Statistic Packages Overview 2025

- Regression & Correlation box script demos
- 7 Statistic Methods & 5 Diagramtypes
- Stats Package TeeChart & mrMath, Delphi AI
- Sklearn & Statsmodels (P4D)
- AGSI, JSON, EdgeView2 and Weather Reports
- Alternate Packages, Top & Flops Statistic AI
- This session shows various ways of using statistic methods and plots in your apps.

https://blogs.embarcadero.com/why-a-data-scientist-chooses-delphi-for-powerful-real-world-visualizations/


SQL as Statistic Query Language 😅

- We’ve all seen it — those dreamy job descriptions filled with buzzwords like: 📊
- SQL. Python. Spark. Azure. Airflow. Machine Learning and Data Science.
- The post makes you feel like you’re applying to work at NASA (we all love numbers).
- You land the job (finally).
- And then…
- You spend 90% of your time writing SQL queries and cleaning Excel sheets. 😅

A Base Class

```
Proc SIRegister_TStatisticClass(CL: TPSPascalCompiler);
begin
  //with RegClassS(CL,'TOBJECT', 'TStatisticClass') do
 with CL.AddClassN(CL.FindClass('TOBJECT'),'TStatisticClass') do begin
   Proc Assign( const S : TStatisticClass);
   Func Duplicate : TStatisticClass');
   Proc Clear');
   Func IsEqual( const S : TStatisticClASS):Bool');
   Proc Add( const V : MFloat);
   Proc Add1( const V : array of MFloat);
   Proc Add2( const V : TStatisticClass);
   Proc AddNegated( const V : TStatisticClass);
   Proc Negate');
   Count', 'Integer', iptr);
   Min', 'MFloat', iptr);
   Max', 'MFloat', iptr);
   Sum', 'MFloat', iptr);
   SumOfSquares', 'MFloat', iptr);
   SumOfCubes', 'MFloat', iptr);
   SumOfQuads', 'MFloat', iptr);
   Func Range : MFloat');
   Func Mean : MFloat');
   Func PopulationVariance : MFloat');
   Func PopulationStdDev : MFloat');
   Func Variance : MFloat');
   Func StdDev : MFloat');
   Func M1 : MFloat');
   Func M2 : MFloat');
   Func M3 : MFloat');
   Func M4 : MFloat');
   Func Skew : MFloat');
   Func Kurtosis : MFloat');
   Func GetAsString :Str');
  end;
end;
```

EKON

A Base Class

System.Math.MomentSkewKurtosis is declared as

```
1  procedure MomentSkewKurtosis(const Data: array of Double; var M1,M2,M3,M4,
2
3  Ex.: MomentSkewKurtosis ([1,2,3,4,5,5,5,6,7,8,9] ,M1,M2,M3,M4,Skew,Kurtosi
4  writ(format('%2f %2f %2f %2f %2f %2f',[M1, M2, M3, M4, Skew, Kurtosis]))
5  5.00 5.45 0.00 64.36 0.00 2.16
```

MomentSkewKurtosis: Calculates the core factors of statistical analysis: the first four moments plus the coefficients of skewness and kurtosis.

- M1 is the Mean.
- M2 is the Variance.
- Skew reflects symmetry of distribution: **M3** / (M2**(3/2))
- Kurtosis reflects flatness of distribution: **M4** / Sqr(M2)
- Look at `M3` and `M4` (the 3rd and 4th order moment) as helper variables in order to calculate skew and kurtosis (extracted from source):

More Base Units

https://github.com/fundamentalslib/fundamentals5/blob/master/Source/Maths/flcStatistics.pas

https://github.com/project-jedi/jcl/blob/master/jcl/source/common/JclStatistics.pas

https://github.com/TurboPack/SysTools/tree/master/source

https://github.com/TurboPack/SysTools/blob/master/source/StStat.pas

https://github.com/fundamentalslib/fundamentals5/blob/master/Source/Maths/flcStatistics.pas

- Random number generators
- Cryptographic (Symmetric: AES, DES, RC2, RC4; Asymmetric: RSA, Diffie-Hellman, Elliptic Curve)
- Data structures (array, dictionary and map classes)
- Mathematics (Rational number, complex number, vector, matrix, statistics)

A Brain Class Test Unit

https://github.com/maxkleiner/maXbox5/blob/main/EKON29/1442_Statistics_unittest.pas

7 Statistical Methods Criteria

- 1. Bayesian Inference (p(A))
- 2. Hypothesis Testing (t-test)
- 3. Factor Analysis (Loading Analysis)
- 4. Cluster Analysis (K-means)
- 5. Logistic Regression Classifier
- 6. Monte Carlo Simulation (Rnd)
- 7. Time Series Analysis (ARIMA)

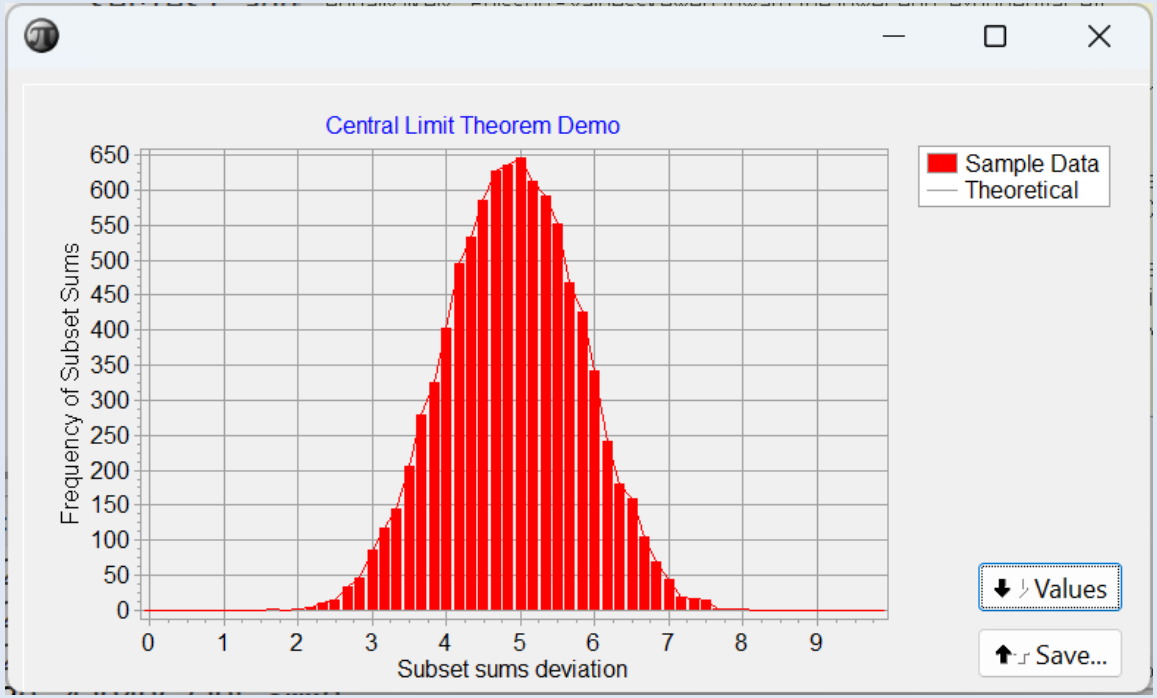
Data science is a multidisciplinary field that combines math, statistics, computer science, machine learning, and domain expertise to extract insights from data. While data science algorithms often put the spotlight, a solid foundation in statistical methods can be just as pivotal.

https://softwareschule.code.blog/2025/02/20/7-data-science-statistical-methods/

Demo:1382_data_science_stuff140_12_py_uc.txt

Common Theorie

- Trade-off between probability and precision
- Therefore, probability (confidence level) and accuracy (precision) are not negatively correlated in the classical sense, but rather they have a trade-off relationship: greater certainty (probability) often means lower accuracy due to wider intervals.



5 Important Diagram Types

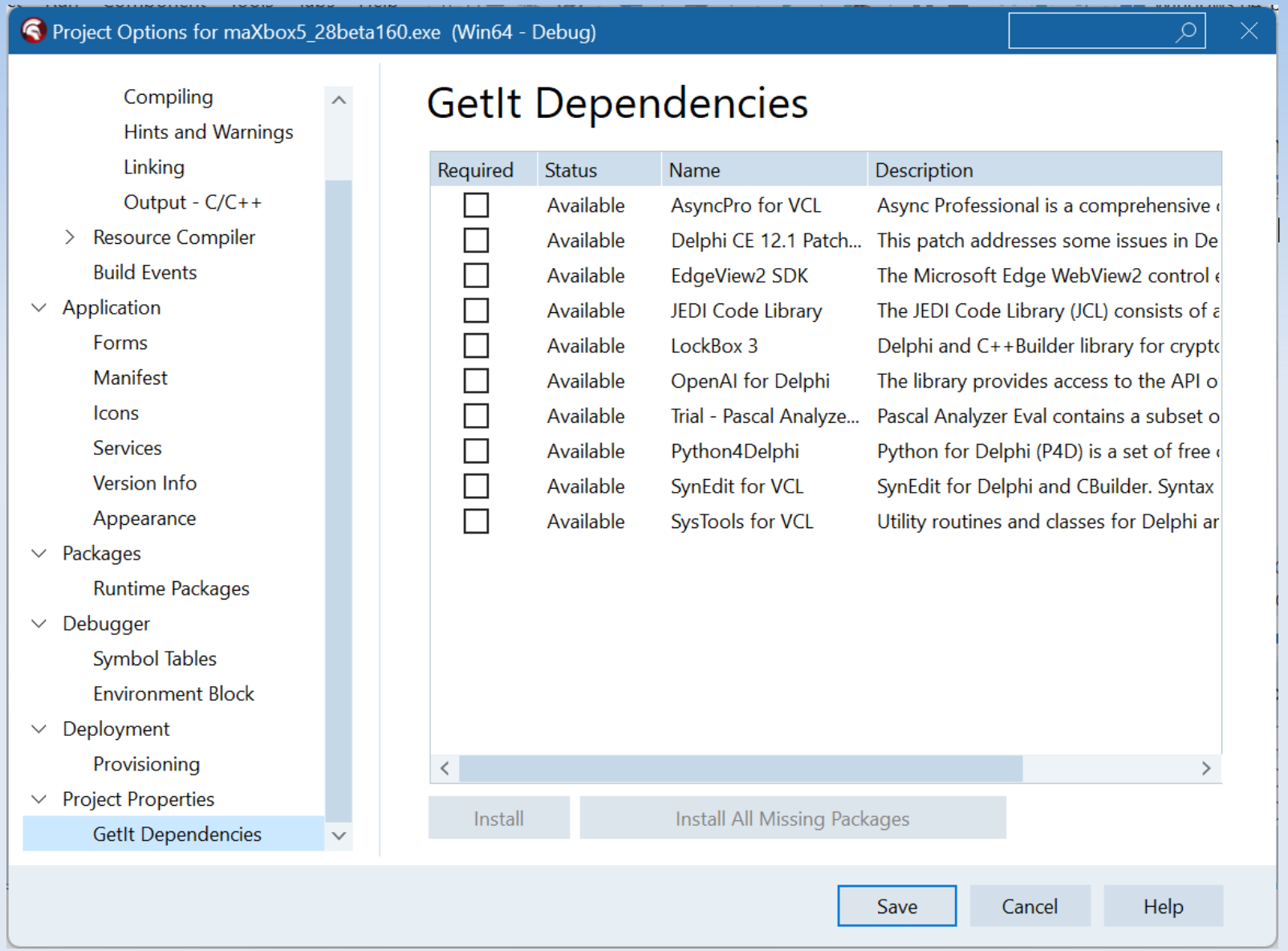
Here, I'll show you how to analyze a runtime script created dataset and extract meaningful insights with 5 diagram types:

1) Bar Chart
2) Scatter Plot
3) Histogram
4) Box Plot
5) Correlation Matrix

- **Bar charts:** Used to show comparisons between different categories.
- **Line charts:** Used to show trends over time or across different categories.
- **Pie charts:** Used to show proportions or percentages of different categories.
- **Histograms:** Used to show the distribution of a single variable.
- **Heatmaps:** Used to show the correlation between different variables.
- **Scatter plots:** Used to show the relationship between two continuous variables.
- **Box plots:** Used to show the distribution of a variable and identify outliers.
- **Correlation Matrix** shows relations to each other variable

https://softwareschule.code.blog/2025/02/20/7-data-science-statistical-methods/

Stats Dependencies



Cross-platform class TChart

- **TeeChart is a powerful charting component developed by Steema Software, widely used in Delphi and C++Builder environments. It provides a comprehensive set of tools for creating various types of charts, including line, bar, area, pie, point, and horizontal bar charts, which can be viewed in both 2D and 3D formats.**

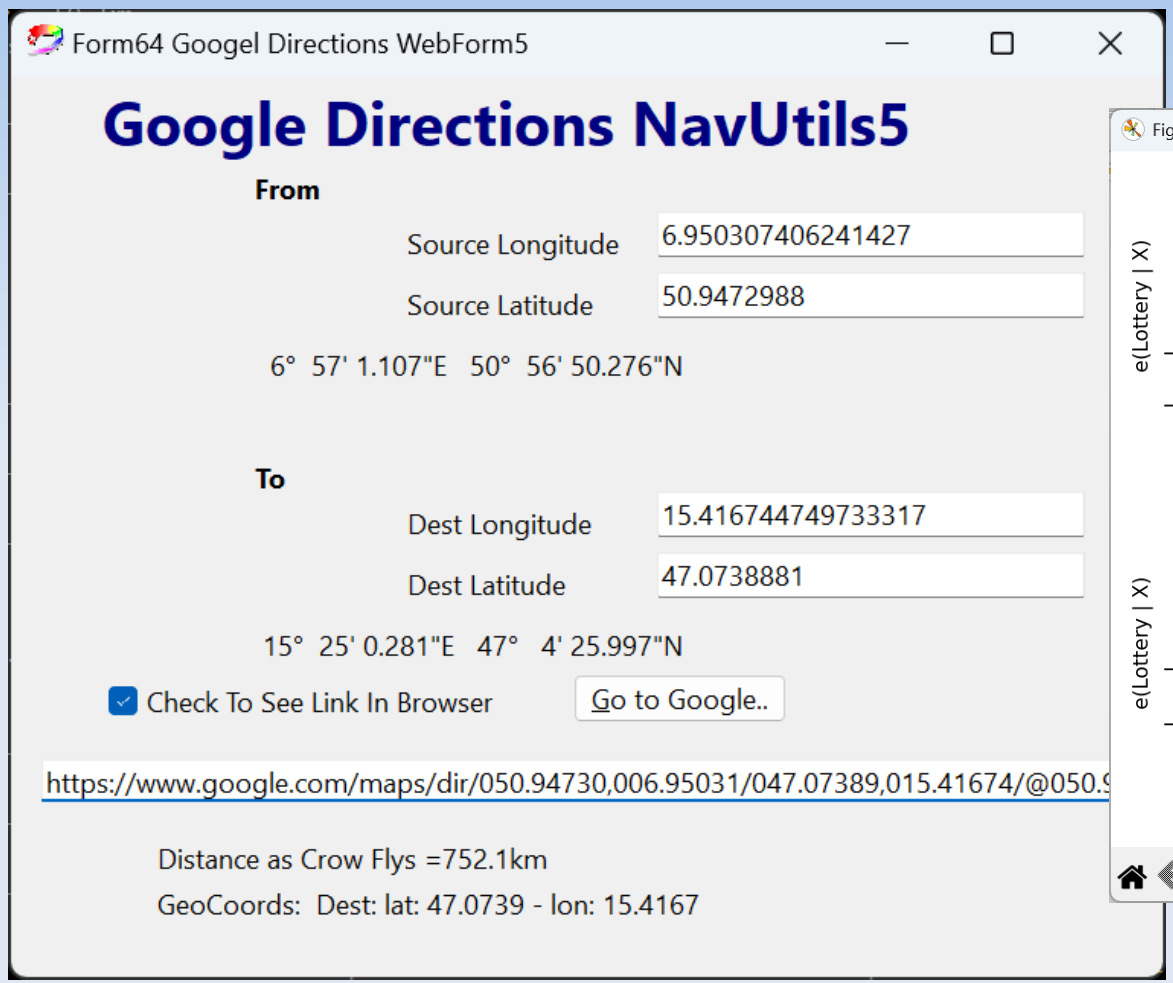
Key Features

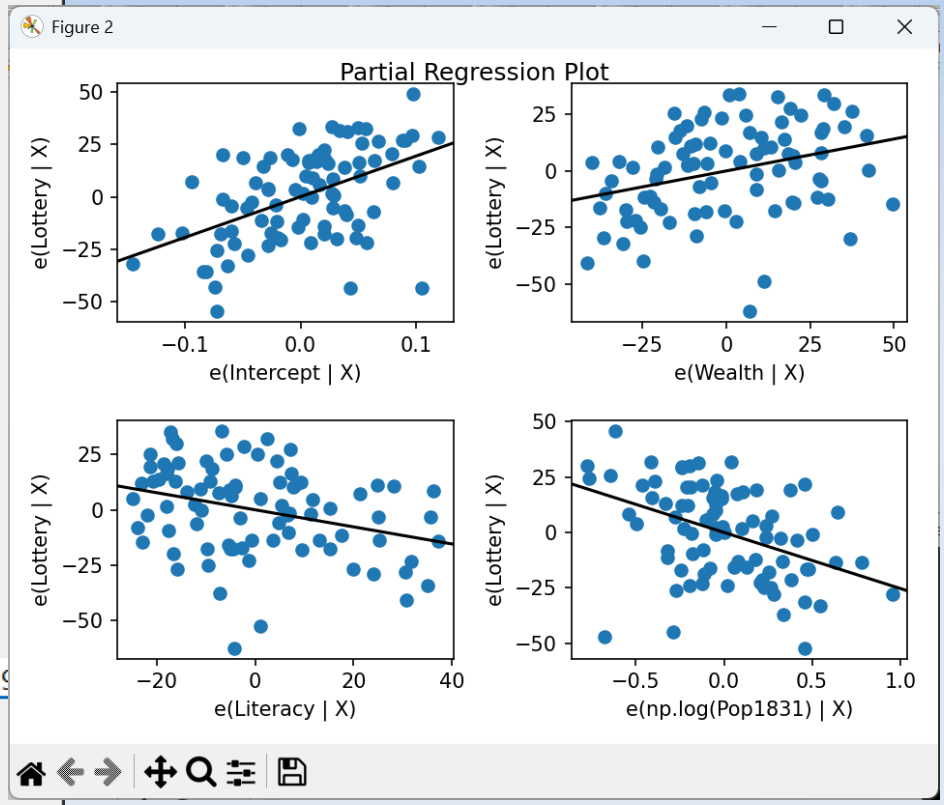
- Versatility: TeeChart supports a wide range of chart types and can be used for business, real-time, financial, and scientific applications. It is included in most Delphi and C++Builder products and supports both VCL and FM.
- Cross-Platform Support: TeeChart can be used to create applications for Windows (x32/x64) and macOS. For Apple iOS support, TeeChart Pro.
- Interactivity: TeeChart allows users to interact with charts by zooming, panning, and exporting to various file types. This makes it suitable for applications that require real-time data visualization and user interaction.

https://blogs.embarcadero.com/why-a-data-scientist-chooses-delphi-for-powerful-real-world-visualizations/

Demo: 820_U_ProbabilityDist2ekon29_1_64.pas
942_gaussen_data_science3_64.txt

GUI or Console?






```
1  latlong:=  TAddressGeoCodeOSM5('Hauptbahnhof, Graz, Austria');
2  writeln('OSM5 res back_: '+latlong.descript);
3  >>> OSM5 res back_: Coords: lat 47.07391  lng 15.41681 Hauptbahnhof, Europ
4
5  writeln('get geoCoords: '+format(' lat: %.4f - lon: %.4f',[latlong.lat,lat
6  OpenWeb('https://www.latlong.net/c/?lat='+flots(latlong.lat)+'&long='+flot
```

Mr Math Package

The mrMath, mrStats, mrMatrix, mrImgUtils package includes:

- Standard Fisher LDA classifier
- Robust (and Fast Robust) version of this classifier
- Incremental (and Robust) Fisher LDA classifier learning.
- Support Vector Machines (least squares and lagrangian learning)
- Naive Bayes
- Simple Decission stumps
- Radial basis function
- C4.5 Decission trees.
- K-means cluster
- Ensemble classifiers: AdaBoost, Gentle Boost, Bagging
- Simple feed forward Neural Nets

Important: On top of these classifiers there exists a few image database handling routines and an 1D, 2D Haar Feature extractor which is based on an integral image approach.

Demo: TestdemoStatisticClassifier64bit.exe

Mr Math II Test

• A testing app TestClassifier.dpr which shows the usage and performance of these classifiers on various tasks (e.g. face recognition) which we slightly improved, compiled and signed can be found on git or sourceforge:

•

•https://github.com/maxkleiner/mrai_mx5/tree/master/TestApp

•

•

•https://sourceforge.net/projects/maxbox5/files/binaries/TestClassifier64bit.exe/download

•

```
15  uses
16    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, For
17    Dialogs, BaseClassifier, ExtCtrls, StdCtrls, Matrix, ComCtrls, Haar2DAd
18    Haar2DImageSweep, Image2DSweep, Types, Vcl.Imaging.jpeg;
19
20  type
21    TTrainSetType = (stGauss, stXOR, stCircles);
22  type
23    TfrmClassifierTest = class(TForm)
```

Code Control II

Write a stateful function, class, generator or co-routine that takes a series of floating point numbers, one at a time, and returns the running standard deviation of the series.

Use this to compute the standard deviation of this demonstration set, {2,4,4,4,5,5,7,9} which is 2. {\displaystyle \{2,4,4,4,5,5,7,9\}}

```
1  var dat: array of double; //Extended;
2      statclass: TStatisticClass;
3
4  setlength(dat,7);
5  dat:= [2,4,4,4,5,5,7,9]
6  statclass:= TStatisticClass.create;
7  for it:= 0 to high(dat) do
8     formatF('%2.0f -> %18.16f',[dat[it],AppendAndCalcStat(statclass,dat[it]
9  statclass.Free;
```

The **AppendAndCalcStat()** takes the statistic object to get filled and the continues dataset:

```
1  function AppendAndCalcStat(statobj: TStatisticClass; newVal:Extended):Exte
2  begin
3    statobj.add(newval);
4    result:= statobj.PopulationStdDev;
5  end;
```

Gas storage timeline AGSI dataset

The data science tutorial explains the so called AGSI data storage and his visualization of the timeline. AGSI is the Aggregated Gas Storage Inventory and offers you the possibility to be kept up to date whenever a new service announcement or update from one of our data providers is posted on the website.

- Mean **Function**: Calculates the average **of** an **array**.
- Covariance **Function**: Computes the covariance between two arrays.
- Correlation: **Uses** covariance **and** standard deviations for the correlation coefficient.
- ComputeCorrelationMatrix: Iterates through all variable pairs **to** compute corr-matrix.
- PrintMatrix: Outputs the matrix **to** the console.

Delphi does **not** have a built-**in** correlation matrix **function** with a heatmap too, but you can implement one using standard math operations.

https://blogs.embarcadero.com/why-a-data-scientist-chooses-delphi-for-powerful-real-world-visualizations/

Demo: 1154_energy_api_agsi_plot14_64bit.txt

Statsmodels

Statsmodels is a Python library designed for statistical modeling, hypothesis testing, and data exploration with high reference.

It provides a wide range of statistical models, including linear regression, time series analysis, and generalized linear models.
The library supports both formula-based modeling (similar to R) and direct use of NumPy arrays.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                Lottery   R-squared:                       0.414
Model:                            OLS   Adj. R-squared:                  0.392
Method:                 Least Squares   F-statistic:                     19.30
Date:                Thu, 26 Jun 2025   Prob (F-statistic):           1.47e-09
Time:                        17:35:22   Log-Likelihood:                -375.28
No. Observations:                  86   AIC:                             758.6
Df Residuals:                      82   BIC:                             768.4
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
```

Demo:
1417_statsmodels_64_delphi_python3.12.4debug130_EKON29.txt

Statsmodels Code

- We download the Guerry dataset, a collection of historical data used in support of Andre-Michel Guerry's 1833 Essay on the Moral Statistics of France. The data set is hosted
- online in comma-separated values format (CSV) by the Rdatasets repository. We could download the file locally and then load it using read_csv, but pandas takes care of all of this automatically for us:

```
1  Execstring('import statsmodels.api as sm; import numpy as np');
2
3  Execstr('model = smf.ols("Lottery ~ Wealth + Literacy + np.log(Pop1831)",
4  execstr('print(model.summary())');
5  execstr('sm.graphics.plot_partregress_grid(model)');
```

Execstr('model = smf.ols("Lottery ~ Wealth + Literacy + np.log(Pop1831)", data=df).fit()');

Demo: 1382_data_science_stuff140_12_py_uc.txt

Code Behind Dataframe

To transform the CSV data from file to matrix and dataframe you need 4 steps:

```
S:= TStringList.Create;
  try
    //S.StrictDelimiter := True;
    S.LineBreak := #10;
    //S1.Delimiter := ',';
    s.loadfromfile(exepath+'\examples\1417_export_dataframe.csv');
    writ('size: '+itoa(s.count));
    SetMatrixLength(mData, 86, 6);
    TStringListToMatrix(s, mData);
```

Input:
The `StringList` contains rows of data as strings, separated commas.

Output: The matrix is printed to verify the conversion.

Parsing: Each string is split into columns using `CommaText` of a temporary `TStringList`.

Matrix Population: The parsed values are stored in a 2D array (`Matrix`).

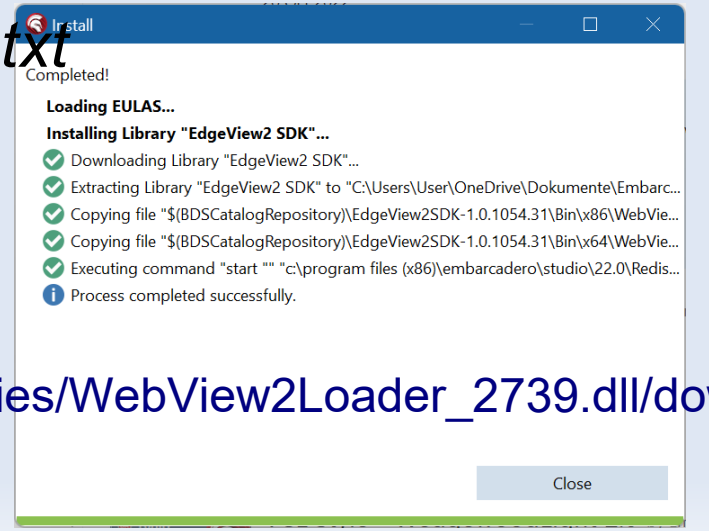
P4D Primer

Python Solution

```
1  procedure STDDEv_PySolution(loc: string);
2  begin
3   with TPythonEngine.Create(Nil) do begin
4     //pythonhome:= PYHOME64;
5     loaddll;
6     autofinalize:= false;
7     try
8       Execstring('from math import sqrt');
9       Execstr('import io, sys');
10      //Execstring('import plotly.express as px');
11      Execstr(DEFSD);
12      execstr('output = io.StringIO()')
13      execstr('sys.stdout = output')
14      execstr('sd.sum = sd.sum2 = sd.n = 0 '+LF+
15              'for value in (2,4,4,4,5,5,7,9):'+LF+
16              '  print (value, sd(value))');
17      println(stringreplace(evalstr('output.getvalue().strip()'),
18                            #10,#13#10,[rfReplaceAll, rfIgnoreCase]));
19    except
20       raiseError;
21    finally
22       UnloadDll;
23       Free;
24    end;
25  end;
26 end;
```

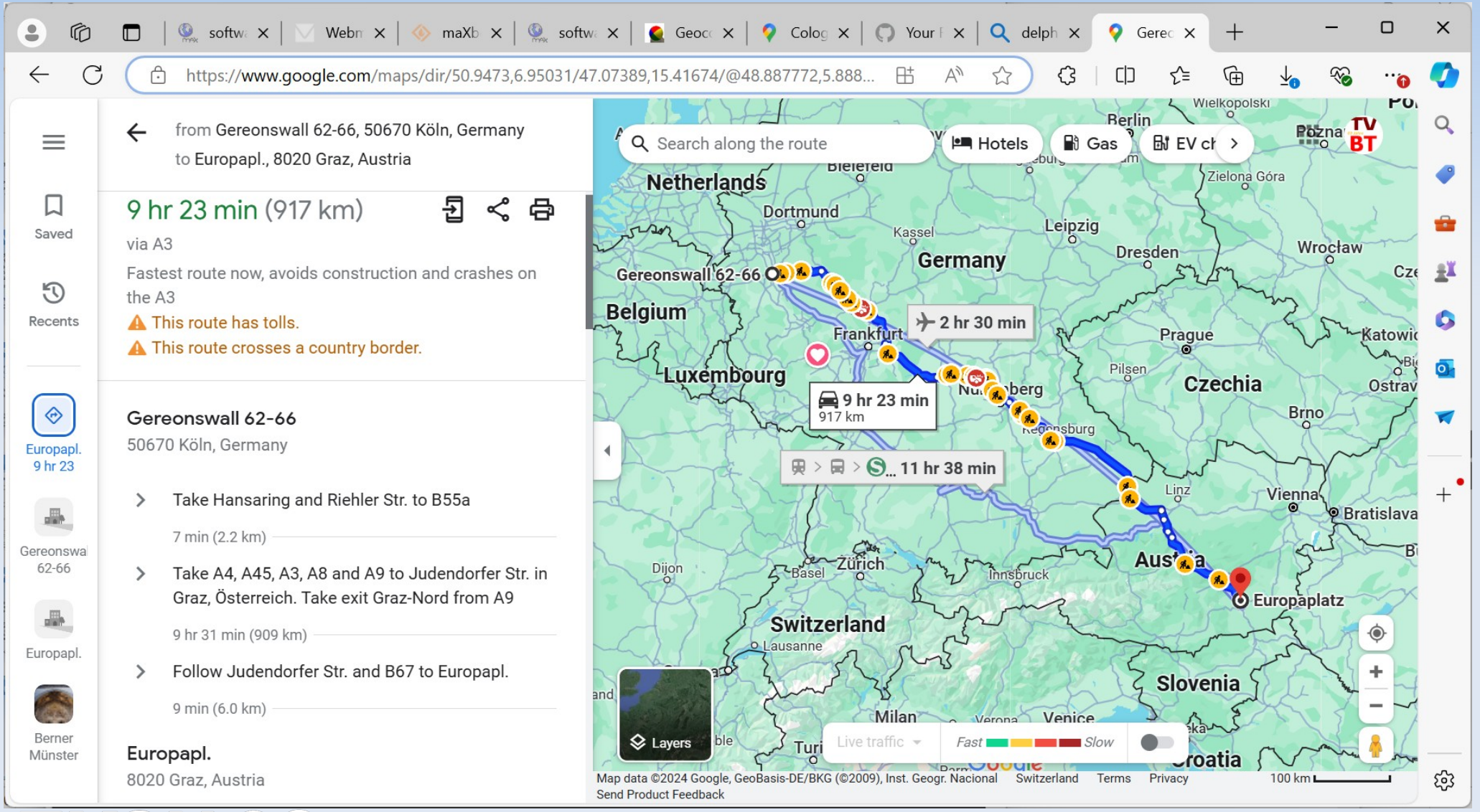
EdgeView2 SDK for API Stats

- McJson use a Simple Object-Pascal native code using TList as internal data structure with a Single-pass string parser and is just one unit (McJSON) and just one class(TMcJsonItem) integrated in maXbox or Lazarus.
- RAD Studio 10.4 Sydney brings support for working with web content through the Chromium-based Edge WebView2 browser control in VCL applications via the new TEdgeBrowser component.
- Demo: *1324_weatherbox_regex_64_maincities.txt*
  1415_weather_listview56.txt



https://sourceforge.net/projects/maxbox5/files/binaries/WebView2Loader_2739.dll/download

Stat Control




JS-Pas-Solution

- These projects have been selected based on their popularity, technical significance, and impact on the Delphi development community. They represent a mix of developer tools, frameworks, and components.
- Integrate dynamic, interactive, custom maps, statistic data and geospatial data into your apps by embedding JS:
- Weather, Navigation or Trafic Data uses huge statistical data with representation

```
48  const
49  HTMLStr: String = //i put The code for the web page page wich load the google maps in
50  '<html> '+
51  '<head> '+
52  '<meta name="viewport" content="initial-scale=1.0, user-scalable=yes" /> '+
53  '<script type="text/javascript" src="http://maps.google.com/maps/api/js?sensor=true">
54  '<script type="text/javascript"> '+
55  ''+
56  ''+//Declare the globals vars to be used in the javascript functions
57  '  var geocoder; '+
58  '  var map;  '+
59  '  var trafficLayer;'+
60  '  var bikeLayer;'+
61  ''+
```

Other Packages

Several statistical packages and libraries are available for Delphi, ranging from commercial solutions to open-source and freeware options.
These packages support a wide range of statistical computations, from basic descriptive statistics to advanced multivariate analysis.

Package	Core Features	Licensing	Advanced Methods
MtxVec + Stats Master	Broad stats, regression, ANOVA, PCA	Commercial	Multivariate, design of exp.
TMS Analytics Pack	Math expression parsing, stats, ODE	Commercial	Symbolic, numeric, physics
sqsBasicStats	Basic real-time statistics	Freeware/VCL	Focus on simplicity
DataLab	Statistical tool (end user app)	Application	Data editing, visualization
Python4Delphi (pandas)	Uses Python's stats libraries	FOSS/Hybrid	All Python-supported methods


Tops & Flops Stats AI

- **1. Awesome Delphi***
- **2. Delphi IDE Theme Editor**
- **3. DelphiMVCFramework (DMVCFramework)**
- **4. Delphi REST Client API***
- **5. Virtual-TreeView**
- **6. Delphi-OpenCV***
- **7. DUnitX**
- **8. Delphi IDE Colorizer**
- **9. CEF4Delphi***
- **10. OmniThreadLibrary**

These projects have been selected based on their popularity, technical significance, and impact on the Delphi development community. They represent a mix of developer tools, frameworks, and components (perplexity,chatgpt,gemini prompts).

https://github.com/maxkleiner/maXbox/blob/master/objectdetector3.ipynb


Tops & Flops Explain

1. Virtual-TreeView
A highly flexible and powerful tree view control, Virtual-TreeView has become a staple in both open-source and commercial Delphi applications. Its performance and extensibility have set the standard for advanced UI components in Delphi, making it one of the most influential libraries in the ecosystem[1].

2. OmniThreadLibrary
This threading library greatly simplified the development of multi-threaded applications in Delphi. Its ease of use and robust feature set have made it the go-to solution for Delphi developers needing parallelism and concurrency, ensuring Delphi apps remain competitive on modern multi-core hardware[1].

3. DelphiMVCFramework (DMVCFramework)
As web development became critical, DMVCFramework provided Delphi developers with a modern, powerful framework for building RESTful web applications and APIs. Its adoption has helped keep Delphi relevant in server-side and web API development[1].

4. CEF4Delphi
By enabling the embedding of Chromium-based browsers in Delphi applications, CEF4Delphi has allowed developers to integrate modern web technologies and content directly into native apps, bridging the gap between desktop and web[1].

5. DUnitX
Unit testing is essential for modern software development, and DUnitX brought advanced, NUnit-inspired testing capabilities to Delphi. Its adoption has fostered better software quality and test-driven development practices in the Delphi community[1].

6. Delphi-OpenCV
This project brought the power of the OpenCV computer vision library to Delphi developers, opening up new possibilities in image processing and AI within the Delphi ecosystem[1].

7. Delphi REST Client API
As RESTful APIs became the backbone of modern applications, this library enabled Delphi developers to easily consume web services, ensuring Delphi's continued viability for connected applications[1].

8. Awesome Delphi
While not a library or component, this curated list of high-quality Delphi resources has become an essential starting point for developers, helping to foster community and knowledge sharing[1].

9. Delphi IDE Theme Editor & Delphi IDE Colorizer
These tools have modernized the developer experience, allowing for extensive customization and theming of the Delphi IDE, which has contributed to the productivity and satisfaction of Delphi programmers[1].

https://github.com/maxkleiner/maXbox/blob/master/objectdetector3.ipynb


Flops (from API to AI)

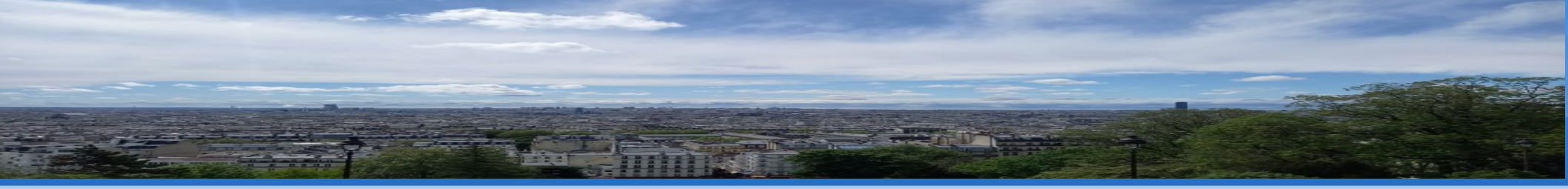
- **1. Delphi 8 and Delphi 2005 Releases**
- **2. Dreaded Sorts (THSorts) Demo**
- **3. Component Colossus Projects**
- **4. Projects Stuck on Old Delphi Versions**
- **5. Projects Dependent on Discontinued Technologies**
- **6. DLL Hell Projects**
- **7. Over-Componentized “Flavor of the Month” Projects**
- **8. Projects Broken by Unicode Transition**
- **9. Projects with Inflexible Architectures**
- **10. “Reinventing the Wheel” Projects**


Conclusion

- https://softwareschule.code.blog/2025/02/20/7-data-science-statistical-methods/
- **Config and spec features of DMath, SKLearn, Statsmodels, mrMath library**
- **Troubleshooting: Typical miscalculations and their solutions with cleaned data**
- **Typical Features of these Packages**
- **Descriptive stats: mean, median, mode, variance, percentiles**
- **Probability distributions (PDF, CDF, inverse CDF)**
- **Random number generation**
- **Hypothesis testing (t-tests, chi-squared, ANOVA)**
- **Regression (linear, logistic, ridge, Poisson, non-linear)**
- **Multivariate techniques (PCA, factor analysis)**
- **Data visualization, function approximation**
- Check the statistic package like TEE in feature-matrix
- https://www.embarcadero.com/docs/rad-studio-feature-matrix.pdf

Method: Correlation
Plot: **Chart** or Matrix
View: Form or Browser

A must see Operation System history
https://lnkd.in/e_KjeJdd



Many Thanks for coming!

Dienstag, 28. Oktober 2025
11:45 – 12:45, Max Kleiner

Materials:

https://github.com/maxkleiner/maXbox5/tree/main/EKON29

http://www.softwareschule.ch/download/maxbox_starter140.pdf

http://www.softwareschule.ch/download/maxbox_starter149.pdf

https://blogs.embarcadero.com/why-a-data-scientist-chooses-delphi-for-powe

https://sourceforge.net/projects/maxbox5/files/EKON29/

https://maxbox4.wordpress.com/2025/10/02/ekon-29-statistic-packages-slides/

EKON 29