What does SAS® Visual Statistics on SAS® Viya™ do?
It enables you to interactively explore data, and build descriptive and predictive models using either a visual drag-and-drop interface or a programming interface. Data scientists can collaborate with business analysts to refine models for better insights. Distributed, in-memory processing shortens data exploration and model development time.

Why is SAS® Visual Statistics on SAS® Viya™ important?
With SAS Visual Statistics on SAS Viya, changing market conditions or shifts in customer behavior can be rapidly reflected in analytical models. Fast experimentation and iterative building and refinement of well-qualified models for each segment or group produces valuable insights that can be appropriately acted on. It enhances productivity and collaboration for analytics and business teams, while supporting governance and centralized administration for IT.

For whom is SAS® Visual Statistics on SAS® Viya™ designed?
SAS Visual Statistics is primarily designed for use by statisticians, data scientists, programmers and citizen data scientists who need to build, refine and evaluate predictive models to get powerful insights.

Benefits

- **Surface new opportunities faster and beat your competition.** Data scientists and statisticians can operate on observations at a granular level using the most appropriate analytical modeling technique for the problem they are trying to solve. The result? Unprecedented speed in uncovering insights and finding new ways to increase revenue.

- **Improve the productivity of your analytics staff.** SAS Visual Statistics reduces manual experimentation and improves collaboration. Multiple users can visually interact with data – adding or changing variables, removing outliers, etc. – and instantly see how those changes affect predictive power. If a model does not meet the needs of business or domain experts, your analytics staff can quickly go back and refine it.

- **Dramatically shorten model development time and put better models into action sooner.** It’s easy to build and refine models to target specific groups or segments, and run numerous scenarios simultaneously. Analytical professionals can ask more what-if questions and get relevant answers because refined models produce better results. Then put those results into action with automatically generated score code for more timely outcomes.

- **Empower users with the programming language of their choice.** Python, Java, R and Lua programmers can experience the power of SAS Visual Statistics without having to learn how to program in SAS. Give them the flexibility to access trusted and tested SAS machine-learning and statistical algorithms from other coding environments.

- **Gain ultimate scalability.** The distributed, in-memory environment means IT can easily scale up and out as data needs grow, numbers of users increase or they tackle more complex problems. Built-in failover management in distributed environments guarantees submitted jobs always finish and computing resources are available whenever needed.
Overview

SAS Visual Statistics on SAS Viya provides a visual drag-and-drop interface for creating descriptive and predictive models on data of any size - rapidly. It also provides an interactive programming interface for those who want to code in SAS, or in other languages while taking advantage of powerful SAS statistical modeling and machine-learning techniques. These analytical modeling techniques are used to predict outcomes that result in better, more targeted actions.

SAS Visual Statistics takes advantage of the SAS Viya engine for even faster results. SAS Viya brings new enhancements to the SAS Platform, including high availability, faster in-memory processing and native cloud support. Data and analytical workload operations are automatically distributed across the cores of a single server or the nodes of a massive compute cluster, taking advantage of parallel processing. It also provides a single, integrated, scalable environment that can be easily managed, maintained and governed.

A visual data exploration and discovery environment

SAS Visual Statistics is an add-on to SAS Visual Analytics, meaning the products share the same interactive data exploration and predictive modeling interface. You get an integrated process for going from data to analytically derived insights.

Both the point-and-click interface and the programming interface let you easily identify predictive drivers among multiple variables. This is the power of SAS Visual Statistics.

Figure 1: Use the point-and-click visual interface to build and refine clusters.

Key Features

Visual data exploration and discovery (available through SAS Visual Analytics)

- Quickly interpret complex relationships or key variables that influence modeling outcomes within large data sets.
- Filter observations and understand a variable’s level of influence on overall model lift.
- Detect outliers and/or influence points to help you determine, capture and remove them from downstream analysis (e.g., models).
- Explore data using bar charts, histograms, box plots, heat maps, bubble plots, geographic maps and more.
- Derive predictive outputs or segmentations that can be used directly in other modeling or visualization tasks. Outputs can be saved and passed to those without model-building roles and capabilities.

Visual interface access to analytical techniques

- Clustering:
  - K-means clustering.
  - Parallel coordinate plots to interactively evaluate cluster membership.
  - Scatter plots of inputs with cluster profiles overlaid for small data sets and heat maps with cluster profiles overlaid for large data sets.
  - Detailed summary statistics (means of each cluster, number of observations in each cluster, etc.).
  - Generate on-demand cluster ID as a new column.
  - Support holdout data (training and validation) for model assessment.

- Decision trees:
  - Computes measures of variable importance.
  - Supports classification and regression trees.
  - Based on a modified C4.5 algorithm or cost-complexity pruning.
  - Interactively grow and prune a tree. Interactively train a subtree.
  - Set tree depth, max branch, leaf size, aggressiveness of tree pruning and more.
  - Use tree map displays to interactively navigate the tree structure.
  - Generate on-demand leaf ID as a new column.
  - Support holdout data (training and validation) for model assessment.
exploratory variables, and visually discover and understand outliers and data discrepancies.

Visual data exploration makes it much easier to understand relationships in your data, derive new variables and select relevant variables to improve your model development efforts. Find out which variables are relevant as inputs to your model and which variables best define your segmentation strategy.

With integrated model building and visual data discovery, you can maintain an uninterrupted workflow, cycling quickly between hypotheses and verification. This will boost your modeling confidence, productivity and accuracy.

GUI-based access to predictive modeling techniques

With the visual web browser interface, it’s a simple drag-and-drop process for citizen data scientists to experiment with and create powerful descriptive and predictive models.

And if faced with a backlog of modeling projects or the need to experiment with lots of granular segments, programmers and data scientists could find the visual GUI useful for creating models quickly.

The visual interface provides point-and-click access to linear regression, logistic regression, generalized linear models and two-dimension reduction techniques – clustering and decision trees. When working with large and complex data, dimension reduction techniques can improve modeling accuracy. Explore and evaluate segments for further analysis using k-means clustering, scatter plots and detailed summary statistics. Decision trees can be built for both

Key Features

- Linear regression:
  - Influence statistics.
  - Variable selection.
  - Frequency and weight variables.
  - Residual diagnostics.
  - Summary table includes overall ANOVA, model dimensions, fit statistics, model ANOVA, Type III test and parameter estimates.
  - Generate on-demand predicted values and residuals as new columns.
  - Support holdout data (training and validation) for model assessment.

- Logistic regression:
  - Models for binary data with logit and probit link functions.
  - Influence statistics.
  - Variable selection.
  - Frequency and weight variables.
  - Residual diagnostics.
  - Summary table includes model dimensions, iteration history, fit statistics, convergence status, Type III tests, parameter estimates and response profile.
  - Generate on-demand predicted labels and predicted event probabilities as new columns. Adjust the prediction cutoff to label an observation as event or non-event.
  - Support holdout data (training and validation) for model assessment.

- Generalized linear models:
  - Distributions supported include beta, normal, binary, exponential, gamma, geometric, Poisson, inverse Gaussian and negative binomial.
  - Offset variable support.
  - Frequency and weight variables.
  - Residual diagnostics.
  - Summary table includes model summary, iteration history, fit statistics, Type III test table and parameter estimates.
  - Informative missing option for treatment of missing values on the predictor variable.
  - Generate on-demand predicted values and residuals as new columns.
  - Support holdout data (training and validation) for model assessment.
classification and regression. After creating a
decision tree, you can interactively prune
trees and train subtrees.

Open, code-based model development

While the visual GUI of SAS Visual Statistics is
powerful and appealing, many statisticians,
data scientists and quantitative specialists
prefer to code their own predictive models
and take advantage of more options to fine-
tune the models.

Analytical actions running in SAS Visual
Statistics on SAS Viya can be programmatic-
ally accessed from the SAS Studio
programming interface, or can be called
from other languages such as Python, R, Lua
and Java.

In addition, public REST APIs enable coders
to add proven SAS Analytics to existing
applications.

This puts the power of SAS in the hands of
programmers who may not be familiar with
SAS – but know it’s the best analytics
software available.

SAS Visual Statistics on SAS Viya provides
programming access to many tested and
proven SAS algorithms running in the
distributed in-memory environment,
including those for:

• Data manipulation.
• Variable binning.
• Missing value imputation.
• Supervised and unsupervised variable
  selection.
• Clustering using k-means and k-modes.
• Decision trees.
• Principal component analysis.
• Linear and logistic regression.
• Nonlinear regression.
• Generalized linear regression.
• Ordinary least squares.
• Partial least squares.
• Quantile regression.

Key Features (continued)

Programming access to analytical techniques

- Programmers and data scientists can access SAS Viya (CAS server) from SAS Studio
  using SAS procedures (PROCs) and other tasks.
- Programmers can execute CAS actions using PROC CAS or use different programming
  environments like Python, R, Lua and Java.
- Users can also access SAS Viya (CAS server) from their own applications using public
  REST APIs.
- Provides native integration to Python Pandas DataFrames. Python programmers can
  upload DataFrames to CAS and fetch results from CAS as DataFrames to interact with
  other Python packages such as Pandas, matplotlib, Plotly, Bokeh, etc.

- Principal component analysis (PCA):
  - Performs dimension reduction by computing principal components.
  - Provides the eigenvalue decomposition, NIPALS and ITERGS algorithms.
  - Outputs principal component scores across observations.
  - Creates scree plots and pattern profile plots.

- Decision trees:
  - Supports classification trees and regression trees.
  - Supports categorical and numerical features.
  - Provides criteria for splitting nodes based on measures of impurity and
    statistical tests.
  - Provides the cost-complexity and reduced-error methods of pruning trees.
  - Supports partitioning of data into training, validation and testing roles.
  - Supports use of validation data for selecting the best subtree.
  - Supports the use of test data for assessment of final tree model.
  - Provides various methods of handling missing values, including surrogate rules.
  - Creates tree diagrams.
  - Provides statistics for assessing model fit, including model-based (resubstitution)
    statistics.
  - Computes measures of variable importance.
  - Outputs leaf assignments and predicted values for observations.

Figure 3: Use open source languages, including Python, to build descriptive and
predictive models.
• Descriptive statistics.
• Supervised model assessment.

This open programming environment provides flexibility for data scientists and statisticians, based on their programming skills and preferences, to easily access the power of SAS for data manipulation and advanced analytics.

Dynamic group-by processing
With SAS Visual Statistics, many users can concurrently build numerous models and process results for each group or segment without having to sort or index data each time. The grouping variables, or their properties, can change from one action to the next, and groups are processed without shuffling or reordering the data.

This means more models can be quickly created for more segments or groups on the fly without additional processing overhead. The result? Models that meet the unique needs of individual segments or groups.

Model comparison and assessment
After models have been created, they can be easily compared and assessed using a variety of statistical comparison summaries such as lift charts, ROC charts, concordance statistics and misclassification tables on one or more models from either the visual or programming interface.

And, from the visual interface, an interactive slider lets you manipulate cutoff thresholds so you can easily and visually evaluate lift at different percentiles. Combine model fitting with model diagnostics to quickly see and understand impacts on performance.

Figure 4: Dynamic group-by processing computes results for each group, partition or segment without having to sort or index data each time.

Key Features (continued)
• Clustering:
  • Provides the k-means algorithm for clustering continuous (interval) variables.
  • Provides the k-modes algorithm for clustering nominal variables.
  • Provides various distance measures for similarity.
  • Provides the aligned box criterion method for estimating the number of clusters.
  • Outputs cluster membership and distance measures across observations.
• Linear regression:
  • Supports linear models with continuous and classification variables.
  • Supports various parameterizations for classification effects.
  • Supports any degree of interaction and nested effects.
  • Supports polynomial and spline effects.
  • Supports forward, backward, stepwise, least angle regression and lasso selection methods.
  • Supports information criteria and validation methods for controlling model selection.
  • Offers selection of individual levels of classification effects.
  • Preserves hierarchy among effects.
  • Supports partitioning of data into training, validation and testing roles.
  • Provides a variety of diagnostic statistics.
  • Generates SAS code for production scoring.
• Logistic regression:
  • Supports binary and binomial responses.
  • Supports various parameterizations for classification effects.
  • Supports any degree of interaction and nested effects.
  • Supports polynomial and spline effects.
  • Supports forward, backward, fast backward and lasso selection methods.
  • Supports information criteria and validation methods for controlling model selection.
  • Offers selection of individual levels of classification effects.
  • Preserves hierarchy among effects.
  • Supports partitioning of data into training, validation and testing roles.
  • Provides variety of statistics for model assessment.
  • Provides variety of optimization methods for maximum likelihood estimation.

• Clustering:
  • Provides the k-means algorithm for clustering continuous (interval) variables.
  • Provides the k-modes algorithm for clustering nominal variables.
  • Provides various distance measures for similarity.
  • Provides the aligned box criterion method for estimating the number of clusters.
  • Outputs cluster membership and distance measures across observations.
• Linear regression:
  • Supports linear models with continuous and classification variables.
  • Supports various parameterizations for classification effects.
  • Supports any degree of interaction and nested effects.
  • Supports polynomial and spline effects.
  • Supports forward, backward, stepwise, least angle regression and lasso selection methods.
  • Supports information criteria and validation methods for controlling model selection.
  • Offers selection of individual levels of classification effects.
  • Preserves hierarchy among effects.
  • Supports partitioning of data into training, validation and testing roles.
  • Provides a variety of diagnostic statistics.
  • Generates SAS code for production scoring.
• Logistic regression:
  • Supports binary and binomial responses.
  • Supports various parameterizations for classification effects.
  • Supports any degree of interaction and nested effects.
  • Supports polynomial and spline effects.
  • Supports forward, backward, fast backward and lasso selection methods.
  • Supports information criteria and validation methods for controlling model selection.
  • Offers selection of individual levels of classification effects.
  • Preserves hierarchy among effects.
  • Supports partitioning of data into training, validation and testing roles.
  • Provides variety of statistics for model assessment.
  • Provides variety of optimization methods for maximum likelihood estimation.
Model assessment features let you compare models to identify the ones that produce the best lift and ROI.

Model scoring
After you have determined which model performs best, your champion model can be easily applied against new data.

Moving all of the data preparation tasks and algorithms of a sophisticated model from a development environment to an operational system is usually one of the most difficult aspects of predictive modeling and machine learning.

With SAS Visual Statistics, you can export your models as SAS DATA step code and easily apply them to new data. Putting your predictive models into production produces the insights needed for making better decisions and taking optimal actions.

Distributed, in-memory analytical processing with SAS* Viya™
SAS Visual Statistics runs on SAS Viya, a new high-performance runtime engine that uses in-memory analytical processing to provide answers to a range of business questions in a single, scalable and governed environment. It takes advantage of distributed, parallel processing for blazingly fast speed, dramatically reducing data exploration and model development time.

SAS Viya delivers high availability and scalable processing so IT can scale computing capacity up and out to meet the needs of more users who are dealing with more data and increasingly complex analytical problems.

You need analytical processing power you can count on. The fault-tolerant design of SAS Viya automatically detects server failure, even in multiplatform processing environments, and redistributes processing as needed. It also manages several copies of data on the processing cluster. If a node in

Key Features (continued)

- Generalized linear models:
  - Supports responses with variety of distributions, including binary, normal, Poisson and gamma.
  - Supports various parameterizations for classification effects.
  - Supports any degree of interaction and nested effects.
  - Supports polynomial and spline effects.
  - Supports forward, backward, fast backward, stepwise and group lasso selection methods.
  - Supports information criteria and validation methods for controlling model selection.
  - Offers selection of individual levels of classification effects.
  - Preserves hierarchy among effects.
  - Supports partitioning of data into training, validation and testing roles.
  - Provides variety of statistics for model assessment.
  - Provides a variety of optimization methods for maximum likelihood estimation.

- Nonlinear regression models:
  - Fits nonlinear regression models using least squares or maximum likelihood estimation.
  - Supports standard distributions for responses such as binary, Poisson and normal.
  - Supports programming syntax for specifying customized distributions for responses.
  - Supports programming syntax for specifying models and expressions of parameters.
  - Provides a variety of optimization methods for parameter estimation.
  - Computes analytical derivatives for parameter estimation.
  - Supports bounds for functions of parameters.
  - Supports linear restrictions on parameter estimates.
  - Provides confidence limits for functions of parameters.
  - Constructs predictions of expressions across observations.

- Quantile regression models:
  - Supports quantile regression for single or multiple quantile levels.
  - Supports multiple parameterizations for classification effects.
  - Supports any degree of interactions (crossed effects) and nested effects.
  - Supports hierarchical model selection strategy among effects.
  - Provides multiple effect-selection methods.
  - Provides effect selection based on a variety of selection criteria.
  - Supports stopping and selection rules.

- Predictive partial least squares models:
  - Provides programming syntax with classification variables, continuous variables, interactions and nestings.
  - Provides effect-construction syntax for polynomial and spline effects.
  - Supports partitioning of data into training and testing roles.
  - Provides test set validation to choose the number of extracted factors.
  - Implements the following methods: principal component regression, reduced rank regression and partial least squares regression.

Descriptive statistics
- Distinct counts to understand cardinality.
- Box plots to evaluate centrality and spread, including outliers for one or more variables.
- Correlations to measure the Pearson’s correlation coefficient for a set of variables.
- Cross-tabulations, including support for weights.
- Contingency tables, including measures of associations.
the cluster becomes unavailable or fails, the required data is retrieved from another block to quickly continue processing. These self-healing mechanisms ensure high availability for uninterrupted processing and automated recovery.

**Concurrent access to data in memory**
SAS Viya enables you to load and persist data in memory on demand, and execute multipass analytical computations. All data, tables and objects are held in memory as long as required, whether it’s for interactive visual investigations or advanced analytical processing. Many users can collaborate to explore the same raw data and build models simultaneously.

In-memory processing eliminates unnecessary and expensive data shuffling during iterative steps or requests. This means that the results from changes to models (e.g., adding new variables or removing outliers) are instantly visible.

**Flexible deployment options**
SAS Visual Statistics offers deployment options for organizations with any size data, different workloads and varying performance requirements. Deployments can scale from a single-machine environment for departmental workgroups or small to midsize businesses up to large distributed systems with hundreds of nodes in a cluster for large organizations.

It can run in both private and public cloud infrastructures.

SAS Visual Statistics on SAS Viya is also available in the Cloud Foundry environment. Cloud Foundry is an open source cloud application platform for easily deploying, managing, orchestrating and updating enterprise cloud applications. SAS Visual Statistics is available as a collection of Cloud Foundry BOSH releases for customers to deploy into their own Cloud Foundry environment.

**Key Features (continued)**

- Histograms with options to control binning values, maximum value thresholds, outliers and more.
- Multidimensional summaries in a single pass of the data.
- Percentiles for one or more variables.
- Summary statistics such as number of observations, number of missing values, sum of nonmissing values, mean, standard deviation, standard errors, corrected and uncorrected sums of squares, min and max, and the coefficient of variation.
- Kernel density estimates using normal, tri-cube and quadratic kernel functions.

**Group-by processing**
- Build models, compute and process results on the fly for each group or segment without having to sort or index the data each time.
- Build segment-based models instantly (i.e., stratified modeling) from a decision tree or clustering analysis.

**Model comparison and assessment**
- Generate model comparison summaries such as lift charts, ROC charts, concordance statistics and misclassification tables for one or more models.
- Interactively slide the prediction cutoff for automatic updating of assessment statistics and classification tables.
- Interactively evaluate lift at different percentiles.

**Model scoring**
- Export models as SAS DATA step code to integrate models with other applications. Score code is automatically concatenated if a model uses derived outputs from other models (e.g., leaf ID, cluster ID, etc.).

---

Figure 5: Evaluate multiple models to find the best performer.
Key Features (continued)

SAS® Viya™ in-memory runtime engine
- SAS Viya features an in-memory server called CAS (Cloud Analytic Services). It provides the ability to perform processing in memory and distribute processing across nodes in a cluster.
- User requests (expressed in a procedural language) are translated into actions with necessary parameters to process in a distributed environment. The result set and messages are passed back to the procedure for further action by the user.
- Data in CAS is managed in blocks and can be loaded in memory on demand. If your workload requires you to work with tables that exceed the memory capacity, the server caches the blocks on disk. Data and intermediate results are held in memory as long as required, across jobs and user boundaries, reducing latency.
- The distributed CAS server adds highly efficient node-to-node communication and uses an algorithm to determine the optimal number of nodes for a given job.
- CAS has a communication layer that supports fault tolerance. It can continue processing requests even after losing connectivity to some nodes, guaranteeing that submitted jobs always finish. The communication layer also enables you to remove or add nodes from or to a server while it is running.

Deployment options
- On-premises deployments:
  - Single-machine mode to support the needs of small to midsize organizations.
  - Distributed mode to meet growing data, workload and scalability requirements.
- Cloud deployments:
  - Enterprise hosting; private or public cloud (e.g., BYOL in Amazon) infrastructure; Cloud Foundry platform as a service (PaaS) to support multiple cloud providers.

Figure 6: Decision trees illustrate the most likely outcomes and consequences. Build them using the visual interface or with code.