**FSDS MAY BATCH 2022(ML Assignment -8)**

**Submitted by: Shubham Tiwari**

Q1: What exactly is a feature? Give an example to illustrate your point.

Ans: A feature is a characteristic or attribute of an object or item that is used as input for a model. In other words, it is a specific piece of information that describes an object. Features are used to represent an object so that a machine learning model can make predictions or decisions about it.

For example, in a problem of identifying the type of fruit based on its features, the features could be color, shape, size, weight and texture. The color feature could be represented by a number between 0 and 255 (in a RGB color space) and the shape feature could be represented by a set of values such as roundness, flatness and elongation.

Another example is in a dataset of customers of a company, the features could be age, income, location, marital status, etc. These features can be used to make predictions about customer behavior, such as likelihood of purchasing a product, or to segment customers into different groups for targeted marketing.

Q2: What are the various circumstances in which feature construction is required?

Ans: Feature construction is often required when the raw data is not in a form that can be easily used as input to a machine learning model. There are several circumstances in which feature construction is required:

1. **Missing or incomplete data:** If the data is missing or incomplete, new features can be constructed from the available data to fill in the missing values.
2. **Data in different units or scales:** If the data is in different units or scales, new features can be constructed to normalize or standardize the data.
3. **Data in different forms:** If the data is in different forms, such as text or images, new features can be constructed by extracting useful information from the data, such as by converting text to numerical values or by extracting features from images.
4. **Combining multiple data sources:** If multiple data sources are available, new features can be constructed by combining the data from these sources.
5. **Handling categorical variables:** If the data contains categorical variables, new features can be constructed by encoding these variables in a way that a machine learning model can understand.
6. **Improving model performance:** New features can be constructed to improve the performance of a machine learning model. This could be done by creating interaction features, adding polynomial features, etc.
7. **Domain knowledge:** By using domain knowledge, new features can be constructed that capture important information that was not captured in the original data.

Q3: Describe how nominal variables are encoded.

Ans: Nominal variables are categorical variables that do not have a natural order or ranking. They are typically encoded using one-hot encoding, where each category is represented by a binary column, with a 1 indicating that the observation belongs to that category, and a 0 indicating that it does not. For example, if a variable has three categories (A, B, and C), three binary columns would be created, one for each category. If an observation belongs to category A, then the value for the A column would be 1 and the values for the B and C columns would be 0. This method allows the model to understand the categorical nature of the variable without assuming any order or ranking between the categories.

Q4: Describe how numeric features are converted to categorical features.

Ans: Numeric features can be converted to categorical features in several ways, depending on the nature of the data and the desired outcome. Some common methods include:

1. **Binning**: This method involves dividing the range of numeric values into a set of bins or intervals, and then mapping each numeric value to the corresponding bin or interval. For example, you can divide a range of ages into bins such as "under 18", "18-24", "25-34", "35-44", etc.
2. **Discretization:** This method involves dividing a continuous variable into a set of discrete values, based on certain rules or thresholds. For example, you can convert a feature like "income" into discrete categories such as "low", "medium", "high"
3. **One-hot encoding**: This method involves converting a numeric feature into a set of binary features, one for each unique value of the numeric feature. Each binary feature represents whether the original feature had a specific value or not.
4. **Ordinal encoding**: This method involves assigning an ordinal value to each unique value of a numeric feature. For example, if you have a feature that contains values like "small", "medium", "large", you can assign an ordinal value of 1,2 and 3 respectively.
5. **Dummy variable**: This method involves creating a new binary variable for each category of the original feature. For example, if you have a feature that contains values like "red", "green", "blue", you can create three new binary variables: "is\_red", "is\_green", "is\_blue".
6. The choice of method depends on the nature of the data, the desired outcome, and the algorithm that will be used to analyze the data.

Q5: Describe the feature selection wrapper approach. State the advantages and disadvantages of this approach?

Ans: The feature selection wrapper approach is a method of evaluating the performance of a model while iteratively selecting subsets of features. It uses an external performance metric, such as accuracy or F1 score, to evaluate the performance of the model with different subsets of features.

The basic process of the feature selection wrapper approach is as follows:

1. Initialize a subset of features
2. Train a model using the current subset of features.
3. Evaluate the model's performance using a performance metric.
4. Add or remove features from the subset, and repeat steps 2-3.
5. Select the subset of features that yields the best performance.

The advantages of the feature selection wrapper approach include:

* It can be used with any machine learning algorithm
* It takes into account the interactions between features and the model
* It can be useful in cases where the number of features is large, and it is not clear which features are most important

The disadvantages of the feature selection wrapper approach include:

* It can be computationally expensive, especially when the number of features is large
* It is sensitive to the choice of performance metric and the specific model
* It may be prone to overfitting, as it selects a subset of features based on the specific training data used
* It may be biased towards features that are easily separable, even if they are not the most informative.

Overall, the feature selection wrapper approach can be a powerful technique for feature selection, but it should be used with caution, and in conjunction with other feature selection methods.

Q6: When is a feature considered irrelevant? What can be said to quantify it?

Ans: A feature is considered irrelevant when it does not contribute to the overall goal or purpose of the system or program. It can be identified through user research, testing, or other forms of user feedback, by examining if users are using it, or if the feature is not aligned with the target user's needs or expectations.

Criteria that can be used to identify irrelevant features include:

* Lack of usage or engagement with the feature.
* Feedback or complaints from users indicating that the feature is not useful or relevant to them.
* The feature not being aligned with the overall goals or objectives of the system or program.
* The feature not being aligned with the target user's needs or expectations.
* The feature not being aligned with the current business or product strategy.

Quantifying the relevance of a feature can be done by measuring its usage, user engagement or feedback, and its impact on the overall performance, usability or satisfaction of the system or program.

Q7: When is a function considered redundant? What criteria are used to identify features that could be redundant?

Ans: **A function is considered redundant when it serves no purpose or adds no value to the system or program**. It can be identified through code review or testing, by examining its inputs and outputs, and determining if they can be produced by other existing functions or if they are not used by other parts of the program.

Criteria that can be used to identify redundant features include:

* Lack of usage or calls to the function
* Functionality that is already provided by other functions or libraries
* Functionality that can be achieved through simpler or more efficient means
* Functions that are no longer necessary due to changes in requirements or design
* Overlapping or duplicate functionality with other functions in the program.

Q8: What are the various distance measurements used to determine feature similarity?

Ans: There are several distance measurements that can be used to determine feature similarity. Some common ones include:

* **Euclidean distance**: This measures the straight-line distance between two points in a multidimensional space.
* **Manhattan distance:** This measures the distance between two points by summing the absolute differences of their coordinates.
* **Cosine similarity:** This measures the cosine of the angle between two vectors and ranges from -1 (completely dissimilar) to 1 (completely similar).
* **Jaccard similarity**: This measures the similarity between two sets by calculating the size of the intersection divided by the size of the union of the sets.
* **Mahalanobis distance**: This is a measure of distance that takes into account the covariance of the data.
* **Minkowski distance:** This is a generalization of Euclidean and Manhattan distance.

These are just a few examples and there are many other distance metrics available. The choice of which distance metric to use will depend on the specific data and problem we are working on.Top of Form

Q9: State difference between Euclidean and Manhattan distances?

Ans: Euclidean distance is a measure of the straight-line distance between two points in Euclidean space, while Manhattan distance is a measure of the distance between two points in a grid-like pattern, such as the layout of streets in Manhattan. In simpler terms, Euclidean distance is the straight-line distance between two points, while **Manhattan distance is the distance between two points when you can only move along grid lines, like a rook in chess.**

Q10: Distinguish between feature transformation and feature selection.

Ans: Feature transformation is the process of modifying or transforming the features in a dataset in order to make them more informative or suitable for a particular machine learning algorithm. This can include techniques such as normalization, scaling, or dimensionality reduction.

Feature selection, on the other hand, is the process of selecting a subset of the most relevant features from a dataset to use in a machine learning model. This can be done using techniques such as feature importance ranking, mutual information or correlation based feature selection.

In summary, feature transformation is modifying the feature set to improve its representation, while feature selection is selecting a subset of features from the set to be used.

Q11: Make brief notes on **any two** of the following:

1.SVD (Standard Variable Diameter Diameter).

Ans: xxxxxxxxxxxxxxxxx

2. Collection of features using a hybrid approach.

Ans: xxxxxxxxxxxxxxxxxxxxx

3. The width of the silhouette.

Ans: In machine learning, the silhouette is a method of evaluation for determining the quality of a clustering algorithm. The silhouette width is a measure of how similar an object is to its own cluster compared to other clusters. The silhouette width is defined as the difference between the average distance between an object and all other objects in its own cluster (intra-cluster distance) and the average distance between an object and all other objects in the nearest cluster (inter-cluster distance).The silhouette width ranges from -1 to 1, with a width of 1 indicating that the object is perfectly matched to its own cluster and a width of -1 indicating that the object is better matched to another cluster. A value close to 0 indicates that the object is on or close to the decision boundary between two clusters. In general, the wider the silhouette, the better the clustering algorithm is performing.

4. Receiver operating characteristic curve.

Ans: A receiver operating characteristic (ROC) curve is a graphical representation of the performance of a binary classifier system as the discrimination threshold is varied. The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The true positive rate is the number of true positive instances divided by the number of true positive plus false negative instances, while the false positive rate is the number of false positive instances divided by the number of false positive plus true negative instances. A perfect classifier would have a TPR of 1 and a FPR of 0, resulting in a point in the top left corner of the ROC space.