

Predicting Divorce

This project uses various classifier models to predict whether a couple is likely to divorce based on responses from the Divorce Predictors Scale (DPS) questionnaire. The models performed include logistic regressions, decision tree classifiers, random forest classifiers, and support vector classifiers. In this project, we eliminate redundant features, select the most relevant features, and perform grid searches to optimize each model's performance. While all models provided accurate predictions, some outperformed others in terms of precision and recall.

Citation

Please cite the dataset authors if you use these data. The dataset can be cited as follows: Rabie El Kharoua. (2020). *Split or stay: Divorce predictor dataset*. Kaggle.
<https://www.kaggle.com/datasets/rabieelkharoua/split-or-stay-divorce-predictor-dataset?resource=download>

Data Key

This project uses a CSV file from Kaggle that is made up of responses to the DPS. The dataset includes the following columns:

- Atr1-54: Responses to individual questionnaire items in numeric data types
- Class: Indicates whether the couple is married (1) or divorced (0)

Project Features

These models evaluate the likelihood of a couple being divorced based on their DPS responses. This project runs multiple classification models to predict the outcome, performs feature selection to enhance model efficiency, and conducts grid searches to fine-tune the models for better performance.

After processing the data, the system outputs the accuracy and classification report for each model, allowing the user to compare their effectiveness. The project is designed to be flexible, allowing for the addition or removal of models, as well as freedom for parameter customization.

Installations and Requirements

The following Python libraries are required for this project:

- pandas
- matplotlib.pyplot
- seaborn
- sklearn.feature_selection: RFECV, RFE
- sklearn.model_selection: train_test_split, GridSearchCV
- sklearn.metrics: accuracy_score, classification_report
- sklearn.linear_model: LogisticRegression

- sklearn.preprocessing: StandardScaler
- sklearn.tree: DecisionTreeClassifier
- sklearn.ensemble: RandomForestClassifier
- sklearn.svm: SVC

Using the Classifier Models

These models were created in Jupyter Notebook, so that is where we recommend using it. However, you could use this on any Python IDE you prefer, such as PyCharm. If you wish to use it in Jupyter Notebook, download the .ipynb file for use in your own Jupyter Notebook or copy each cell into your Jupyter Notebook.

This project will load the dataset, preprocess the data, perform feature selection, and then apply each classifier to predict whether a couple is divorced based on the questionnaire responses. After executing each cell, the IDE will display the performance metrics of each classifier model.

Contact

For any questions or concerns, please feel free to contact me, Ahria Dominguez, at ahriadominguez@outlook.com.