

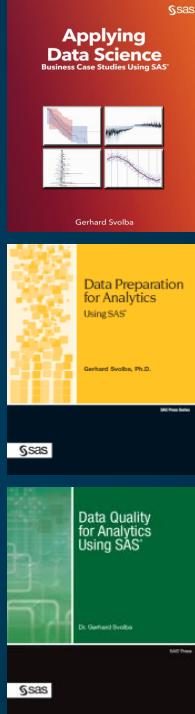
# Wie bringe ich 4 unterschiedliche Analytik-Benutzergruppen an einen Tisch? – Die Offenheit von SAS Viya ermöglicht eine Analyseplattform für unterschiedliche Benutzertypen

Gerhard Svolba, SAS Austria

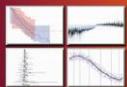
Mannheim, 2. März 2018 - KSFE 2018



<https://github.com/gerhard1050/>

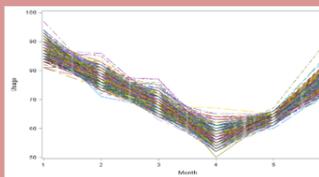
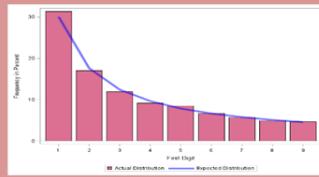






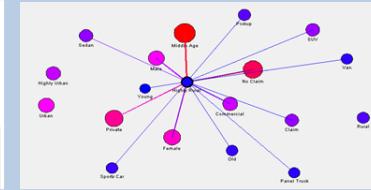
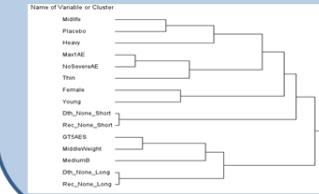
## Checking the Alignment with Predefined Pattern

*Which customers show a behaviour which is far from what you expected?*



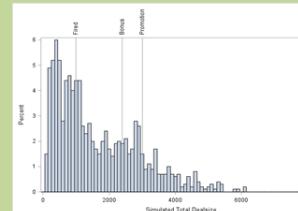
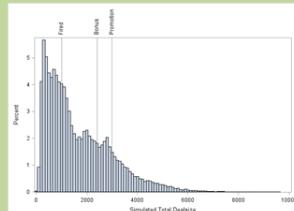
## Listen to Your Data – Discover Unknown Relationships

*Can your data tell you stories, even if you don't ask them?*



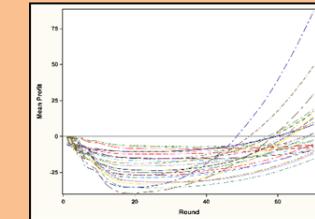
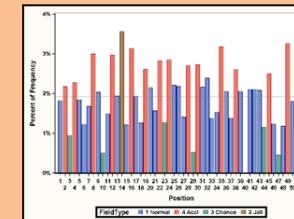
## Using Monte Carlo Simulations to Understand the Outcome Distribution

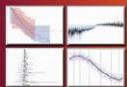
*Will the Sales Manager keep his job (when you look at his sales pipeline)?*



## Studying Complex Systems – Simulate the Monopoly® Board Game

*How can you simulate complex environments to get insight in the most frequent processes?*

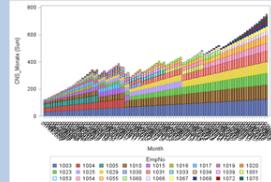
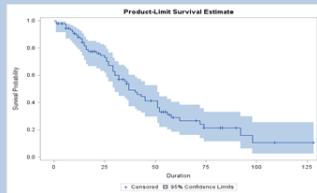




Gerhard Stolla

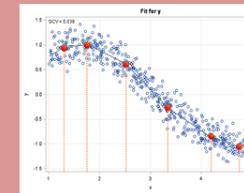
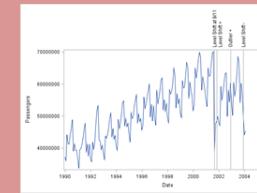
## Performing Headcount Survival Analysis for Employee Retention

*Can you make assumptions about the average length of time intervals, even if most of the endpoints have not yet been observed?*



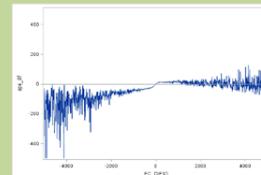
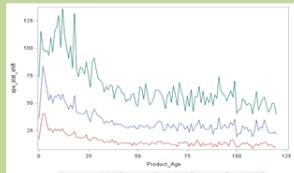
## Detecting Outliers and Structural Changes in Longitudinal Data

*Can you automatically detect events and changes in the course of your data over time?*



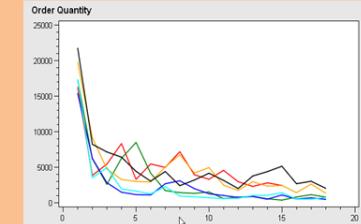
## Explaining Deviations and Forecast Errors

*Do the demand planners really improve forecast accuracy with their manual overwrites?*



## Forecasting the Demand for New Products

*Can you assess the expected demand of products that are introduced right now?*

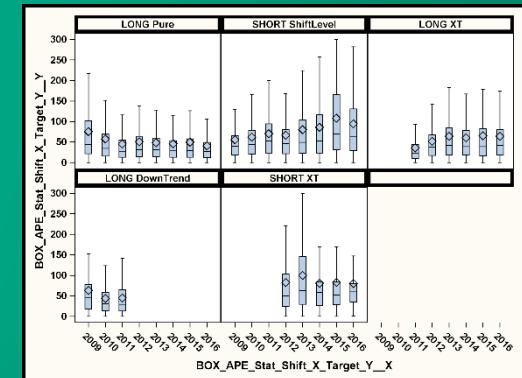
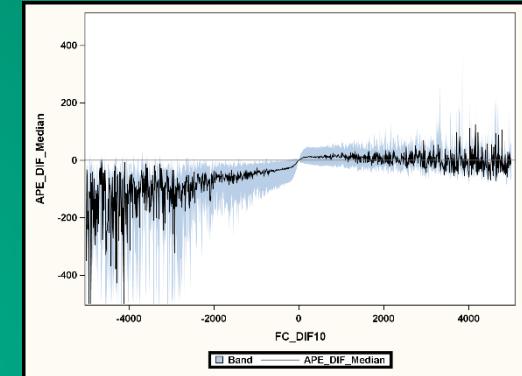


# Data Science in Action: #4

## Explaining Forecast Errors and Deviations

*Do the demand planners really improve forecast accuracy with their manual overwrites?*

Linear Regression  
Quantile Regression  
Descriptive Statistics



# „Was bisher geschah“ (1)

Fachliche Fragestellung im Unternehmen

- Monatliche Nachfrage wird mittels Zeitreihenmodellen vorhergesagt.
- Produkte mit langer und kurzer Absatzhistorie (langjährige Artikel sowie Fashion)
- Analyse des Vorhersage-Fehlers
  - Welcher Vorhersage-Fehler wird von 50%, 75% meiner Vorhersagen nicht überschritten?
  - Welchen Faktoren führen zu geringen Vorhersagefehlern?
  - Gibt es zeitliche Trends und saisonale Muster im Vorhersagefehler?
- Faktoren
  - Stammdaten: PRODUCT\_AGE, PRICE\_INDEX, LAUNCH\_MONTH, PRODUCT\_GROUP
  - Stat. Forecasting: MODEL, LEAD\_TIME, TARGET\_YEAR, TARGET\_CALENDAR\_MONTH

# „Was bisher geschah“ (2)

## Datenaufbereitung

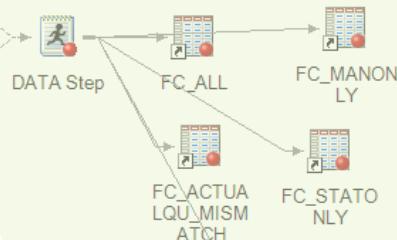
### Statistischer Forecast



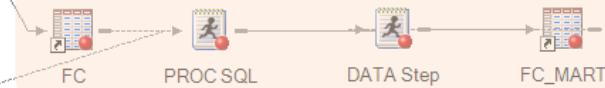
### Manueller Forecast



### Datenqualitätsprüfung



### Berechnung abgeleiteter Variablen



### Artikel-Stammdaten



$APE\_STAT = \frac{abs(statfc - actual)}{actual} * 100$   
→ Absolute Percentage Error

# 4 different user roles analyse a business question

One Integrated Solution for Different User Types



Gerhard Svolba  
SAS Visual Analytics



Gernot Engel  
SAS Studio Tasks



Franz Helmreich  
SAS Studio Program



Matthias Svolba  
Python  
(Jupyter Notebook)



IT and Application Mngt.



# Opening the SAS Analytic Platform via Different Interfaces



Start the SAS Cloud Analytic Server

```
jupyter cif_10_tech_exchng_demo Last Checkpoint: 30 minutes ago (unsaved changes)
File Edit View Insert Kernel Help
s = sw.CAS('cas01.unx.sas', 11775)
s.sessionprop.setsessopt(caslib='CASUSER',timeout=3.1536E7)
s.loadactionset('deepLearn')

Define the Network Architecture

Input Layer 32 Feature Maps Sub-sampling 32 Feature Maps Sub-sampling 64 Feature Maps Sub-sampling 64 Hidden Neurons Output Layer 10 Classes
24x24x3 5x5 12x12x32 5x5 6x6x32 5x5 3x3x64 10 Classes


In [ ]:
s.createModel(model=dict(name='convNet', replace=True), type='CNN')
s.addLayer(model='convNet', name='data', type='input',
           inputOpts=dict(channels=3, width=24, height=24, scale=1))
s.addLayer(model='convNet', name='conv1', type='convolution',
           convOpts=dict(filters=32, width=5, height=5, stride=1, init='msra2'), srcLayer=None)
s.addLayer(model='convNet', name='pool1', type='pooling',
           poolingOpts=dict(width=2, height=2, stride=2, pool='max'), srcLayers=['conv1'])
s.addLayer(model='convNet', name='conv2', type='convolution',
```

Define the Network Layers

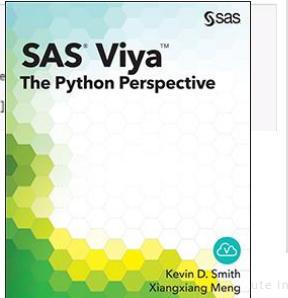
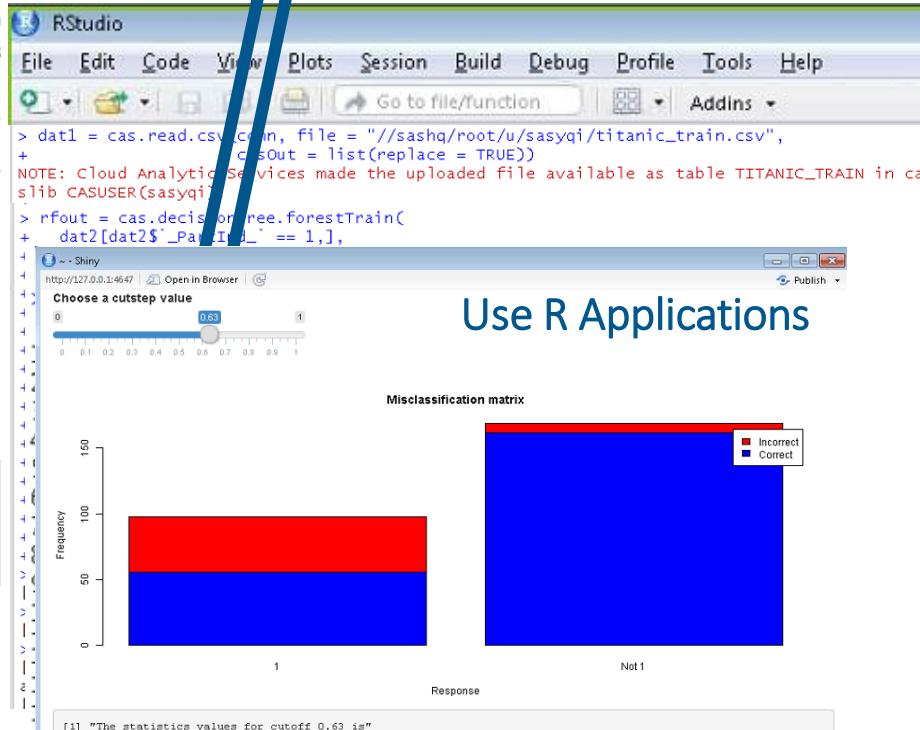


#SASF17



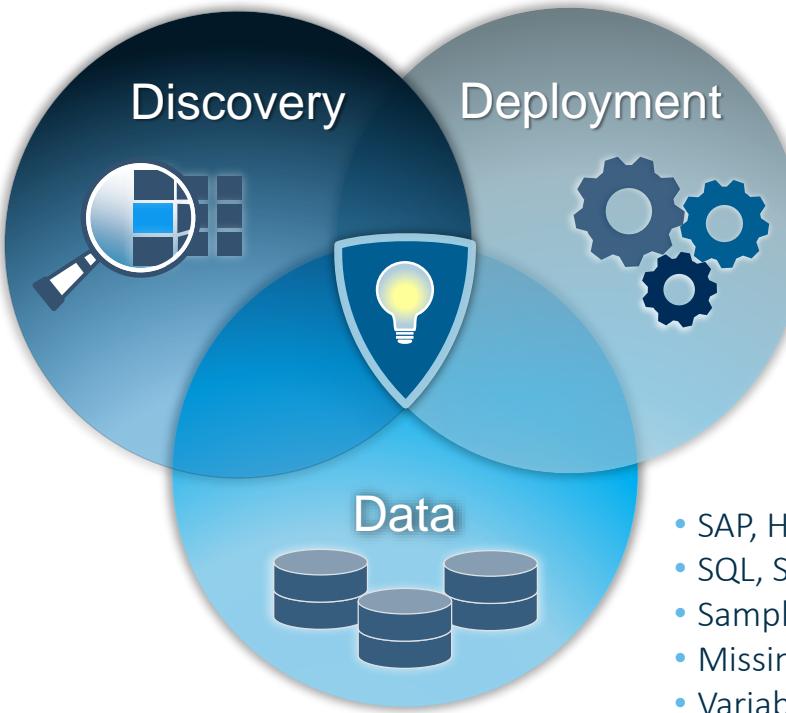
Load the DeepLearning Action Set

Use the CAS Random Forest  
Display the Results in R-Studio



# Data Mining und Machine Learning mit der SAS Analytic Plattform

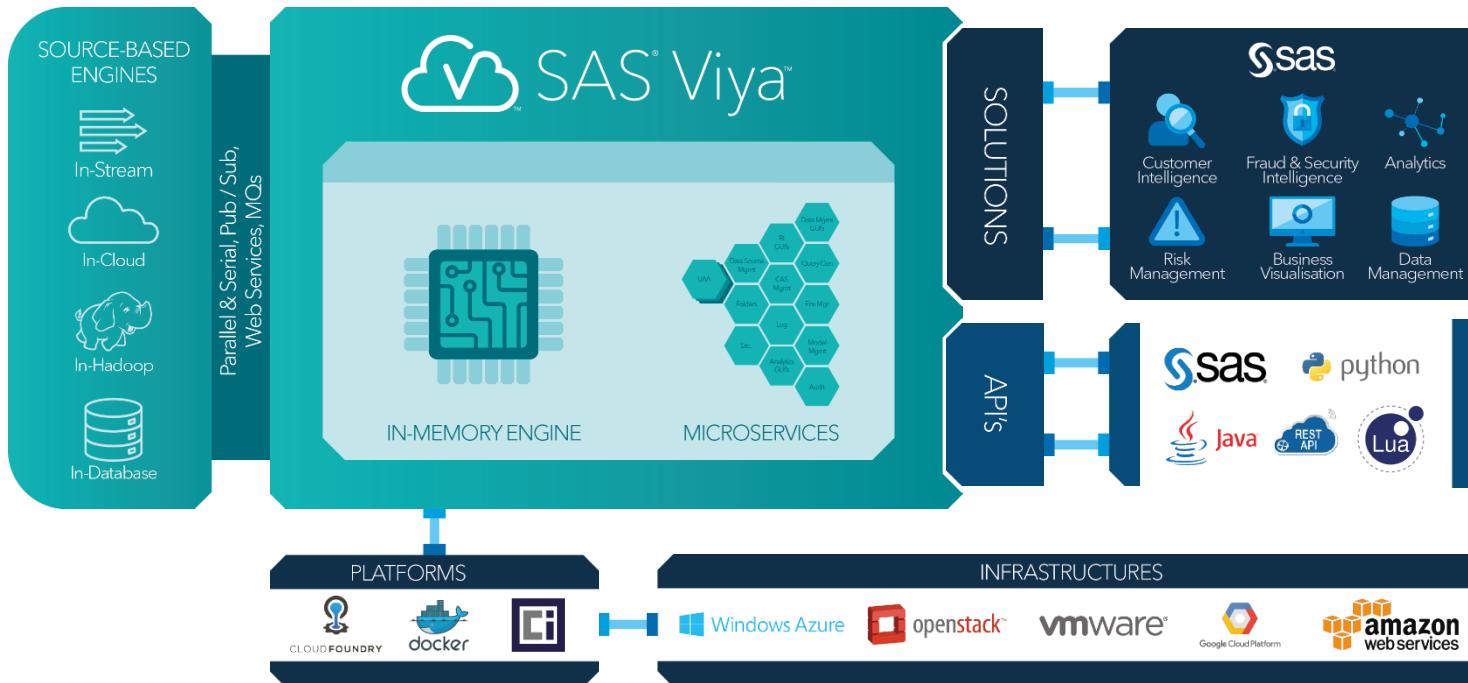
- Logistic Regression
- Linear Regression
- Generalized Linear Models
- Nonlinear Regression
- Ordinary Least Squares Regression
- Decision Trees
- Partial Least Squares Regression
- Quantile Regression
- K-means and K-modes Clustering
- Principal Component Analysis
- Random Forest
- Gradient Boosting
- Neural Networks
- Support Vector Machines
- Factorization Machines
- Network Analytics/Community Detection
- Text Mining
- Boolean Rules
- Auto-tuned Hyper-parameters



- Assess Supervised Models
- Modellverwaltung
- Deployment
- Laufende Validierung
- Modell-Retirement
- Retraining

- SAP, Hadoop, Streaming, rel.DB, ...
- SQL, SAS Datastep, Matrix
- Sampling and Partitioning
- Missing Value Imputation
- Variable Binning
- Variable Selection
- Transpose

# Überblick über die SAS Analytic Plattform



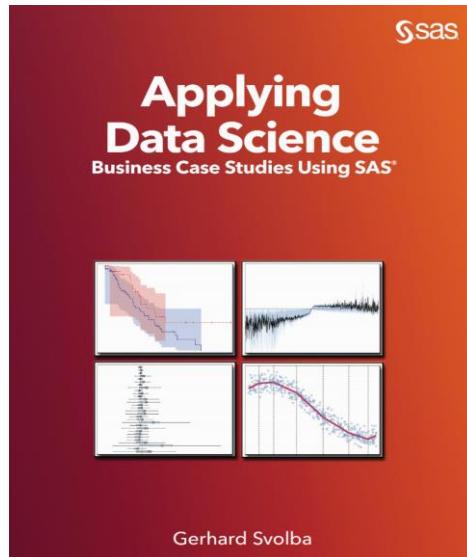
# More Information



Gerhard Svolba – Principal Analytic Solutions Architect

[sastools.by.gerhard@gmx.net](mailto:sastools.by.gerhard@gmx.net)

<https://github.com/gerhard1050/>



- Applying Data Science – Business Case Studies Using SAS, SAS Press 2017
- Eight Case Studies showing how Data Science and Analytics can be applied to provide insight into your data and improve your business decisions
- [http://www.sascommunity.org/wiki/Applying\\_Data\\_Science - Business Case Studies Using SAS](http://www.sascommunity.org/wiki/Applying_Data_Science - Business Case Studies Using SAS)

# SAS Viya Technical Primer

Deep Learning Toolkit

Image Processing Toolkit

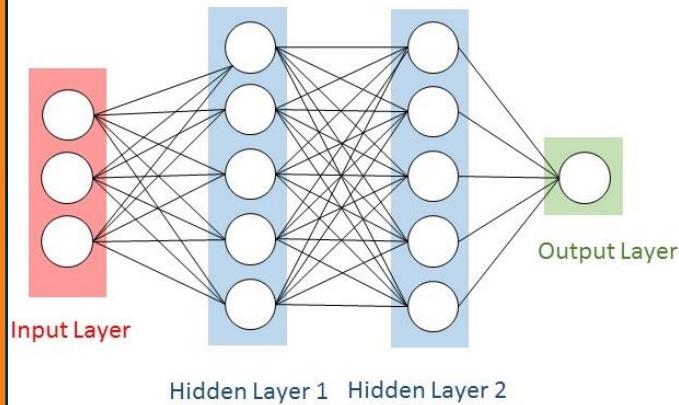
Natural Language Toolkit

# Deep Learning Toolkit

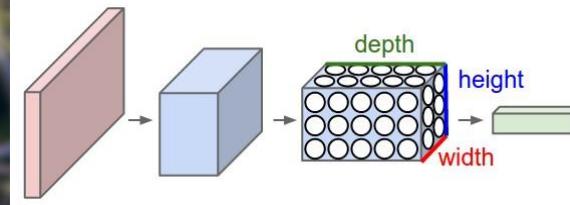
Ships with VDMML license as CAS actions

CAS Action, but also built  
into MS pipeline for Neural  
Nets with more than 5  
layers

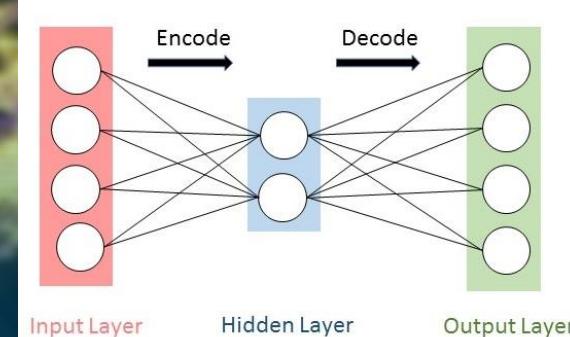
## DEEP FORWARD



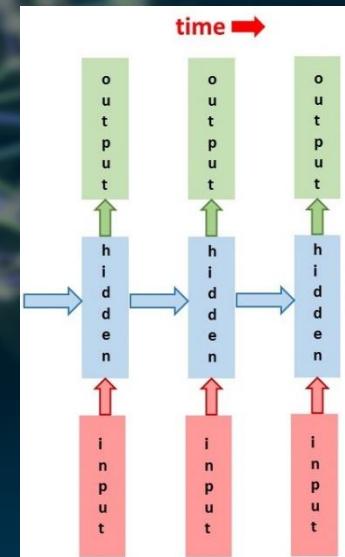
## CONVOLUTIONAL



## AUTOENCODERS



## RECURRENT



# Image Processing Toolkit

Ships with VDMML license as CAS actions



## Image Action Set (Image)

- **LoadImages** – loads images from a path
- **SavelImages** – writes images to a table
- **CompareImages** – compares two sets of images
- **ProcessImages** – performs core functions

## Biomedical Action Set (bioMedImage)

- **buildSurface** – This action can process an image table containing biomedical image data and generate 3D surfaces required to visualize the images.
- **Biomedical extensions to loadImages & savelImages** – Both of these actions are being extended to support loading and saving biomedical images (ex. DICOM images).

# Analyse der NBA 1997 Daten durch 4 verschiedene Benutzer-Rollen

Offenheit der SAS Analytic Plattform für unterschiedliche Zugriffsarten

One Integrated Solution for Different User Types



Business Analyst



New-to-SAS Statistician



SAS Data Scientist



Open Source Data Scientist



IT and Application Mngt.

# Analyse der NBA 1997 Daten durch 4 verschiedene Benutzer-Rollen

SAS Visual Analytics und SAS Visual Statistics für den Business Analyst

One Integrated Solution for Different User Types



Business Analyst



New-to-SAS Statistician



SAS Data Scientist



Open Source Data Scientist

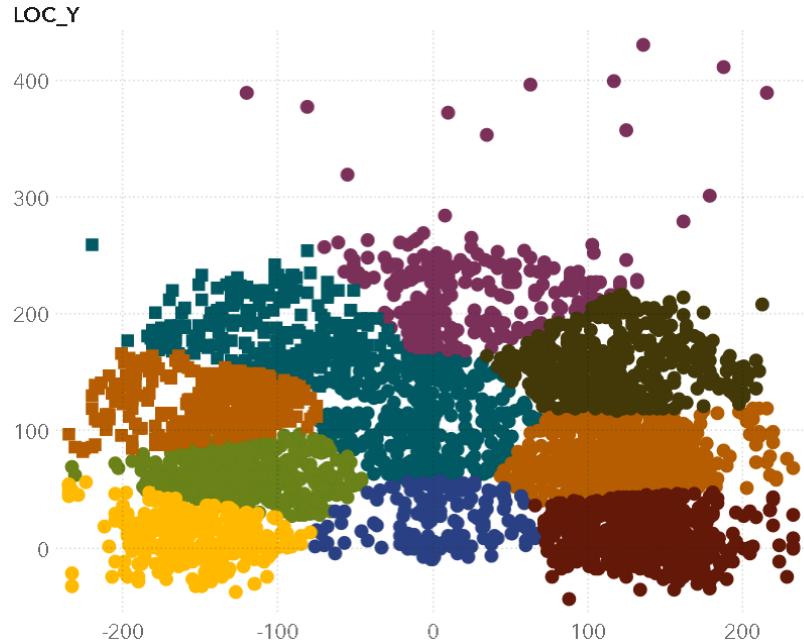
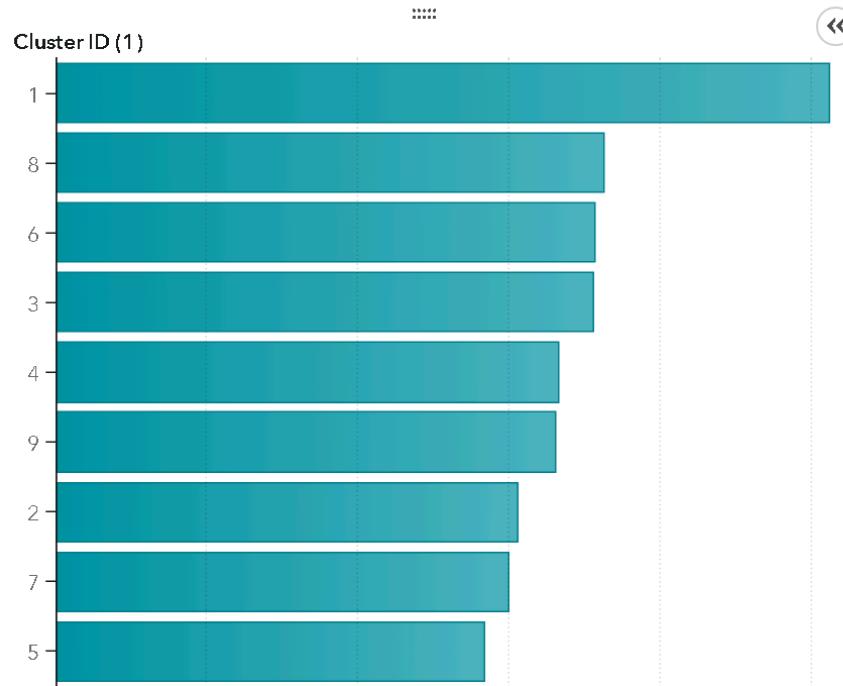


IT and Application Mngt.

# SAS Visual Statistics für den Business Analyst

## Point&Click Zugriff auf Machine Learning Methoden

Drop a data item or control to create a page prompt



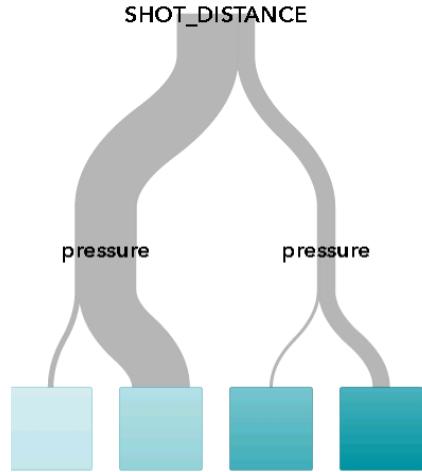
# SAS Visual Statistics für den Business Analyst

## Point&Click Zugriff auf Machine Learning Methoden

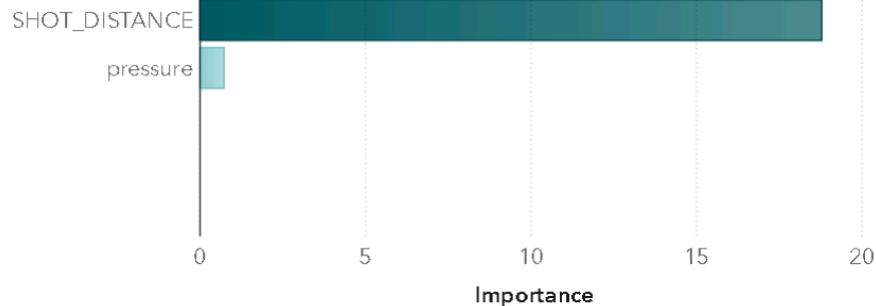
Drop a data item or control to create a page prompt

Decision Tree **SHOT\_MADE\_FLAG** ASE 0.245162 Observations Used 4,809

Tree



Variable Importance



Assessment

**SHOT\_MADE\_FLAG**



# Analyse der NBA 1997 Daten durch 4 verschiedene Benutzer-Rollen

SAS-Python Integration für den Open Source Data Scientist

One Integrated Solution for Different User Types



Business Analyst



New-to-SAS Statistician



SAS Data Scientist



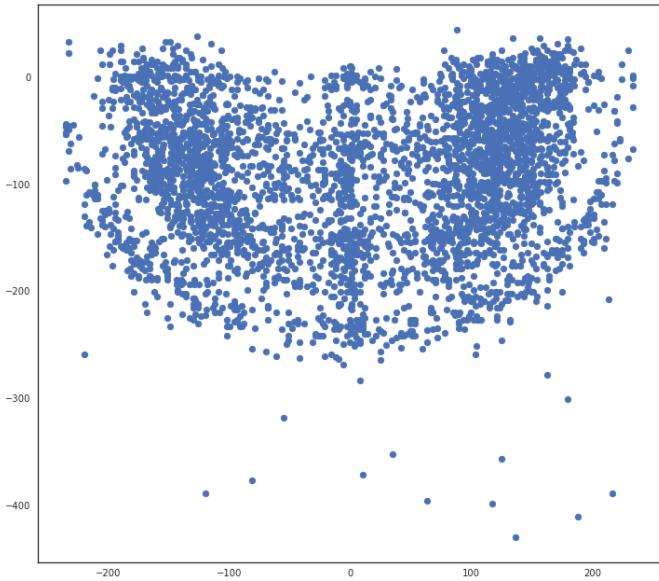
Open Source Data Scientist



IT and Application Mngt.

# How good was Jordan under pressure?

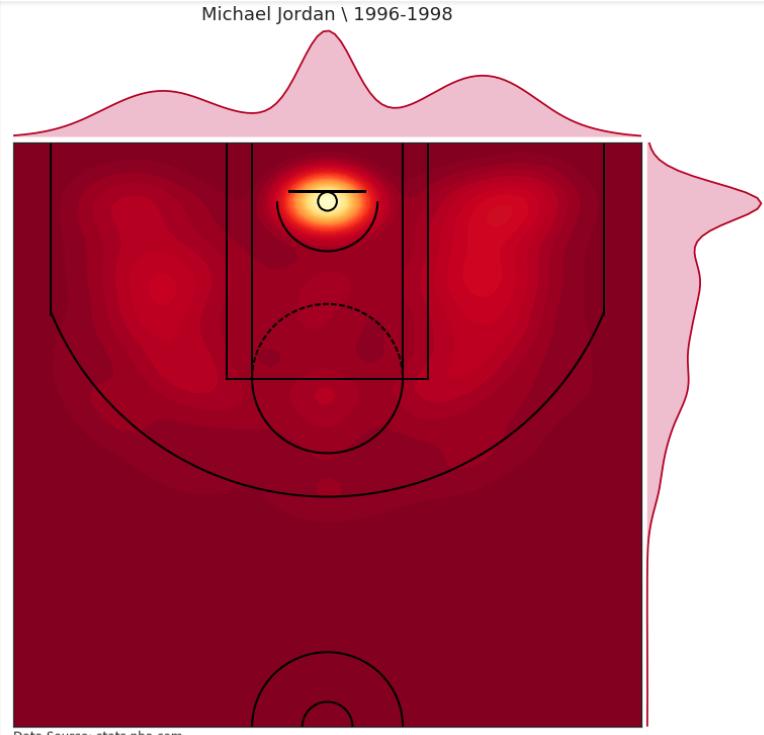
## Some Graphs benefiting from the open source community



<http://savvastjortjoglou.com/nba-shot-sharts.html>



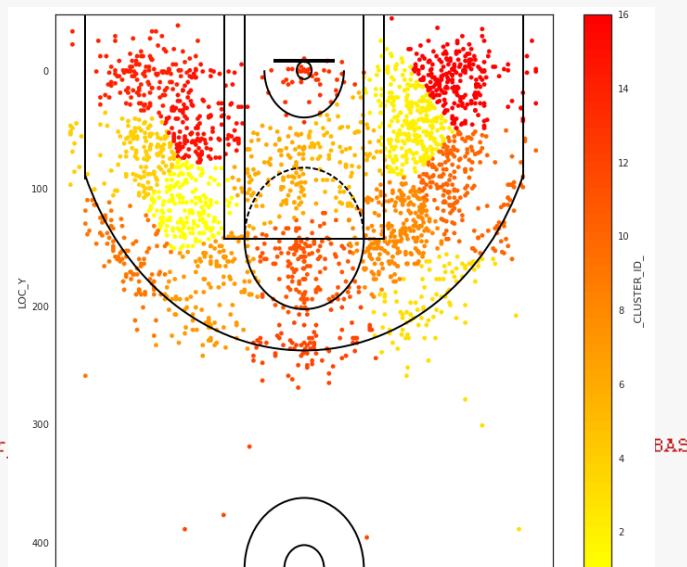
SASF17 / SASForum



# How good was Jordan under pressure?

## Clustering the Court into Shot Zones using CAS action set

```
clust=sess.clustering.kClus(  
    table={  
        "name":"jordan_mining"  
    },  
    inputs={"LOC_X","LOC_Y_MINUS"},  
    nClusters=30,  
    maxIters=10,  
    distanceNom="RELATIVEFREQ",  
    estimateNClusters={  
        "method":"ABC",  
        "B":10,  
        "minClusters":15,  
        "criterion":"ALL",  
        "align":"PCA"  
    },  
    kPrototypeParams={  
        "method":"USERGAMMA",  
        "value":10  
    },  
    output={"CasOut":{"name":"kClusOutputScore", "replace":True},  
            "copyVars":{"LOC_X","LOC_Y","LOC_Y_MINUS","Made","Lead_Player"},  
            ,  
            display={"names":{"Modelinfo", "ClusterSumIntNom"}},  
    }  
}
```



# How good was Jordan under pressure?

## Using CAS Regression to investigate performance under pressure

Logistic

```
In [69]: lr = sess.regression.logistic(  
    table={"name":"kClusOutputScore"},  
    classVars=[{"vars":{_CLUSTER_ID_,"Lead_Player_Before", "Homegame", "ACTION_TYPE", "end_of_game","Overtime","close_game", "}},  
    model={  
        "depVars":[{"name":"Made", "options":{"event":"1"} }],  
        "effects":[{"vars":("SHOT_DISTANCE", _CLUSTER_ID_, "ACTION_TYPE", "pressure 1", "pressure 0", "Overtime Regular", "Overtime Overtime", "Tip Shot", "Slam Dunk Shot", "Running Jump Shot", "Layup Shot", "Jump Shot", "Hook Shot", "Dunk Shot", "Driving Layup Shot", "Driving Dunk Shot") }]  
    },  
    outputTables={"names":"parameterestimates"}  
)  
sess.dataStep.runCode(  
    code="""data round; set parameterestimates(keep=Parameter DF  
do i = 1 to dim(_nums);  
    _nums(i) = round(_nums(i),.001);  
end;  
drop i;  
run;"""  
)  
sess.fetch(table="round")  
Out[69]: § Fetch
```

Selected Rows from Table ROUND

|    | Parameter                      | DF  | Estimate | StdErr  | ChiSq | ProbChiSq |
|----|--------------------------------|-----|----------|---------|-------|-----------|
| 0  | Intercept                      | 1.0 | 14.322   | 94.876  | 0.023 | 0.880     |
| 1  | pressure 1                     | 1.0 | -0.385   | 0.168   | 4.193 | 0.041     |
| 2  | pressure 0                     | 0.0 | 0.000    | NaN     | NaN   | NaN       |
| 3  | Overtime Regular               | 1.0 | 0.382    | 0.370   | 1.069 | 0.301     |
| 4  | Overtime Overtime              | 0.0 | 0.000    | NaN     | NaN   | NaN       |
| 5  | SHOT_DISTANCE                  | 1.0 | -0.035   | 0.014   | 6.500 | 0.011     |
| 6  | ACTION_TYPE Tip Shot           | 1.0 | -13.135  | 94.875  | 0.019 | 0.890     |
| 7  | ACTION_TYPE Slam Dunk Shot     | 1.0 | 0.027    | 128.564 | 0.000 | 1.000     |
| 8  | ACTION_TYPE Running Jump Shot  | 1.0 | -12.634  | 94.876  | 0.018 | 0.894     |
| 9  | ACTION_TYPE Layup Shot         | 1.0 | -13.064  | 94.875  | 0.019 | 0.890     |
| 10 | ACTION_TYPE Jump Shot          | 1.0 | -14.106  | 94.875  | 0.022 | 0.882     |
| 11 | ACTION_TYPE Hook Shot          | 1.0 | -12.552  | 94.882  | 0.017 | 0.895     |
| 12 | ACTION_TYPE Dunk Shot          | 1.0 | -10.540  | 94.875  | 0.012 | 0.912     |
| 13 | ACTION_TYPE Driving Layup Shot | 1.0 | -11.510  | 94.875  | 0.015 | 0.903     |
| 14 | ACTION_TYPE Driving Dunk Shot  | 0.0 | 0.000    | NaN     | NaN   | NaN       |

# Analyse der NBA 1997 Daten durch 4 verschiedene Benutzer-Rollen

SAS Procedures für den SAS Data Scientist

One Integrated Solution for Different User Types



Business Analyst



New-to-SAS Statistician



SAS Data Scientist

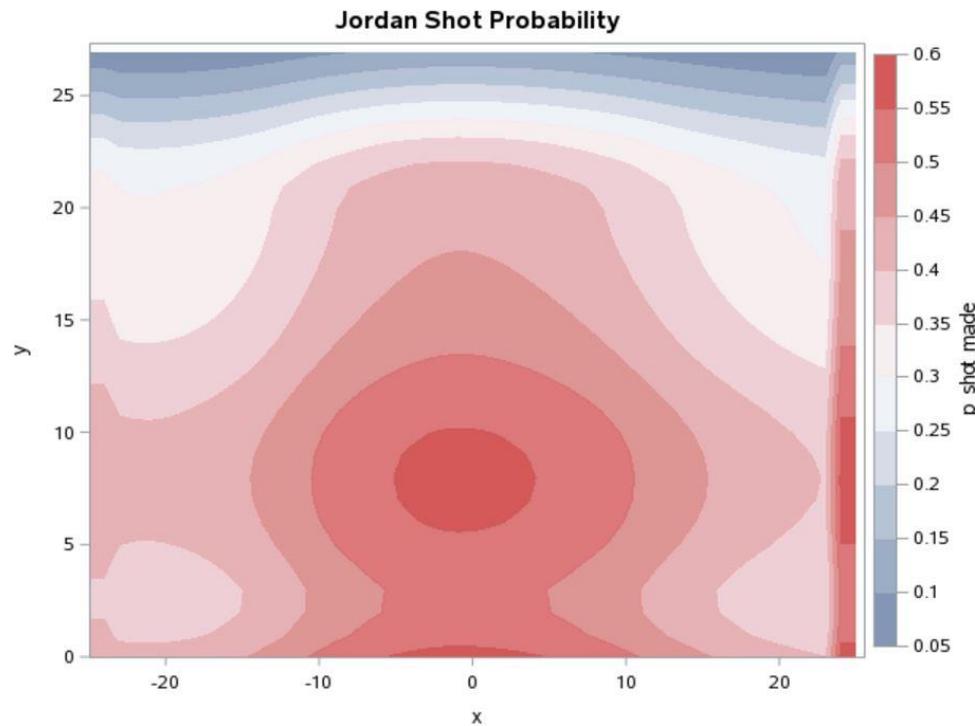


Open Source Data Scientist



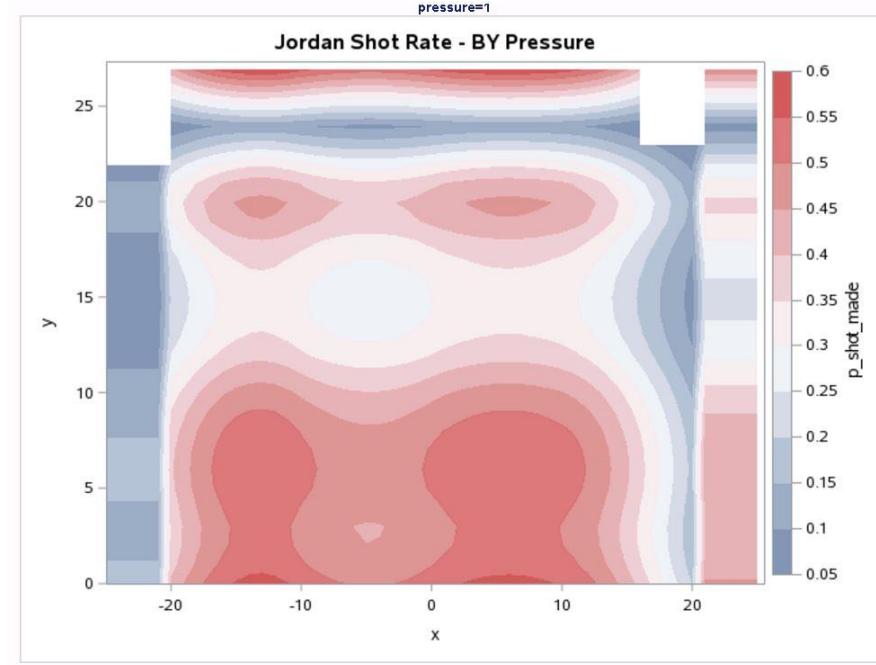
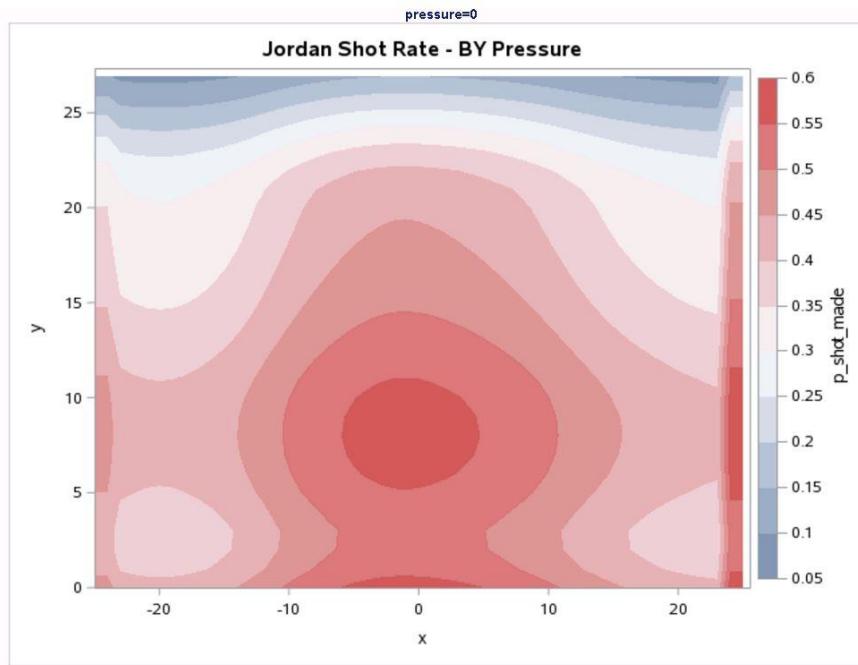
IT and Application Mngt.

# Vorhersage der Treffer-Wahrscheinlichkeit von Jordan im [-25,25]x[0,27] Grid



```
proc logselect data=sfdcas.Jordan;
where Shot_Distance <= 30;
effect spl = spline(X Y / degree=2);
model Shot_Made(event='1') = spl ;
output out=sfdcas.Jordan_pred
pred=p copyvars=(x y shot_distance
shot_made);
Code file='/opt/sasinside/
DemoData/SFD/JordanPred_0.sas';
run;
```

# Vorhersage der Treffer-Wahrscheinlichkeit von Jordan getrennt nach Pressure ja/nein



# Analyse der NBA 1997 Daten durch 4 verschiedene Benutzer-Rollen

Data Mining/Machine Learnings Tasks im SAS Studio für den „New-to-SAS“ Statistician

One Integrated Solution for Different User Types



Business Analyst



New-to-SAS Statistician



SAS Data Scientist



Open Source Data Scientist



IT and Application Mngt.

## Vordefinierte Tasks

The screenshot shows the SAS Studio interface with a blue header bar. In the top left, there's a blue callout box labeled "Vordefinierte Tasks". The main area has a sidebar on the left containing categories such as "Utilities", "Tasks", "Snippets", "Bibliotheken", and "Dateiverknüpfungen". The main panel shows a configuration dialog for a "Forest" task. It includes sections for "METHODEN" (Methods) with dropdowns for "Anzahl Bäume" (100), "Maximale Baumtiefen" (20), and "Bootstrap-Stichprobenanteil" (0,6); "TEILUNGSKRITERIEN" (Splitting Criteria) with dropdowns for "Mindestanzahl Beobachtungen je Blatt" (5) and "Maximale Anzahl Zweige je Knoten" (2); and "Variablenbedeutungsmethoden" (Variable Importance) set to "Gini (Standard)". A large blue callout box labeled "Optionen" is overlaid on the bottom left of the configuration panel.

# SAS Studio

## Code Generierung mit Tasks

The screenshot shows the SAS Studio interface with a blue header bar. In the top right, there's a blue callout box labeled "Code Generierung". The main area shows a code editor with generated SAS code for a forest model. The code includes PROC FOREST and PROC SGPLOT statements. A small blue callout box labeled "Eigene Tasks/Forest\_Jordan.ctk" is at the bottom left of the code editor. The status bar at the bottom right shows "Zeilene 1, Spalte 1" and "Meldungen: 17 Benutzer: sasdemo".

```
/*
*
* Taskcode generiert von SAS Studio 4.2
*
* Generiert am '26.06.17 16:28'
* Generiert von 'sasdemo'
* Generiert auf Server 'SASSERVER.DEMO.SAS.COM'
* Generiert auf SAS Platform 'Linux LIN X64 2.6.32-2.6.32-0-MP03082017'
* Generiert mit SAS-Version 'V.03.02MP03082017'
* Generiert mit Browser 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) sas.com/SASStudio/main?locale=de_DE&zone=GMT%252B02%253B00'
* Generiert mit Webclient 'http://racesx06045.e16.x86_64'
*/
ods noproctitle;
proc forest data=MYCASLIB.JORDAN_MINING;
partition fraction(validate=0.3 seed=1234);
target Made / level=nominal;
input SHOT_DISTANCE Margin_Player_Before / level=interval;
input season_type ACTION_TYPE SHOT_TYPE SHOT_ZONE_BASIC SHOT_ZONE_AREA
      SHOT_ZONE_RANGE Homegame Lead_Player_Before Overtime Last_quarter close_game
      pressure / level=nominal;
ods output FitStatistics=Work._Forest_FitStats_
      VariableImportance=Work._Forest_VarImp_;
score out=mycaslib.scored copyvars(_all_);
run;
proc sgplot data=Work._Forest_FitStats;
title3 'Fehlklassifikationen nach Anzahl der Bäume';
title4 'Out-of-Bag vs. Training vs. Validierung';
series x=Trees y=MiscTrain;
series x=Trees y=MiscOob /lineattrs=(pattern=shorthash thickness=2);
series x=Trees y=MiscValid /lineattrs=(pattern=dot thickness=2);
yaxis label='Fehlklassifikationsanteil';
label Trees='Anzahl Bäume';
label MiscTrain='Training';
label MiscOob='Out-of-Bag';
label MiscValid='Valid';
run;
```

## Vordefinierte Tasks

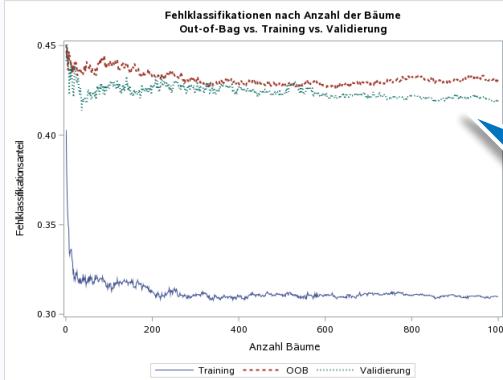
The screenshot shows the SAS Studio interface with a sidebar titled 'Vordefinierte Tasks'. The main area displays a configuration panel for a 'Decision Tree' task. The panel includes sections for 'METHODEN' (Methods) and 'PLOTS' (Plots). In the 'METHODEN' section, options like 'Anzahl Bäume' (Number of Trees), 'Maximale Baumtiefen' (Maximal Tree Depth), and 'Bootstrap-Stichprobenanteil' (Bootstrap Sample Fraction) are set. The 'PLOTS' section contains a checkbox for 'Fehlklassifikationen nach Anzahl der Bäume' (Classification errors by number of trees). A large blue callout box labeled 'Optionen' points to this section.

# SAS Studio

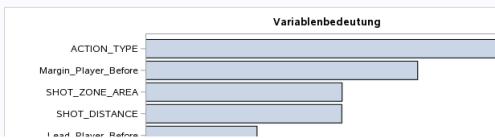
## Code Generierung mit Tasks

CODE LOG ERGEBNISSE AUSGABEDATEN

Inhalt



Ergebnisse



Meldungen: 18 Benutzer: sasdemo

## Vordefinierte Tasks

The screenshot shows the SAS Studio interface with a sidebar containing a tree view of available tasks. The 'Tasks' section is expanded, showing sub-categories like 'Vorbereiten und Untersuchung', 'Zusammenfassung', 'Datentransformation', etc. A specific section for 'Auto-Tune' is highlighted with a blue callout box.

# SAS Studio

## Code Generierung mit Tasks

Aktivierung der  
Autotuning  
Funktionalität

The screenshot shows the 'CODE' tab of the SAS Studio interface. A large blue callout box covers the top half of the screen, pointing to the 'Auto-Tune' configuration section. The configuration panel contains several input fields for tuning parameters like 'Startwert', 'Untergrenze', and 'Obergrenze'.

|                                   |         |
|-----------------------------------|---------|
| Anzahl evaluierte Konfigurationen | 45      |
| Gesamt-Tuning-Dauer (in Sekunden) | 19.5001 |
| Paralleles Tuning Speedup         | 2.4544  |

| Task                    | Seconds | Percent |
|-------------------------|---------|---------|
| Modell Training         | 37.08   | 77.12   |
| Modell Scoring          | 6.88    | 14.31   |
| Summe Zielevaluierungen | 43.98   | 91.40   |
| Tuner                   | 4.11    | 8.54    |
| Gesamt-CPU-Zeit         | 49.08   | 100.00  |

| Best Configuration                 |            |
|------------------------------------|------------|
| Evaluierung                        | 12         |
| Number of Trees                    | 80         |
| Number of Variables to Try         | 5          |
| Bootstrap                          | 0.79525748 |
| Maximum Tree Levels                | 14         |
| Fehlklassifizierungsfehler Prozent | 40.08      |

| Tuner Results<br>Default and Best Configurations |                     |                 |                            |           |                          |            |
|--|---------------------|-----------------|----------------------------|-----------|--------------------------|------------|
| Evaluation                                       | Maximum Tree Levels | Number of Trees | Number of Variables to Try | Bootstrap | Missclassification Error | Percentage |
| 0  | 21                  | 100             | 14                         | 0.600000  | 41.30                    |            |
| 12   | 14                  | 80              | 5                          | 0.795207  | 40.08                    |            |
| 45   | 14                  | 80              | 7                          | 0.725718  | 40.54                    |            |
| 39   | 12                  | 72              | 7                          | 0.705657  | 40.61                    |            |
| 30   | 12                  | 70              | 7                          | 0.277778  | 40.68                    |            |
| 3  | 2                   | 20              | 14                         | 0.277778  | 40.68                    |            |
| 17   | 5                   | 34              | 12                         | 0.403667  | 40.68                    |            |
| 22   | 4                   | 22              | 13                         | 0.335816  | 40.68                    |            |
| 24   | 5                   | 34              | 12                         | 0.364741  | 40.68                    |            |
| 32   | 2                   | 20              | 14                         | 0.219658  | 40.68                    |            |
| 33   | 8                   | 33              | 12                         | 0.415225  | 40.68                    |            |

Tuner History  
All Evaluated Configurations

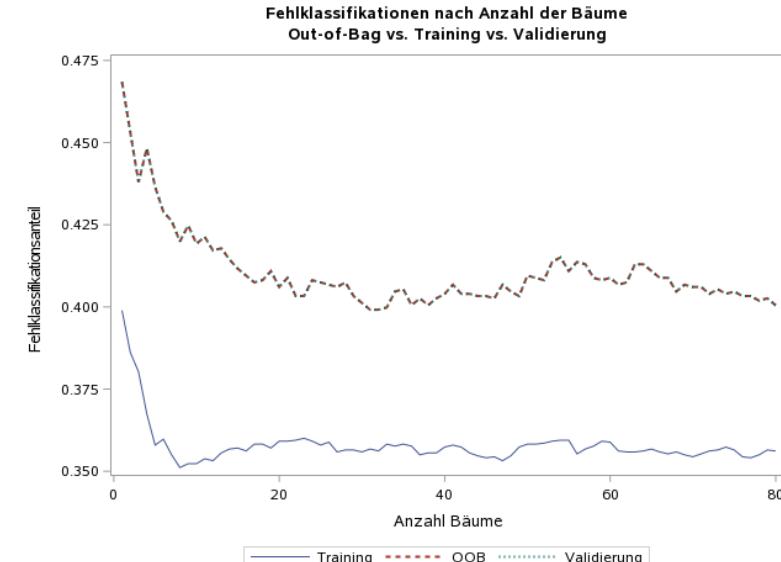
Ergebnisse

Meldungen: 19 Benutzer: sasdemo

# SAS Studio

## Code Generierung mit Tasks

| Tuner Results<br>Default and Best Configurations |                     |                 |                            |           |                                    |
|--|---------------------|-----------------|----------------------------|-----------|------------------------------------|
| Evaluation                                       | Maximum Tree Levels | Number of Trees | Number of Variables to Try | Bootstrap | Misclassification Error Percentage |
| 0  | 21                  | 100             | 14                         | 0.600000  | 41.30                              |
| 12   | 14                  | 80              | 5                          | 0.795287  | 40.06                              |
| 45   | 14                  | 80              | 5                          | 0.791425  | 40.26                              |
| 39   | 12                  | 72              | 7                          | 0.725718  | 40.54                              |
| 30   | 12                  | 70              | 7                          | 0.705657  | 40.61                              |
| 3  | 2                   | 20              | 14                         | 0.277778  | 40.68                              |
| 17   | 5                   | 34              | 12                         | 0.403667  | 40.68                              |
| 22   | 4                   | 22              | 13                         | 0.335616  | 40.68                              |
| 24   | 5                   | 34              | 12                         | 0.394741  | 40.68                              |
| 32   | 2                   | 20              | 14                         | 0.218658  | 40.68                              |
| 33   | 6                   | 33              | 12                         | 0.415225  | 40.68                              |



# SAS Studio

## Code Generierung mit Tasks

