Longitudinal data analysis in R: Residuals, model diagnostics, and variable importance

Nelson Nazzicari

(bioinformatician & ML engineer) **CREA**, Lodi (Italy)

Analysis of residuals

Reminder:

- X: features (predictors, independent variables)
- y: target (dependent variable)
- $\hat{y}$: predictions (fitted values)

- assumption: $y = f(X) + N(0, \sigma^2)$

- practical: $\hat{y}$ = TRAINED_MODEL(X)

- residuals := $y - \hat{y}$

Analysis of residuals

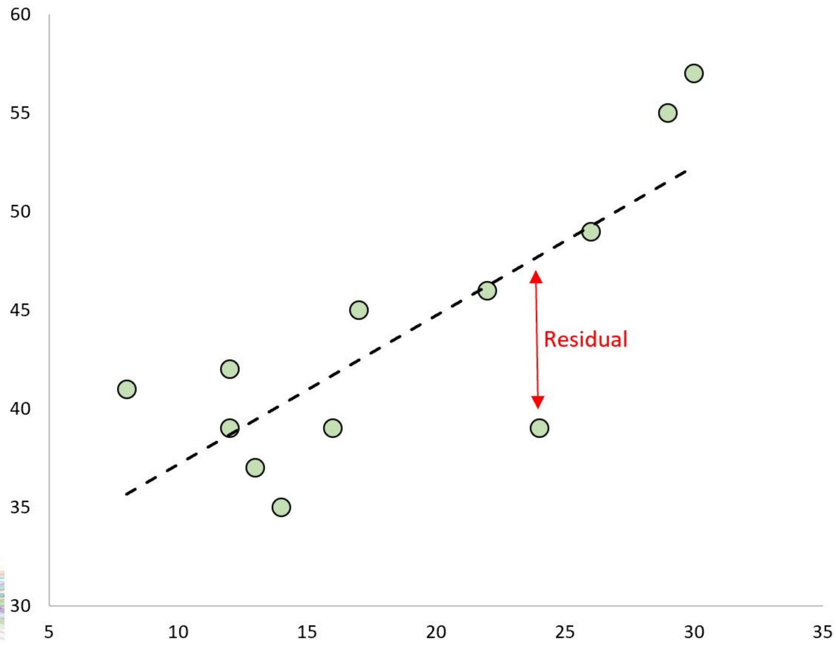
Reminder:

- X: features (predictors, independent variables)
- y: target (dependent variable)
- ŷ: predictions (fitted values)

- assumption: $y = f(X) + N(0, \sigma^2)$

- practical: ŷ = TRAINED_MODEL(X)

- residuals := y - ŷ



Analysis of residuals

Well behaved residuals:

- Zero mean
- Symmetrically distributed
  - Actually, *normally* distributed
- Independent from fitted values

Notebook on analysis of residuals

Model diagnostics (traditional)

- Information criteria (expected negative, the lower the better)
- Relative measures, useful for comparison
- Akaike information criterion
  - $AIC = 2k - 2\, ln(L)$
- Bayesian information criterion
  - $BIC = k\, ln(n) - 2\, ln(L)$

With:

- k: number of parameters in the model
- *ln*: natural logarithm
- L (Likelihood): measures how well the model fits the data. Higher is better
  - (it’s the probability of obtaining the data given the model)
- n: number of samples used to train the model

Model diagnostics (ML)

- Crossvalidation, overfitting
- Training cost (hardware, software, time)
  - Prediction cost
- No model is “true”
- Error analysis
  - Confusion matrix
  - Sample subsets
  - Combining models

Feature importance

Some natural questions

- what are the most important variables?
- what can be removed from the analysis?

Linear model: easy

$$\hat{y} = w_1 * x_1 + w_2 * x_2 + w_2 * x_2 + \ldots + w_M * x_M$$

Random forest: easy (sort of)

- Internally, ensemble learners MUST have metrics for feature importance
    - They must do something called “impurity reduction”
- Depending on the library, it can be easy (or hard) to extract the info

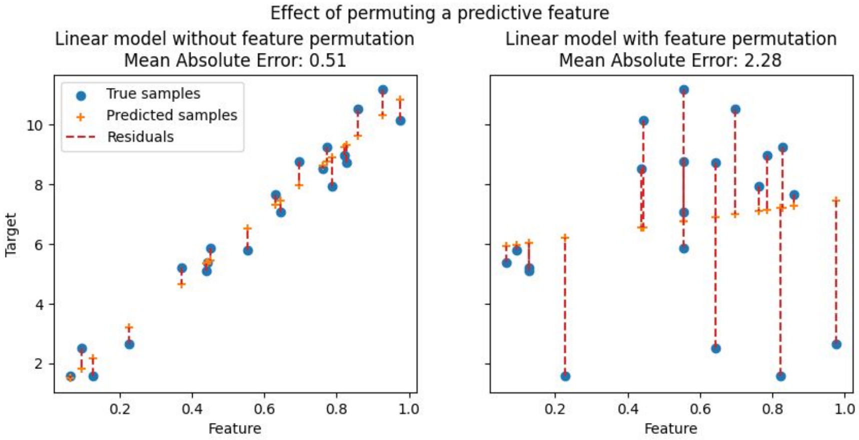
```
from sklearn.ensemble import RandomForestRegressor
model = RandomForestRegressor()
model.fit(X, y)
importance = pd.Series(model.feature_importances_)

import xgboost as xgb
model = xgb.XGBClassifier()
model.fit(X, y)
xgb.plot_importance(model)
```

Generic model: permutation

- Record the model performance
- Permute (shuffle) one feature
- Record the performance drop
- Maybe repeat the process a few times

Generic model: permutation

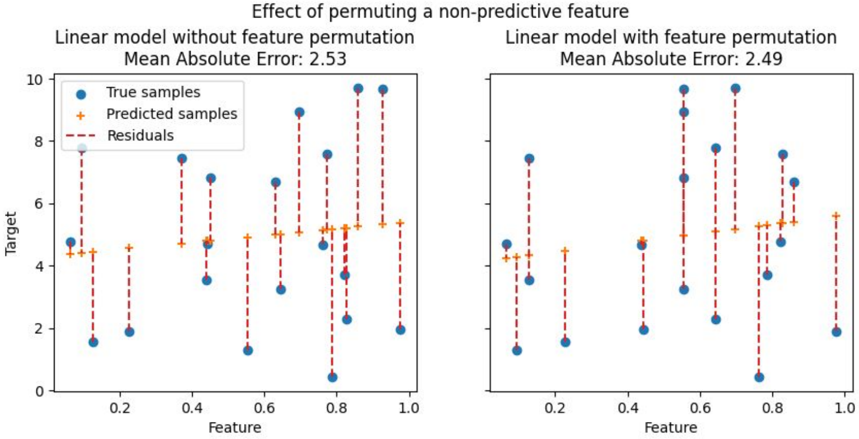
- Record the model performance
- Permute (shuffle) one feature
- Record the performance drop
- Maybe repeat the process a few times



Effect of permuting a predictive feature

Linear model without feature permutation
Mean Absolute Error: 0.51

Linear model with feature permutation
Mean Absolute Error: 2.28



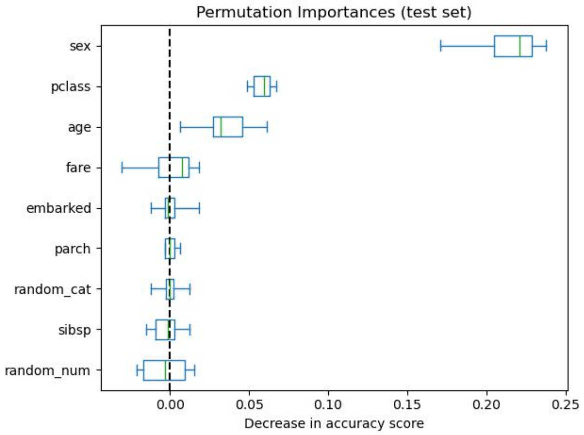
Effect of permuting a non-predictive feature

Linear model without feature permutation
Mean Absolute Error: 2.53

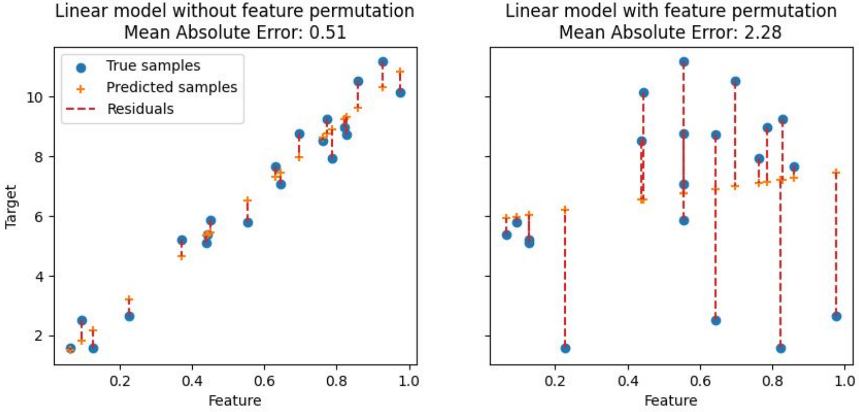
Linear model with feature permutation
Mean Absolute Error: 2.49

Generic model: permutation

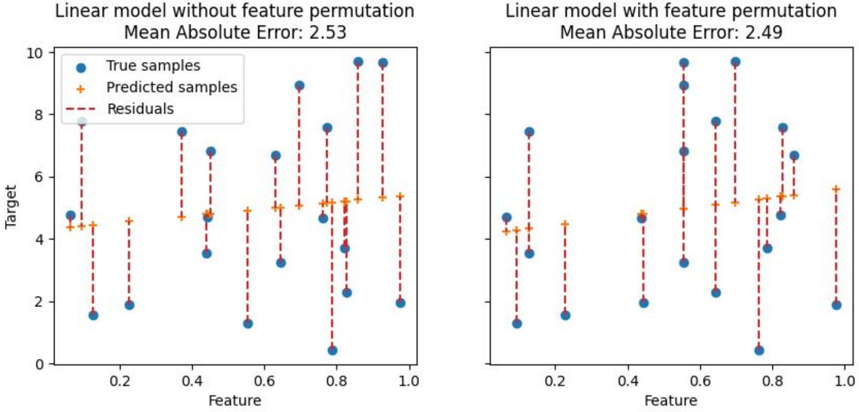
- Record the model performance
- Permute (shuffle) one feature
- Record the performance drop
- Maybe repeat the process a few times



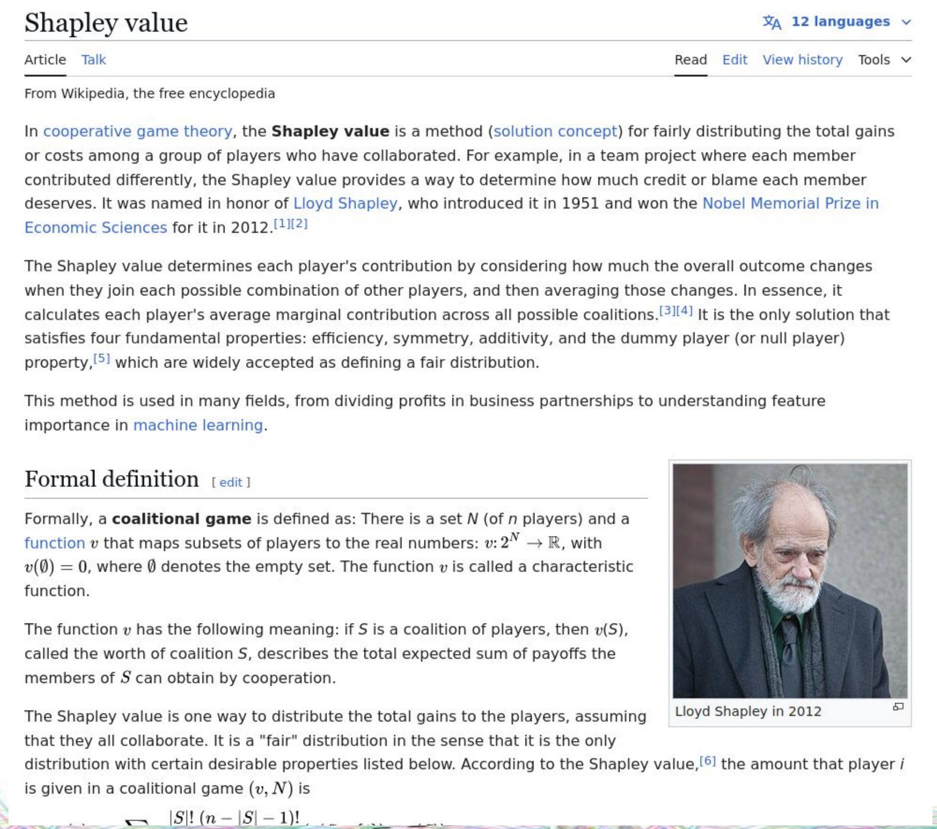
Effect of permuting a predictive feature



Effect of permuting a non-predictive feature



Generic model: Shapley values



Shapley value

From Wikipedia, the free encyclopedia

In cooperative game theory, the **Shapley value** is a method (solution concept) for fairly distributing the total gains or costs among a group of players who have collaborated. For example, in a team project where each member contributed differently, the Shapley value provides a way to determine how much credit or blame each member deserves. It was named in honor of Lloyd Shapley, who introduced it in 1951 and won the Nobel Memorial Prize in Economic Sciences for it in 2012.[1][2]

The Shapley value determines each player's contribution by considering how much the overall outcome changes when they join each possible combination of other players, and then averaging those changes. In essence, it calculates each player's average marginal contribution across all possible coalitions.[3][4] It is the only solution that satisfies four fundamental properties: efficiency, symmetry, additivity, and the dummy player (or null player) property,[5] which are widely accepted as defining a fair distribution.

This method is used in many fields, from dividing profits in business partnerships to understanding feature importance in machine learning.

Formal definition

Formally, a **coalitional game** is defined as: There is a set *N* (of *n* players) and a function v that maps subsets of players to the real numbers: $v\colon 2^N \to \mathbb{R}$, with $v(\emptyset) = 0$, where $\emptyset$ denotes the empty set. The function v is called a characteristic function.

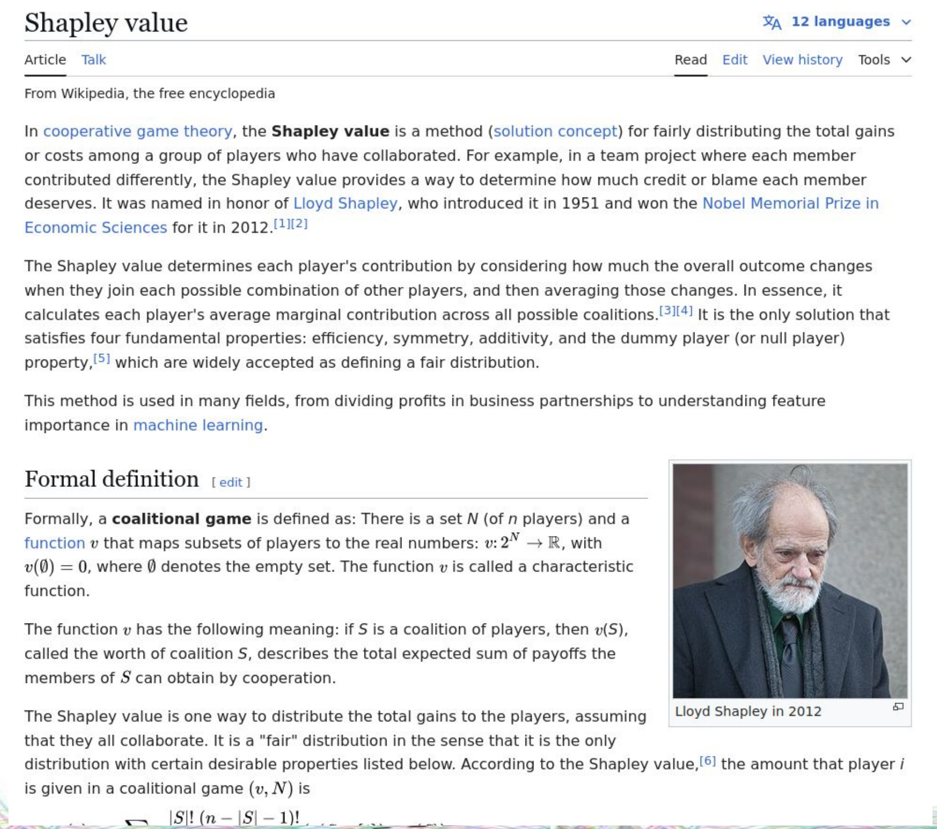
The function v has the following meaning: if *S* is a coalition of players, then $v(S)$, called the worth of coalition *S*, describes the total expected sum of payoffs the members of S can obtain by cooperation.

Lloyd Shapley in 2012

The Shapley value is one way to distribute the total gains to the players, assuming that they all collaborate. It is a "fair" distribution in the sense that it is the only distribution with certain desirable properties listed below. According to the Shapley value,[6] the amount that player *i* is given in a coalitional game (v, N) is

$$|S|!\,(n-|S|-1)!$$

Generic model: Shapley values



Shapley value

From Wikipedia, the free encyclopedia

In cooperative game theory, the **Shapley value** is a method (solution concept) for fairly distributing the total gains or costs among a group of players who have collaborated. For example, in a team project where each member contributed differently, the Shapley value provides a way to determine how much credit or blame each member deserves. It was named in honor of Lloyd Shapley, who introduced it in 1951 and won the Nobel Memorial Prize in Economic Sciences for it in 2012.[1][2]

The Shapley value determines each player's contribution by considering how much the overall outcome changes when they join each possible combination of other players, and then averaging those changes. In essence, it calculates each player's average marginal contribution across all possible coalitions.[3][4] It is the only solution that satisfies four fundamental properties: efficiency, symmetry, additivity, and the dummy player (or null player) property,[5] which are widely accepted as defining a fair distribution.

This method is used in many fields, from dividing profits in business partnerships to understanding feature importance in machine learning.

Formal definition [edit]

Formally, a **coalitional game** is defined as: There is a set *N* (of *n* players) and a function v that maps subsets of players to the real numbers: $v\colon 2^N \to \mathbb{R}$, with $v(\emptyset) = 0$, where $\emptyset$ denotes the empty set. The function v is called a characteristic function.

The function v has the following meaning: if *S* is a coalition of players, then $v(S)$, called the worth of coalition *S*, describes the total expected sum of payoffs the members of S can obtain by cooperation.

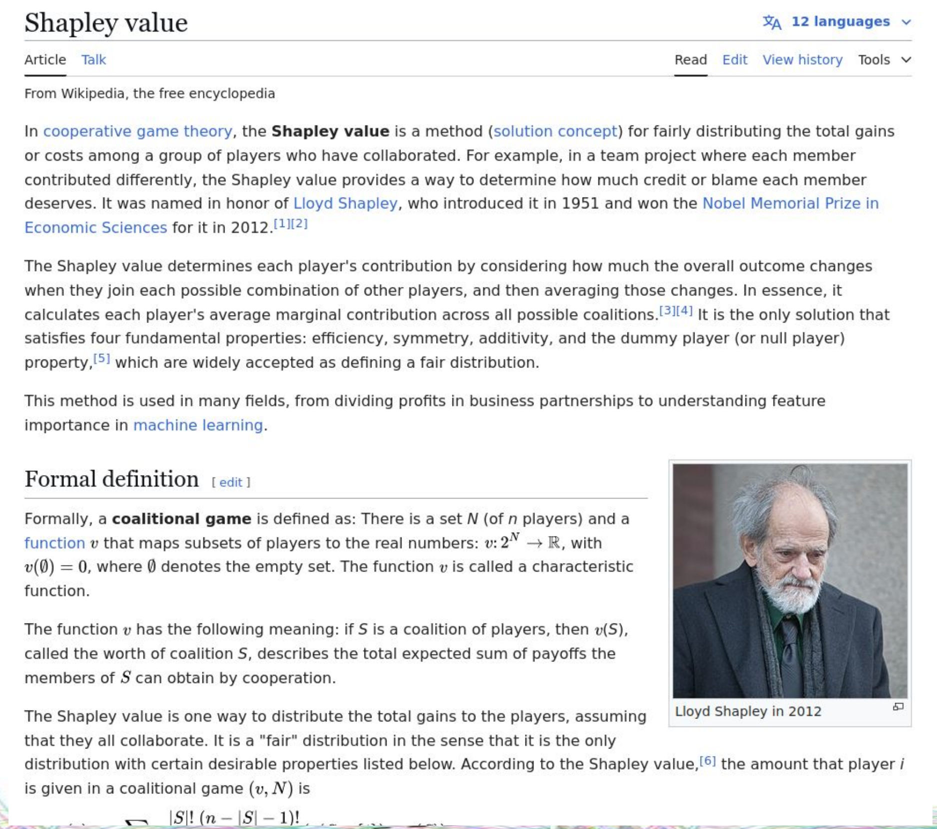
Lloyd Shapley in 2012

The Shapley value is one way to distribute the total gains to the players, assuming that they all collaborate. It is a "fair" distribution in the sense that it is the only distribution with certain desirable properties listed below. According to the Shapley value,[6] the amount that player *i* is given in a coalitional game (v, N) is

$|S|!\,(n-|S|-1)!$

- Game theory approach
- Player → Feature
- Distribute contribution
  - The glove game
- How much a feature contributes to a prediction, averaged over all possible combinations of features

Generic model: Shapley values



Shapley value

From Wikipedia, the free encyclopedia

In cooperative game theory, the **Shapley value** is a method (solution concept) for fairly distributing the total gains or costs among a group of players who have collaborated. For example, in a team project where each member contributed differently, the Shapley value provides a way to determine how much credit or blame each member deserves. It was named in honor of Lloyd Shapley, who introduced it in 1951 and won the Nobel Memorial Prize in Economic Sciences for it in 2012.[1][2]

The Shapley value determines each player's contribution by considering how much the overall outcome changes when they join each possible combination of other players, and then averaging those changes. In essence, it calculates each player's average marginal contribution across all possible coalitions.[3][4] It is the only solution that satisfies four fundamental properties: efficiency, symmetry, additivity, and the dummy player (or null player) property,[5] which are widely accepted as defining a fair distribution.

This method is used in many fields, from dividing profits in business partnerships to understanding feature importance in machine learning.

Formal definition [edit]

Formally, a **coalitional game** is defined as: There is a set *N* (of *n* players) and a function v that maps subsets of players to the real numbers: $v\colon 2^N \to \mathbb{R}$, with $v(\emptyset)=0$, where $\emptyset$ denotes the empty set. The function v is called a characteristic function.

The function v has the following meaning: if *S* is a coalition of players, then $v(S)$, called the worth of coalition *S*, describes the total expected sum of payoffs the members of S can obtain by cooperation.

Lloyd Shapley in 2012

The Shapley value is one way to distribute the total gains to the players, assuming that they all collaborate. It is a "fair" distribution in the sense that it is the only distribution with certain desirable properties listed below. According to the Shapley value,[6] the amount that player *i* is given in a coalitional game (v, N) is

$|S|!\,(n-|S|-1)!$

- Game theory approach
- Player → Feature
- Distribute contribution
  - The glove game
- How much a feature contributes to a prediction, averaged over all possible combinations of features

Fairness: fair distribution of credit among features
Model-agnostic: can be applied to any model (tree-based, neural networks, etc.)
Handles interaction effects: a feature's contribution might depend on the presence of others

Notebook on feature importance