AWS Certified Machine Learning Engineer - Associate Exam Guide

# Section 1.2 — Transform Data & Perform Feature Engineering (In-Depth)

This section covers transforming raw data and creating features for machine learning models. It includes cleaning, feature engineering, encoding, and AWS services to operationalize these steps.

## 1. Data Cleaning

• Handle missing values: imputation (mean, median, mode), deletion, placeholder tokens

• Handle outliers: capping, removal, transformations (log, Box-Cox)

• Deduplication: remove duplicates, ensure uniqueness in IDs/keys

• Data validation: schema checks, constraints

## 2. Feature Engineering

• Scaling: Min-max, z-score normalization, robust scaling

• Splitting: train/validation/test partitions, stratified sampling

• Binning: bucketization of continuous variables (quantiles, custom bins)

• Normalization/Standardization: improve performance for gradient-based algorithms

• Log transforms: stabilize variance, handle skew

• Polynomial & interaction features

## 3. Encoding Techniques

• One-hot encoding: for categorical variables (works with most algorithms, may cause high dimensionality)

• Label encoding: convert categories into integers (works for tree-based models)

• Binary encoding: compact representation for high-cardinality categorical features

• Embeddings: dense vector representations (used for text, categorical, and image data)

• Tokenization: break text into words, subwords, or characters

## 4. AWS Tools for Transformation & Feature Engineering

• AWS Glue → ETL transformations at scale

• AWS Glue DataBrew → no-code data profiling & cleaning

• Amazon EMR (with Spark) → distributed data transformations

• SageMaker Data Wrangler → visual data prep, feature engineering, export to pipelines

• SageMaker Processing Jobs → run preprocessing scripts at scale

• SageMaker Feature Store → manage and share features consistently between training and inference

## 5. Labeling & Validation

• SageMaker Ground Truth → scalable data labeling with human-in-the-loop

• Mechanical Turk integration for manual labeling

• Data validation → ensure correct splits, schema consistency, and proper labeling

## ✅ Exam Tips

• Use SageMaker Data Wrangler for interactive transformations

• Use Glue DataBrew for quick no-code profiling/cleaning tasks

• Feature Store = consistency across train and inference

• Stratified sampling is critical for imbalanced datasets

• One-hot encoding works for most algorithms, but embeddings are preferred for high-cardinality features

• Always validate datasets after transformation to ensure schema/consistency