AutoFinancer

Wiktoria Boguszewska
Mateusz Zacharecki

22 January 2025

What is AutoFinancer?

AutoFinancer is a package for automated machine learning, which offers an end-to-end solution for data preprocessing, model selection, hyperparameter optimization and result evaluation.

This package is intended for:

- **Bank employees**
- **Financial analysts**
- **Employees of financial institutions**
- **Risk management specialists**
- **Data scientists and machine learning engineers** working in the financial sector.

Package Specialization

The AutoFinancer package supports three main types of machine learning problems:

1. **Binary Classification**
2. **Multiclass Classification**
3. **Regression**

Comparison with other Packages

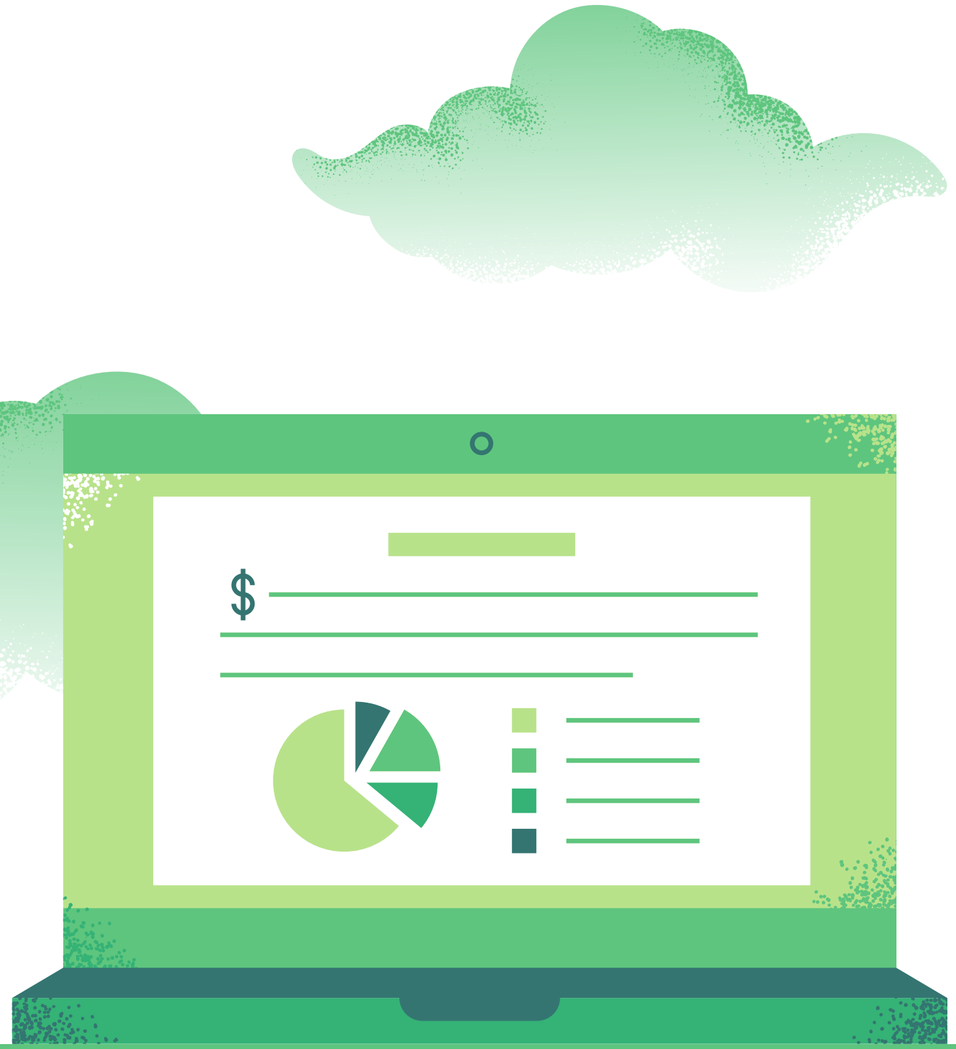
Popular tools such as **AutoGluon** and **mljar** provide similar functionalities.

However, **AutoFinancer** offers several key differentiators:

- Customization and Flexibility
- Integrated Report Generation
- Specified Target Audience
- Performance Tuning
- Problem Type Specialization

Components

1. **Automated Model Training and Optimization** (`train` method):
   - **Data Preprocessing**: Prepares the input data (`X`) and target variable (`y`) for model training. This includes handling missing values, feature scaling, and encoding categorical variables.
   - **Model Selection and Optimization**: Selects the best model from a pool of candidates and tunes its hyperparameters, using methods like `RandomizedSearchCV` and `GridSearchCV` (optional).
   - **Report Generation**: Produces detailed performance reports summarizing the training and optimization process.
2. **Prediction** (predict method).
3. **Probability Prediction** (predict_proba method).
4. **Prediction and Report Generation** (predict_and_report method)



Preprocessing

AutoFinancer includes comprehensive preprocessing functions to handle missing data, normalize features, and encode categorical variables. This section describes the key methods responsible for preparing the data before model training.

Key methods:

- Converting target variable y,
- Converting input data X,
- Handling missing values in y,
- Processing target variable (y),
- Identifying binary, numerical, categorical, and datetime columns,
- Handling missing data,
- Standardizing numerical features,
- Encoding categorical variables,
- Feature selection

AutoFinancer uses multiple feature selection techniques to identify the most relevant features:

- **Correlation-based Feature Removal**: The package identifies pairs of highly correlated features and removes one of them if their correlation exceeds a specified threshold (default: 0.9).
- **Random Forest-based Feature Selection**: This method uses the feature importances provided by a *RandomForestClassifier* to rank and select the top-k features.
- **Statistical Feature Selection with SelectKBest**: This method uses statistical tests, specifically ANOVA F-tests (*f_classif*), to rank and select the top-k features based on their relationship with the target variable.
- **Combined Method**: The combined method merges the results from both the Random Forest-based and statistical selection methods. Only the features selected by both approaches are retained for the final model.

Feature Selection

Model Selection and Optimization

AutoFinancer supports automatic model selection and hyperparameter tuning using two primary methods:

- **Random search**
- **Grid search** (optional)

It also uses **k-fold cross-validation** for model evaluation during hyperparameter optimization

The package supports following models:

- **Random Forest**
- **Decision Tree**
- **XGBoost**
- **GradientBoosting**
- **Logistic Regression**
- **Lasso Regression**
- **RidgeRegression**
- **LDA**
- **QDA**
- **Linear Regression**

The package automatically selects appropriate metrics based on the problem type:

Classification Metrics:

- Default Metric: *accuracy*
- *roc_auc*
- *precision*
- *recall*
- *f1*
- *balanced_accuracy*

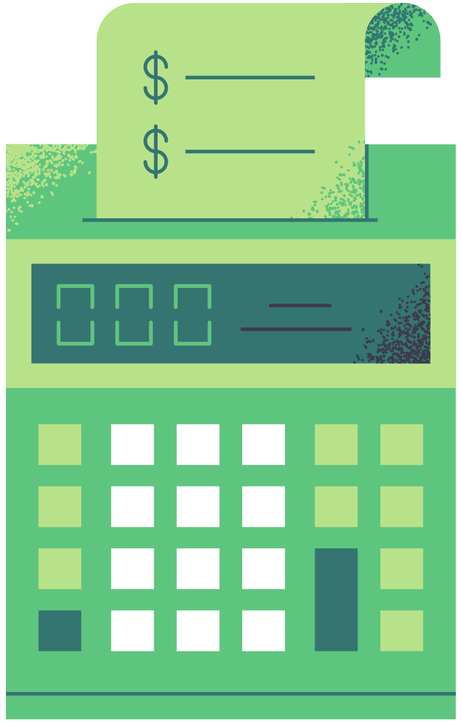
Regression Metrics:

- Default Metric: *r2*
- *neg_mean_squared_error*
- *neg_mean_absolute_error*

Reporting

After model training, the package generates a comprehensive report that includes:

- Summary with basic results for the best model
- A tabular summary showing the model names, best parameters, scores, evaluation metrics, and training times
- Overall descriptive statistics of the processed dataset for numerical variables
- Histogram for target variable y with its distribution
- Classes breakdown for target variable y
- Permutation Importance
- Partial Dependence Profiles (PDP)
- Accumulated Local Effects (ALE)
- Break Down plots
- Ceteris Paribus plots
- Repots based on specifics of each model



Innovative Approach

- Advanced model interpretability based on **dalex** package methods
- Adaptability to several problem types
- Customizable hyperparameter optimization
- Feature selection customization
- Advanced reporting:
  - Summary reports
  - Individual model reports
  - Reporting on test dataset

Tutorial

The tutorial will guide you through:

1. Loading and exploring data.
2. Creating and configuring the **AutoFinancer** object.
3. Training models and generating a report.
4. Making predictions on new data.
5. Generating report on test dataset.

```python
from autofinancer import AutoFinancer

pipeline = AutoFinancer()
pipeline.train(X, y)
```

THANK YOU