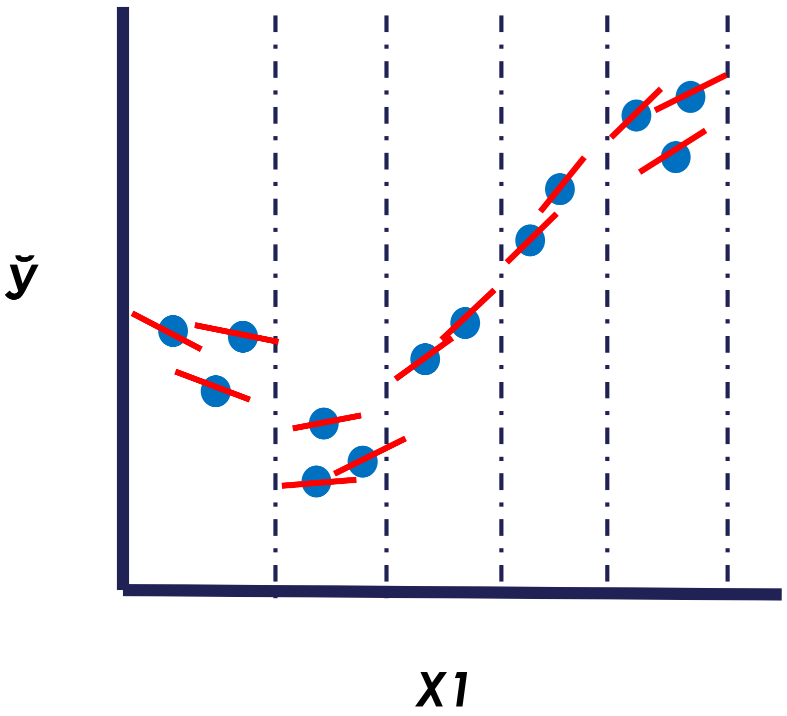
Accumulated local effects

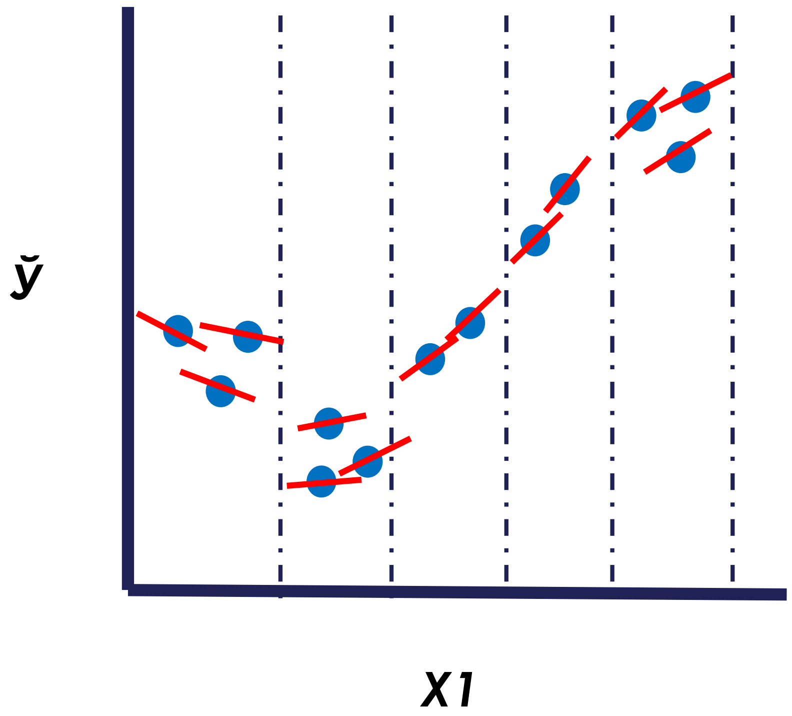
Discrete and categorical variables

ALE – Estimation

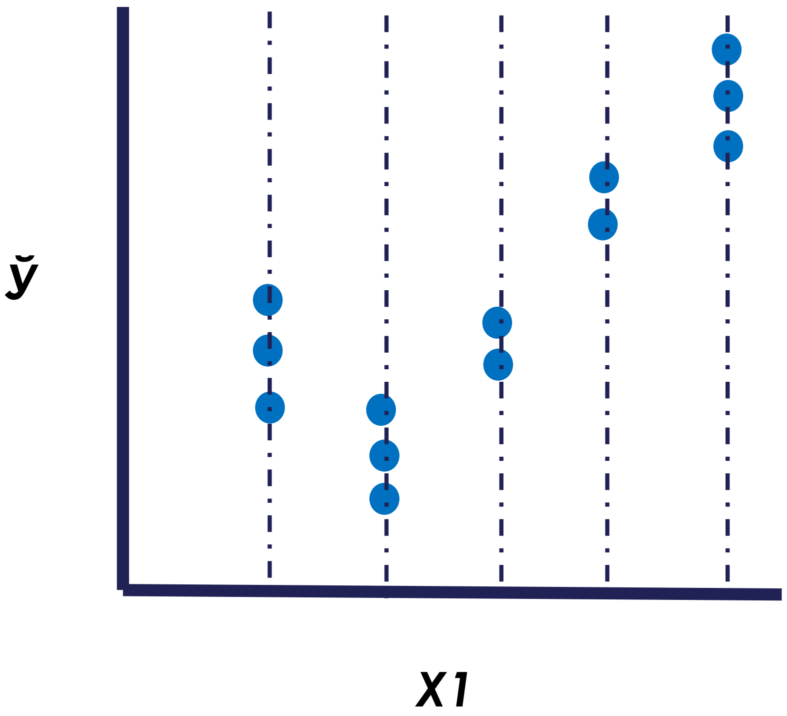


1. Sort variable into intervals
2. Set observations to the interval's maximum an obtain a prediction.
3. Set observations to the interval's minimum an obtain a prediction.
4. Obtain the difference between predictions.
5. Average differences.
6. Determine the cumulative distribution of the differences.

ALE: designed for continuous features!

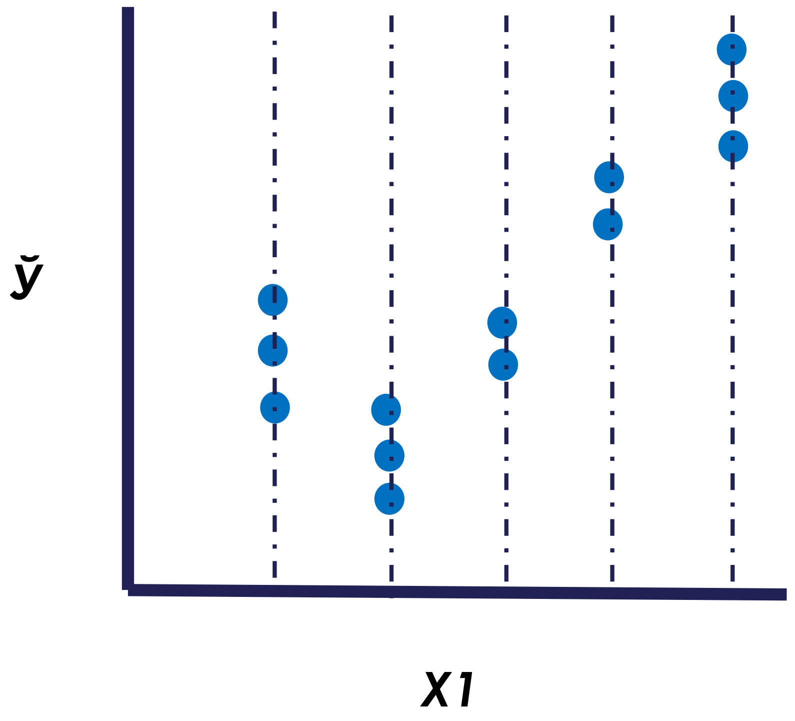


ALE – discrete features



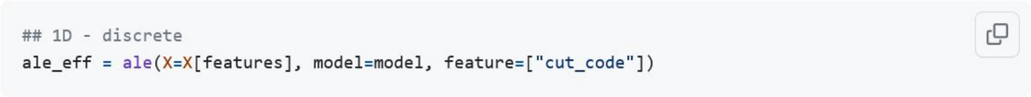
What happens when features are discrete?

ALE – discrete features

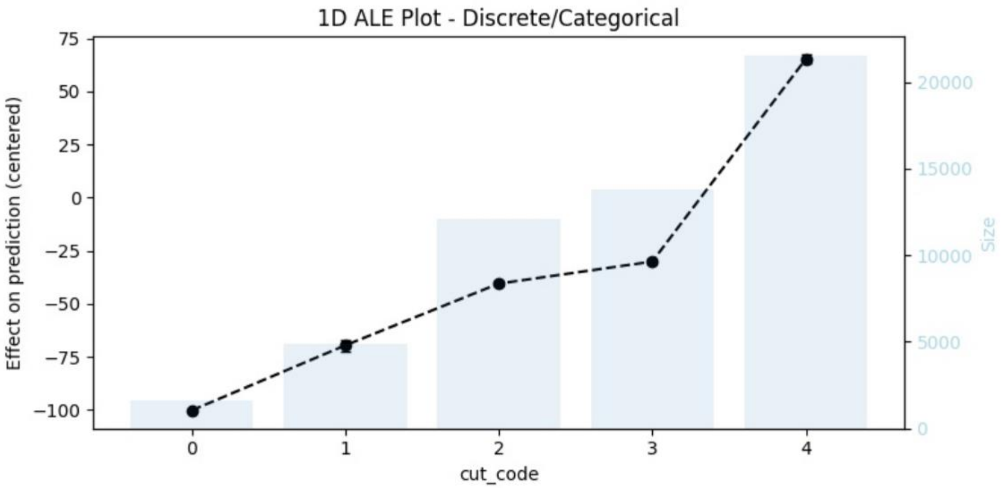


1. Set the observations to its nearest point to the right, and obtain prediction.
2. Set observation to its nearest point to the left and obtain prediction.
3. Determine the difference in prediction.
4. Calculate the cumulative distribution.

Discrete features with PyALE



```
## 1D - discrete
ale_eff = ale(X=X[features], model=model, feature=["cut_code"])
```



```
ale(
    X,
    model,
    feature,
    feature_type="auto",
    grid_size=20,
)
```

Features with 10 unique values or less will be treated as discrete.

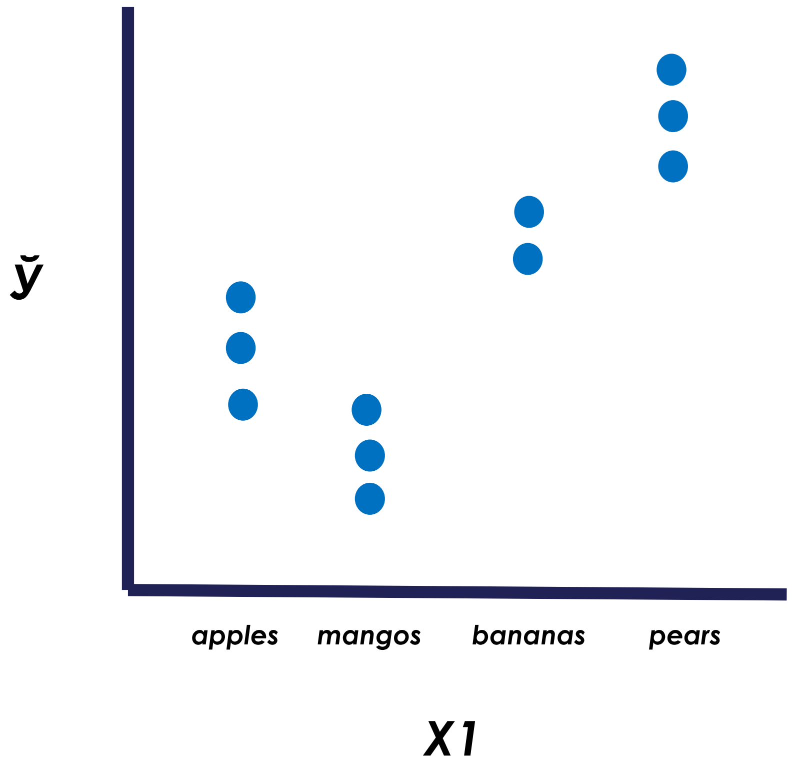
Alternatively, set `feature_type` to “continuous” or “discrete”.

Discrete features with Alibi

```
ALE(
    check_feature_resolution=True,
    low_resolution_threshold=10,
)
```

Features with less unique values than `low_resolution_threshold` will be treated as discrete. To override, set `check_feature_resolution` to False.

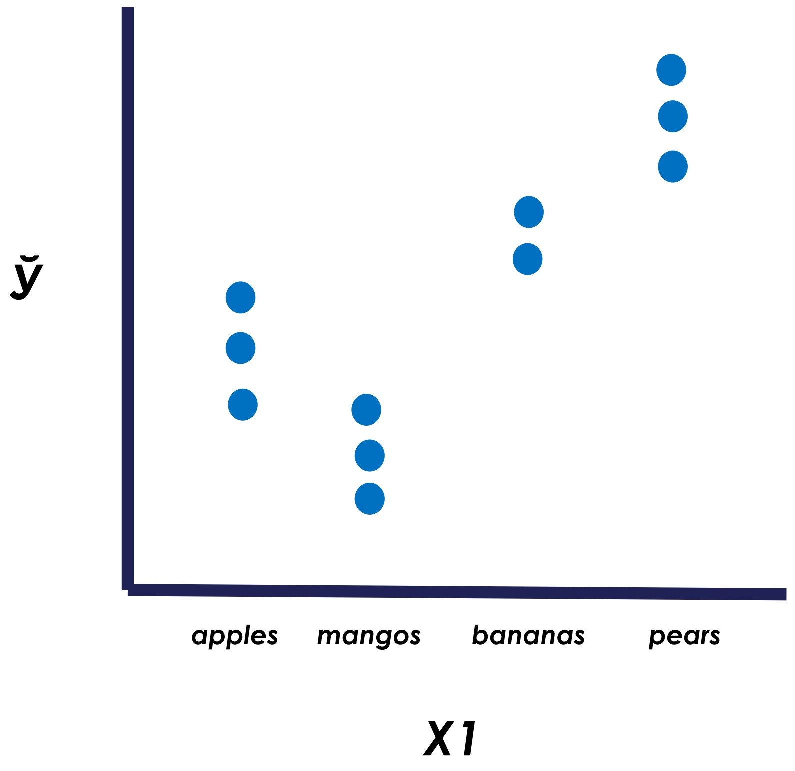
ALE – categorical features



What happens when features are categorical?

ALE plots make no sense.

ALE – categorical features

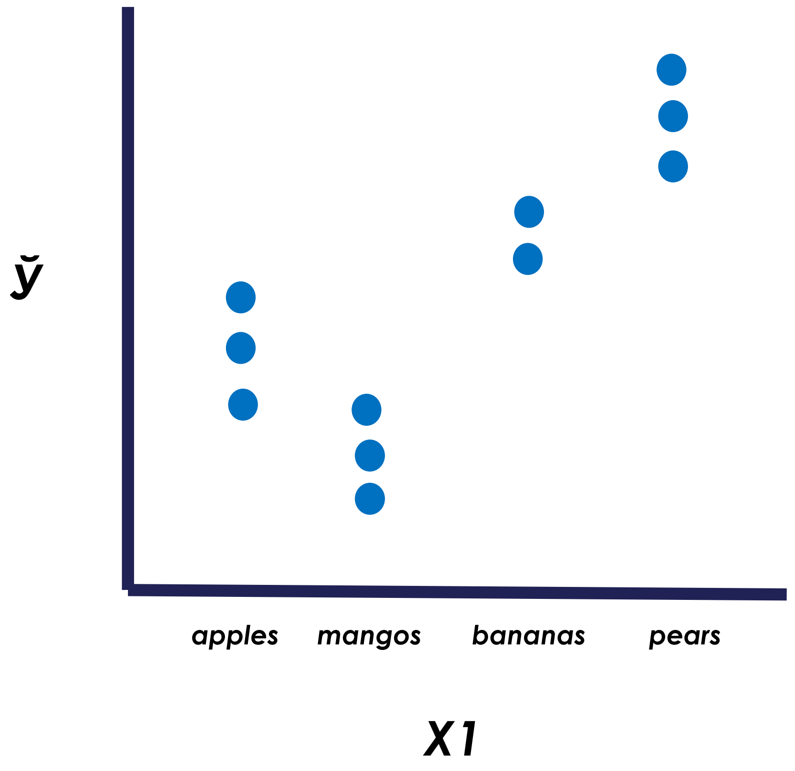


To train a model, we probably encoded the variables ➔ we assigned a somewhat arbitrary order.

We could proceed as if they were discrete.

For ALE plots, the order is important, so not having an intrinsic order defeats the point of ALE plots.

ALE – categorical features



In the original article, there is some discussion about assigning order to categories using distances calculated over the other features.

Not implemented in any open-source library. Presumably because we encode the variables in any case before training a model.