Mini Research Paper on "Predicting Well Being with Machine Learning and Explainable AI"

Name: Aditya Dobhal

Roll No: R2142230377

SAP ID: 500119731

Abstract

This paper suggests designing a few socio-economic factors-based machine learning model to predict well-being. Using XAI techniques improves interpretability, making the predictions more transparent and actionable for decision-makers. The project will support the achievement of United Nations SDG 3, Good Health and Well-being. The model incorporates a Random Forest algorithm with SHAP (SHapley Additive exPlanations) to highlight what specific features contribute to predictions in well-being and thus reveals more deeply layered factors that might come into determining well-being outcomes.

Introduction

Over the last few years, there has been a significant spotlight on the need for interpretable machine learning models, especially for applications that deal with social and economic data where a clear explanation and understanding of factors resulting in predictions are involved. Even though it can be extremely accurate, black-box models lack transparency. For such an application domain, where trust and accountability are of great importance, it is not practical enough to be used. Here, the tools about explainable AI, like SHAP, come into a scenario and help explain what makes model predictions the way they do, implying for the user which variables influence specific outcomes. This project applies the range of machine learning algorithms and XAI to well-being predictions based on socioeconomic indicators for SDG 3: Good Health and Well-being.

Data and Methodology

Data Source and Preprocessing

This is the training set for the model, which is the 2024 World Happiness Report from Kaggle. The dataset captures a wide range of the socioeconomic factors that would impact the well-being of a country while linking different indicators such as Gross Domestic Product per capita, social support, life expectancy, freedom in making life choices, generosity, and perceptions of corruption. After an initial inspection for missing values removed in order to maintain consistency while training, it features:

1). Gross domestic product per capita

2). Social support

3). Life expectancy in good health

4). Freedom to make your own choices

5). Generosity

6). Perceptions of corruption

The dependent variable, "Ladder score", is the well-being score that the model forecasts for quantifying the life satisfaction as perceived.

Code Snippet: Data Loading and Preprocessing

```python
# Load the dataset
url = 'WHR2024.csv'
df = pd.read_csv(url)

# Drop rows with missing values and select features
df = df.dropna()
X = df.drop(columns=['Country name', 'Ladder score', 'upperwhisker', 'lowerwhisker'])
y = df['Ladder score']

# Standardize numerical features
scaler = StandardScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)
```

Model Training and Resampling with SMOTE

To obtain balanced sets and further enhance model performance, the Synthetic Minority Over-sampling Technique, if applicable, was applied. SMOTE creates synthetic samples of underrepresented classes to better represent the full dataset and promote balanced learning. Once resampling was done, data was divided into training and testing sets in preparation for further evaluation.

Code Snippet: SMOTE Application And Data Splitting

```python
# Apply SMOTE if applicable
try:
    smote = SMOTE(sampling_strategy='auto', k_neighbors=2)
    X_resampled, y_resampled = smote.fit_resample(X, y)
except ValueError as e:
    print(f"SMOTE could not be applied: {e}")
    X_resampled, y_resampled = X, y

# Split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled, test_size=0.2,
```

Model Selection and Evaluation

In this research, the three regression models that were considered include Random Forest, Gradient Boosting and Support Vector Regression known as SVR. The 5-fold cross validation technique was utilized to evaluate models for generalisability over unseen data. Evaluations of the models targeted three primary metrics: Mean Squared Error, Mean Absolute Error, and the R^2 score; these measure the accuracy and reliability of the predictions provided by the models. The final selection of the model was a Random Forest model because it performed well across all metrics.

```python
regressors = {
    "Random Forest": RandomForestRegressor(random_state=42),
    "Gradient Boosting": GradientBoostingRegressor(random_state=42),
    "SVM": SVR()
}
kf = KFold(n_splits=5, shuffle=True, random_state=42)

for name, model in regressors.items():
    cv_scores = cross_val_score(model, X_train, y_train, cv=kf, scoring='neg_mean_squared_err
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)

    mse = mean_squared_error(y_test, y_pred)
    mae = mean_absolute_error(y_test, y_pred)
    r2 = r2_score(y_test, y_pred)

    print(f"{name} Regressor:")
    print(f"  Cross-Validation MSE: {-cv_scores.mean():.4f}")
    print(f"  Test MSE: {mse:.4f}")
    print(f"  Test MAE: {mae:.4f}")
    print(f"  Test R^2: {r2:.4f}\n")
```

The model was saved, by making use of the joblib library, enabling the effective integration into a Flask Web Application that was designed for deployment as well as interaction with the actual end-users.

Code Snippet: Model Saving

```python
import joblib
joblib.dump(model, 'random_forest_model.pkl')
joblib.dump(scaler, 'scaler.pkl')
```

SHAP-based Explanatory AI

SHAP values-based Interpretations can explain how each of the features contributes to the prediction on the well-being score. The study used SHAP's KernelExplainer to visualize the impact of each feature on individual predictions. This is the way through which end-users might trace how factors like GDP per capita or social support come into influence on specific well-being scores, thus making the model behavior more transparent and more understandable. The use of SHAP will bring an important element of interpretability that helps to recognize the most informative socioeconomic factors in determination levels by policymakers and researchers.

Code Snippet: SHAP Analysis and Visualization

```python
import shap
import matplotlib.pyplot as plt
from io import BytesIO
import base64

# Generate SHAP values for visualization
explainer = shap.KernelExplainer(model.predict, shap.sample(X_test, 100))
shap_values = explainer.shap_values(X_test)

plt.figure()
shap.force_plot(explainer.expected_value, shap_values[0], X_test.iloc[0], matplotlib=True, show=False)
buf = BytesIO()
plt.savefig(buf, format="png", bbox_inches="tight")
buf.seek(0)
shap_img = base64.b64encode(buf.getvalue()).decode("utf-8")
```

Conclusion

This paper demonstrates the development of a machine learning model that can accurately predict well-being in an interpretable manner. This project is an example of the practical framework for delivering transparent predictions of well-being by using the Random Forest model with SHAP inside a Flask application. Such a tool shall contribute to achieving SDG 3: Good Health and Well-being since it will make the above factors better understood and addressed in more detail by policymakers and researchers. The approach used here could extend to a broad range of other domains, where drivers behind model predictions would need to be understood for actionable insights-that further reinforces the role of interpretable machine learning in socially relevant applications.