Seventeenth Session

Alireza Moradi

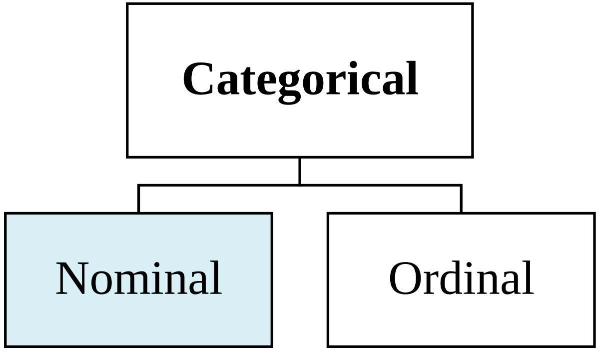
Encoding

- **Encoding** is the process of converting original data into a form that can be understood and used by machine learning models in a digital environment.

- **Machine learning models** can only work with numerical values.

- For this reason, it is necessary to transform the categorical values of the relevant features into numerical ones. This process is called feature encoding.

- **One-hot encoding** is a technique in machine learning that turns nominal data, into numerical data for machines to understand.



Encoding; Examples

- Suppose we want to predict the sentiment of a comment, which can be negative, neutral, or positive, how can we encode this?

Value	Replace with
"Negative"	
"neutral"	
"positive"	

- In another problem, we're dealing with predicting different respiratory illnesses. Specifically, patients arrive at the hospital with a persistent cough, fever, and shortness of breath. These symptoms could indicate Pneumonia, Influenza, or Tuberculosis.

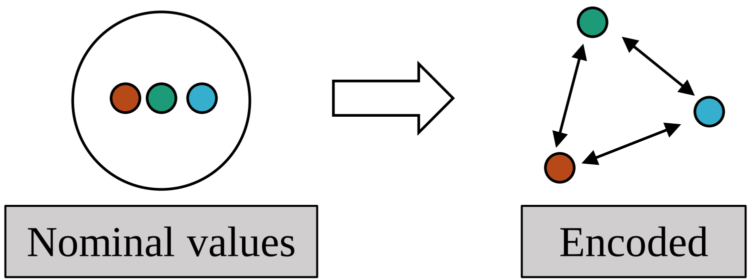
Value	Replace with
Pneumonia	
Influenza	
Tuberculosis	

✓ How can we encode in this scenario?

One-hot Encoding

- **One-hot encoding** creates new binary columns for each category, with a 1 marking the presence of that category and 0s elsewhere.

- This method eliminates the hierarchical order that numerical encoding might imply, allowing models to treat each category with equal importance.



- **One-hot encoding** is most useful when the number of categorical features is relatively small, as the increase in dimensionality is then more manageable

One Hot Encoding; Example

- Consider a categorical feature "color" with four categories: Red, Green, Yellow, and Blue.

Index	Color
1	Red
2	Green
3	Blue
4	Red
5	Yellow
6	Blue
7	Yellow

One-hot Encoding

One-hot Encoded Vector
[1 , 0 , 0 , 0]
[0 , 1 , 0 , 0]
[0 , 0 , 1 , 0]
[1 , 0 , 0 , 0]
[0 , 0 , 0 , 1]
[0 , 0 , 1 , 0]
[0 , 0 , 0 , 1]

Red	Green	Blue	Yellow
1	0	0	0
0	1	0	0
0	0	1	0
1	0	0	0
0	0	0	1
0	0	1	0
0	0	0	1

- So, We have converted the nominal feature "color" into four numerical features, making it suitable for use in ML models.

One-hot Encoding

- One-hot encoding can lead to the "**curse of dimensionality**".

- Suppose we want to predict import/export volume using *countries* and *HS codes*:

Feature name	HS code	country
Categories	5,000	70

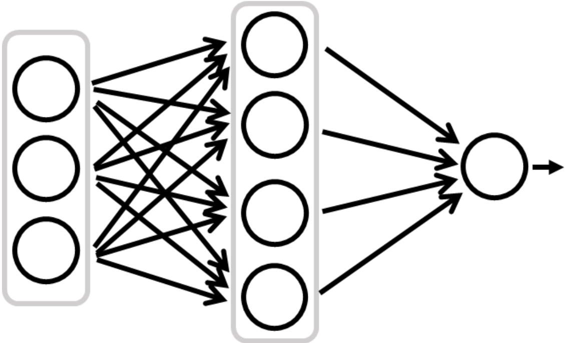
✓ How many new numerical features should we create?

- It creates a **sparse matrix**, which can be computationally intensive for some models to handle.

Parameters vs Hyperparameters

- Recall that in a machine learning model, **parameters** are the learnable values that the model adjusts during training using a learning algorithm.

- In ANNs, the parameters are the **weights** and **biases** that the architecture learns through the **backpropagation algorithm**.



✓ What are parameters in a linear regression model?

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 \cdot X_i$$

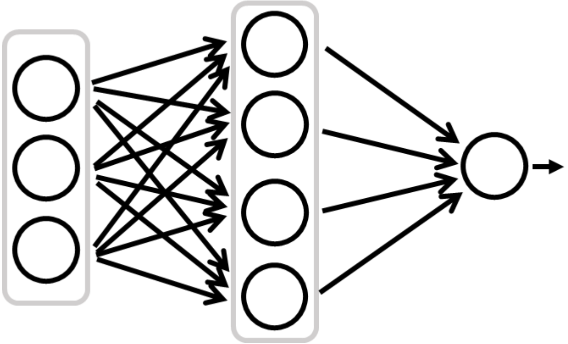
Parameters vs Hyperparameters

- **Hyperparameters** are external configuration variables that we use to manage machine learning model training.

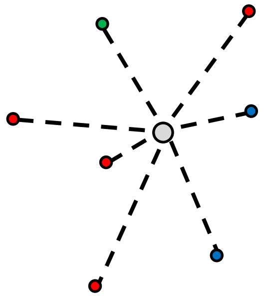
- The prefix **'hyper'** suggests that they are 'top-level' parameters.

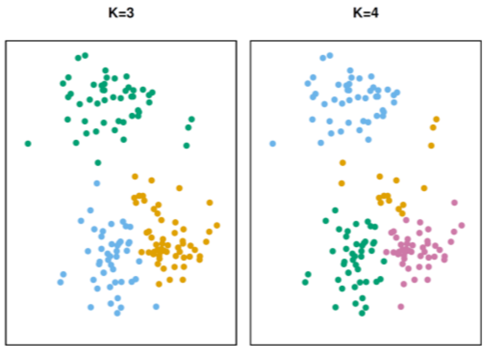
- A **hyperparameter value** is set before the learning process begins so, hyperparameters are not learned from the data.

- Example:

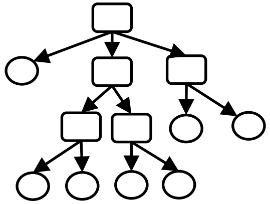


Hyperparameters; Examples





- K-NN;
  - The number of neighbors (K)
- K-means;
  - Number of clusters (K),
  - Distance metric (E.g., Euclidean or Manhattan Distance)



- Random Forest;
  - Number of trees (B),
  - Number of features used in each tree (m)
  - maximum depth of a decision tree
- SVM;
  - Penalty parameter (C)
  - Kernel type

$$\underset{\beta_0,\beta_1,\dots,\beta_p}{maximise}\ M$$

$subject$ to:

$$\sum_{j=1}^{p} \beta_j^2 = 1$$

$$y_i \cdot \left(\beta_0 + \beta_1.X_{i1} + \beta_2.X_{i2} + \ \dots + \beta_p.X_{ip}\right)$$
$$\geq M(1 - \varepsilon_i)$$
$$\varepsilon_i \geq 0 \quad \forall i = 1,2,\cdots,n$$
$$\textstyle\sum_{i=1}^{n} \varepsilon_i \leq C$$

Hyperparameter optimization

- ✓ Since hyperparameters are not learned via an algorithm, how do we determine the appropriate value to choose?

- **Hyperparameter optimization** finds a tuple of hyperparameters that yields an optimal model which minimizes a predefined loss function on given independent data.

- For example, when using SVM for a binary classification problem, we need to find the best model by exploring all combinations of kernels and penalty parameter (C).

$$(Kernel^*, C^*) \longrightarrow \text{Best tuple of hyperparameters}$$

- This tuples of hyperparameters optimizes the model performance.

Hyperparameter optimization

- In the ML world, there are many Hyperparameter optimization techniques are available.

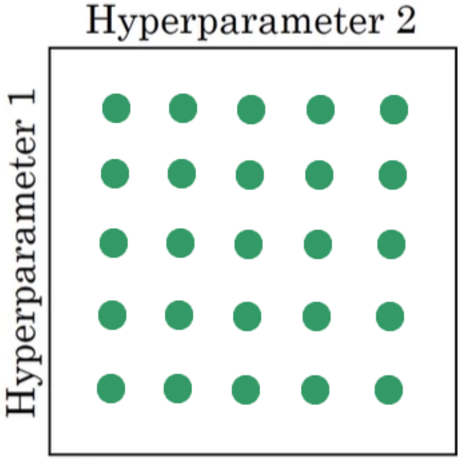
    - Manual Search
    - Random Search
    - Grid Search
    - Automated Hyperparameter tuning
    - Bayesian Optimization
    - …

- Hyperparameter Optimization = Hyperparameter Tuning

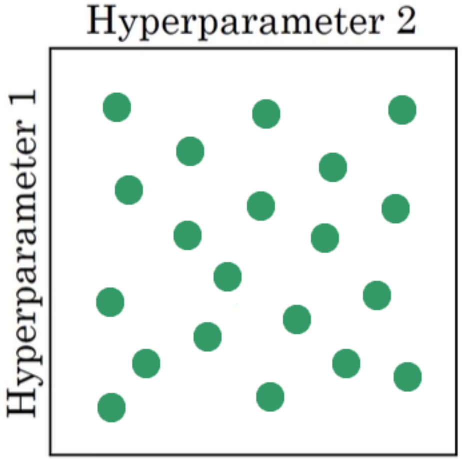
Grid search

- An example of hyperparameter tuning is a **grid search**.

- In **grid search**, we define a set of hyperparameter values **to search over**, and the algorithm tries all possible combinations of these values.



Grid search



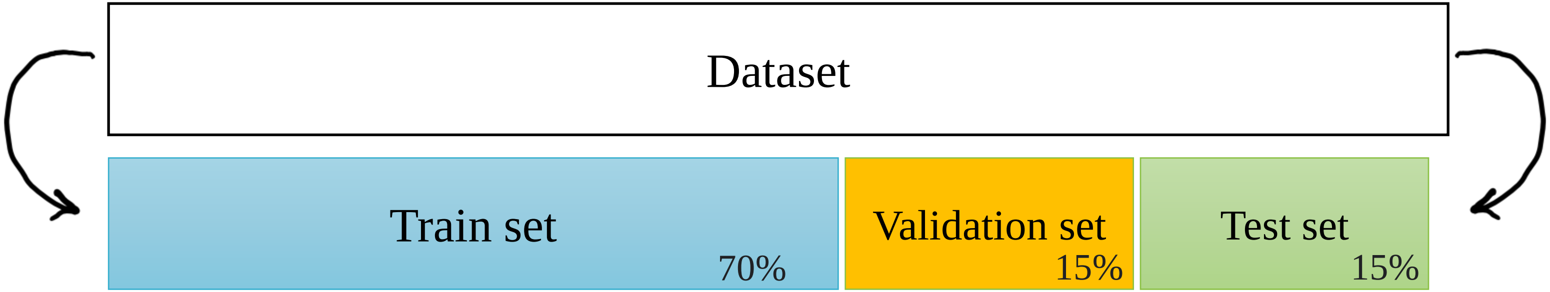
Random search

Train, validation, and test

- The **validation set** is used to select the best hyperparameters.

- It serves as an unbiased evaluation set during the model development process.

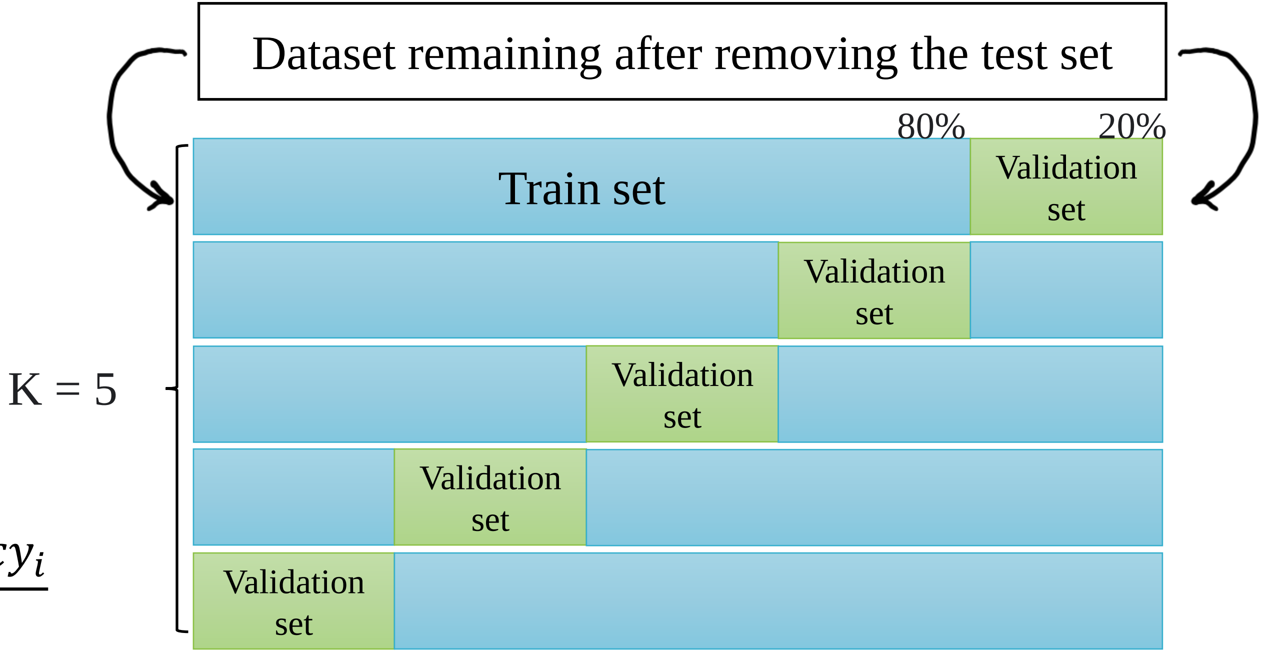
- By evaluating the model's performance on the validation set, we can compare different hyperparameter settings and select the best configuration.



✓ Why do we need test set?

Cross validation; Recall

- K-fold cross-validation:



$$\text{Accuracy} = \frac{\sum_{i=1}^{k} Accuracy_i}{k}$$

- We choose our model based on cross validation and estimate its performance using test set.