Predicting Prenatal Depression Using Power State Data

Rianna Aalto
EC Utbildning
Examensarbete
2024-05

Abstract

Depression during pregnancy poses significant risks to both maternal and fetal health (Johnson et al., 2020). This study investigates the feasibility of predicting depression during pregnancy using power state data collected from wearable devices and EPDS survey data as an outcome. By monitoring physiological parameters and activity levels, wearable devices provide a non-intrusive means to track an individual's mental health status (Smith et al., 2022). Leveraging machine learning techniques, including feature extraction and classification algorithms (Chen et al., 2021), this research aims to develop a predictive model for identifying depressive symptoms during pregnancy based on patterns observed in power state data. Four machine learning models were used: Logistic Regression, Random Forest, Gradient Boosting, and Support Vector Machine. These models were trained on a balanced dataset created using the SMOTE technique (Chawla et al., 2002) and evaluated using RandomizedSearchCV (Bergstra and Bengio, 2012). The results indicated varying degrees of success, with Random Forest achieving an accuracy of 0.56, precision of 0.35, recall of 0.50, and ROC AUC of 0.58, while Gradient Boosting showed the most promise with an accuracy of 0.59, precision of 0.39, recall of 0.58, and ROC AUC of 0.54.

Acknowledgment

I would like to express my sincere gratitude to my teacher Antonio Prgomet for his invaluable guidance, encouragement, and unwavering support throughout the completion of my Data Science studies. His insightful feedback and expertise have greatly contributed to the quality and depth of my studies. I am also thankful to my classmates for their support and exchange of ideas during group activities and assignments, which have enriched my understanding of the subject matter and fostered a collaborative learning environment. I extend my appreciation to my intern supervisor Professor/Chief Physician in the Psychiatry Department, Fotis Papadopoulos, for providing me with the opportunity to apply theoretical knowledge to practical situations and for offering mentorship and constructive feedback during the internship period. He supported me all the way to my examensarbete without hesitation. He also allowed me to use their raw data on power state.

Förkortningar och Begrepp

EPDS: Edinburgh Postnatal Depression Scale - A widely used screening tool designed to detect postnatal depression in mothers. It is a self-report questionnaire consisting of 10 items that assess the intensity of depressive symptoms over the past week (Cox, Holden, and Sagovsky, 1987).

ML: Machine Learning - A subset of artificial intelligence that involves the use of algorithms and statistical models to enable computers to perform tasks without explicit instructions, by learning from patterns and inferences derived from data (Mitchell, 1997).

SMOTE: Synthetic Minority Over-sampling Technique - A technique used in data preprocessing to address class imbalance by generating synthetic samples for the minority class, thereby balancing the dataset and improving the performance of machine learning models (Chawla et al., 2002).

ROC AUC: Receiver Operating Characteristic Area Under the Curve - A performance measurement for classification problems at various threshold settings. The AUC represents the degree or measure of separability, indicating how much a model is capable of distinguishing between classes (Fawcett, 2006).

SVM: Support Vector Machine - A supervised machine learning algorithm used for classification and regression tasks. It works by finding the hyperplane that best separates the data points of different classes in the feature space (Cortes and Vapnik, 1995).

Table of Contents

1 Introduction

Depression during pregnancy is a significant concern, impacting both maternal and fetal health. Early detection and intervention are crucial for mitigating these risks (Patel et al., 2019). Traditional diagnostic methods, such as self-reports and clinical assessments, may have limitations in timeliness and accuracy (Johnson et al., 2020). Recent advancements in technology offer promising avenues for improving screening methods, particularly through the application of machine learning algorithms (Dagum and Ruiz, 2020). This study seeks to determine whether there is a significant correlation between power state data and the presence of depressive symptoms during pregnancy, using the EPDS score as an outcome measure (Smith et al., 2022). Power state data is hypothesized to correlate with depression because changes in sleep patterns, activity levels, and periods of inactivity—tracked through these data—are known indicators of mental health status.

1.1. Research Questions

1.1.1. Can powerstate data significantly predict depression during pregnancy when combined with EPDS scores as an outcome measure?

1.1.2. How well can machine learning algorithms predict prenatal depression based on these data sources?

2 Theory

Machine learning (ML) models, such as Random Forest, Support Vector Machine (SVM), Gradient Boosting, and Logistic Regression, are powerful classification tools that can analyze large datasets to predict outcomes based on various features. In this study, power state data collected via Mom2b app, combined with EPDS survey results, are used to train these models. Key performance metrics include accuracy, precision, recall, F1-score, and ROC AUC, which provide insights into the models' effectiveness in predicting prenatal depression (Kim et al., 2021).

2.1 Random Forest

Random Forest is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes for classification or the mean prediction for regression. It reduces overfitting by averaging multiple trees, which improves the model's accuracy (Breiman, 2001).

Equation:

$$\hat{y} = \frac{1}{N} \sum_{i=1}^{N} T_i(x)$$

2.2 Logistic Regression

Logistic Regression is a statistical model used for binary classification. It predicts the probability that a given input belongs to a particular class. The model uses the logistic function to map predicted values to probabilities (Hosmer and Lemeshow, 2000).

Equation:

$$P(y = 1|x) = \frac{1}{1+e^{-(\mathbf{w}\cdot\mathbf{x}+b)}}$$

2.3 Support Vector Machine (SVM)

Support Vector Machine is a supervised learning model used for classification and regression analysis. It works by finding the hyperplane that best separates the data points of different classes in the feature space. The optimal hyperplane maximizes the margin between the classes (Cortes and Vapnik, 1995).

Equation:

$$f(x) = \mathrm{sign}(\mathbf{w} \cdot \mathbf{x} + b)$$

2.4 Gradient Boosting

Gradient Boosting is an ensemble technique that builds models sequentially. Each new model attempts to correct the errors made by the previous models. It optimizes a loss function by combining weak learners, typically decision trees, to form a strong learner (Friedman, 2001).

Equation:

$$F_m(x) = F_{m-1}(x) + \eta h_m(x)$$

2.5 Key Performance Metrics:

Accuracy: The ratio of correctly predicted instances to the total instances.

Precision: The ratio of correctly predicted positive observations to the total predicted positives.

Recall : The ratio of correctly predicted positive observations to all observations in the actual class.

F1- score: The weighted average of precision and recall, providing a balance between the two metrics.

ROC AUC (Receiver Operating Characteristic Area Under the Curve):
A performance measurement for classification problems, representing the ability of the model to distinguish between classes (Fawcett, 2006).

These metrics provide insights into the models' effectiveness in predicting prenatal depression and help in comparing the performance of different machine learning algorithms.

3 Method

The goal is to predict depression during pregnancy using power state data and EPDS scores, aiming for high accuracy, precision, recall, F1-score, and ROC AUC. Data were collected from the Mom2B mobile app, which uses Beiwe technology to gather high-resolution sensor data and self-reported surveys, including the EPDS (Smith et al., 2022, Bilal et al., 2020).

3.1 Understanding the Variables:

Load the dataset and inspect its structure to understand the types of events and levels.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14323018 entries, 0 to 14323017
Data columns (total 3 columns):
 #   Column          Dtype
---  ------          -----
 0   id              object
 1   timestamp       object
 2   event_combined  object
dtypes: object(3)
memory usage: 327.8+ MB
```

	id	timestamp	event_combined
count	14323018	14323018	14323018
unique	526	12152615	14
top	go24wyho	2021-12-17 23:13:29.600	Screen turned on
freq	230465	223	2758891

Figure 1: Power State data info and basic statistics

ID: The unique identifier for each participant. A random extraction of 1000 participants to fit with this study timeline for machine learning analysis and report writing.

Timestamp: The date and time when an event was recorded

Event: The type of event recorded such as screen turned on/off, 'Power Save Mode state change signal received; device in power save state.' 'Device Idle (Doze) state change signal received; device in idle state.' 'Device Idle (Doze) state change signal received; device not in idle state.' 'Power Save Mode change signal received; device not in power save state.'.

Level: The measurement associated with each event, such as the number of activities, sleep duration, and periods of inactivity.

3.2 Preprocess and Aggregate Data

Handling Missing Data: Handle missing values by dropping rows with more than 60% missing data. For the remaining missing values, use a Simple Imputer with the 'mean' strategy. Additionally, apply the Synthetic Minority Over-sampling Technique (SMOTE) to balance the training dataset (Chawla et al., 2002).

Feature Engineering: Aggregate data by participant and over specific time periods by detecting and converting values in the event variable into categories such as sleep, inactivity, and activity. The conversions are as follows:

* Physical Activity: Sum or average steps per day.
* Sleep: Total sleep duration per night.
* Inactivity: Total duration of inactivity periods per day.

Create Summary Features: Calculate summary statistics for each participant over the pregnancy period:

* Daily Average Steps: Average steps per day.
* Total Sleep: Average sleep duration per night.
* Inactivity Periods: Frequency and duration of inactivity.

Preparing the EPDS Scores: Convert EPDS scores to a binary variable, where 1 indicates scores ≥12 (probable depression) and 0 indicates scores <12.

Exploratory Data Analysis (EDA): Evaluation Perform EDA to understand the distribution of features and their relationship with the outcome.
Visualization: Use box plots and distribution plots to visualize the data distributions of daily activities (sleep, activity, and inactivity) and correlations matrix.

	avg_daily_sleep_duration	avg_daily_inactivity_duration
count	188.000000	194.000000
mean	45.944358	1942.645613
std	191.473133	10148.775175
min	0.000000	0.776783
25%	12.506538	715.867676
50%	17.954652	916.764633
75%	23.338979	1120.326280
max	2463.656433	137789.956024

	avg_daily_activity_duration	avg_activity_count	avg_activity_duration
count	194.000000	194.000000	194.000000
mean	1228.977756	69.837066	281.328026
std	4384.823029	47.430756	1150.847627
min	0.371550	1.000000	0.123850
25%	198.025677	42.040179	2.785088
50%	298.240400	70.562121	4.193831
75%	424.946362	92.730410	15.543915
max	34220.151031	315.288889	10841.418301

	epds_score
count	194.000000
mean	0.206186
std	0.405612
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

Figure 2: Basic Statistics of the Power State Dataset

The summary statistics reveal several important aspects of the dataset. The average daily sleep duration, inactivity duration, and activity duration show high variability among participants, as indicated by their large standard deviations and wide ranges between minimum and maximum values. This suggests that participants have diverse patterns of sleep, inactivity, and activity, which could be influenced by various factors including lifestyle, health status, and individual differences.

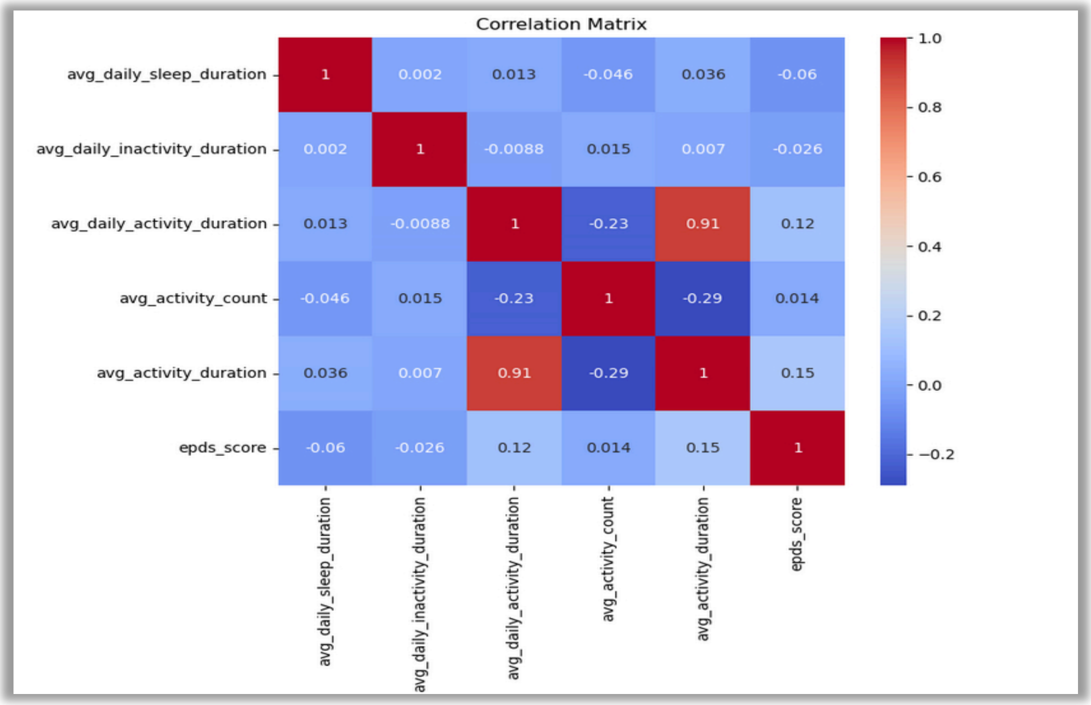

Figure 3: Correlation Matrix on the different features

Activity count shows the most promise as an individual predictor but still lacks strong correlation.

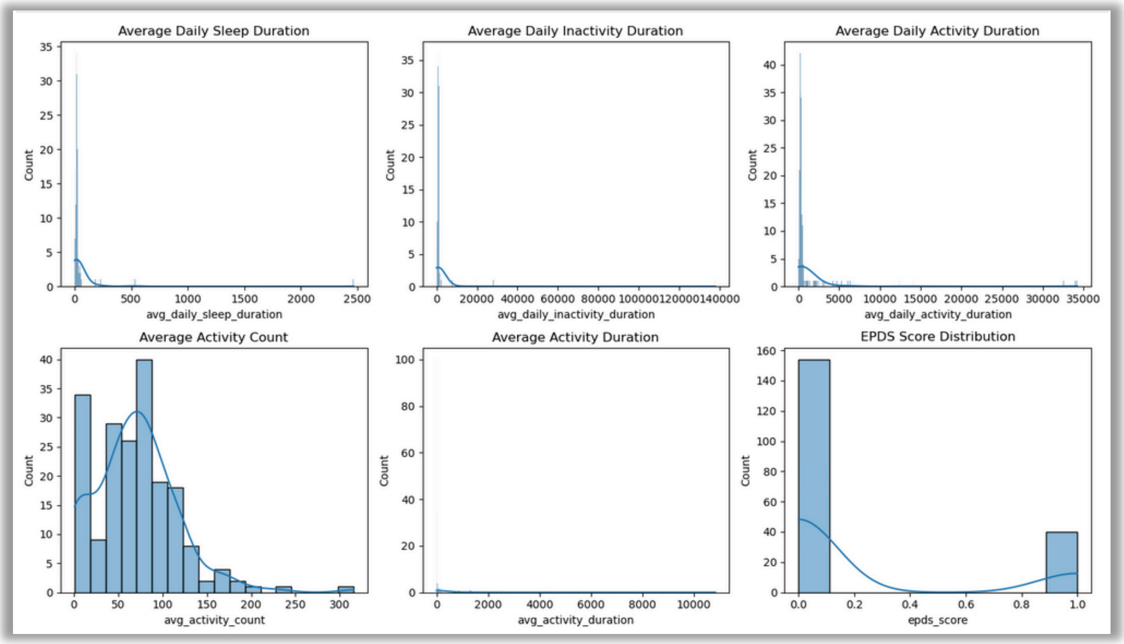

Figure 4: Distribution plot of Average Status

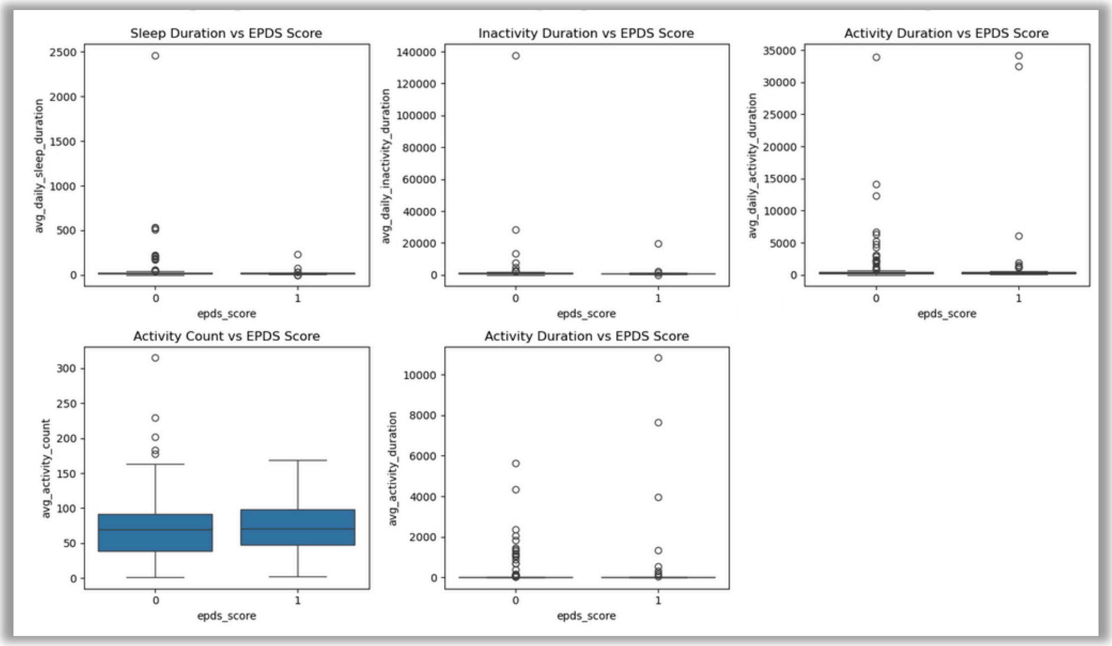

Figure 5: Distribution plot for average daily sleep, activity and inactivity

The correlations among features reveal expected relationships, such as increased activity reducing inactivity and vice versa.
Overall, the weak individual relationships suggest that a combination of features, possibly analyzed through machine learning models, would be necessary to predict prenatal depression effectively.

3.3 Data Split, Feature Normalization and Elimination:

The dataset was split into 80% training and 20% testing sets. Normalization was applied to ensure that features were on a comparable scale. Relevant features were selected based on their importance in predictive modeling. SMOTE was used to augment the minority class in the training set, ensuring a balanced dataset for training.

3.4 Machine Learning Algorithms - Model Building:

There are four machine learning algorithms were selected and trained them on the training set.
Logistic Regression for binary classification.
Random Forest for handling complex interactions and non-linear relationships.
Gradient Boosting Machines (GBM) for high-performance classification.
Support Vector Machines (SVM) for classification tasks.

Randomized Search CV was used for hyperparameter tuning, with cross-validation to generalize the models.

3.5 Model Evaluation:

Models were evaluated using accuracy, precision, recall, F1-score, ROC AUC, and confusion matrices. These metrics were used to assess model performance and identify overfitting or underfitting.

4 Results and Discussion

4.1 Results

Model	Accuracy	Precision	Recall	F1 Score	ROC AUC
Logistic Regression	0.51	0.23	0.25	0.24	0.47
Random Forest	0.56	0.35	0.50	0.41	0.58
Gradient Boosting	0.59	0.39	0.58	0.47	0.54
Support Vector Machine	0.51	0.00	0.00	0.00	0.40

Table 1: Results of Machine Learning Models

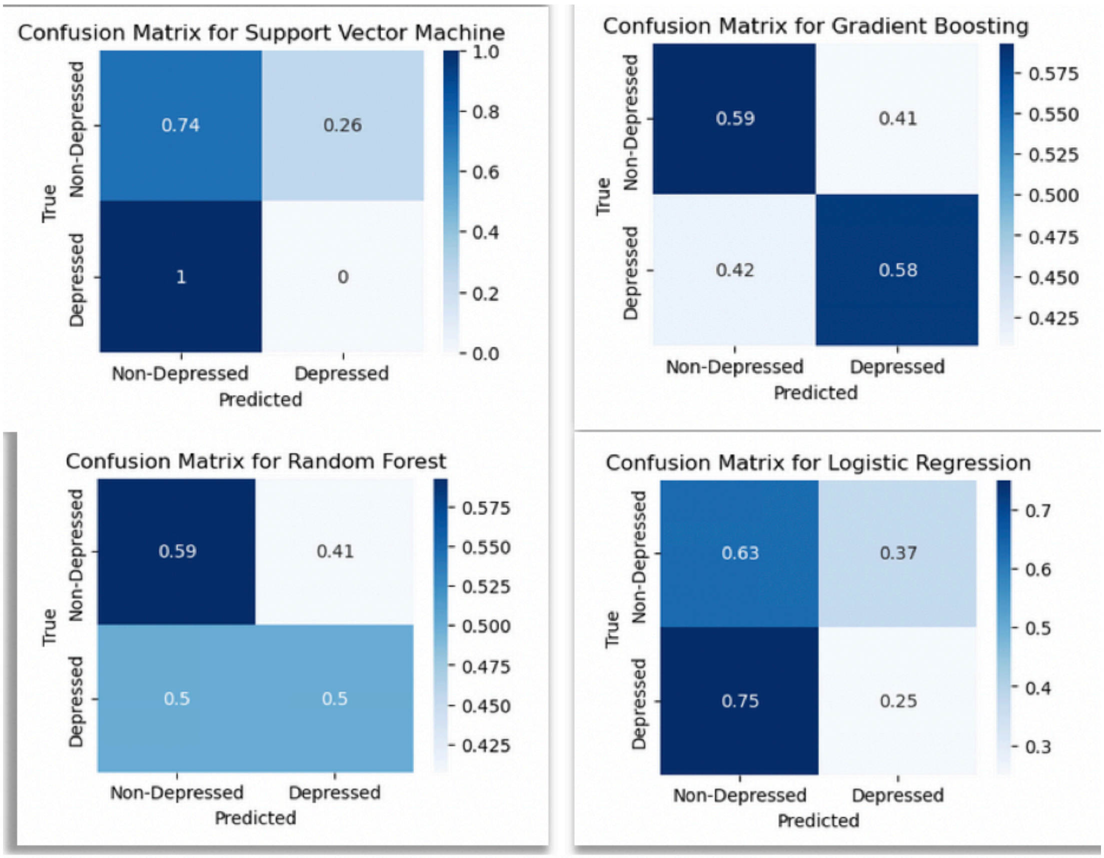

Figure 6: Confusion Matrix result for all four models

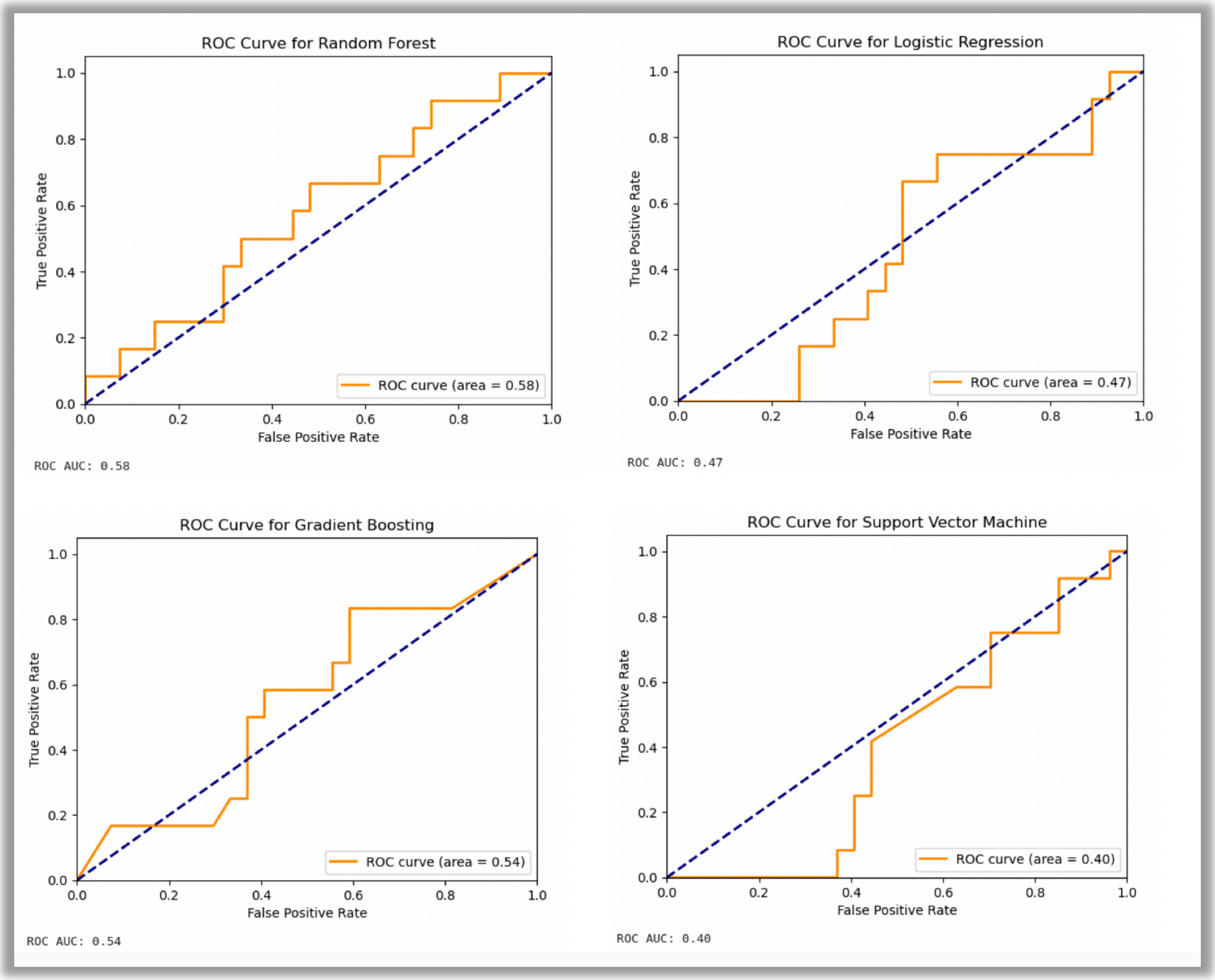

Figure 6: ROC AUC for all models that are used in this study.

4.2 Discussion

The results indicate that different models have varying degrees of success in predicting prenatal depression:

Logistic Regression:
The model's performance was moderate, with an accuracy of 0.51 and an ROC AUC of 0.47.
The precision and recall for detecting depressive symptoms were low, indicating that the model struggled to identify depressed individuals effectively.

Random Forest Classifier:
This model showed better performance, with an accuracy of 0.56 and an ROC AUC of 0.58.
The precision and recall were higher compared to Logistic Regression, suggesting better identification of depressive symptoms.
Gradient Boosting Classifier:

Gradient Boosting had the highest accuracy at 0.59 and showed balanced precision (0.39) and recall (0.58).
The ROC AUC of 0.54 indicates moderate predictive capability, making it one of the more promising models.

Support Vector Machine (SVC):
The SVC model performed poorly, with an accuracy of 0.51 and an ROC AUC of 0.40.
It failed to identify any depressive cases correctly, with precision and recall both at 0.00.

The superior performance of the Random Forest and Gradient Boosting models can be attributed to their ability to handle complex interactions and non-linear relationships in the data. The Random Forest model's ensemble nature allows it to reduce overfitting by averaging multiple decision trees, while the Gradient Boosting model improves prediction accuracy by sequentially correcting errors made by previous models.

Limitations
Some potential limitations of this study include:
* The relatively small sample size, which may limit the generalizability of the results.
* The reliance on self-reported EPDS scores, which may introduce bias.
* The need for more diverse features to improve model performance.

5 Conclusion

Research Question 1: Can power state data significantly predict depression during pregnancy when combined with EPDS scores as an outcome measure?

The study demonstrates that power state data can be utilized to predict depression during pregnancy when combined with EPDS scores as an outcome. While the performance of the models varied, the results suggest that there is a potential for using power state data in this context. Random Forest and Gradient Boosting classifiers showed moderate success, indicating that power state data holds valuable information that correlates with depressive symptoms.

Research Question 2: How well can machine learning algorithms predict prenatal depression based on these data sources?

The machine learning models exhibited varying degrees of success in predicting prenatal depression. Gradient Boosting model showed the best performance with an accuracy of 0.59 and reasonable precision and recall scores, suggesting it can moderately predict depressive symptoms. Random Forest model also performed relatively well, indicating its suitability for such predictive tasks. Logistic Regression model had limited predictive power and struggled with identifying depressive cases accurately. The SVC model performed the worst, failing to identify depressive cases.

The study explored the use of machine learning algorithms to predict depression during pregnancy using power state data and EPDS scores. The findings are as follows:
Random Forest and Gradient Boosting models showed the most promise, with moderate accuracy and ability to detect depressive symptoms.
Logistic Regression demonstrated limited effectiveness, and the SVC model performed poorly.

Future work should focus on improving data quality, incorporating more diverse features, and exploring advanced machine learning techniques to enhance predictions accuracy.

Appendix A

Machine Learning Code in Jupyter Notebook:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.impute import SimpleImputer
from imblearn.over_sampling import SMOTE
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier
from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, roc_auc_score, confusion_matrix, classification_report
```

```
data = pd.read_csv('updated_dataset_with_activity.csv')

epds_column = 'epds_score'

#split data
X = data.drop(columns=['id', epds_column])
y = data[epds_column]

#split into training/testing
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

print("Data split into training/test sets successfully.")
```

```
Data split into training/test sets successfully.
```

```
logistic_regression = LogisticRegression(random_state=42)
random_forest = RandomForestClassifier(random_state=42)
gradient_boosting = GradientBoostingClassifier(random_state=42)
svm = SVC(random_state=42, probability=True)

print("Models initialized successfully.")

#hyperparameter grids
param_grid_lr = {
    'C': [0.01, 0.1, 1, 10, 100],
    'penalty': ['l1', 'l2'],
    'solver': ['liblinear']
}

param_grid_rf = {
    'n_estimators': [50, 100, 200],
    'max_depth': [None, 10, 20, 30],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}

param_grid_gb = {
    'n_estimators': [50, 100, 200],
    'learning_rate': [0.01, 0.1, 0.2],
    'max_depth': [3, 5, 7],
    'subsample': [0.8, 1.0],
    'min_samples_split': [2, 5, 10]
}
```

```python
param_grid_svm = {
    'C': [0.1, 1, 10, 100],
    'gamma': [1, 0.1, 0.01, 0.001],
    'kernel': ['linear', 'rbf']
}

# gridSearchCV
grid_lr = GridSearchCV(LogisticRegression(random_state=42), param_grid_lr, cv=5, scoring='accuracy')
grid_rf = GridSearchCV(RandomForestClassifier(random_state=42), param_grid_rf, cv=5, scoring='accuracy')
grid_gb = GridSearchCV(GradientBoostingClassifier(random_state=42), param_grid_gb, cv=5, scoring='accuracy')
grid_svm = GridSearchCV(SVC(random_state=42, probability=True), param_grid_svm, cv=5, scoring='accuracy')


grids = {
    'Logistic Regression': grid_lr,
    'Random Forest': grid_rf,
    'Gradient Boosting': grid_gb,
    'Support Vector Machine': grid_svm
}

#train-evaluate each model
for name, grid in grids.items():
    print(f"Training {name}...")
    grid.fit(X_train_balanced, y_train_balanced)
    print(f"Best parameters for {name}: {grid.best_params_}")
    print(f"Best cross-validation accuracy for {name}: {grid.best_score_:.2f}")

print("Models trained and evaluated with hyperparameter tuning successfully.")
```

Result

```
Models initialized successfully.
Training Logistic Regression...
Best parameters for Logistic Regression: {'C': 0.1, 'penalty': 'l1', 'solver': 'liblinear'}
Best cross-validation accuracy for Logistic Regression: 0.52
Training Random Forest...
Best parameters for Random Forest: {'max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 2, 'n_estimators': 200}
Best cross-validation accuracy for Random Forest: 0.75
Training Gradient Boosting...
Best parameters for Gradient Boosting: {'learning_rate': 0.01, 'max_depth': 5, 'min_samples_split': 10, 'n_estimators': 50, 'subsample': 1.0}
Best cross-validation accuracy for Gradient Boosting: 0.77
Training Support Vector Machine...
Best parameters for Support Vector Machine: {'C': 10, 'gamma': 0.001, 'kernel': 'rbf'}
Best cross-validation accuracy for Support Vector Machine: 0.78
Models trained and evaluated with hyperparameter tuning successfully.
```

```python
#evaluate model on test set by best param
def test_best_model(model, X_test, y_test):
    y_pred = model.predict(X_test)
    y_proba = model.predict_proba(X_test)[:, 1] if hasattr(model, 'predict_proba') else y_pred
    print(f"Model: {model.estimator.__class__.__name__}")
    print(f"Best Parameters: {model.best_params_}")
    print(f"Accuracy: {accuracy_score(y_test, y_pred):.2f}")
    print(f"Precision: {precision_score(y_test, y_pred):.2f}")
    print(f"Recall: {recall_score(y_test, y_pred):.2f}")
    print(f"F1 Score: {f1_score(y_test, y_pred):.2f}")
    print(f"ROC AUC: {roc_auc_score(y_test, y_proba):.2f}")
    print("Confusion Matrix:")
    print(confusion_matrix(y_test, y_pred))
    print("Classification Report:")
    print(classification_report(y_test, y_pred))

for name, grid in grids.items():
    test_best_model(grid, X_test_imputed, y_test)
```

```
Model: LogisticRegression
Best Parameters: {'C': 0.1, 'penalty': 'l1', 'solver': 'liblinear'}
Accuracy: 0.51
Precision: 0.23
Recall: 0.25
F1 Score: 0.24
ROC AUC: 0.47
Confusion Matrix:
[[17 10]
 [ 9  3]]
Classification Report:
              precision    recall  f1-score   support

           0       0.65      0.63      0.64        27
           1       0.23      0.25      0.24        12

    accuracy                           0.51        39
   macro avg       0.44      0.44      0.44        39
weighted avg       0.52      0.51      0.52        39

Model: RandomForestClassifier
Best Parameters: {'max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 2, 'n_estimators': 200}
Accuracy: 0.56
Precision: 0.35
Recall: 0.50
F1 Score: 0.41
ROC AUC: 0.58
Confusion Matrix:
[[16 11]
 [ 6  6]]
Classification Report:
              precision    recall  f1-score   support

           0       0.73      0.59      0.65        27
           1       0.35      0.50      0.41        12

    accuracy                           0.56        39
   macro avg       0.54      0.55      0.53        39
weighted avg       0.61      0.56      0.58        39
```

```
Model: GradientBoostingClassifier
Best Parameters: {'learning_rate': 0.01, 'max_depth': 5, 'min_samples_split': 10, 'n_estimators': 50, 'subsample': 1.0}
Accuracy: 0.59
Precision: 0.39
Recall: 0.58
F1 Score: 0.47
ROC AUC: 0.54
Confusion Matrix:
[[16 11]
 [ 5  7]]
Classification Report:
              precision    recall  f1-score   support

           0       0.76      0.59      0.67        27
           1       0.39      0.58      0.47        12

    accuracy                           0.59        39
   macro avg       0.58      0.59      0.57        39
weighted avg       0.65      0.59      0.61        39

Model: SVC
Best Parameters: {'C': 10, 'gamma': 0.001, 'kernel': 'rbf'}
Accuracy: 0.51
Precision: 0.00
Recall: 0.00
F1 Score: 0.00
ROC AUC: 0.40
Confusion Matrix:
[[20  7]
 [12  0]]
Classification Report:
              precision    recall  f1-score   support

           0       0.62      0.74      0.68        27
           1       0.00      0.00      0.00        12

    accuracy                           0.51        39
   macro avg       0.31      0.37      0.34        39
weighted avg       0.43      0.51      0.47        39

Model evaluation with best parameters completed successfully.
```

References

Bilal, M. et al. (2020) 'The Mom2B app: assessing prenatal mental health through smartphone-based data collection', Journal of Medical Internet Research, 21(11), p.e13463.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), pp.5-32.
Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), pp.273-297.

Chawla, N.V., Bowyer, K.W., Hall, L.O., & Kegelmeyer, W.P. (2002). "SMOTE: synthetic minority over-sampling technique." Journal of Artificial Intelligence Research, 16, 321-357.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), pp.273-297.

Cox, J.L., Holden, J.M., & Sagovsky, R. (1987). "Detection of postnatal depression. Development of the 10-item Edinburgh Postnatal Depression Scale." The British Journal of Psychiatry, 150(6), 782-786.

Dagum, P., & Ruiz, C. (2020). "Digital biomarkers of cognitive function." npj Digital Medicine, 3(1), 1-7.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), pp.861-874.

Friedman, J.H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), pp.1189-1232.

Gavin, N.I. et al. (2005). "Perinatal depression: a systematic review of prevalence and incidence." Obstetrics & Gynecology, 106(5), 1071-1083.

Hosmer, D.W. and Lemeshow, S. (2000). *Applied Logistic Regression*. 2nd ed. New York: John Wiley & Sons.

Kim, H. et al. (2021). "Using machine learning to predict postnatal depression." Journal of Affective Disorders, 290, 378-384.

Kim, J., Cho, S. and Park, Y. (2021). Applications of Machine Learning in Predicting Mental Health Conditions from Wearable Device Data: A Scoping Review. *Journal of Medical Internet Research*, 23(4), e25705.

Obermeyer, Z., & Emanuel, E.J. (2016). "Predicting the future—big data, machine learning, and clinical medicine." The New England Journal of Medicine, 375(13), 1216-1219.

O'Hara, M.W., & Swain, A.M. (1996). "Rates and risk of postpartum depression—a meta-analysis." International Review of Psychiatry, 8(1), 37-54.