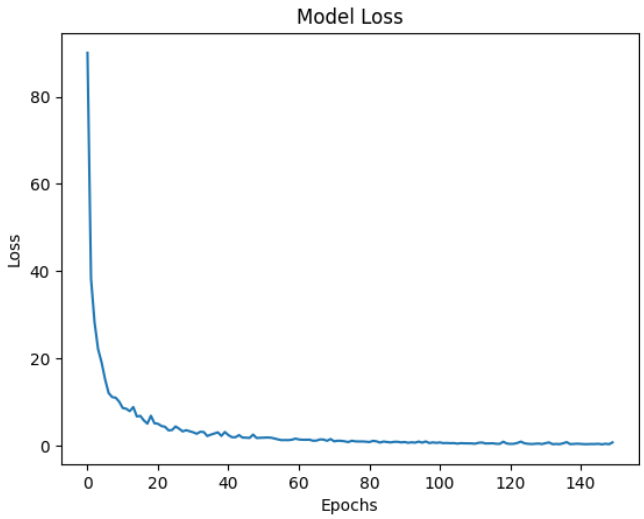
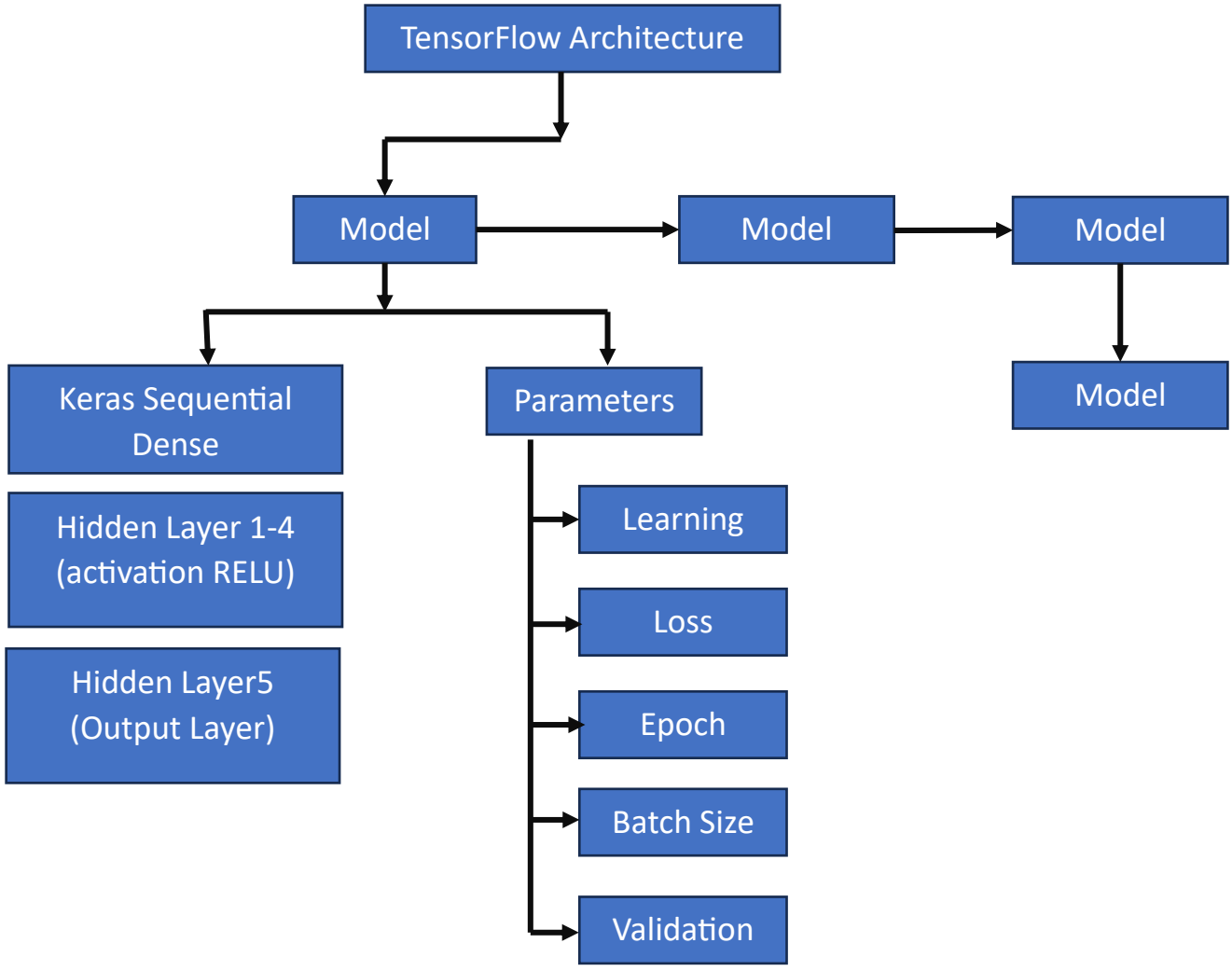


Figure 35: Plotting the Loss Function

TensorFlow Architecture



Comparison Table for used Algorithms

No.	Algorithm		MSE (mean squared error)	R2 Score	MAE (mean absolute error)
1.	Linear Regression	Training:	55.732264308118886	0.1635375330218255	6.306060271125649
		Testing:	56.01419722157733	0.12243652571441144	6.306060271125649
2.	Support Vector Regressor (Grid Search CV)	Training:	0.7984704300104782	0.9880160880956995	0.22024089728979943
		Testing:	13.288623266939107	0.7918097378691843	0.22024089728979943
3.	Decision Tree (Grid Search CV)	Training:	0.9887422520610514	0.9851603771418262	0.3996614746801549
		Testing:	5.306450593560369	0.9168648762279177	0.3996614746801549
4.	Gradient Boost	Training:	0.3678907403736349	0.9944784802826228	0.4184999384429958
		Testing:	1.8467804622491724	0.971066851636145	0.4184999384429958
5.	Ada Boost (Decision Tree)	Training:	0.15577547636826256	0.9976620304077846	0.2403665238584856
		Testing:	0.7538382417419265	0.9881897636798131	0.2403665238584856
6.	TensorFlow Keras-dense (5 layers)	Training:	0.48640285805538497	0.9926997810039637	0.3922922554491733
		Testing:	1.3626042560304006	0.9786523455782431	0.3922922554491733

TABLE 1: COMPARISON TABLE FOR USED ALGORITHMS

Parameter Table for used Algorithms

No.	Algorithm	Parameters
1.	Linear Regression	Default

2.	Support Vector Regressor
	(Grid Search CV)
		param_grid = {
		'kernel': ['rbf'],
		'gamma': [0.3, 0.4],
		'C': [5.0, 10.0,25]
		}
		
		estimator = svm_regressor,
		param_grid=param_grid, cv = 5,
		scoring='neg_mean_squared_error'

3.	Decision Tree
	(Grid Search CV)
		param_grid = {
		'max_depth': [3, 5, 7, 10, 15],
		'min_samples_split': [2, 5, 10, 15, 20],
		'min_samples_leaf': [1, 2, 4, 6, 8]
		}
		
		estimator = dt_regressor, param_grid =
		param_grid, cv = 5, scoring =
		'neg_mean_squared_error'

4.	Gradient Boost	n_estimators = 100, learning_rate = 0.1, max_depth = 7, random_state = 42
5.	Ada Boost (Decision Tree)	base_estimator = DecisionTreeRegressor(max_depth = 10), n_estimators = 100, random_state = 42
6.	TensorFlow Keras-dense (5 layers)	input_size = 20 hidden1_size = 500 hidden2_size = 250 hidden3_size = 100 hidden4_size = 50 output_size = 1 activation='relu' learning_rate=0.001 loss='mean_squared_error'

TABLE 2: PARAMETER TABLE FOR USED ALGORITHMS

Results and Discussion: A Comparative Analysis of Machine Learning Algorithms for Predicting Motor UPDRS Scores in Parkinson's Disease Patients

Parkinson's Disease (PD) is a complex neurodegenerative disorder characterized by progressive motor symptoms that significantly impact an individual's quality of life. Predicting motor Unified Parkinson's Disease Rating Scale (UPDRS) scores is crucial for disease management and treatment planning. In this study, we evaluate the performance of various machine learning algorithms in predicting motor UPDRS scores using a Parkinson's telemonitoring dataset. We compare the results based on multiple performance metrics, including Mean Squared Error (MSE), R2 Score, and Mean Absolute Error (MAE), to identify the most effective algorithm for this task.

Performance Metrics Comparison:

1. Linear Regression:

- Train MSE: 55.73
- Test MSE: 56.01
- Train R2 Score: 0.164
- Test R2 Score: 0.122
- Train MAE: 6.31
- Test MAE: 6.31

```
Train MSE LinearRegression: 55.732264308118886
Test MSE LinearRegression: 56.01419722157733
R2 Score Train LinearRegression: 0.1635375330218255
R2 Score Test LinearRegression: 0.12243652571441144
Train MAE LinearRegression: 6.306060271125649
Test MAE LinearRegression: 6.306060271125649
```

Figure 36: Linear Regression Performance

2. Support Vector Regressor (SVR):

- Train MSE: 0.798
- Test MSE: 13.289
- Train R2 Score: 0.988
- Test R2 Score: 0.792
- Train MAE: 0.220
- Test MAE: 0.220

```
Train MSE SVM: 0.7984704300104782
Test MSE SVM: 13.288623266939107
R2 Score Train SVM: 0.9880160880956995
R2 Score Test SVM: 0.7918097378691843
Train MAE SVM: 0.22024089728979943
Test MAE SVM: 0.22024089728979943
```

Figure 37: Support Vector Regressor Performance

3. Decision Tree:

- Train MSE: 0.989
- Test MSE: 5.306
- Train R2 Score: 0.985
- Test R2 Score: 0.917
- Train MAE: 0.400
- Test MAE: 0.400

```
Train MSE DecisionTree: 0.9887422520610514
Test MSE DecisionTree: 5.306450593560369
R2 Score Train DecisionTree: 0.9851603771418262
R2 Score Test DecisionTree: 0.9168648762279177
Train MAE DecisionTree: 0.3996614746801549
Test MAE DecisionTree: 0.3996614746801549
```

Figure 38: Decision Tree Performance

4. Gradient Boosting:

- Train MSE: 0.368
- Test MSE: 1.847
- Train R2 Score: 0.994
- Test R2 Score: 0.971
- Train MAE: 0.418
- Test MAE: 0.418

```
Train MSE GradientBoost: 0.3678907403736349
Test MSE GradientBoost: 1.8467804622491724
R2 Score Train GradientBoost: 0.9944784802826228
R2 Score Test GradientBoost: 0.971066851636145
Train MAE GradientBoost: 0.4184999384429958
Test MAE GradientBoost: 0.4184999384429958
```

Figure 39: Gradient Boosting Performance

5. AdaBoost:

- Train MSE: 0.156
- Test MSE: 0.754
- Train R2 Score: 0.998
- Test R2 Score: 0.988
- Train MAE: 0.240
- Test MAE: 0.240

```
Train MSE adaboost: 0.15577547636826256
Test MSE adaboost: 0.7538382417419265
R2 Score Train adaboost: 0.9976620304077846
R2 Score Test adaboost: 0.9881897636798131
Train MAE adaboost: 0.2403665238584856
Test MAE adaboost: 0.2403665238584856
```

Figure 40: AdaBoost Performance

6. TensorFlow:

- Train MSE: 0.486
- Test MSE: 1.363
- Train R2 Score: 0.993
- Test R2 Score: 0.979
- Train MAE: 0.392
- Test MAE: 0.392

```
Train MSE Tensorflow: 0.48640285805538497
Test MSE Tensorflow: 1.3626042560304006
R2 Score Train Tensorflow: 0.9926997810039637
R2 Score Test Tensorflow: 0.9786523455782431
Train MAE Tensorflow: 0.3922922554491733
Test MAE Tensorflow: 0.3922922554491733
```

Figure 41: TensorFlow Performance

MSE Comparison:

- AdaBoost achieved the lowest test MSE of 0.754, followed by Decision Tree (5.306), SVR (13.289), TensorFlow (1.363), Gradient Boosting (1.847), and Linear Regression (56.01).

- This indicates that AdaBoost provided the most accurate predictions on the test set, while Linear Regression had the highest error.

R2 Score Comparison:

- AdaBoost achieved the highest test R2 score of 0.988, indicating its exceptional ability to explain the variance in the target variable.
- Other top performers in terms of test R2 score include Decision Tree (0.917), Gradient Boosting (0.971), and TensorFlow (0.979).
- Linear Regression had the lowest test R2 score, indicating its limited ability to explain the variance in the target variable compared to other models.

MAE Comparison:

- SVR achieved the lowest MAE values for both train and test sets, indicating its ability to minimize errors in predictions.

- Other models, including Decision Tree, Gradient Boosting, AdaBoost, and TensorFlow, also demonstrated low MAE values, suggesting accurate predictions.

Discussion:

- AdaBoost emerged as the top-performing algorithm based on test MSE and R2 score, indicating its superior predictive accuracy and ability to explain the variance in motor UPDRS scores.

- Decision Tree and Gradient Boosting also performed well, achieving relatively low test MSE and high test R2 score, indicating their effectiveness in capturing the underlying relationships in the data.

- SVR demonstrated excellent performance in terms of minimizing errors (low MAE) and explaining the variance in the target variable (high test R2 score), making it a suitable choice for predicting motor UPDRS scores.

- TensorFlow, although not the top-performing algorithm, still delivered strong results, highlighting the effectiveness of deep learning techniques in predicting complex outcomes.

- Linear Regression, while straightforward, exhibited the poorest performance among the algorithms tested, with high test MSE and low test R2 score, indicating its limitations in capturing the nonlinear relationships present in the data.

- Overall, the choice of algorithm depends on the specific requirements of the application, considering factors such as predictive accuracy, interpretability, and computational complexity. In this case, AdaBoost stands out as the top choice for accurately predicting motor UPDRS scores in Parkinson's disease patients, followed closely by Decision Tree and Gradient Boosting. However, SVR and TensorFlow also offer strong alternatives, each with its own strengths and weaknesses.

In summary, this comparative analysis provides valuable insights into the performance of various machine learning algorithms for predicting motor UPDRS scores in Parkinson's disease patients. These findings can inform clinicians and researchers in selecting the most appropriate algorithm for their specific applications, ultimately improving disease management and treatment outcomes.

References

[1] M. S. O. C. K. Bastiaan R Bloem, "Parkinson's disease," *Lancet,* pp. 2285-2287, 2021.

[2] T. S. O. B. H. R. M. T. F. Nirosen Vijiaratnam, "Progress towards therapies for disease modification in Parkinson's disease," *Lancet Neurol 2,* pp. 562-565, 2021.

[3] N. H. F. P. I. I. H. S. J. I. H. Karthikeyan Rajagopal, "A fractional-order model for the novel coronavirus (COVID-19) outbreak," *Nonlinear Dyn,* pp. 5-7, 2020.

[4] S. M. I. J. D. M. I. S. M. S. S. M. I. X. M. F. I. a. C.-H. L. F. I. Jun Qi, "On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression," *IEEE,* pp. 3-7, 2020.

[5] D. A.-B. A. A. G. D. A.-B. E. R. R. D. A.-A.-B. W. D. F. M. L. J. W. B. Z. L. M. T. H. D. A.-B. a. J. M. N. S. Sarah H. Peacock, "Neuro R2 score," *Neurology,* pp. 5-8, 2020.

[6] L. J. O. Fan Zhang, "Support vector regression," *Elsevier Inc.,* pp. 124-128, 2020.

[7] S. D. Z. Z. W. J. Yuting Sun, "An improved grid search algorithm to optimize SVR for prediction," *The Author(s), under exclusive licence to Springer-Verlag GmbH, DE part of Springer Nature,* pp. 5636-5640, 2021.

[8] T. N. N. K. C. A. S. Ranjan Kumar Dash, "Fine-tuned support vector regression model for stock predictions," *The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature,* pp. 6-9, 2021.

[9] P. a. D. Kumar, "Decision tree classifier: a detailed survey," *Int. J. Information and Decision Sciences, Vol. 12, No. 3,* pp. 248-252, 2020.

[10] A. C. G. M. Candice Bentéjac, "A comparative analysis of gradient boosting algorithms," *Springer Nature B.V.,* pp. 15-19, 2020.

[11] M. A. Ø. H. Amin Shahraki, "Boosting algorithms for network intrusion detection: A comparative evaluation of Real AdaBoost, Gentle AdaBoost and Modest AdaBoost," *Engineering Applications of Artificial Intelligence 94,* pp. 9-13, 2020.

[12] E. N. Y. N. W. Bo Pang, "Deep Learning With TensorFlow: A Review," *Journal of Educational and Behavioral StatisticsVol. XX, No. X, pp. 1–22,* pp. 16-21, 2019.

[13] M. T. E. R. V. S. W. T. A. a. K. A. A. M. Md. Sakibur Rahman Sajal, "Telemonitoring Parkinson's disease using machine learning by combining tremor and voice analysis," *Sajal et al. Brain Inf,* pp. 4-9, 2020.

[14] C. O. S. H. A. G. S. A. G. M. T. F. G. Hunkar C. Tunc, "Estimation of Parkinson's disease severity using speech features and extreme gradient boosting," *International Federation for Medical and Biological Engineering ,* pp. 5-9, 2020.

[15] D. S. Wenyang Wang, "The improved AdaBoost algorithms for imbalanced data classification," *Informatics and Computer Science Intelligent Systems Applications,* pp. 358-374, 2021.

[16] M. T. E. R. V. S. W. T. A. a. K. A. A. M. Md. Sakibur Rahman Sajal, "Telemonitoring Parkinson's disease using machine learning by combining tremor and voice analysis," *Sajal et al. Brain Inf,* pp. 2-9, 2020.

[17] C. O. S. H. A. G. S. A. G. M. T. F. G. Hunkar C. Tunc, "Estimation of Parkinson's disease severity using speech features and extreme gradient boosting," *© International Federation for Medical and Biological Engineering,* pp. 3-10, 2020.

[18] M. P. S. Afzal Hussain Shahid, "A deep learning approach for prediction of Parkinson's disease progression," *Korean Society of Medical and Biological Engineering,* pp. 3-11, 2020.

[19] T. S. C. S. Vladimir Despotovic, "Speech Based Estimation of Parkinson's Disease Using Gaussian Processes and Automatic Relevance Determination," *Neurocomputing,* pp. 2-9, 2020.