SAS Text Analytics, Time Series, Experimentation and Optimization Exam

Text Analytics - 30%

Create data sources for text mining

- Create data sources that can be used by SAS Enterprise Miner Projects
- Identify data sources that are relevant for text mining

Import data into SAS Text Analytics

- Process document collections and create a single SAS data set for text mining using the Text Import Node
- Merge a SAS data set created from Text Importer with another SAS data set containing target information and other non-text variables
- Compare two models, one using only conventional input variables and another using the conventional inputs and some text mining variables

Use text mining to support forensic linguistics using stylometry techniques

Retrieve information for Analysis

- Use the Interactive Text Filter Viewer for information retrieval
- Use the Medline medical abstracts data for information retrieval

Parse and quantify Text

- Provide guidelines for using weights
- Use SVD to project documents and terms into a smaller dimension metric space
- Discuss Text Topic and Text Cluster results in light of the SVD

Perform predictive modeling on text data

- Explain the trade-off between predictive power and interpretability
- Set up Text Cluster and Text Topic nodes to affect this trade-off
- Perform predictive modeling using the Text Rule Builder node

Use the High-Performance (HP) Text Miner Node

- Identify the benefits of the HP Text Miner node
- Use the HPTMINE procedure

Time Series - 30%

Identify and define time series characteristics, components and the families of time series models

- Transform transactional data into time series data (Accumulate) using PROC TIMESERIES
  - Transactional Data Accumulation and Time Binning
- Define the systematic components in a time series (level, seasonality, trend, irregular, exogenous, cycle)
- Describe the decomposition of time series variation (noise and signal)
- List three families of time series models
  - exponential smoothing (ESM)
  - autoregressive integrated moving average with exogenous variables (ARIMAX)
  - unobserved components (UCM)
- Identify the strengths and weaknesses of the three model types
  - usability
  - complexity
  - robustness
  - ability to accommodate dynamic regression effects

Diagnose, fit and interpret ARIMAX Models

- Analyze a time series with respect to signal (system variation) and noise (random variation)
- Explain the importance of the Autocorrelation Function Plot and the White Noise Test in ARMA modeling
- Compare and contrast ARMA and ARIMA models
- Define a stationary time series and discuss its importance
- Describe and identify autoregressive and moving average processes
- Estimate an order 1 autoregressive model
- Evaluate estimates and goodness-of-fit statistics
- Explain the X in ARMAX
- Relate linear regression with time series regression models
- Recognize linear regression assumptions
- Explain the relationship between ordinary multiple linear regression models and time series regression models
- Explain how to use a holdout sample to forecast
- Given a scenario, use model statistics to evaluate forecast accuracy
- Given a scenario, use sample time series data to exemplify forecasting concepts

Diagnose, fit and interpret Exponential Smoothing Models

- Describe the history of ESM

- Explain how ESMs work and the types of systematic components they accommodate
- Describe each of the seven types of ESM formulas
- Given a sample data set, choose the best ESM using a hold-out sample, output fit statistics, and forecast data sets

Diagnose, fit and interpret Unobserved Components Models

- Describe the basic component models: level, slope, seasonal
- Be able to explain UCM strengths and when it would be good to use UCM
    - Example: Visualization of component variation
- Given a sample scenario, be able to explain how you would build a UCM
    - Adding and deleting component models and interpreting the diagnostics

Experimentation & Incremental Response Models - 20%

Explain the role of experiments in answering business questions

- Determine whether a business question should be answered with a statistical model
- Compare observational and experimental data
- List the considerations for designing an experiment
- Control the experiment for nuisance variables
- Explain the impact of nuisance variables on the results of an experiment
- Identify the benefits of deploying an experiment on a small scale

Relate experimental design concepts and terminology to business concepts and terminology

- Define Design of Experiments (DOE) terms (response, factor, effect, blocking, etc)
- Map DOE terms to business marketing terms
- Define and interpret interactions between factors
- Compare one-factor-at-a-time (OFAT) experiment methods to factorial methods
- Describe the attributes of multifactor experiments (randomization, orthogonality, etc)
- Identify effects in a multifactor experiment
- Explain the difference between blocks and covariates

Explain how incremental response models can identify cases that are most responsive to an action

- Design the experimental structure to assess the impact of the model versus the impact of the treatment
- Explain the effect of both the model and the message from assessment experiment data

- Describe the standard customer segments with respect to marketing campaign targets
- Explain the value of using control groups in data science
- Define an incremental response

Use the Incremental Response node in SAS Enterprise Miner

- List the required data structure components of the Incremental Response node
- Explain Net Information Value (NIV) and Penalized Net Information Value (PNIV) and their use in SAS Enterprise Miner
- Explain Weight of Evidence (WOE) and Net Weight of Evidence (NWOE) and their use in SAS Enterprise Miner
- Use stepwise regression with the Incremental Response node
- Adjust model properties for various types of incremental revenue analysis
- Compare variable/constant revenue and cost models
- Understand and explain the value of difference scores in the combined incremental response model
- Use difference scores to compare treatment and control

Optimization - 20%

Optimize linear programs

- Explain local properties of functions that are used to solve mathematical optimization problems
- Use the OPTMODEL procedure to enter and solve simple linear programming problems
- Formulate linear programming problems using index sets and arrays of decision variables, families of constraints, and values stored in parameter arrays
- Modify a linear programming problem (changing bounds or coefficients, fixing variables, adding variables or constraints) within the OPTMODEL procedure
- Use the Data Envelope Analysis (DEA) linear programming technique

Optimize nonlinear programs

- Describe how, conceptually and geometrically, iterative improvement algorithms solve nonlinear programming problems
- Identify the optimality conditions for nonlinear programming problems
- Solve nonlinear programming problems using the OPTMODEL procedure
- Interpret information written to the SAS log during the solution of a nonlinear programming problem
- Differentiate between the NLP algorithms and how solver options influence the NLP algorithms

Note: All 17 main objectives will be tested on every exam. The 87 expanded objectives are provided for additional explanation and define the entire domain that could be tested.