Understanding Interactions in Statistical Models: Unravelling Complex Relationships with SHAP

Guillermo Dominguez, Joana Levtcheva, Mays Azeez, Minh Tuan Nguyen
Imperial College London

INTRODUCTION

From simple linear regression to large language models, Machine learning algorithms have been applied and contributed to the development of technology companies, medical trials, political research, agriculture research, etc. The rise of these methods creates ethical questions one of which is interpretability: Do we really understand what the model is saying about our data? One aspect that often gets overlooked is interaction. This phenomenon may occur when analyzing data with 2 or more predictors. It is when the effect of one of the predictors on the response variable depends on the value of other predictors. Interactions are crucial in understanding complex relationships that cannot be explained by the main effects from only single predictors. They often make it more difficult to assess the impact of changing the value of a single variable. Ignoring interactions may lead to devastating misinterpretation of data. For example, clinical trials may fail to notice the fact that a treatment only benefits younger patients or, even worse, the fact that it can have adverse effects on older people.

In this poster, we summarize the definition and classification of interactions, and give an overview of methods to detect and interpret them. We also dive into the concept of Shapley values and SHAP, and provide an example of how it can be applied to detect and analyze interactions using the SHAP Python package.

BACKGROUND

Definition:

A purely additive linear model with two predictors x_1, x_2 and one response variable y will take the form:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon.$$

This model works under the assumption that the effect of each predictor is independent of one another. However, in practice, an interaction may arise when a predictor's effect depends on the value of other predictors. In this case, an interaction between x_1 and x_2 may occur and an interaction term $\beta_3 x_1 x_2$, representing their joint effect on the response y should be added to our model:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + \epsilon\,,$$

Interaction can occur between any pair of variables or any groups of variables in multivariate datasets. It is important to identify interactions to avoid model misspecification.

Main classifications of interactions :

- Interactions between **numerical – categorical** variables: Interpreting interactions between only numerical variables is straightforward. From the fitted model's coefficients, we can deduce how the response is expected to change when one predictor varies, and the others are fixed. On the other hand, when dealing with interactions involving categorical variables, these variables can be either one-hot encoded and/or other methods such as analysis of variance (ANOVA) can be applied.

- Quantitative – qualitative (positive – negative) interactions:
  - **Quantitative interaction:** the magnitude of the effect of one predictor depends on the values of the others and its direction is constant. For example, a beneficial treatment can be increasingly effective towards younger patients.
  - **Qualitative interaction:** both magnitude and direction of the effect of one predictor depends on the value of the others. For example, two beneficial treatments can have adverse effects when applied together.

BACKGROUND

Shapley values and Shapley interaction index:

We are interested in how each feature affects the prediction of a data point. A linear model prediction for one data instance is

$$\hat{f}(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2,$$

where each $x_j, j = 1,2$ is a feature value, with contribution ϕ_j on the prediction $\hat{f}(x)$ defined as

$$\phi_j(\hat{f}) = \beta_j x_j - E(\beta_j x_j) = \beta_j x_j - \beta_j E(x_j),$$

where $E(\beta_j x_j)$ is the mean effect estimate for feature j. The contribution is the difference between the feature effect minus the average effect.

The **Shapley value** of a feature value is its contribution to the payout, weighted and summed over all possible feature value combinations:

$$\phi_j(val) = \sum_{S \subseteq \{1,\dots,p\} \setminus \{j\}} \frac{|S|!\,(p - |S| - 1)!}{p!} \left(val_x(S \cup \{j\}) - val_x(S)\right),$$

where S is a subset of the features used , x is the vector of feature values of the instance to be explained, p is the number of features, and $val_x(S)$ is the prediction for feature values in set S that are marginalized over features that are not included in set S:

$$val_x(S) = \int \hat{f}(x_1 x_2) dP_{x \notin S} - E_X(\hat{f}(X))$$

The interaction effect is the additional combined feature effect after accounting for the individual feature effects. The **Shapley interaction index** is defined as:

$$\phi_{i,j} = \sum_{s \subseteq \{i,j\}} \frac{|S|!\,(p - |S| - 2)!}{2(p-1)!} \delta_{ij}(S), \qquad i \neq j,$$

where

$$\delta_{ij}(S) = \hat{f}_x(S \cup \{i,j\}) - \hat{f}_x(S \cup \{i\}) - \hat{f}_x(S \cup \{j\}) + \hat{f}_x(S).$$

SHAP:

SHAP (SHapley Additive exPlanations) is a method to explain individual predictions and is based on the game theoretically optimal Shapely values.

- **SHAP Summary plot:**
  - Combines feature importance with feature effects and showcases the first indications of the relationship between the value of a feature and the impact on the prediction
  - Each point on the summary plot is a Shapley value for a feature and an instance
  - The color represents the value of the feature from low to high
  - The features are ordered according to their importance

- **SHAP Dependence plot:**
  - An alternative to partial dependence plots (PDP) and accumulated local effects (ALE) helping to see the exact form of relationships
  - While PDP and ALE plot show average effects, SHAP dependence also shows the variance on the y-axis
  - When there are interactions, the SHAP dependence plot will be much more dispersed in the y-axis
  - With the help of the Shapley interaction index the dependence plot can be improved by highlighting these feature interactions

CASE STUDY

A dataset related to a study into the effect of a drug methotrexate, as a treatment for rheumatoid arthritis is being explored. The available variables are:

- **dose** - total weight-standardized dose of methotrexate received (in mg/kg).
- **response** - change from baseline in blood concentration of an inflammation marker (in mmol/L).
- **variant** – binary, indicating whether the subject has the more common form of the genetic variant rs4673993

With the help of the SHAP Python package we aim to detect whether there is interaction between the two features dose and variant.

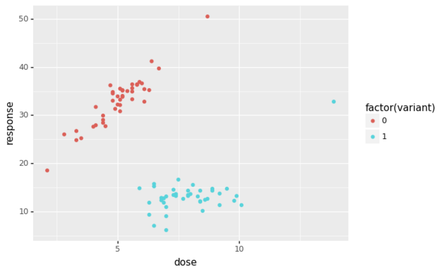

Fig 1. Scatter plot of the data

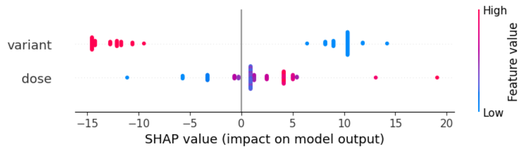

Fig 2. SHAP Summary plot: The variant is more important than dose, and forms two distinct groups having either higher or lower feature values. Higher doses increase the change from baseline in blood concentration of an inflammation marker.

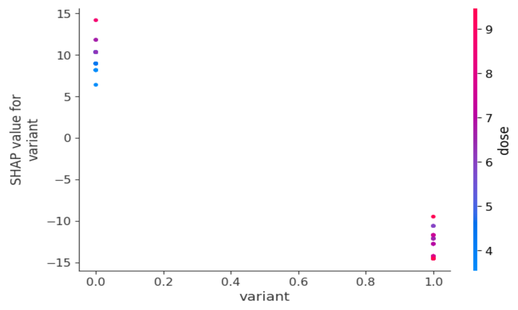

Fig 3. SHAP Dependence plot for the feature variant: There is a clear dispersion in the y-axis for both variants, indicating an interaction between the two features.

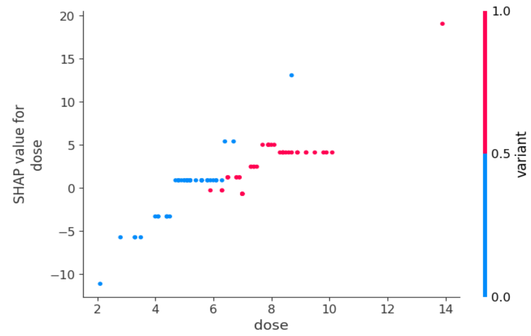

Fig 4. SHAP Dependence plot for the feature dose: the trends defined from both variants hint that they are crossing, considered to be an indication, although in this case it is not very clear.

CONCLUSION

The poster shows the importance of interactions in statistical models using the Shapley values and the SHAP package. The Shapley values provide a fair way of measuring the contribution of each feature in a model, while the SHAP package allows us to calculate them efficiently and provides a way to do meaningful and easy to understand visualizations. These visualizations can help with the detection of interactions between different features, as well as to help ease and improve the interpretation of their relationships and impact on the model outcome. This method is especially useful in complex interactions, where it can be difficult to determine the importance of each feature on its own. This can help make more informed decisions when building and refining models, leading to better results and more accurate predictions.

REFERENCES

- Cox, D. R. (1984). Interaction. International Statistical Review / Revue Internationale de Statistique, 52(1), 1–24. https://doi.org/10.2307/1403235
- Fisher, R.A. (1992). Statistical Methods for Research Workers. In: Kotz, S., Johnson, N.L. (eds) Breakthroughs in Statistics. Springer Series in Statistics. Springer, New York, NY. https://doi.org/10.1007/978-1-4612-4380-9_6
- Peto, D. P. (1982). "Statistical aspects of cancer trials". Treatment of Cancer (First ed.). London: Chapman and Hall. ISBN 0-412-21850-X.
- Christoph Molnar. (2019). Interpretable machine learning : a guide for making Black Box Models interpretable. Lulu.
- Basic SHAP Interaction Value Example in XGBoost — SHAP latest documentation. (2018). Readthedocs.io. https://shap.readthedocs.io/en/latest/example_notebooks/tabular_examples/tree_based_models/Basic%20SHAP%20Interaction%20Value%20Example%20in%20XGBoost.html