DIGITAL FINANCE

This project has received funding from the Horizon Europe research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 101119635

Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

State Secretariat for Education, Research and Innovation SERI

Funded by the European Union

MARIE CURIE ACTIONS

DIGITAL

Basic Explainability - Feature Importance, PDP, ICE

Faizan Ahmed

DIGITAL

Funded by
the European Union

Reading

- **Mandatory Reading Material**
  - Molnar, Christoph. *Interpretable machine learning*.2020.[Section 23,2419,20,13] https://christophm.github.io/interpretable-ml-book/
- **Recommended Reading Material**
  - Lipton, Zachary C. "The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery." *Queue* 16.3 (2018): 31-57. https://arxiv.org/abs/1606.03490
  - **If you wanted to know a lot more:**
  - Fisher, Aaron, Cynthia Rudin, and Francesca Dominici. "All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously." http://arxiv.org/abs/1801.01489 (2018).
  - Wei, Pengfei, Zhenzhou Lu, and Jingwen Song. "Variable importance analysis: a comprehensive review." Reliability Engineering & System Safety 142 (2015): 399-432
  - Kim, Been, Rajiv Khanna, and Oluwasanmi O. Koyejo. "Examples are not enough, learn to criticize! Criticism for interpretability." Advances in Neural Information Processing Systems (2016). [Very mathematical]
    - The talk is interesting: https://www.youtube.com/watch?v=bQfYRcXc9F0&ab_channel=MicrosoftResearch
- **Libraries**
  - MMD-critic https://github.com/BeenKim/MMD-critic
  - ALE Plots: https://github.com/blent-ai/ALEPython

DIGITAL

Permutation Importance

- Measures the increase in the prediction error of the model after the feature values are permuted
- How: **only a column (feature) of the training data is shuffled and make the prediction again but with the shuffled values.**
- Note: we are creating a mismatch from the true data by shuffling only one column, i.e. the whole row is not shuffled.
- By shuffling a particular column only, if the output predictions falls significantly, then we know the feature was very important and vice versa, if the feature wasn't important then the performance does not fall.

f1	f2	f3	…	fn	y
2.29	3.47	2.55		3.17	0
2.86	2.38	0.72		3.37	0
0.95	0.44	0.08		1.61	0
1.28	0.48	0.10		3.12	1
0.74	1.32	1.41		3.42	1

DIGITAL

Permutation Feature Importance

(Fisher, Rudin, and Dominici)

- **Input:** Trained model $\hat{f}$, feature matrix X, target vector y, error measure $L(y, \hat{f})$

$$e_{orig} = L\left(y, \hat{f}\right)$$

- For each feature $j \in \{1, \cdots, p\}$ do
  - Generate X_{perm} by permuting feature j
  - Estimate error $e_{perm} = L(Y, \hat{f}(X_{perm}))$
  - Computer feature importance $FI_j = \frac{e_{perm}}{e_{orig}}$ or $FI_j = e_{perm} - e_{orig}$

- Sort feature by descending FI_j

f1	f2	f3	…	fn	y
2.29	3.47	2.55		3.17	0
2.86	2.38	0.72		3.37	0
0.95	0.44	0.08		1.61	0
1.28	0.48	0.10		3.12	1
0.74	1.32	1.41		3.42	1

Fisher, Aaron, Cynthia Rudin, and Francesca Dominici. "All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously." http://arxiv.org/abs/1801.01489 (2018).

DIGITAL

Permutation Feature Importance

Penguin Sex Classification: Logistic Regression Models

- Trained 3 logistic regression models to predict penguin sex
- Used 2/3 of the data for training, 1/3 fo feature importance evaluation
- Measured error using **log loss**

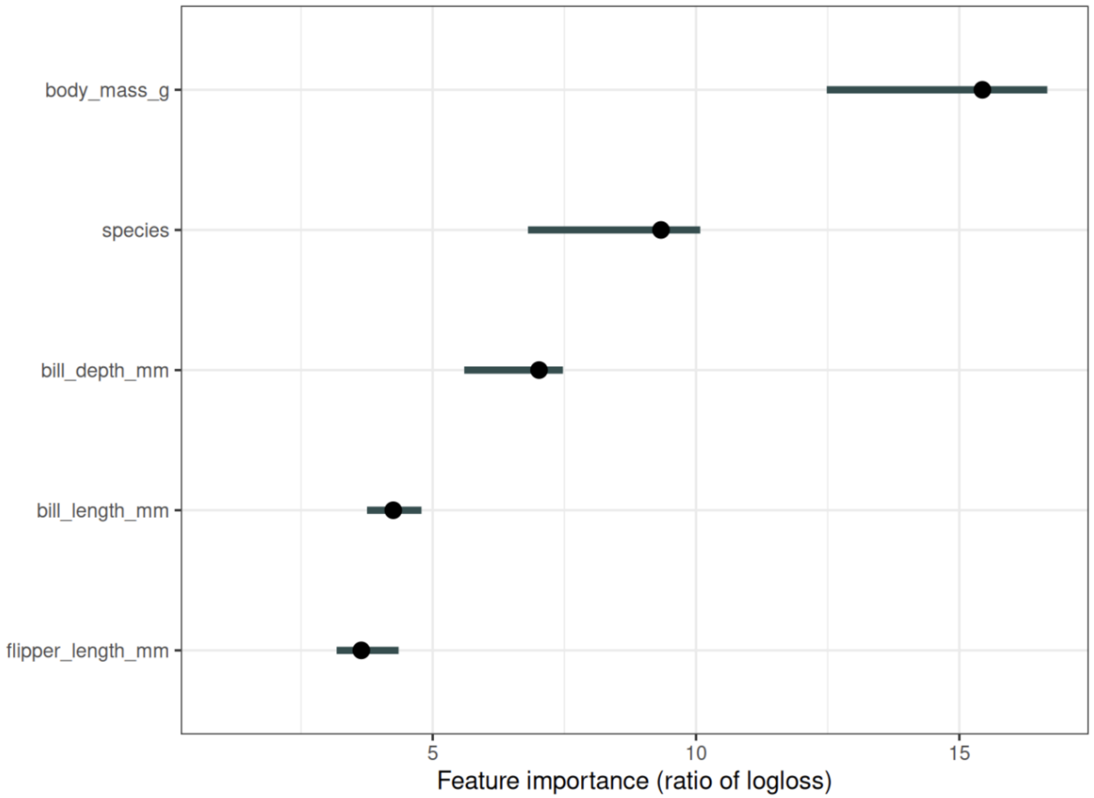


Figure: Permutation feature importance values for the penguin classification task. [Source]

DIGITAL

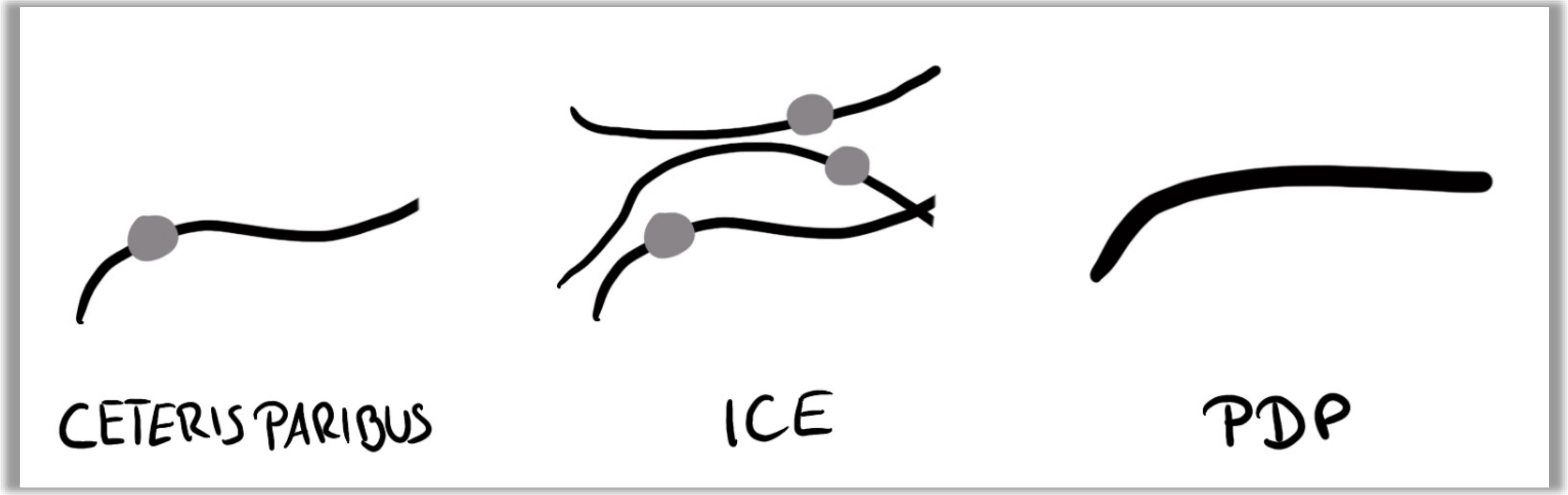
Permutation Feature Importance

- **Nice Interpretation**
- **Comparable across different problems.**
- **Need access to the true outcome**
- **Can be biased by unrealistic data instances**

Further reading: https://christophm.github.io/interpretable-ml-book/feature-importance.html

If you really want to know all about it: Wei, Pengfei, Zhenzhou Lu, and Jingwen Song. "Variable importance analysis: a comprehensive review." Reliability Engineering & System Safety 142 (2015): 399-432

Ceteris Paribus Plots



Partial Dependence Plot

- **Partial Dependence Plot (PDP)**, sketches the functional form of the relationship between an input feature and the target.
  - Show the average effect on predictions as the value of feature changes.

- **Assumption**: the feature of interest are independent from the complement features
  - this method is applied to a model which is already trained (can be used in conjunction with permutation importance)
  - use it to see "how" the predictions are changed by changes in a feature.

DIGITAL

Partial Dependence Plot

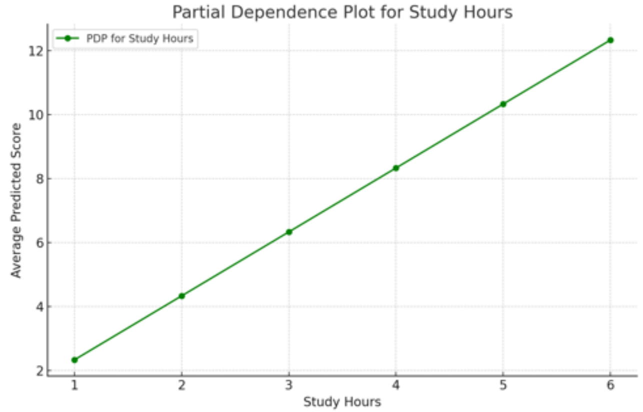
- **Step 0:** Select feature.
- **Step 1:** Define grid.
- **Step 2:** Per grid value:
  - Replace feature with grid value and
  - Average predictions.
- **Step 3:** Draw curve.

Study hours (x1)	Breaks (x2)	Sleep(x3)	grade
1	2	7	5
2	2	6	6
3	1	7	7
4	1	6	8
5	0	7	9
6	0	5	9

X1	X2	X3	Y_pred
1	2	7	5
1	2	6	4
1	1	7	3
1	1	6	2
1	0	7	1
1	0	5	-1
Average			14/6

X1	X2	X3	Y_pred
2	2	7	7
2	2	6	6
2	1	7	5
2	1	6	4
2	0	7	3
2	0	5	1
Average			26/6

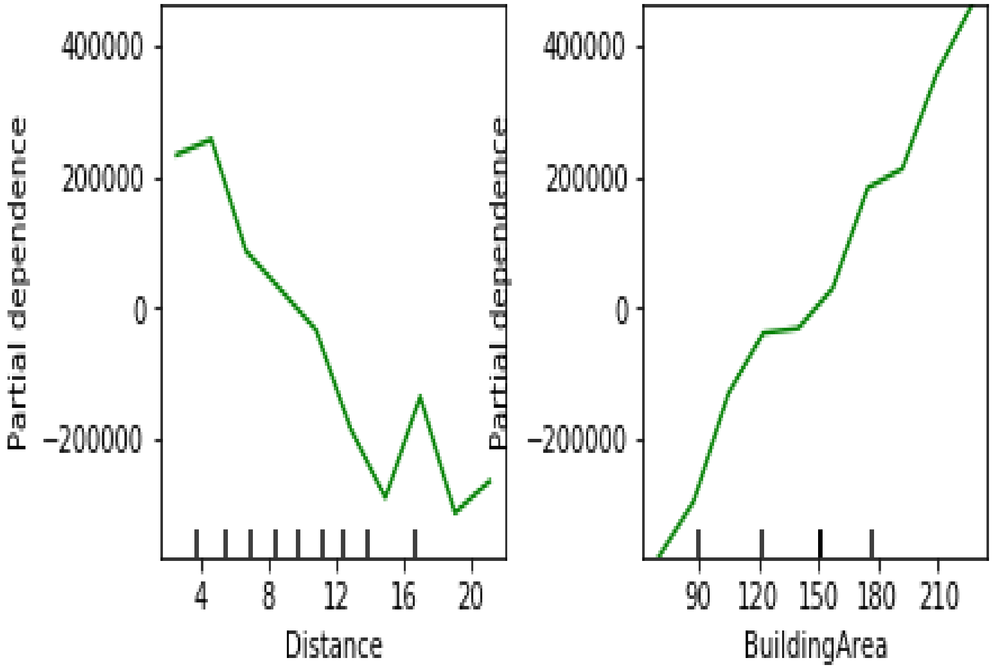
X1	Y(x1)
1	2.33
2	4.33
3	6.33
4	8.33
5	10.33
6	12.33



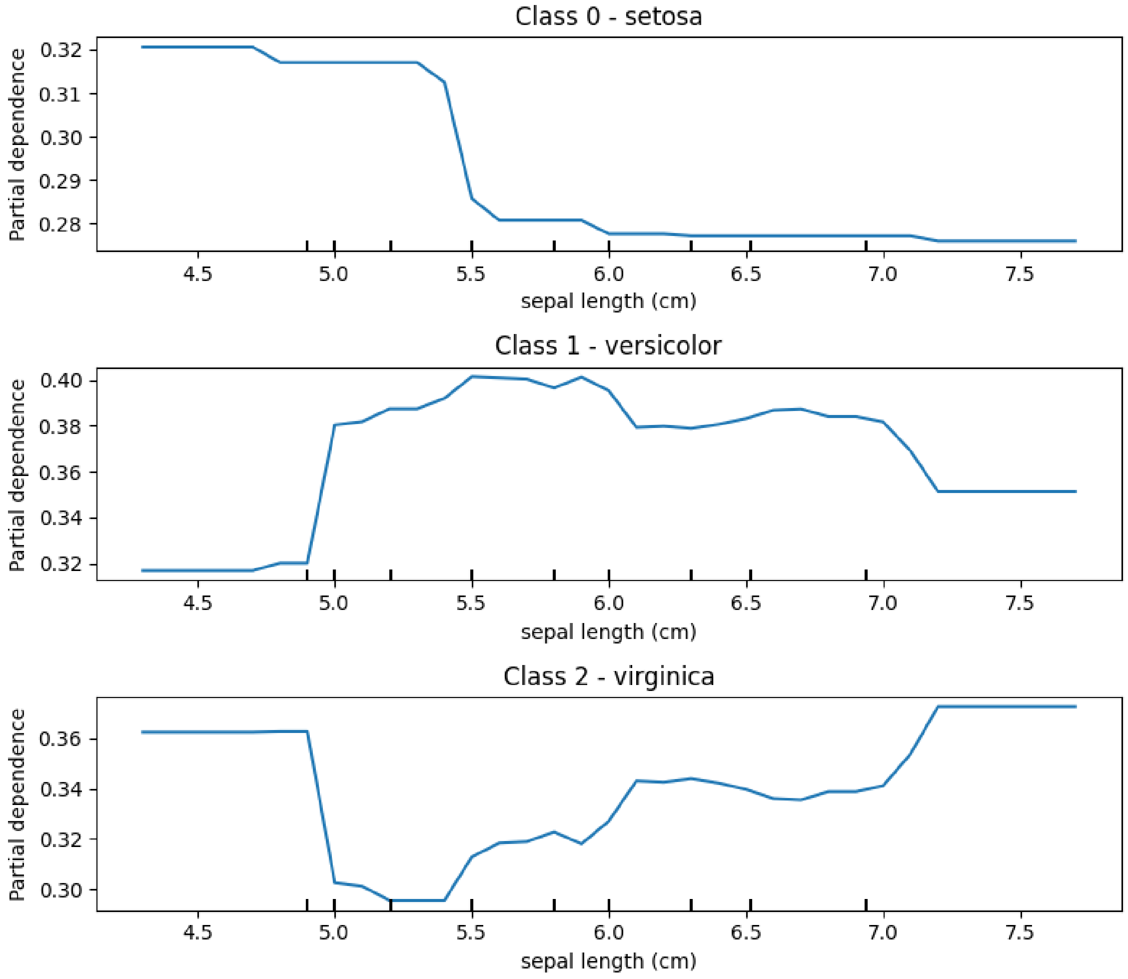
Source: https://christophm.github.io/interpretable-ml-book/pdp.html

DIGITAL

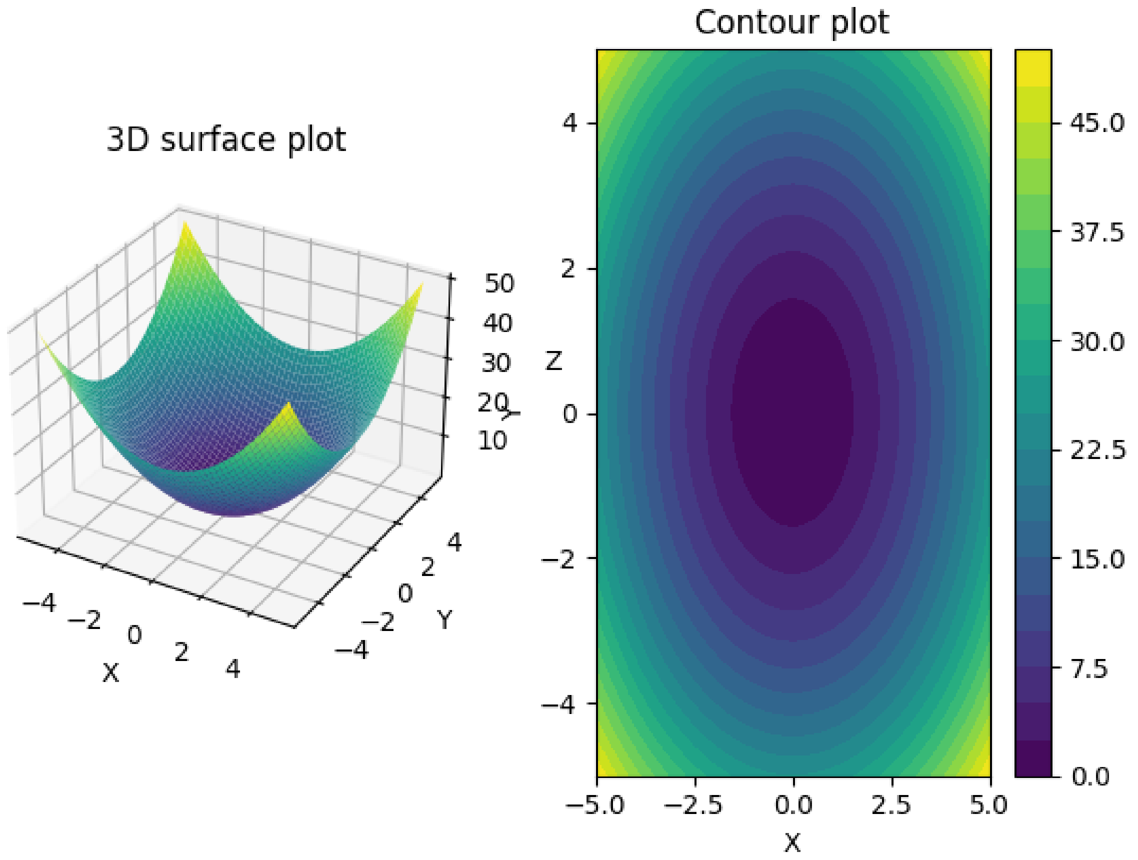
Partial Dependence Plot



The relationship (according to our model) between Price and a couple variables from the Melbourne Housing dataset. [Source]



DIGITAL

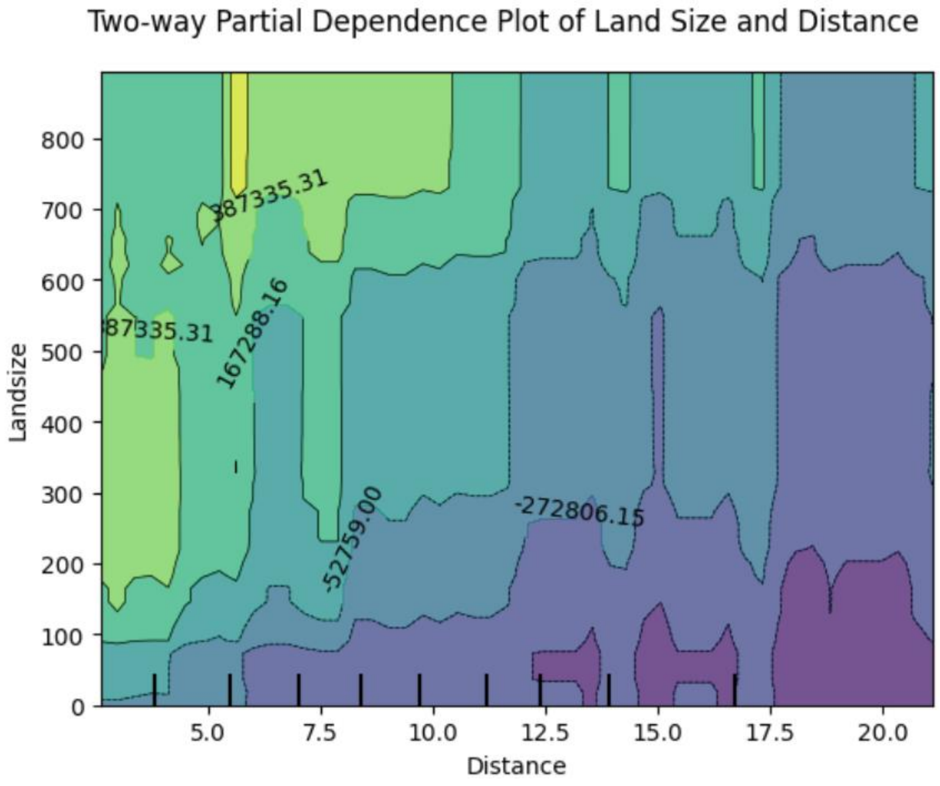


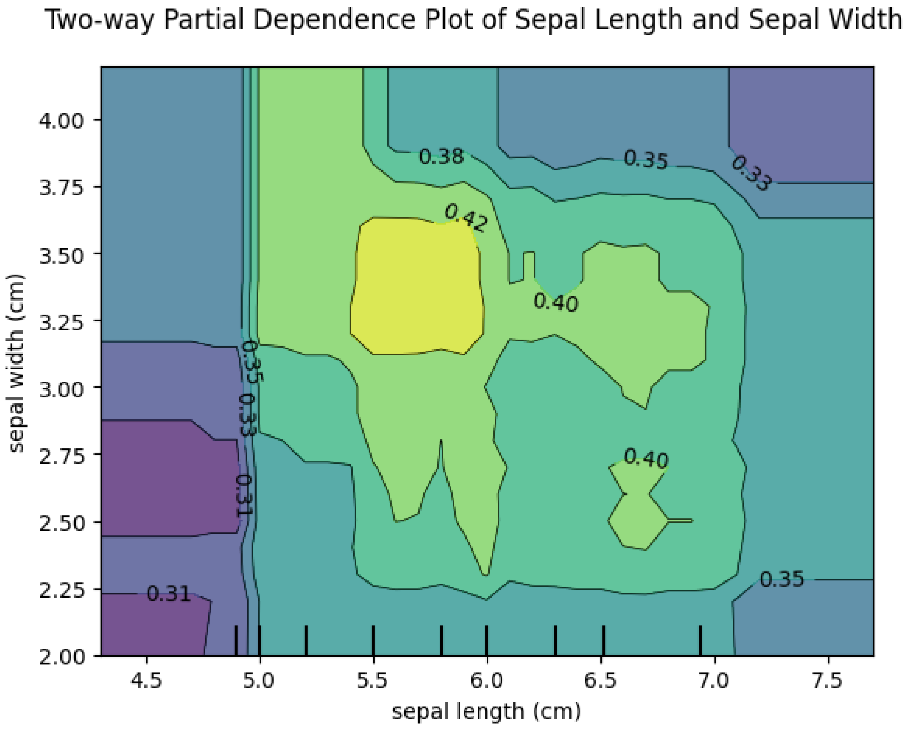
Contour Map

DIGITAL

Partial Dependence Plot

- One-way PDPs tell us about the interaction between the target response and an input feature of interest
- Two-way PDPs show the interactions among the two features.





See also

DIGITAL

Property	Assessment
Completeness	Interpretability achieved with agnostic method, completeness is low, limited possibility of anticipating model predictions (we can just look at goal scored as rough indicator)
Expressive power	Good in terms of getting evidence of the most important feature but on average and without details of feature interactions (or limited)
Translucency	Low, we don’t have insight into model internals
Portability	High, the method doesn’t rely on the ML model specs
Algorithmic complexity	Low, no need of complex methods to generate explanations
Comprehensibility	Good level of human understandable explanations

Partial Dependence Plot

DIGITAL

Partial Dependence Plot

+ Computation is intuitive
+ Interpretation is clear (Caution: Uncorrelated)
+ Causal interpretation
- Maximum number of features
- Omitting the feature distribution can be misleading
- Assumption of independence
- Heterogeneous effects might be hidden

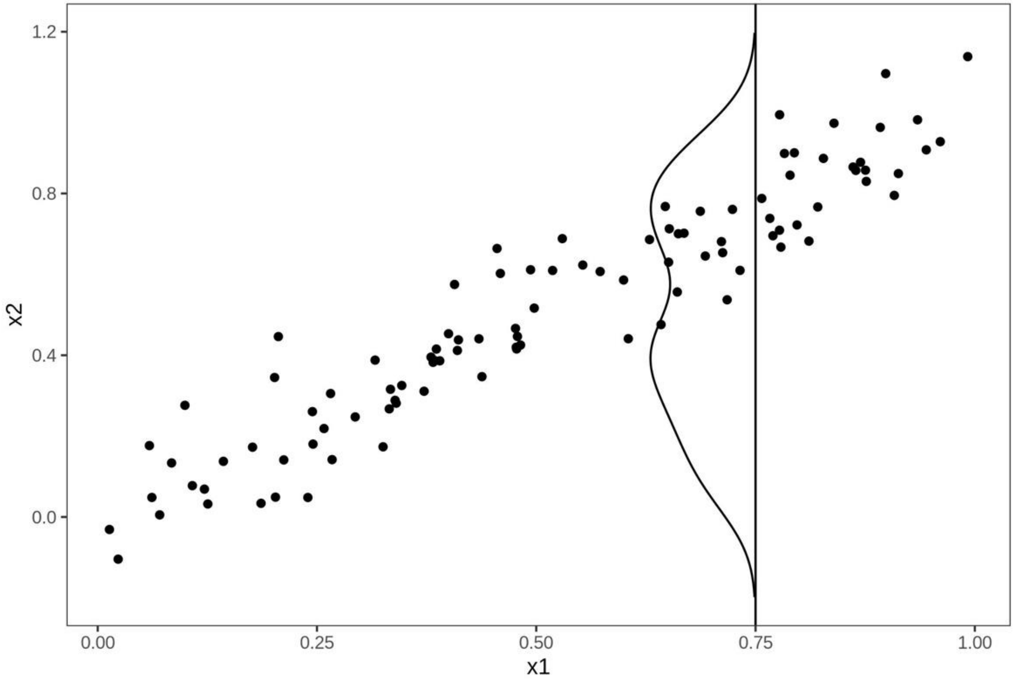
DIGITAL

From PDP to Accumulated Local Effects

What will happen if we have 24 (an unrealistic value) here instead of 7?

PDP

Marginal distribution P(x2)

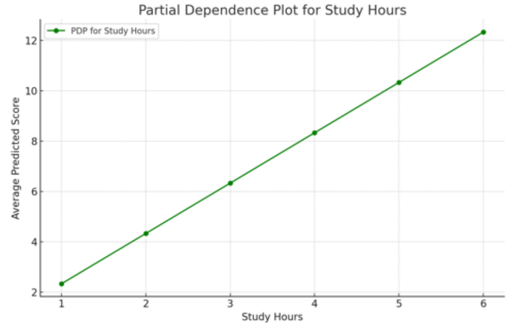


Study hours (x1)	Breaks (x2)	Sleep(x3)	grade
1	2	7	5
2	2	6	6
3	1	7	7
4	1	6	8
5	0	7	9
6	0	5	9

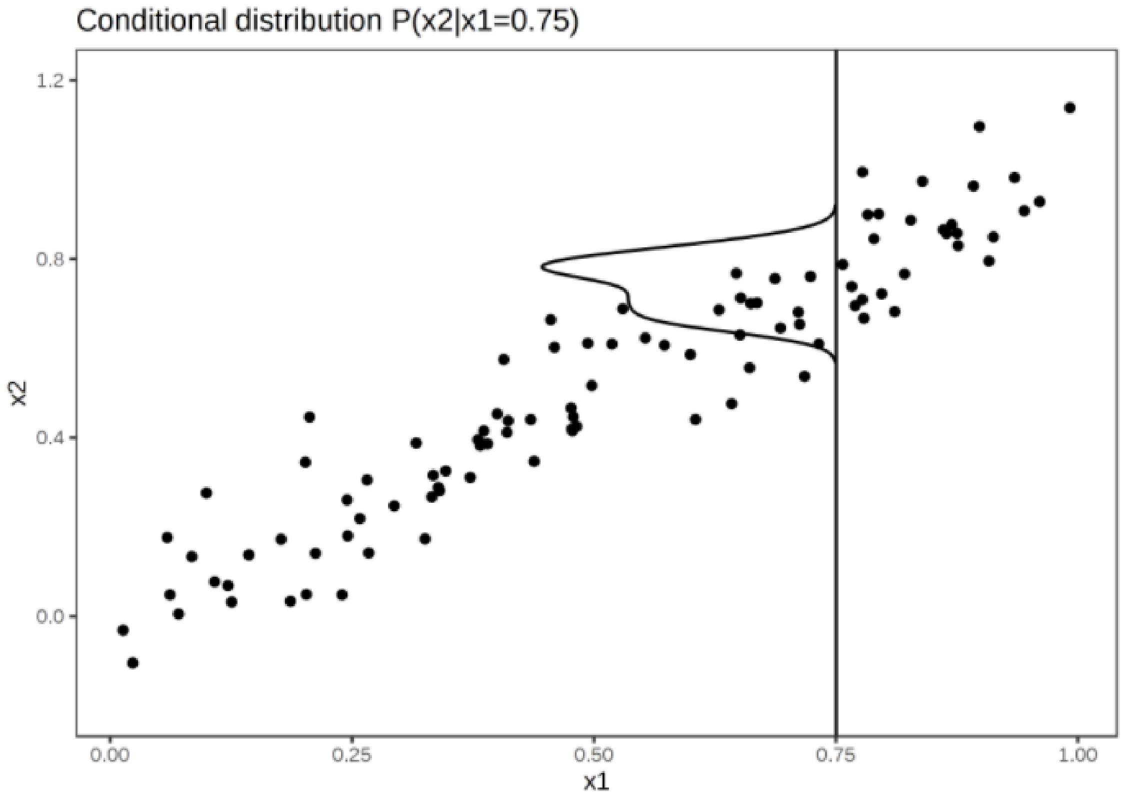
X1	X2	X3	Y_pred
1	2	7	5
1	2	6	4
1	1	7	3
1	1	6	2
1	0	7	1
1	0	5	-1
Average			14/6

X1	X2	X3	Y_pred
2	2	7	7
2	2	6	6
2	1	7	5
2	1	6	4
2	0	7	3
2	0	5	1
Average			26/6

X1	Y(x1)
1	2.33
2	4.33
3	6.33
4	8.33
5	10.33
6	12.33



From PDP to Accumulated Local Effects



- **Solution:** To find the feature effects of correlated features, we can average over the conditional distribution of the feature, meaning at a grid value of x_1, we average the predictions of instances with a similar x_1 value.
- The solution for calculating feature effects using the conditional distribution is called Marginal Plots or M-Plots.
- Issue: We are combining the effects. Which means more sleep (or less sleep) lead to worst grades.
- ***M-Plots*** avoid averaging predictions of unlikely data instances, but they mix the effect of a feature with the effects of all correlated features.
- ***ALE plots*** show how features impact predictions by accumulating the local effects of features across the data distribution.
- Focus on local effects, reducing the smearing effect seen in PDPs due to averaging over the data distribution.

DIGITAL

Algorithm 1 Accumulated Local Effects (ALE) Plots

```
Require: Trained prediction model, model
Require: Feature index for ALE plot, feature_index
Require: Dataset containing features and outputs, data
Require: Number of intervals, num_intervals
Ensure: ALE plot of feature x_j
Calculate quantile bounds for the feature x_j over the specified number of
    intervals, num_intervals
Initialize arrays local_effects and all_effects to zeros with length equal to the
    number of data instances
for k = 1 to num_intervals do
    Determine bounds z_{k-1,j} and z_{k,j} for the current interval
    Create modified datasets data_lower and data_upper by replacing x_j in
    all instances with z_{k-1,j} and z_{k,j}, respectively
    Compute model predictions for both modified datasets: predictions_lower
    and predictions_upper
    Calculate differences Δf̂_{i,k} = f̂(z_{k,j}, x_{-j}) − f̂(z_{k-1,j}, x_{-j})
    for each data instance i do
        if data[i, feature_index] ≥ z_{k-1,j} and data[i, feature_index] < z_{k,j}
        then
            Accumulate effects: local_effects[i] += Δf̂_{i,k}
        end if
    end for
end for
Calculate the mean of local_effects:
```

$$mean\_effect = \frac{1}{N}\sum_{i=1}^{N} local\_effects[i]$$

```
Adjust each element in local_effects by subtracting the mean effect:
    all_effects[i] = local_effects[i] − mean_effect
Plot all_effects against feature x_j values to visualize the ALE plot
```

Accumulated Local Effects

Number of Intervals: More intervals can provide finer resolution but might introduce noise. Experiment with the number of intervals for the best clarity.

Quantiles: Using quantiles to define intervals ensures even distribution of data points across intervals, which is beneficial when the feature distribution is skewed.

Two steps:

- Accumulate difference for each data point where x_{ij} falls within interval K:

$$\tilde{f}_{j,ALE}(x_j) = \sum_{k: z_{k-1,j} \le x_{ij} \le z_{k,j}} \Delta \tilde{f}_{i,k}$$

- Adjust ALE to have zero mean across the dataset

$${}_{,ALE}(x_i)$$

$$\tilde{f}_{j,ALE}(x_i) - \frac{1}{N}\sum_{i=1}^{N} \tilde{f}_{j,ALE}(x_i) \quad N \text{ is the number of instances}$$

Accumulated Local Effects

Sample Data			
age	bmi	heart_disease	P of stroke
2	12	0	**20**
3	15	0	**21**
6	11	0	**20**
22	24	0	**30**
24	21	0	**31**
27	24	0	**29**
45	23	0	**40**
43	25	0	**41**
47	25	0	**45**
66	30	1	**93**
68	28	1	**88**
63	29	1	**95**

- **Data Type:**
  - **Numerical Features**: ALE is calculated by dividing the feature into intervals, computing prediction differences for small changes within these intervals, and accumulating these to get the ALE curve.
  - **Categorical Features**: Special methods like ordering categories based on similarity (using metrics like the Kolmogorov-Smirnov distance) are required since categorical data doesn't naturally fit into intervals

DIGITAL

Age 3			
Age interval 2-6 (Lower)			
age	bmi	heart_disease	P of stroke
2	12	0	20
2	15	0	22
2	11	0	21
		Average P	21

Age 3			
Age interval 2-6 (Upper)			
age	bmi	heart_disease	P of stroke
6	12	0	22
6	15	0	23
6	11	0	20
		Average P	22

Difference	
2	
1	
-1	
0.67	Average Diff.

Age 24			
Age interval 22-27 (Lower)			
age	bmi	heart_disease	P of stroke
22	24	0	30
22	21	0	29
22	22	0	27
		Average P	29

Age 24			
Age interval 22-27 (Upper)			
age	bmi	heart_disease	P of stroke
27	24	0	31
27	21	0	29
27	22	0	29
		Average P	30

Difference	
1	
0	
2	
1	Average Diff.

Age 45			
Age interval 43-47 (Lower)			
age	bmi	heart_disease	P of stroke
43	23	0	40
43	25	0	42
43	25	0	44
		Average P	42

Age 45			
Age interval 43-47 (Upper)			
age	bmi	heart_disease	P of stroke
47	23	0	42
47	25	0	44
47	25	0	45
		Average P	44

Difference	
2	
2	
1	
1.67	Average Diff.

Age 66			
Age interval 63-68 (Lower)			
age	bmi	heart_disease	P of stroke
63	30	1	93
63	28	1	87
63	29	1	94
		Average P	91

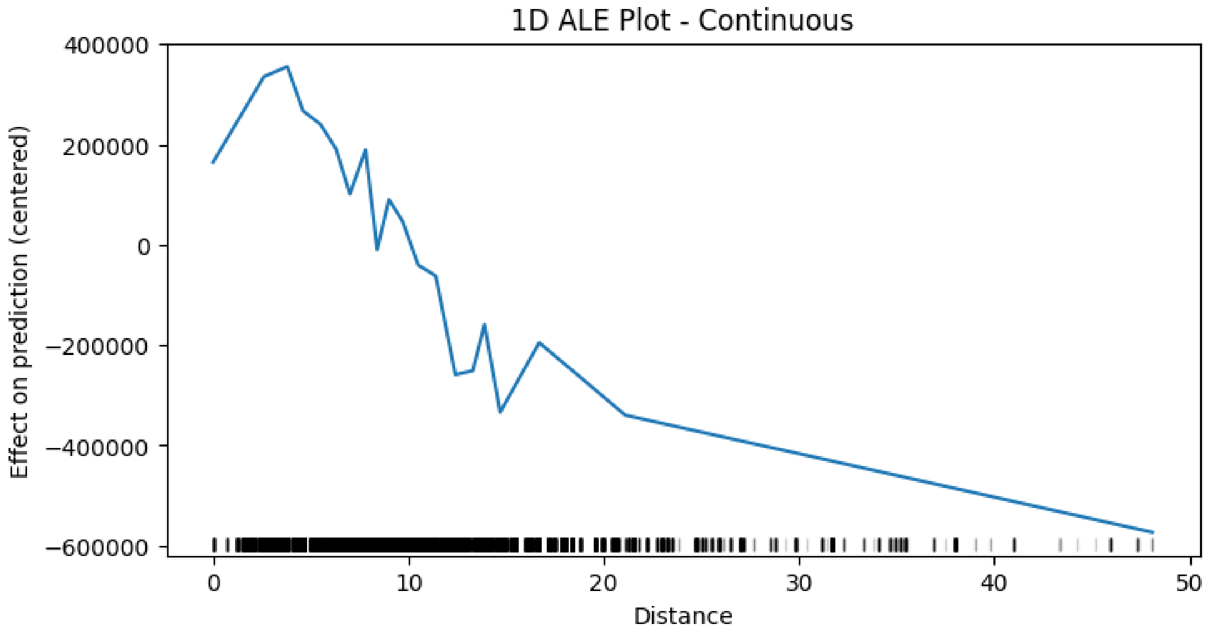
Age 66			
Age interval 63-68 (Upper)			
age	bmi	heart_disease	P of stroke
68	30	1	96
68	28	1	90
68	29	1	95
		Average P	94

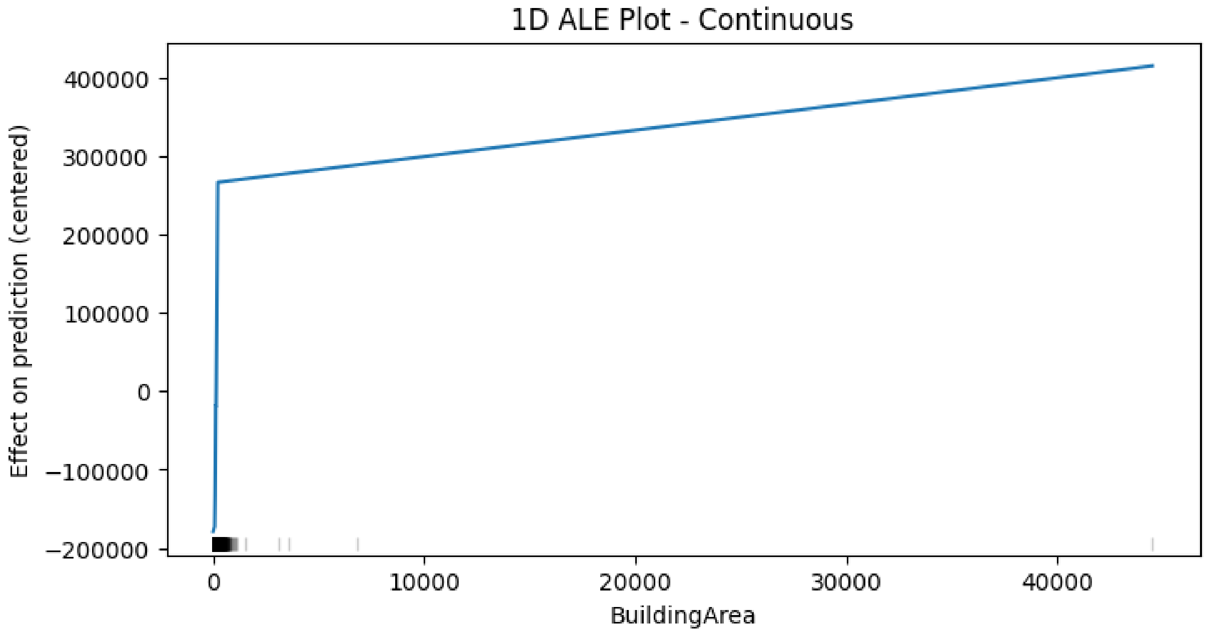
Difference	
3	
3	
1	
2.33	Average Diff.

5.67	Accum Diff.
0.5	Accum Diff / N

*Average Prediction is rounded to the nearest integer value

DIGITAL





ALE Example

Limitations

- **Computational Complexity**: ALE plots require significant computational resources, particularly with large datasets or many feature intervals.
- **Interpretation Challenges**: Interpreting the results of ALE plots can be difficult, especially in complex, high-dimensional models.

DIGITAL

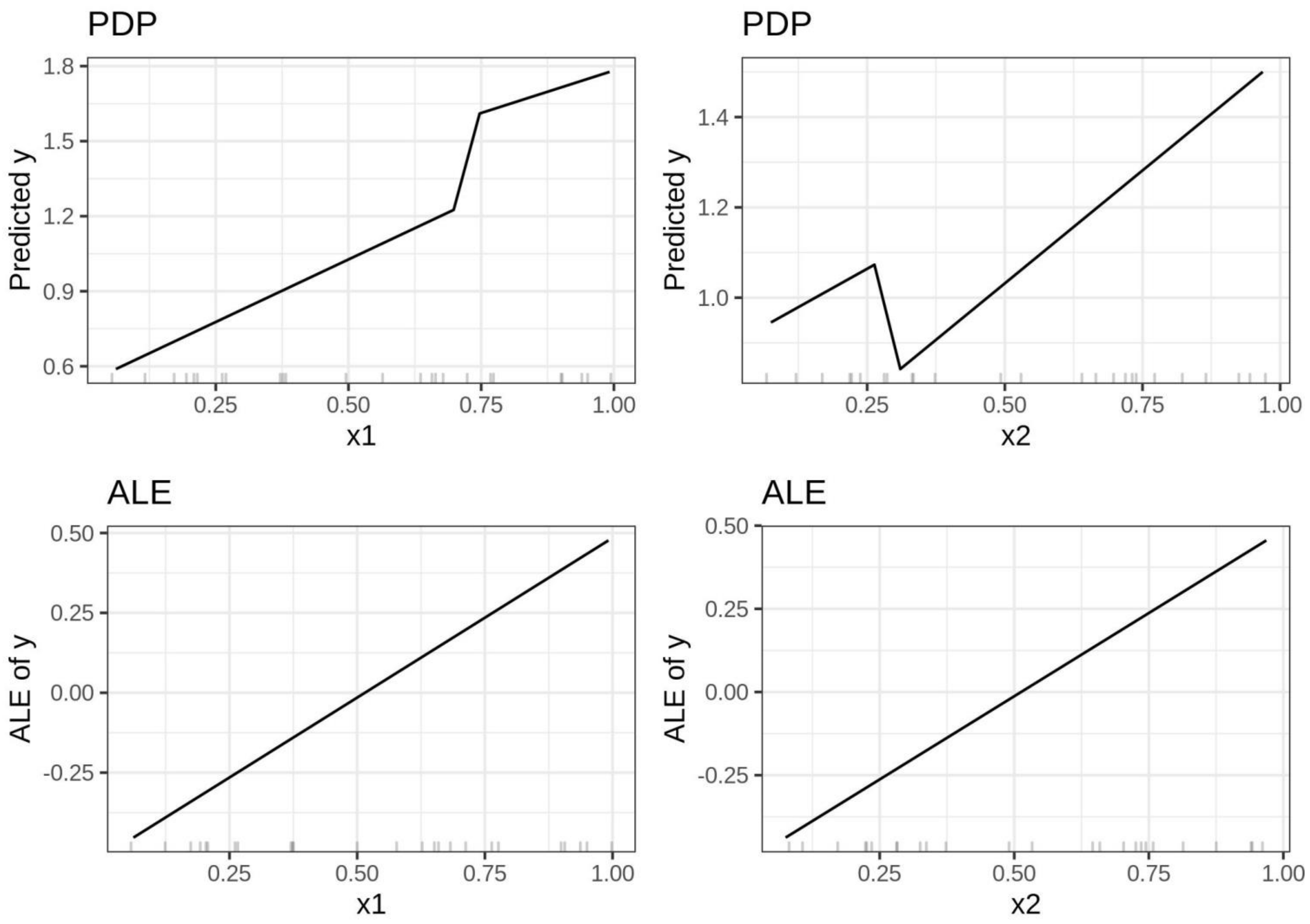
Accumulated Local Effects

- **Partial Dependence Plots**: “Let me show you what the model predicts on average when each data instance has the value v for that feature. I ignore whether the value v makes sense for all data instances.”
- **M-Plots**: “Let me show you what the model predicts on average for data instances that have values close to v for that feature. The effect could be due to that feature, but also due to correlated features.”
- **ALE plots**: “Let me show you how the model predictions change in a small ”window” of the feature around v for data instances in that window.”

Source: https://christophm.github.io/interpretable-ml-book/ale.html

Python: https://github.com/blent-ai/ALEPython

DIGITAL



PDP vs ALE

SOURCE:
HTTPS://CHRISTOPHM.GITHUB.IO/INTERPRETABLE-ML-BOOK/ALE.HTML

DIGITAL

DIGITAL

Funded by
the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or Horizon Europe: Marie Skłodowska-Curie Actions. Neither the European Union nor the granting authority can be held responsible for them.

DIGITAL

This project has received funding from the European Union's Horizon Europe research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 101119635