Other Considerations on Decision Trees

Here we gonna discuss some issues and considerations related to Decision Trees :

Categorical Predictors

When splitting a predictor that have q possible unordered values, there are

$$\text{possiblities} = 2^{q-1} - 1$$

to partition the q value into two groups.

- The computation becomes to expansive for large q possibilities

To optimize and simplify the computation we order the **predictor** by their proportion for each category $p_1 \leq p_2 \leq \cdots \leq p_q$, for example :
Trying to predict loan default $1 \rightarrow$ default , $0 \rightarrow$ n default

- For the predictor **city** it has 50 cities(categories)
- Rank the cities based on their default rate from the highest to the lowest a
- Find the optimal cut point which will results in a binary outcome
  - Split 1: {City F} vs {all other 49 cities}
  - Split 2: {City F, City B} vs {remaining 48 cities
  - Split 3: {City F, City B, City Z} vs {remaining 47 cities}
  - ...
  - Split 49: {first 49 cities} vs {last city}
- We calculate the **Gini index** or **entropy** for each split to measure the impurity $Q(T)$
- The result might be **Cities with default rate** ≤ 0.15 vs **Cities with default rate** > 0.15 giving us only **Left and Right node**
- giving us only 49 splits to evaluate

If we tried to calculate all the possible $2^{q-1} - 1$ splits it will be computationally impossible
with $O(2^q)$ complexity.
While the ordered approach is $O(q \log q)$ for sorting + $O(q)$ for split evaluation $\rightarrow$ computationally feasible

Note : even tho the partitioning algorithm tends to favor categorical predictors , many levels q can grow exponentially and in practice might need feature engineering for large values of q

The Loss Matrix