SAS® Viya® Supervised Machine Learning Pipelines Exam

Data Sources (30 – 36%)

Create a project in Model Studio

- Bring data into Model Studio for analysis
  - Import data from a local source (Import tab)
  - Add data from a stored data source (Data Sources tab)
  - Use an in-memory data source (Available tab)
- Create Model Studio Pipelines with the New Pipeline window
  - Automatically generate pipelines
  - Pipeline templates
- Advanced Advisor options
  - Maximum class level
  - Maximum % missing
  - Interval cut-off
- Partition data into training, validation, and test
  - Explain why partitioning is important
  - Explain the different methods to partition data (stratified vs simple random)
- Use Event Based Sampling for rare events.
- Set up Node Configuration

Explore the data

- Use the DATA EXPLORATION node
- Profile data during data definition
- Preliminary data exploration using the data tab
- Save data with the SAVE DATA node

Modify data

- Explain concepts of replacement, transformation, imputation, filtering, outlier detection
- Modify metadata within the DATA tab
- Modify metadata with the MANAGE VARIABLES node
- Use the REPLACEMENT node to update variable values
- Use the TRANSFORMATION node to correct problems with input data sources, such as variables distribution or outliers
- Use the IMPUTE node to impute missing values and create missing value indicators
- Prepare text data for modeling with the TEXT MINING node
- Explain common data challenges and remedies for supervised learning

Use the VARIABLE SELECTION node to identify important variables to be included in a predictive model

- Unsupervised Selection
- Fast Supervised Selection
- Linear Regression Selection
- Decision Tree Selection
- Forest Selection
- Gradient Boosting Selection
- Create Validation from Training
- Use multiple methods within the same VARIABLE SELECTION node

Building Models (40 – 46%)

Describe key machine learning terms and concepts

- Data partitioning: training, validation, test data sets
- Observations (cases), independent (input) variables/features, dependent (target) variables
- Measurement scales: Interval, ordinal, nominal (categorical), binary variables
- Supervised vs unsupervised learning
- Prediction types: decisions, rankings, estimates
- Curse of dimensionality, redundancy, irrelevancy
- Decision trees, neural networks, regression models, support vector machines (SVM)
- Model optimization, overfitting, underfitting, model selection
- Describe ensemble models
- Explain autotuning

Build models with decision trees and ensemble of trees

- Explain how decision trees identify split points
    - Split search algorithm
    - Recursive partitioning
    - Decision tree algorithms
    - Multiway vs. binary splits
    - Impurity reduction
    - Gini, entropy, Bonferroni, IGR, FTEST, variance, chi-square, CHAID
    - Compare methods to grow decision trees for categorical vs continuous response variables
- Explain the effect of missing values on decision trees
- Explain surrogate rules
- Explain the purpose of pruning decision trees
- Explain bagging vs. boosting methods
- Build models with the DECISION TREE node
    - Adjust splitting options
    - Adjust pruning options
- Build models with the GRADIENT BOOSTING node

    - Adjust general options: number of trees, learning rate, L1/L2 regularization
    - Adjust Tree Splitting options
    - Adjust early stopping
- Build models with the FOREST node
    - Adjust number of trees
    - Adjust tree splitting options
- Interpret decision tree, gradient boosting, and forest results (fit statistics, output, tree diagrams, tree maps, variable importance, error plots, autotuned results)

Build models with neural networks

- Describe the characteristics of neural network models
    - Universal approximation
    - Neurons, hidden layers, perceptrons, multilayer perceptrons
    - Weights and bias
    - Activation functions
    - Optimization Methods (LBFGS and Stochastic Gradient Descent)
    - Variable standardization
    - Learning rate, annealing rate, L1/L2 regularization
- Build models with the NEURAL NETWORK node
    - Adjust number of layers and neurons
    - Adjust optimization options and early stopping criterion
- Interpret NEURAL NETWORK node results (network diagram, iteration plots, and output)

Build models with support vector machines

- Describe the characteristics of support vector machines.
- Build model with the SVM node
    - Adjust general properties (Kernel, Penalty, Tolerance)
- Interpret SVM node results (Output)

Use Model Interpretability tools to explain black box models

- Partial Dependence plots
- Individual Conditional Expectation plots
- Local Interpretable Model-Agnostic Explanations plots
- Kernel-SHAP plots

Incorporate externally written code

- Open Source Code node
- SAS Code node
- Score Code Import node

Model Assessment and Deployment Models (24 – 30%)

Explain the principles of Model Assessment

- Explain different dimensions for model comparison
    - Training speed
    - Model application speed
    - Tolerance
    - Model clarity
- Explain honest assessment
    - Evaluate a model with a holdout data set
- Use the appropriate fit statistic for different prediction types
    - Average error for estimates
    - Misclassification for decisions
- Explain results from the INSIGHTS tab

Assess and compare models in Model Studio

- Compare models with the MODEL COMPARISON node
- Compare models with the PIPELINE COMPARISON tab
- Interpret Fit Statistics, Lift Reports, ROC reports, Event Classification chart
- Interpret Fairness and Bias plots

Deploy a model

- Exporting score code
- Registering a model
- Publish a model
- SCORE DATA node

Note: All 13 main objectives will be tested on every exam. The expanded objectives are provided for additional explanation and define the entire domain that could be tested.