Explanation - depression_stage_prediction_aarav

The 8 steps in the machine learning process, viewed through the lens of machine learning concepts:

1. Data Acquisition and Loading

The machine learning process starts by loading a dataset, which is essentially a collection of structured data. This dataset contains features (inputs like heart rate and SpO2) and labels (outputs like depression stages). This step ensures the ML model has raw material to analyze and learn patterns from.

2. Data Integrity Check

Before any modeling, it is critical to check for missing values or anomalies in the dataset. Missing values can mislead the model during training. If issues are found, imputation or data cleaning strategies are applied to fix the dataset, ensuring the model learns from complete and accurate information.

3. Feature Scaling and Normalization

The features (e.g., heart rate and SpO2) may have different ranges or units. For instance, heart rate values are in the range of 30–120 bpm, while SpO2 values are percentages (0–100%). Scaling or normalizing these features ensures they are on the same scale, which prevents one feature from dominating another and helps optimization algorithms converge faster.

4. Defining Features and Labels

The dataset is split into:

- **Features (X)**: These are the independent variables (e.g., heart rate and SpO2) the model will analyze.
- **Labels (y)**: These are the dependent variables (e.g., depression stages) the model will predict.

This separation helps establish a clear relationship between inputs and outputs during training.

5. Train-Test Split

The dataset is divided into two subsets:

- **Training Set (80%)**: Used to train the model by exposing it to patterns in the data.
- **Testing Set (20%)**: Held back to evaluate the model's generalization ability on unseen data. This step avoids overfitting and provides an unbiased evaluation of the model’s performance.

---

6. Model Selection

Multiple machine learning algorithms (models) are chosen to compare their performance. For example:

- **Logistic Regression**: A simple linear model for classification problems.
- **Random Forest**: An ensemble method using multiple decision trees for improved accuracy.
- **Gradient Boosting**: A sequential ensemble learning method that optimizes errors iteratively.
- **Support Vector Machine (SVM)**: A powerful algorithm for finding hyperplanes to separate data.
- **K-Nearest Neighbors (KNN)**: A non-parametric algorithm that classifies based on the majority vote of neighbors. By testing different models, we can determine which algorithm performs best on the given data.

---

7. Model Training and Evaluation

Each model undergoes training, where it learns to map inputs (features) to outputs (labels). Once trained, models are evaluated using:

- **Accuracy**: Measures the percentage of correct predictions.
- **Precision**: Focuses on how many of the predicted positives are actual positives.
- **Recall**: Measures how many actual positives were correctly predicted.
- **F1-Score**: Combines precision and recall into a single metric for balanced evaluation. Confusion matrices visualize where predictions went wrong, such as misclassifying "Moderate Depression" as "Mild Depression." This helps in understanding model weaknesses.

---

8. Results Visualization and Comparison

After evaluation, the results are summarized in visual form:

- Bar charts compare metrics (accuracy, precision, recall, F1-score) across models using a color palette like "viridis" for clarity.
- These visualizations help identify the best-performing model, making it easier to select a model for deployment.

Final Takeaway

In machine learning, this workflow—loading data, preprocessing, splitting, training models, evaluating, and visualizing results—ensures we create a robust predictive system. By comparing different models, we aim to balance accuracy and computational efficiency for real-world applications.