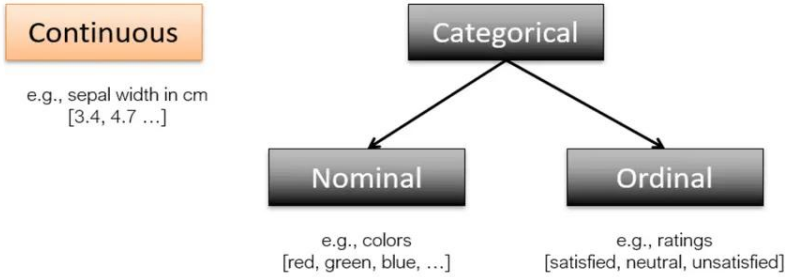
- Introduction: Definition, Real life applications, Introduction to Data in Machine Learning Types of Learning: Supervised Learning Unsupervised Learning, Semi-Supervised Learning, Reinforcement Learning, Concept of Feature, Feature Construction, Feature Selection and Transformation, Curse of Dimensionality. Dataset Preparation: Training Vs. Testing Dataset, Dataset Validation Techniques – Hold-out, kfold Cross validation, Leave-One-Out Cross- Validation (LOOCV)

- **What are the features in machine learning?**
  - Features are nothing but the independent variables in machine learning models.
  - A model for predicting the risk of cardiac disease may have features such as the following:
    - Age
    - Gender
    - Weight
    - Whether the person smokes
    - Whether the person is suffering from diabetic disease, etc.

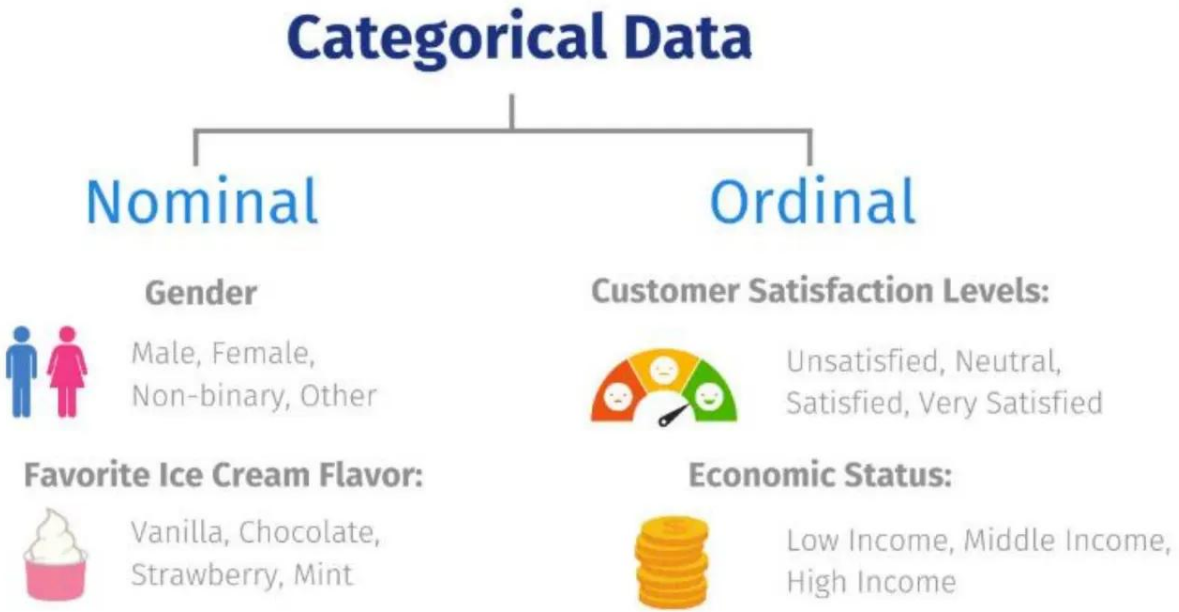
- A model for predicting whether the person is suitable for a job may have features such as the educational qualification, number of years of experience, experience working in the field etc.
- A model for predicting the size of a shirt for a person may have features such as age, gender, height, weight, etc.



- **Continuous features:**
  - Numerical values that can take on any value within a certain range.
  - This type of data is often used to represent things such as time, weight, income, temperature, etc.
  - Continuous features are often used in machine learning applications, since they can provide a more detailed representation of data than discrete or categorical features.
  - For example, imagine that you are trying to predict the weight of an animal based on its height. If you only had discrete data for height (e.g., “short,” “medium,” and “tall”), then your predictions would be less accurate than if you had continuous data (e.g., the animal’s actual height in inches or centimeters).
  - Continuous features can also be more useful than discrete features when it comes to optimizing models.

- **Categorical or discrete features:**
  - Categorical data is data that can be divided into categories, such as “male” and “female” or “red” and “blue.”
  - Categorical features can be **used to help predict what category something belongs to**, based on other features.
  - Categorical data can be thought of as a set of categories, and each category can be represented by a number.
  - For example, if we are predicting the type of animal based on a series of features, the animal’s species would be a categorical feature.

- **Categorical or discrete features:** Categorical features are of two types – **Nominal and Ordinal.**



# Concept of Feature

- **Categorical or discrete or qualitative features:**
  - **Nominal:**
    - Nominal features represent categories with no inherent order
    - Colors: Red, Blue, Green, Yellow
    - Types of cars: Sedan, SUV, Truck, Coupe
    - Eye color: Blue, Brown, Green, Hazel
    - Marital status: Single, Married, Divorced, Widowed

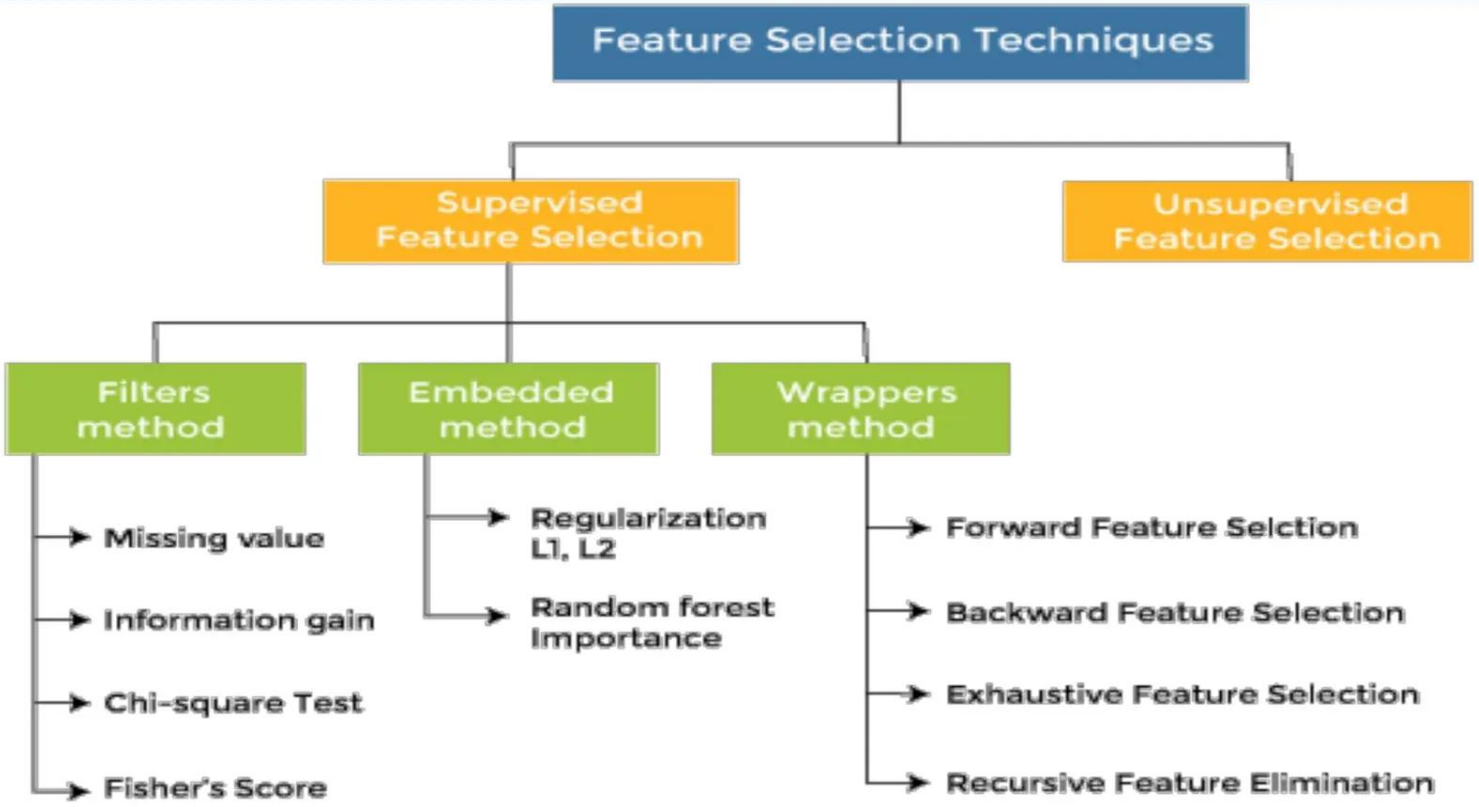
# Concept of Feature

- **Categorical or discrete or qualitative features:**
  - **Ordinal:**
    - Ordinal features, on the other hand, **have a meaningful order or ranking** between their categories
    - Education level: High School, Bachelor's, Master's, PhD
    - Customer satisfaction: Very Unsatisfied, Unsatisfied, Neutral, Satisfied, Very Satisfied
    - Movie ratings: 1 star, 2 stars, 3 stars, 4 stars, 5 stars
    - T-shirt sizes: Small, Medium, Large, X-Large

# Feature Selection

- **Feature Selection:**
  - Feature selection is a process that **chooses a subset of features from the original features** so that the feature space is optimally reduced according to a certain criterion.
  - Adding redundant variables reduces the generalization capability of the model and may also reduce the overall accuracy of a classifier.
  - The goal of feature selection techniques in machine learning is to find the best set of features that allows one to build optimized models of studied phenomena.

# Feature Selection



# Feature Selection

- **Supervised Techniques:**
  - These techniques can be used for labeled data and to identify the relevant features for increasing the efficiency of supervised models like classification and regression.
  - For Example, linear regression, decision tree, SVM, etc.

- **Unsupervised Techniques:**
  - These techniques can be used for unlabeled data.
  - For Example, K-Means Clustering, Principal Component Analysis, Hierarchical Clustering, etc

# Feature Selection

- **Filter Methods (Supervised Technique):**
  - These methods are generally used while doing the pre-processing step.
  - These methods select features from the dataset irrespective of the use of any machine learning algorithm.
  - Selection of feature is evaluated individually which can sometimes help **when features are in isolation** (don't have a dependency on other features) but **will lag when a combination of features can lead to increase in the overall performance** of the model.

Set of all features → Selecting the best subset → Learning algorithm → Performance

# Feature Selection

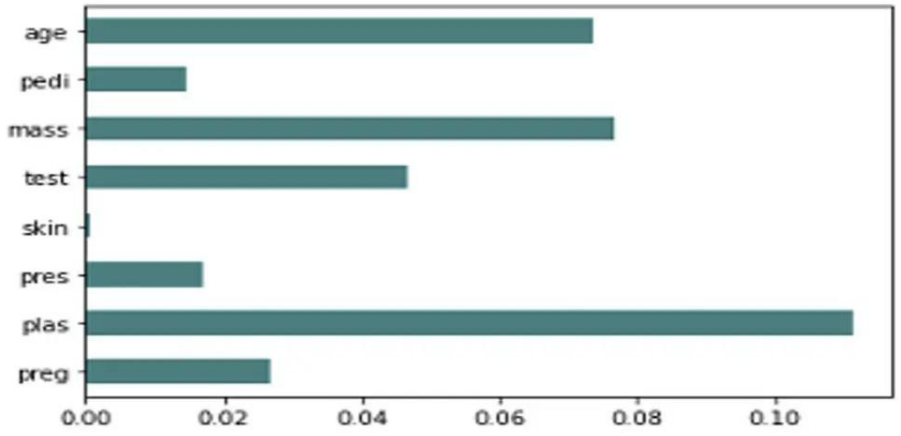
- **Filter Methods (Supervised Technique):**
  - In terms of computation, they are **very fast and inexpensive** and are very good for removing **duplicated, correlated, redundant features** but these methods do not remove multicollinearity.
  - Multicollinearity refers to a phenomenon where independent variables in a regression model are highly correlated with each other. This high correlation can make it difficult to determine the individual impact of each variable on the dependent variable, leading to unstable and unreliable regression results.

# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Information Gain*:**
    - It is defined as the **amount of information provided by the feature** for identifying the target value and measures reduction in the entropy values (entropy is a measure of randomness or disorder within a system).
    - Information gain of each attribute is calculated considering the target values for feature selection.
    - Mutual Information (Information Gain) tells us that **how much "input variable" having relation with "output/target variable".** If its having more Percentage of relation then those features can be considered for Model development and less relationship (indepedent features) features can be dropped.

# Feature Selection

```
1  from sklearn.feature_selection import mutual_info_classif
2  import matplotlib.pyplot as plt
3  %matplotlib inline
4
5  importances = mutual_info_classif(X, Y)
6  feat_importances = pd.Series(importances, dataframe.columns[0:len(dataframe.columns)-1])
7  feat_importances.plot(kind='barh', color = 'teal')
8  plt.show()
```



# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Chi-square test*:**
    - Chi-square method (X^2) is generally used to test the relationship between categorical variables.
    - It compares the observed values from different attributes of the dataset to its expected value.

$$X^2 = \sum \frac{(Observed\ value - Expected\ value)^2}{Expected\ value}$$

# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Chi-square test*:**
    - Chi-square tests check if your observed data matches what you expected. This helps you know if your ideas are on track or if you need to reconsider them.
    - Chi-square tests validate our beliefs based on empirical evidence and boost confidence in our inferences.

# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Chi-square test*:**
    - Example: Income Level vs Subscription Status (Link: https://www.geeksforgeeks.org/machine-learning/ml-chi-square-test-for-feature-selection/)
    - Let us examine a dataset with features including "income level" (low, medium, high) and "subscription status" (subscribed, not subscribed) indicate whether a customer subscribed to a service. The goal is to determine if this feature is relevant for predicting subscription status.

# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Chi-square test*:**

```
1  from sklearn.feature_selection import SelectKBest
2  from sklearn.feature_selection import chi2
3
4  # Convert to categorical data by converting data to integers
5  X_cat = X.astype(int)
6
7  # Three features with highest chi-squared statistics are selected
8  chi2_features = SelectKBest(chi2, k = 3)
9  X_kbest_features = chi2_features.fit_transform(X_cat, Y)
10
11 # Reduced features
12 print('Original feature number:', X_cat.shape[1])
13 print('Reduced feature number:', X_kbest_features.shape[1])
```

```
Original feature number: 8
Reduced feature number: 3
```

# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Fisher's Score:***
    - Fisher's Score selects each feature independently according to their scores under Fisher criterion leading to a suboptimal set of features.
    - The larger the Fisher's score is, the better is the selected feature.

# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Fisher's Score in Text Classification*** *(Ref: https://www.geeksforgeeks.org/machine-learning/fisher-score-for-feature-selection/)*
    - Suppose you are building a spam email classifier using a bag-of-words model.
    - Each email is converted into a vector where each feature represents the frequency of a specific word (e.g., "offer", "free", "meeting", etc.).
    - Now, you want to select the most informative words that help differentiate between the two classes: spam and not spam.
    - **Here's how Fisher Score helps:**
    - For each word (feature), compute the mean frequency of that word in spam emails and in non-spam emails.
    - Also compute the overall mean and standard deviations within each class.
    - Apply the Fisher Score formula to get a score for each word.
    - Words like "free" or "offer" may have high Fisher Scores, indicating they appear frequently in spam and rarely in non-spam, making them highly discriminative.
    - Words like "meeting" or "attached" might appear in both classes similarly, giving them low Fisher Scores, so they can be dropped.

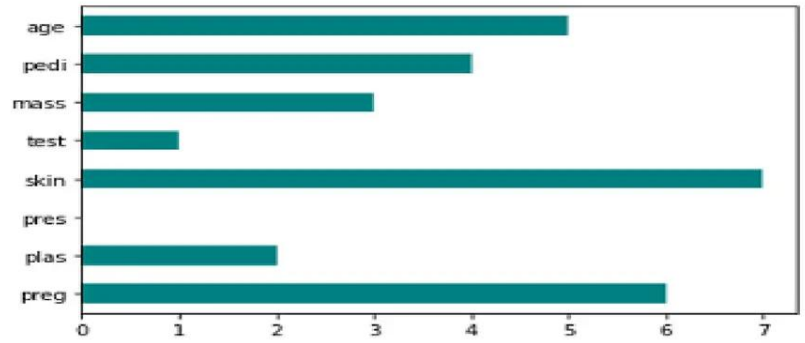
# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Fisher's Score:***

```
from skfeature.function.similarity_based import fisher_score
import matplotlib.pyplot as plt
%matplotlib inline

# Calculating scores
ranks = fisher_score.fisher_score(X, Y)

# Plotting the ranks
feat_importances = pd.Series(ranks, dataframe.columns[0:len(dataframe.columns)-1])
feat_importances.plot(kind='barh', color = 'teal')
plt.show()
```



# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Correlation Coefficient:***
    - Pearson's Correlation Coefficient is **a measure of quantifying the association between the two continuous variables** and the direction of the relationship with its values ranging from -1 to 1.
    - 1 shows a perfect positive correlation where both variables increase or decrease together at a constant rate.
    - -1 shows a perfect negative correlation where one variable increases as the other decreases proportionally.
    - 0 shows no linear relationship means changes in one variable do not predict changes in the other.

# Feature Selection

- **Filter Methods (Supervised Technique):**
  - ***Correlation Coefficient:***
    - r = 0.85 suggests a strong positive correlation such as more study time leading to better test scores.
    - r = - 0.75 shows a strong negative correlation like the inverse relationship between outdoor temperature and heating costs.

# Feature Selection

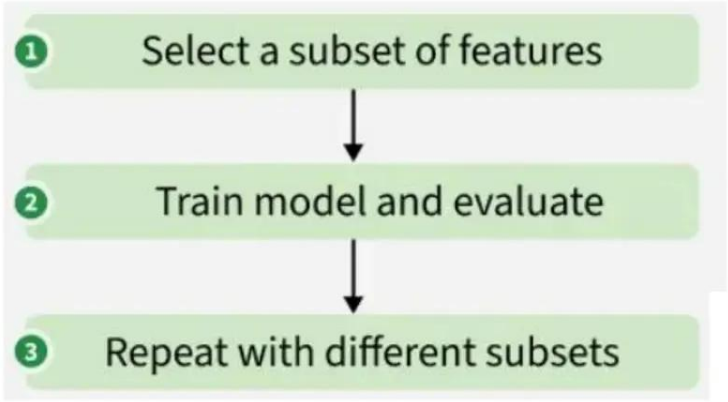
- **Filter Methods (Supervised Technique):**
  - ***Correlation Coefficient:***

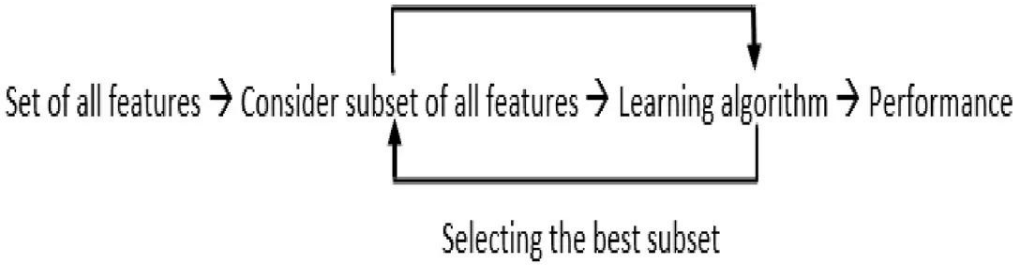
```
1  import seaborn as sns
2  import matplotlib.pyplot as plt
3  %matplotlib inline
4
5  # Correlation matrix
6  cor = dataframe.corr()
7
8  # Plotting Heatmap
9  plt.figure(figsize = (10,6))
10 sns.heatmap(cor, annot = True)
```

```
<AxesSubplot:>
```

- **Wrapper Methods (Supervised Technique):**
  - Wrapper methods, also **referred to as greedy algorithms** train the algorithm by using a subset of features in an iterative manner.
  - Based on the conclusions made from training in prior to the model, addition and removal of features takes place.
  - Stopping criteria for selecting the best subset are usually pre-defined by the person training the model such as when the performance of the model decreases or a specific number of features has been achieved.
  - The main advantage of wrapper methods over the filter methods is that they **provide an optimal set of features for training the model**, thus resulting in better accuracy than the filter methods but are computationally more expensive.

- **Wrapper Methods (Supervised Technique):**



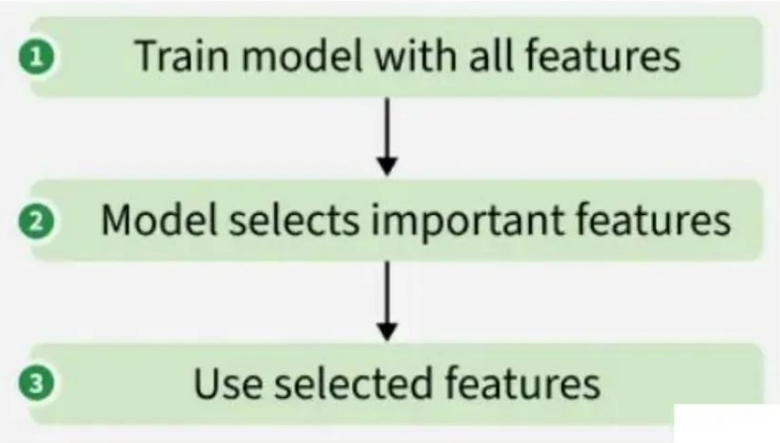



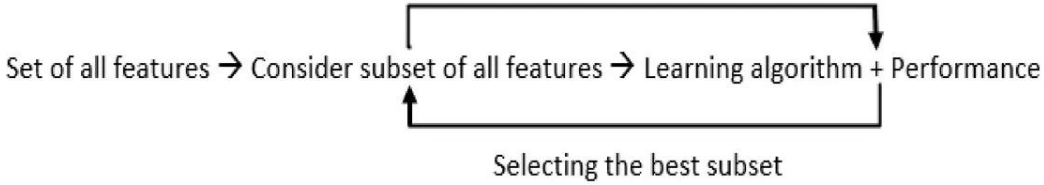
- **Wrapper Methods (Supervised Technique):**
  - **Forward selection**
    - This method is an iterative approach where we initially start with an empty set of features and keep adding a feature which best improves our model after each iteration. The stopping criterion is till the addition of a new variable does not improve the performance of the model.

  - **Backward elimination**
    - This method is also an iterative approach where we initially start with all features and after each iteration, we remove the least significant feature. The stopping criterion is till no improvement in the performance of the model is observed after the feature is removed.

- **Embedded methods (Supervised Technique):**
  - In embedded methods, the feature selection algorithm is blended as part of the learning algorithm, thus having its own built-in feature selection methods.
  - Embedded methods encounter the drawbacks of filter and wrapper methods and merge their advantages.
  - These methods are faster like those of filter methods and more accurate than the filter methods and take into consideration a combination of features as well
  - Works with a specific learning algorithm so the feature selection might not work well with other models

- **Embedded methods (Supervised Technique):**



- **Embedded methods (Supervised Technique):**
  - **Regularization**
    - This method adds a penalty to different parameters of the machine learning model to avoid over-fitting of the model.
    - This approach of feature selection uses Lasso (L1 regularization) and Elastic nets (L1 and L2 regularization).
    - The penalty is applied over the coefficients, thus bringing down some coefficients to zero.
    - The features having zero coefficient can be removed from the dataset.

- **Embedded methods (Supervised Technique):**
  - **Tree-based methods**
    - These methods such as Random Forest, Gradient Boosting provides us feature importance as a way to select features as well.
    - Feature importance tells us which features are more important in making an impact on the target feature.