**Question 1: What is feature engineering, and how does it work? Explain the various aspects of feature engineering in depth.**

Answer: Feature engineering is the process of selecting, transforming, or creating relevant features from raw data to enhance the performance of machine learning models. It involves several aspects:

- Feature Selection: Choosing pertinent features. This enhances model simplicity, reduces overfitting, and speeds up training.

- Feature Transformation: Applying mathematical operations like scaling, normalization, and log transforms. This ensures features are comparable and have the same scale.

- Feature Creation: Generating new features from existing ones. Polynomial features, interaction terms, and domain-specific aggregations can be created.

- Handling Missing Data: Addressing missing values through imputation or adding indicator variables for missingness.

- Encoding Categorical Variables: Converting categorical data into numerical form, often via one-hot encoding or label encoding.

- Handling Text and Time Data: Converting text into numerical representations like TF-IDF or word embeddings. Time data can be broken into day, month, year components.

**Question 2: What is feature selection, and how does it work? What is the aim of it? What are the various methods of function selection?**

Answer: Feature selection involves choosing a subset of relevant features from the original set to improve model performance and reduce complexity, with the goal of avoiding overfitting. Methods include:

- Filter Methods: Evaluate features independently of the model using metrics like correlation, variance, or mutual information.

- Wrapper Methods: Use a machine learning model as evaluator, testing subsets of features and selecting the best set.

- Embedded Methods: Incorporate feature selection within model training, like LASSO regularization.

**Question 3: Describe the function selection filter and wrapper approaches. State the pros and cons of each approach?**

Answer:

- Filter Approaches: Rank features independently using statistical metrics. Pros: Efficiency, computational ease. Cons: Might not capture feature interactions.

- Wrapper Approaches: Use a model to evaluate subsets of features. Pros: Capture interactions, model-specific effects. Cons: Computationally intensive, prone to overfitting.

**Question 4:**

**i. Describe the overall feature selection process.**

**ii. Explain the key underlying principle of feature extraction using an example. What are the most widely used function extraction algorithms?**

Answer:

i. Overall feature selection process involves:

1. Define the problem and dataset.

2. Generate potential features.

3. Use filter methods to rank features.

4. Apply wrapper methods to search for best subset.

5. Evaluate model on a separate test dataset.

ii. Feature extraction aims to reduce dimensionality while retaining relevant information. Example: Principal Component Analysis (PCA) transforms correlated features into orthogonal components that capture maximum variance. Other algorithms include Linear Discriminant Analysis (LDA) and Non-Negative Matrix Factorization (NMF).

**Question 5: Describe the feature engineering process in the sense of a text categorization issue.**

Answer: In text categorization:

1. Tokenization: Split text into words or subword units.

2. Stop Words Removal: Remove common, uninformative words.

3. TF-IDF Transformation: Scale word frequencies by importance.

4. N-grams: Include word sequences for context.

5. Word Embeddings: Convert words into dense vector representations.

**Question 6: What makes cosine similarity a good metric for text categorization? Calculate the cosine similarity between two document-term vectors.**

Answer: Cosine similarity is effective for text categorization because it measures the cosine of the angle between two vectors, indicating their direction similarity regardless of magnitude. It's suitable for comparing text documents with varying lengths. Given vectors (2, 3, 2, 0, 2, 3, 3, 0, 1) and (2, 1, 0, 0, 3, 2, 1, 3, 1), cosine similarity ≈ 0.669.

**Question 7:**

**i. What is the formula for calculating Hamming distance? Calculate the Hamming distance between 10001011 and 11001111.**

**ii. Compare the Jaccard index and similarity matching coefficient for features (1, 1, 0, 0, 1, 0, 1, 1) and (1, 0, 0, 1, 1, 0, 0, 1).**

Answer:

i. Hamming distance formula: Count differing bits. Hamming distance between 10001011 and 11001111 = 2.

ii. Jaccard Index: 2 common / 6 total = 0.333. Similarity Matching Coefficient: 4 matching / 8 total = 0.5.

**Question 8: What is meant by "high-dimensional data set"? Offer examples and describe the challenges.**

Answer: A high-dimensional data set has many features compared to samples. Examples: genomic data, images, text. Challenges include increased computation, sparsity, curse of dimensionality (sparse data and distances), and overfitting. Techniques like dimensionality reduction can address these.

**Question 9: Provide quick notes on:**

**1. PCA is an acronym for Personal Computer Analysis.**

**2. Use of vectors**

**3. Embedded technique**

Answer:

1. Incorrect. PCA stands for Principal Component Analysis, a dimensionality reduction method.

2. Vectors are mathematical representations used in ML for data points in multi-dimensional space.

3. Embedded technique combines feature selection with model training.

**Question 10: Compare:**

**1. Sequential backward exclusion vs. sequential forward selection**

**2. Function selection methods: filter vs. wrapper**

**3. SMC vs. Jaccard coefficient**

Answer:

1. Sequential backward exclusion removes least important features; forward selection adds most important.

2. Filter ranks features based on criteria; wrapper uses a model to evaluate subsets.

3. SMC (Similarity Matching Coefficient) for binary data; Jaccard coefficient for binary and non-binary, based on intersection/union.