Achal Gupta | Pranay Kekre

A00258772 | A00258773

KNIME Analytics

Introduction and Functionalities

Table of Contents

[Introduction 3](#_Toc531091574)

[Five Pillars of Knime 3](#_Toc531091575)

[Integrative 3](#_Toc531091576)

[Transparent and Trustworthy 3](#_Toc531091577)

[Flexible and Agile 3](#_Toc531091578)

[Collaborative 3](#_Toc531091579)

[Powerful 3](#_Toc531091580)

[Knime Ecosystem 3](#_Toc531091581)

[Knime Analytics Platform 4](#_Toc531091582)

[Knime Server 4](#_Toc531091583)

[Knime Extensions 4](#_Toc531091584)

[Knime Integrations 5](#_Toc531091585)

[Community Extensions 5](#_Toc531091586)

[Market Reviews 6](#_Toc531091587)

[Knime Trusted Parthners 7](#_Toc531091588)

[Installation Guide 7](#_Toc531091589)

[Interface Guidelines or Menu Options 9](#_Toc531091590)

[Workflow Projects: 10](#_Toc531091591)

[Node Repository: 10](#_Toc531091592)

[Outlines: 11](#_Toc531091593)

[Console: 11](#_Toc531091594)

[Node Description: 12](#_Toc531091595)

[Other Advanced Data Sources 13](#_Toc531091596)

[Nodes in KNIME 13](#_Toc531091597)

[IO 13](#_Toc531091598)

[Read Node 14](#_Toc531091599)

[Write Node 14](#_Toc531091600)

[Manipulation – Derived Column, Conditional Filtering, Lookup joins, Grouping or Aggregation (Pranay) 15](#_Toc531091601)

[Manipulation: 15](#_Toc531091602)

[Binning: 15](#_Toc531091603)

[Views (Pranay) 16](#_Toc531091604)

[Views: 16](#_Toc531091605)

[Property: 17](#_Toc531091606)

[Utility: 17](#_Toc531091607)

[Analytics 17](#_Toc531091608)

[Statistical 17](#_Toc531091609)

[Classification 18](#_Toc531091610)

[Clustering 18](#_Toc531091611)

[Regression 19](#_Toc531091612)

[Optimization 19](#_Toc531091613)

[Database 19](#_Toc531091614)

[Read/Write 19](#_Toc531091615)

[Connector 20](#_Toc531091616)

[Manipulation 20](#_Toc531091617)

[Utility 21](#_Toc531091618)

[Other Data Types 21](#_Toc531091619)

[Time Series 21](#_Toc531091620)

[Structured Data 21](#_Toc531091621)

[JSON: 21](#_Toc531091622)

[Scripting 22](#_Toc531091623)

[In Scripting: 22](#_Toc531091624)

[Tool & Services 23](#_Toc531091625)

[Rest Web Services 23](#_Toc531091626)

[Workflow Control 23](#_Toc531091627)

[Call Local Workflows 23](#_Toc531091628)

[Variable to Table row 23](#_Toc531091629)

[Catch Errors 24](#_Toc531091630)

[Reporting 24](#_Toc531091631)

[Data to report 24](#_Toc531091632)

[**Real World Example**: 24](#_Toc531091633)

[Dataset: 24](#_Toc531091634)

[Introduction 24](#_Toc531091635)

[Example 1: 24](#_Toc531091636)

[Example 2: 28](#_Toc531091637)

[Opinion on Application 33](#_Toc531091638)

[Experience level required 33](#_Toc531091639)

[Data size 33](#_Toc531091640)

[Visualization capabilities 33](#_Toc531091641)

[Conclusion 33](#_Toc531091642)

[References 34](#_Toc531091643)

# Introduction

KNIME Analytics Platform is an open source for data-driven innovation, designed for discovering the hidden potential of the data. Data mining for new insights and predictive modeling are some of its key strengths. It is developed KNIME headquartered at Zurich

# Five Pillars of Knime

## Integrative

Availability of use of multiple data sources including complex networks, images or texts.

## Transparent and Trustworthy

Creating reusable workflows, customizable interfaces thus allowing even business users to run sophisticated analyses.

## Flexible and Agile

Quick data exploration with no platform or vendor restrictions and simultaneous execution of complex analyses.

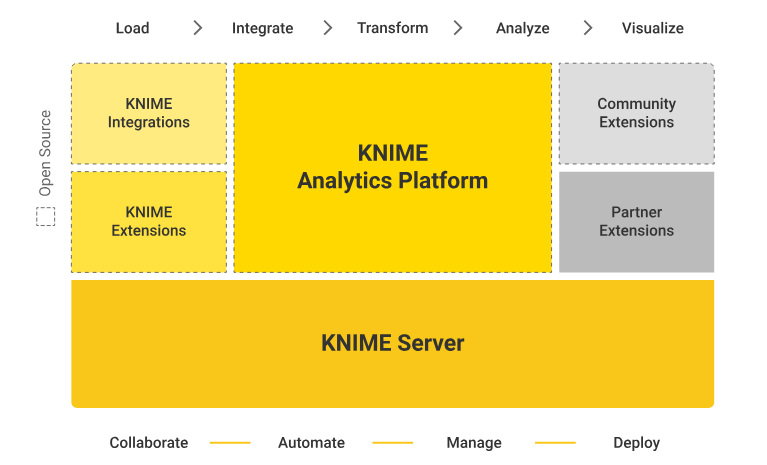
## Collaborative

Open platform makes it easy to share and evolve as a community. Knime also supports collaboration among teams.

## Powerful

With a huge spectrum of analytics tools and visualization Knime is powerful.

# Knime Ecosystem



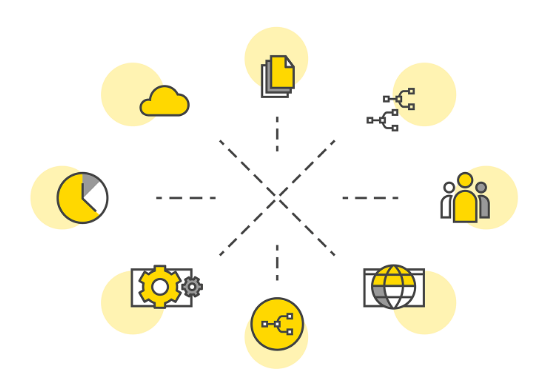
(KNIME, 2018)

## Knime Analytics Platform

Knime Analytics Platform is the core of the entire Knime ecosystem. All the integrations and extensions available in the ecosystem adds up features and functionalities to the platform and makes it one of the most powerful and wide analytics tool available in the market.

## Knime Server

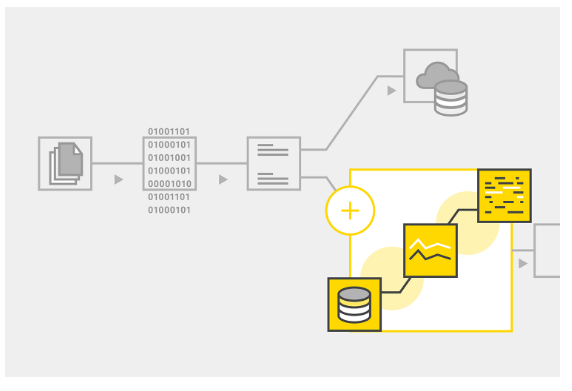
Knime Server is the enterprise software for team-based collaboration, automation, management and deployment of data science workflows, data and guided analytics (KNIME, 2018).



(KNIME, 2018)

## Knime Extensions

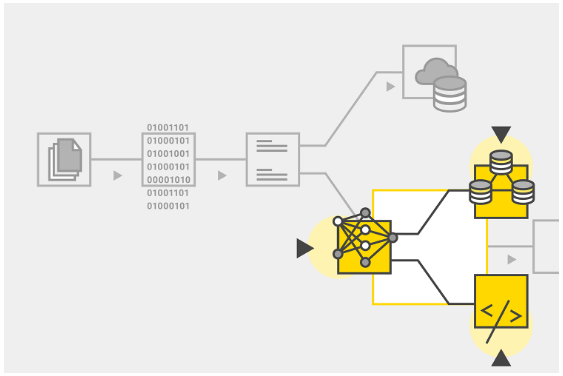
Knime extensions are open source additional functionalities developed and maintained by KNIME, they provide additional features like analyzing complex data types and complex machine learning algorithms. Some of the major extensions include text processing, neural networks, and integration to databases like Amazon Athena, Microsoft SQL etc.



(KNIME, 2018)

## Knime Integrations

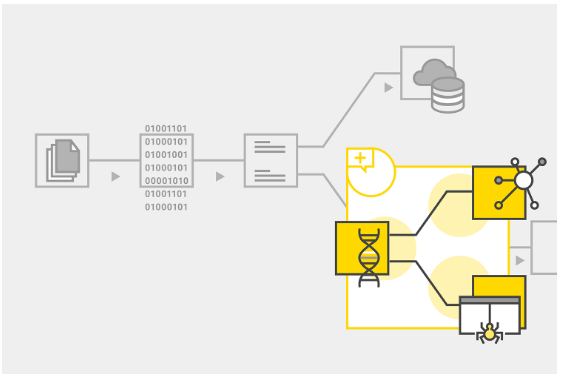
Open source integrations for KNIME Analytics platform provide additional functionality connect large open source projects like Keras for Deep learning, H20 for high-performance machine learning, Apache spark for big data processing and Python and R for scripting. These are also developed and maintained by KNIME.



(KNIME, 2018)

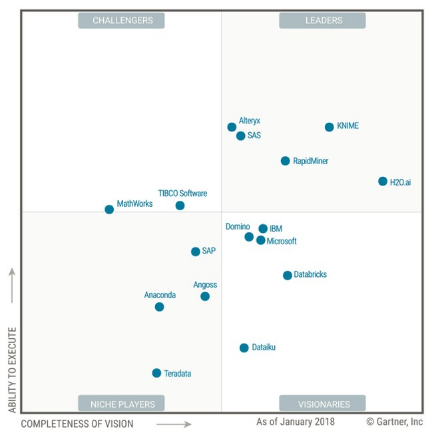
## Community Extensions

This includes additional functionalities developed and maintained by KNIME users which include highly industry specific applications to scientific software integrations.



(KNIME, 2018)

# Market Reviews



**Fig: -** **Gartner 2018 Magic Quadrant for Data Science and Machine Learning Platforms**

(Gartner Inc, 2018)

The 2018 report published by Gartner Inc named “Magic Quadrant for Data Science and Machine-Learning Platforms” clearly shows KNIME as the market leader leaving behind some of the bigger brands like IBM, Microsoft etc. Gartner states following about KNIME.

“The vendor demonstrates a deep understanding of the market, a robust product strategy and strength across all use cases. Together, these attributes have solidified its place as a Leader” (Gartner Inc, 2018).

# Knime Trusted Parthners



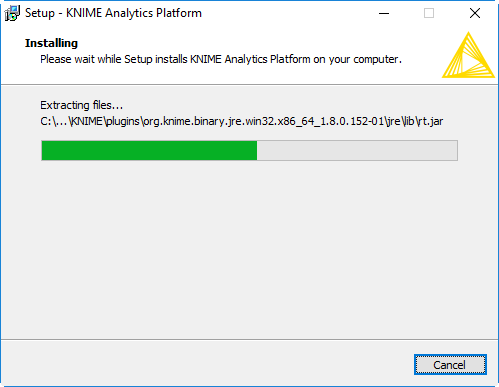
# Installation Guide

Knime Analytics is available to download for Windows/MacOS/Linux and can be downloaded from the below links.

<https://www.knime.com/downloads/download-knime>

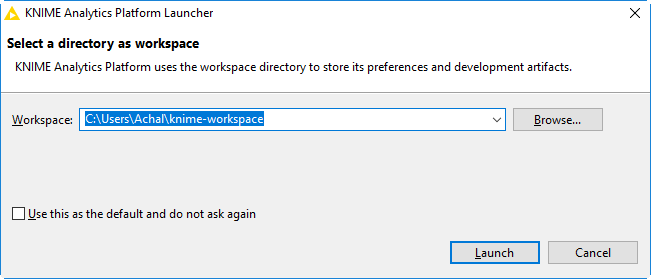
Below is the guide to install Knime Analytics in Windows OS:

* After the download is finished, execute the installer wizard file, agree to license agreement and provide other details and let the installer install Knime automatically

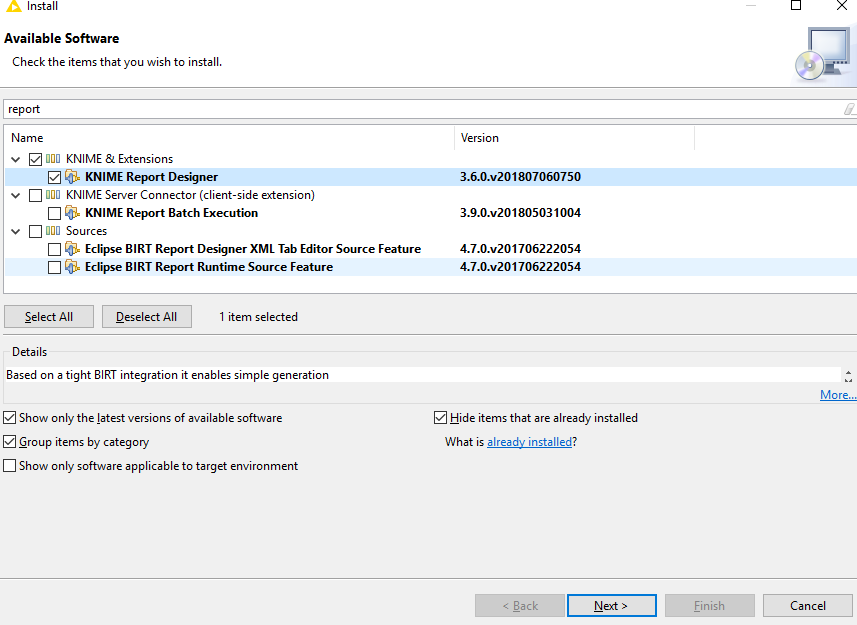


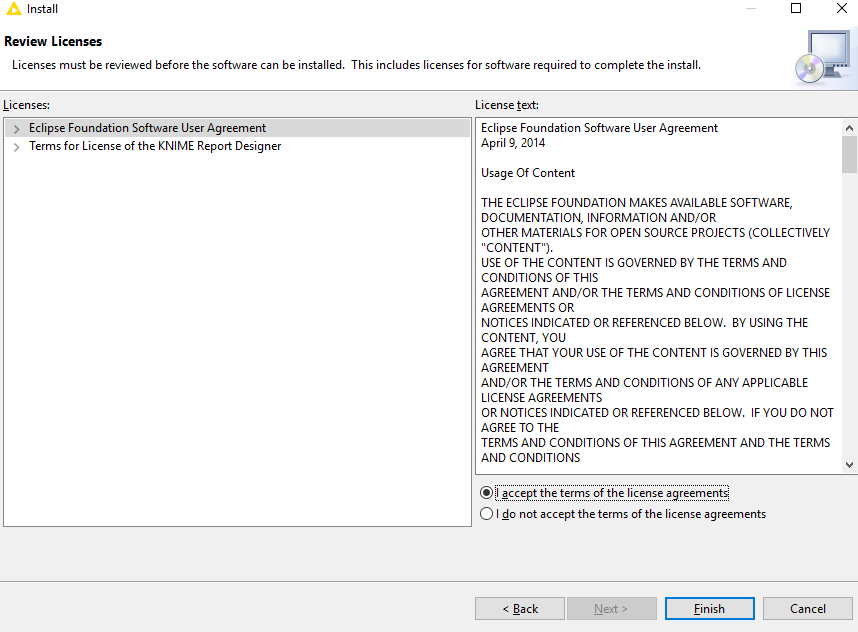
Once the processing is finished click the “Finish” button to complete installation and Launch Knime Analytics software.

* When the software is launched we need to define a workspace, wherein all the files and workflows will be stored. Select the appropriate path and click on Launch to run the software.



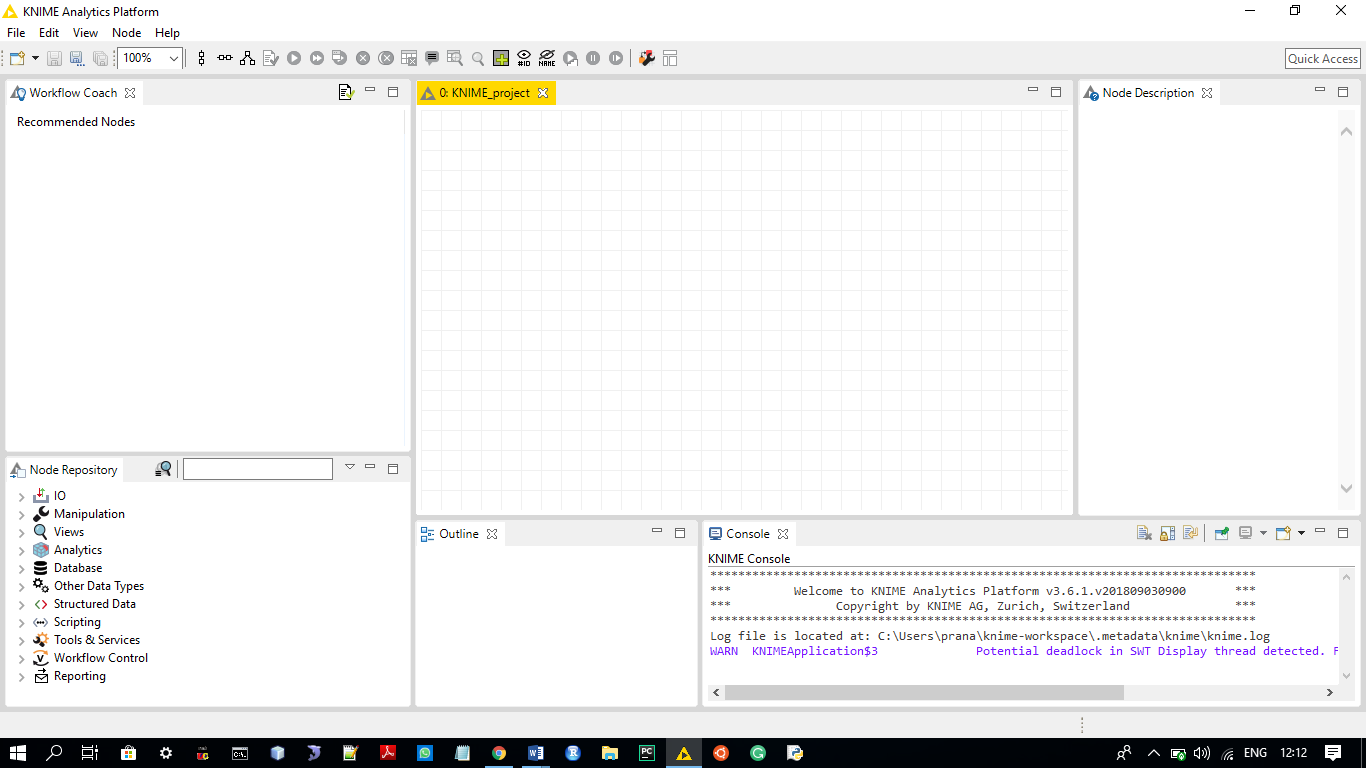
Installing Reporting Extension in KNIME





# Interface Guidelines or Menu Options

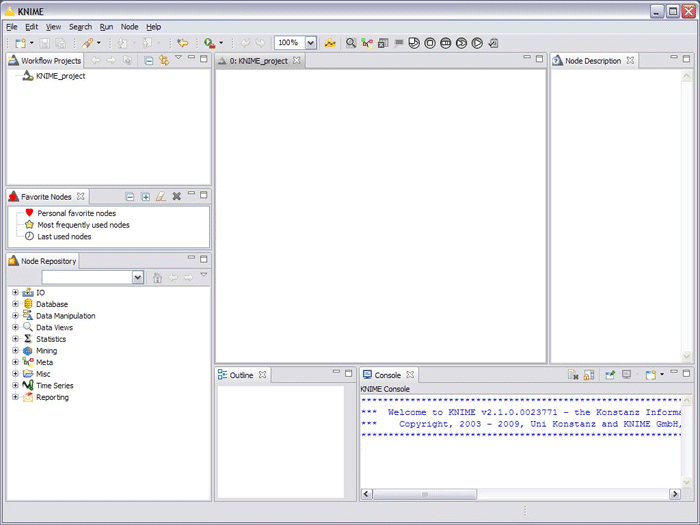
KNIME is not based on scripting languages. A Procedure is implemented by the means of workflow and workflow is made up of nodes. You can see the KNIME workbench. This central area is the workflow editor.



## Workflow Projects:

In this window, all the projects are displayed such that the number of the project made are all displayed here.

You can see the workflow projects window that consist of the project created. The KNIME\_project in Workflow Project section specifies the creation of the project.

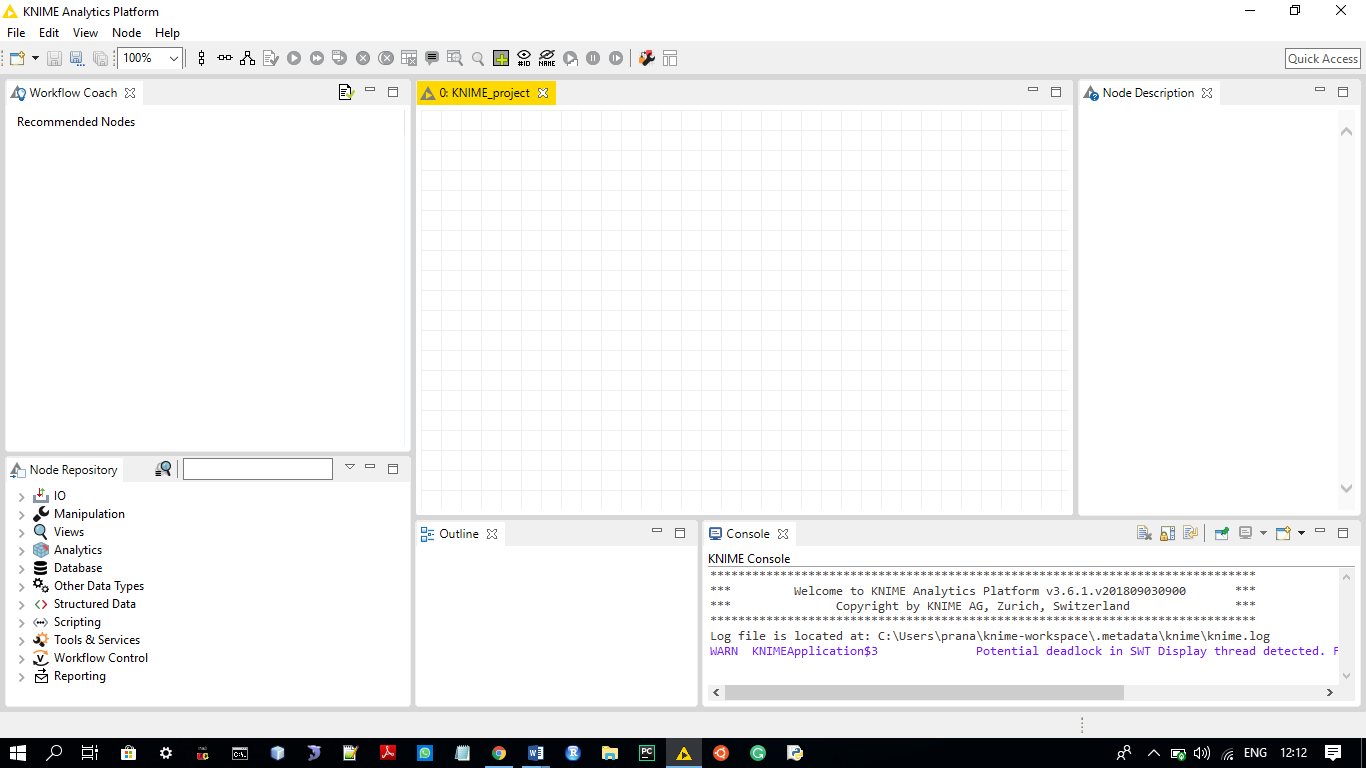


Knime\_project is further categorized in the different subprojects.

## Node Repository:

The Node Repository is located at the lower left corner of the workbench in the KNIME Analytic Platform. It contains the nodes that can be used in your workflow. The Node Repository usually contains over 1000 nodes that can be used as per the latest version available on the KNIME platform. The Nodes are added from the node repository to the workflow to perform the task the user wants. The Nodes are organized in categories where each category represents a specific functionality. The categories that are available below are:

IO, Manipulation, Views, Analytics, Database, Other Data Types, Structured Data, Scripting, Tools & Services, Workflow Control, Reporting.

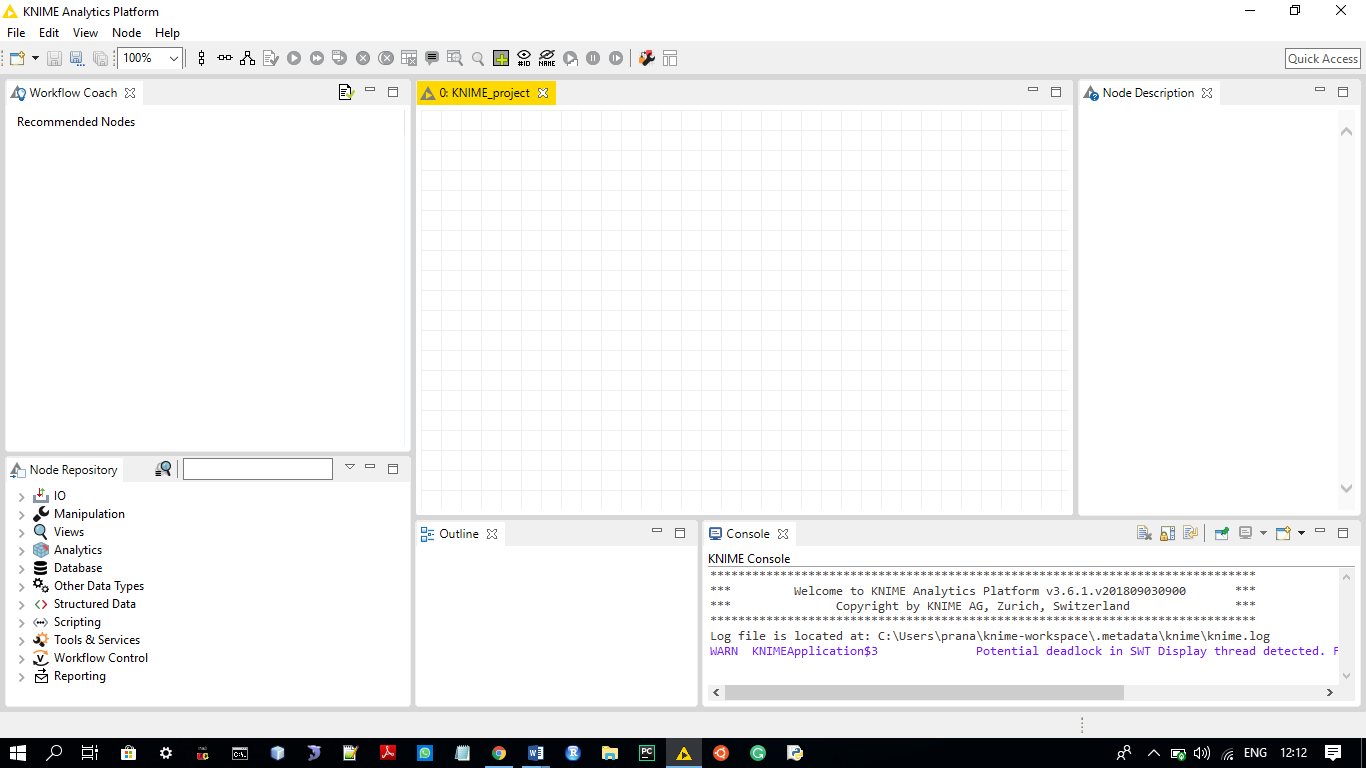


## Outlines:

The Outlines section is there to summary the tasks performed in the workflow area.

The outline works as a navigator of the workflow despite the visibility is too small as compared to the workflow editor, using the outline we can navigate the workflow editor window as per the requirement, the mouse can be used to adjust the workflow editor, the workflow editor may consist of multiple nodes and working on these nodes is impossible with limited visible node, using the outline one can view the node that is on either side of the workflow editor.

This outline keeps the track of each and every node in the workflow area. The outline is just below the workflow area in the KNIME. The outline shows the structure you are working on.

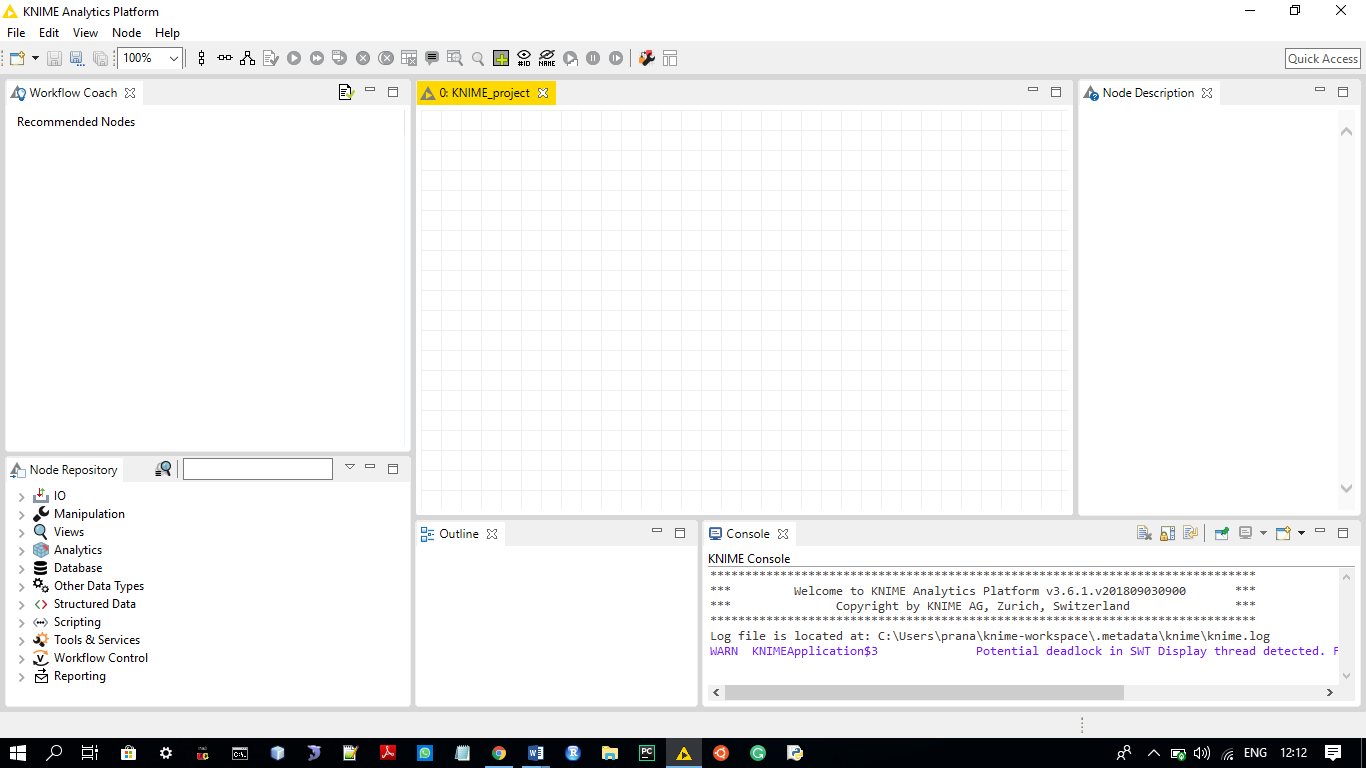


## Console:

The Console views the functions that are performed in the workflow editor such that if a task for example regression is performed on a certain number of variable and set of values then the console will print the result of the end regression.

Now, the result printed in the console can either be an exact output or an error and warning message to provide information regarding what task is performed and task when performed generate.

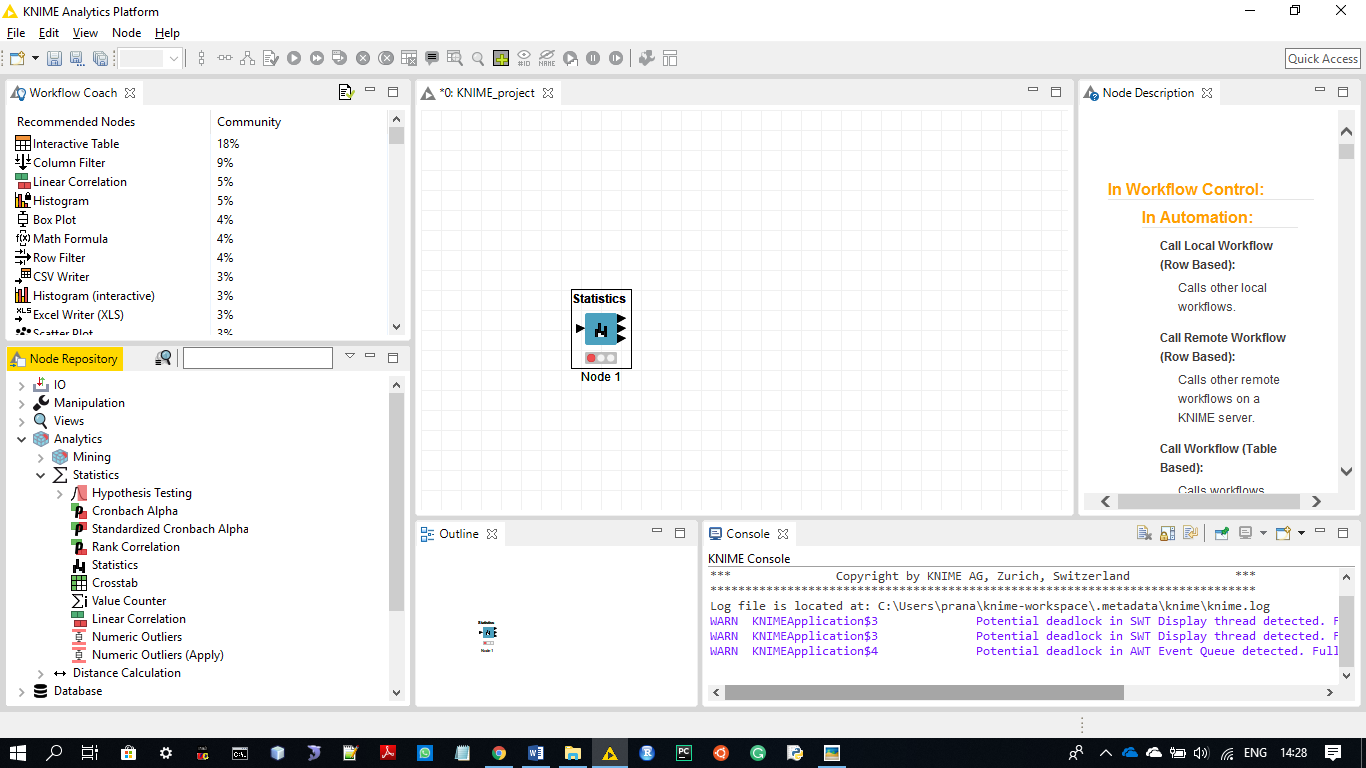
The console provides options such as Debug, Info, Warning, and Error.



## Node Description:

The Node Description section in the KNIME platform is such that the node that the user is using is briefed in the node description section.

The description section shows the dialog options, the available views, the input data format user can enter and the resulting output format.



# Other Advanced Data Sources

The Sources of the KNIME platform by which it is easier to work with the data, this is possible by the KNIME Extension feature, the extension enables this platform to an add-on that is the connectors are there for this platform which allows sharing of the data from external sources to KNIME.

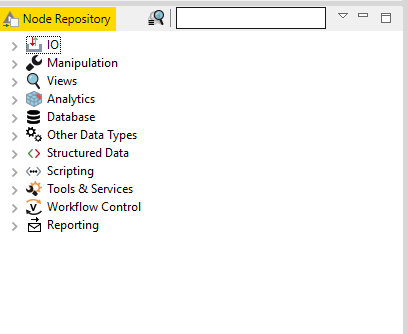
Extensions like Big Data Extension, integration with the Apache Hadoop Platform, Apache Spark platform, KNIME Cluster Executer that is a sort of connecting tool that enables the Analytical Platform, it enables high-performance cluster and distributed across the cluster.

KNIME Collaboration Extensions include KNIME Server Lite that provide the elementary user verification and user privilege, isolated timely execution, report generation, shared data space, workflow repository, and meta-nodes and updates.

Knime Server 4.4 which provide additional features to the Server Lite extension provides much protected and unconventional user care that is web services support and commercial support.

KNIME Teamspace improves squad association work, provided that a means for loading data movement and investigate workflow centrally, to be shared and worked on collaboratively by multiple users at a time by several team members.

# Nodes in KNIME



## IO

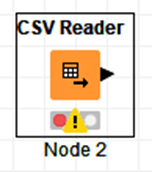
The I/O category contain parser nodes that is responsible of converting texts from various formats, such as DML & SDML, PubMed (XML format), PDF, MSWord, and flat documents.

Parsing as well as analysis the data into the KNIME is the primary work that is compulsory. The result of each and every parser node is the table of data entailing of single column with Document Cells. Every cell has only one file. This list of documents can be used for the input by all the nodes of the enhancement class. The DML and the SDML formats are XML based set-ups to characterise text in structural way. The text obtainable in other XML based formats can be altered into SDML easily by implementation of XML node or sub-nodes in the XML, provided by the KNIME XML plugin (KNIME, 2018).

### Read Node

#### CSV Reader

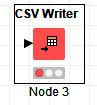
Reads CSV files that the user want to import into the KNIME platform. This node is used if KNIME workflow is used in a server or batch environment and the inserts the files creations change between different supplications. Specifically this node comprises of the variable number of input column. When execution of the node is done the node will examine the input file to regulate number and types of the columns and output a table with the auto-predicted structure.



### Write Node

#### CSV Writer

This node is responsible for the writing of the CSV file into a location or we can say that this the node creates a CSV file into the project directory. The data in the table into a file or to a distant site denoted by an URL that the user specifies. The node provides many options to adapt the yield format. But there is no cross verification done on the backgrounds, so it is thinkable to write out the data in a format that makes it impossible to read it in again.  
Note that if the target site is a remote URL, not all options are available because usually, it's impossible to regulate remote path that exists. In this case, it will always be overwritten.



## Manipulation – Derived Column, Conditional Filtering, Lookup joins, Grouping or Aggregation (Pranay)

Manipulation is one of the most used features of the KNIME platform, this feature is used to manipulate that is either update, delete or insert. In another language this is the one node in the repository that is used to extract, transform and load the data.

The subparts or the sub-categories of manipulation node are Column, rows, and table with PMML as the additional feature.

Coming on the main manipulation feature if we look at the column of the data that is to be manipulated we can do the binning that is none other than the grouping of the numeric data into the intervals.

A derived column is the column that can be made through manipulation of the two different tables and making a new column, here a new table is created using the already existing tables. A specific sort of extraction is done depending on the requirement of the user.

The Filtering is another type of extraction of the data were the unnecessary data is taken out. The more simplified version of filtering is the excluding of the not important column of the table.

This same case is applied to the rows that are the rows get functioning. Filtering, convert and replace the column and rows are possible in the platform.

However, split and combine is also important in a function that is usually performed in the database that is to be used in the project.

The Lookup join can also be performed using the manipulation node of the KNIME. Here, the joins are only applied on the tables and the tables can be linked through the usage of the database. Once a table that is created in the database such as if you make a table in the MYSQL and then using the connector you connect the database node to the MYSQL application then the manipulation on the tables is possible using the KNIME platform.

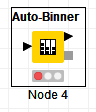
### Manipulation:

In Column the functions are in such a format that the conversion from a data type is done to another using the respective methods in a single operation:

### Binning:

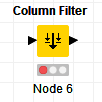
#### Auto-Binner:

The Auto Binner node allows to group numeric data in intervals - called bins, for example, there is a numeric data of around 1-100 then it might generate bins of 1-10, 11-20 and so on.



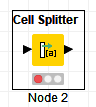
#### Column Filter:

The Column Filter makes columns to be excluded from the input table such that the unwanted columns are eliminated. (KNIME, 2018)



#### Cell Splitter:

Splits the string demonstration of cells in one column of the table into separate columns or into one column containing an assembly of cells, based on a definite delimiter. (KNIME, 2018)

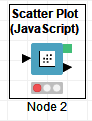


## Views (Pranay)

### Views:

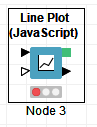
#### Scatter Plot (JavaScript):

A scatter plot using a JavaScript founded in the charting library.



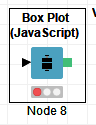
#### Line Plot (JavaScript):

A line plot using a JavaScript present in the charting library.



#### Box Plot (JavaScript):

This node provides a view with a Box Plot implemented with D3.js.



### Property:

#### Colour Manager:

Assigns colours to a certain numeric column.

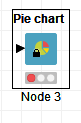


### Utility:

Conversion of one format to another:

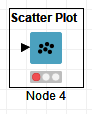
#### Pie chart:

Displays data in a pie chart. Highlighting is not supported.



#### Scatter Plot:

Creates a scatterplot of two selected attributes.



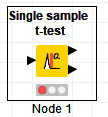
## Analytics

KNIME offers nodes for various analytical algorithms. Below we will explore some of the most common algorithms used in a Data Science project.

### Statistical

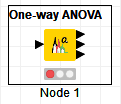
#### Single Sample t-Test

This node evaluates sample t-test. The evidence is gathered to evaluate the Null Hypothesis that the population means is equal to a given number. (KNIME, 2018)



#### One-way ANOVA

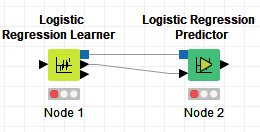
This node calculates statistics for one-way analysis of variance or (ANOVA). It compares the mean of observations in the same column between several groups.



### Classification

#### Logistic Regression

These nodes perform multinomial logistic regression first node is the model generator node and second is the predictor node. Logistic regression is used to create a model and predict a binary outcome i.e. 0 or 1 or Success or failure. E.g. Logistic regression was used to develop Trauma and Injury Score (TRISS) which is widely used to predict mortality in injured patients (Wikipedia, 2018).



### Clustering

#### K Means

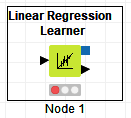
This node displays centers of a predefined number of clusters. K Means algorithm uses Euclidean distance to assign data vectors to clusters, the algorithm terminates when the cluster assignment seizes to change.



### Regression

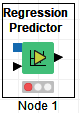
#### Linear Regression

This node generates the predictor variables using linear regression. Independent columns can be included in a regression model which are having numerical and nominal data.



#### Regression Predictor

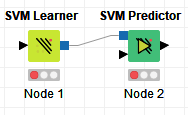
This node predicts the outcome of a dependent variable using the model generated by the learner node.



### Optimization

#### Support Vector Machines

This node trains support vector machine on the input data and generates an optimized model. Predictor node predicts the output of given values using the model generated from the learner node.



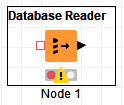
## Database

Knime offers connectivity to most of the databases like Microsoft SQL, SQLite, MySQL etc. and supports advanced big data datasets. Database category of nodes includes all nodes which are responsible for database connectivity and performing various operations on the database.

### Read/Write

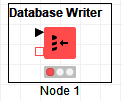
#### Database Reader

This node establishes a database connection to read the data from the database.



#### Database Writer

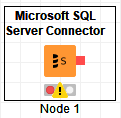
This node establishes a database connection to write data from the database.



### Connector

#### Microsoft SQL Server Connector

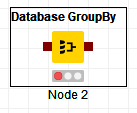
This node creates a connection to Microsoft SQL server using the JDBC driver. Server hostname (IP address), the port, and a database name must be provided to establish the connection. Logins can be provided either directly in configuration or via credentials set on workflow.



### Manipulation

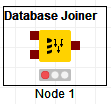
#### Database Group By

This node is used to aggregate the data in a database using the column specified in the GROUP BY clause.



#### Database Joiner

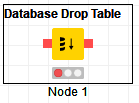
This node is used to join two tables in the database. Join is based on the joining columns of both tables.



### Utility

#### Database Drop Table

This node allows us to run drop table command over the connected database.

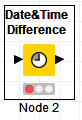


## Other Data Types

### Time Series

#### Date and Time Difference

This node calculates the difference between two dates and time and appends the result in a new column.



## Structured Data

### JSON:

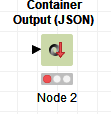
#### Container Input (JSON):

Require a text in the dialog and yield it as a JSON value.

#### 

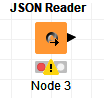
#### Container Output (JSON):

Reads the content of a JSON column and makes it accessible to an external caller.



#### JSON Reader:

Reads .json files to JSON values.



#### JSON Writer:

Writes json files.

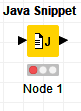


## Scripting

### In Scripting:

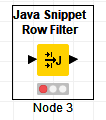
#### Java Snippet:

Calculates a new columns or flow variables based on java code snippets. (KNIME, 2018)



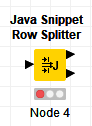
#### Java Snippet Row Filter:

Java Snippet based Row Filter



#### Java Snippet Row Splitter:

Java Snippet based Row Splitter



## Tool & Services

### Rest Web Services

#### Get Request

This node utilizes HTTP Get request method to GET data from a web service.



#### POST Request

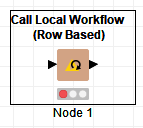
This node utilizes HTTP Post request method to posts request and send data to a web service.



## Workflow Control

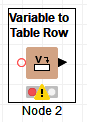
### Call Local Workflows

This node is used call other workflows present in the local workspace. This is a crucial node as it is utilized to maintains execution sequence of various workflows in a project.



### Variable to Table row

This node extracts scope variables and put them into a single row table.



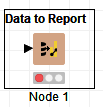
### Catch Errors

Error handling can also be performed in Knime Analytics Platform using various Error Handling nodes provided. Catch Error nodes used to catch errors

## Reporting

### Data to report

This node is used to send incoming data to KNIME Report Designer.



# **Real World Example**:

## Dataset:

<https://www.kaggle.com/hugomathien/soccer>

## Introduction

Soccer is one of the sports with the highest followers all around the world. The below example uses the data of various matches and corresponding players.

The Data used is an SQLite file that is imported into the KNIME platform. It contains five major tables: Player, Player\_Attributes, Match, League, and Team.

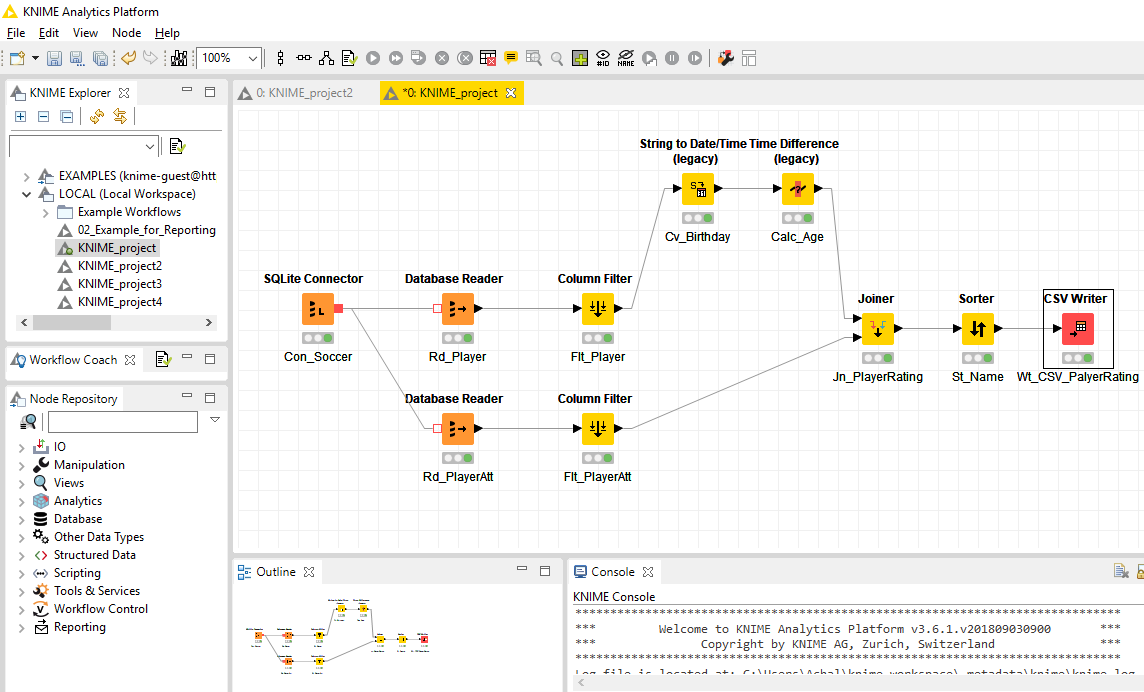
SQLite Connector allow us to connect the database file for further operations.

Moreover, the SQLite file taken is neither converted nor manipulated.

The case study is such that, the attributes of the various tables that are present in the database are taken to evaluate and generate various statistical components such as different graphs. The data is extracted in such a way that the visualization produced provides much of the information.

## Example 1:

Identify Height and Weight distribution of various soccer players and analyse if the overall rating of a player is dependent on Stamina of the player.



The workflow above specifies various nodes used to analyze the dataset.

**Con\_Soccer**: This is the initial node that is taken to perform the database connectivity of the KNIME platform and the SQLite file. Using this node, we upload the dataset to the KNIME. This process of connectivity is in such a way that the data firstly get inserted before which the red-light signal of the node is on and then this data is processed during this the node shows the yellow signal on the signal symbol and once it is processed the green light appears.

**Database Reader:** The Database reader is the node that reads the tables that are present in the database. There can be multiple tables in the database, therefore when the reader node is activated the table that the user wants to access must be specified. In the Two Above Reader Node, the two different tables are inherited, these tables are Player and Player\_Attribute read by Rd\_Player Node and Rd\_PlayerAtt node respectively.

**Flt\_Player:** This node has the functionality of filtering the columns in the Player Table present in the database. The column that is extracted from the table is “Birthday” of the Player that is in the table.

**Flt\_PlayerAtt:** The node is responsible for extracting the data from the Player\_Attribute table, here the columns that are extracted include varying attributes of the Player in the Player\_Attribute table.

**Cv\_Birthday:** This node is concerned with the birthdate conversion, the tables that are present in the database consist of the birthdates of the player in the string format that is not much useful for generating graph that are based on the quantitative data, therefore, we are using the String to Date/Time converter node to get the numerical quantitative data.

**Calc\_Age:** The Calc\_Age node is the one of the most important nodes that calculates the age of the player by undertaking the birthdates of all the player and the dates undertaken are evaluated by the current date so that the age of the player is generated.

**Jn\_Player\_Rating:** The Joiner node taken here is the node that joins the data generated from the two previous nodes: Calc\_Age and Flt\_PlayerAtt. The node joins the player age and the other attributes such as the player name etc.

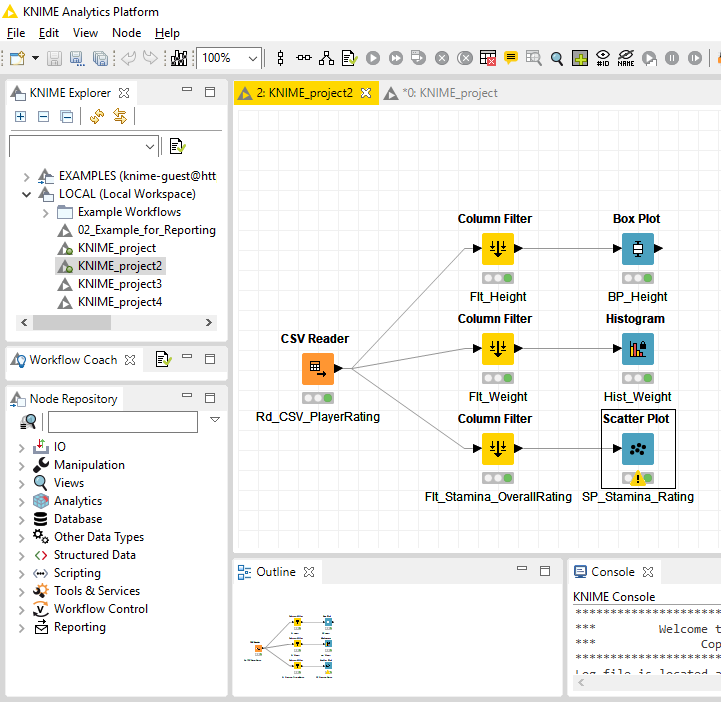
**St\_Name:**  The St\_Name node is the node that sorts the names of the player in the alphabetical order such that the names appear alphabetically in the end.

**CSV\_Writer:** The CSV\_Writer is the node that writes the data into the CSV file, here it depends on the user at what location the file should be saved for the future use.

****

Only a few entries are shown from the CSV generated.

**Generating Reports:**



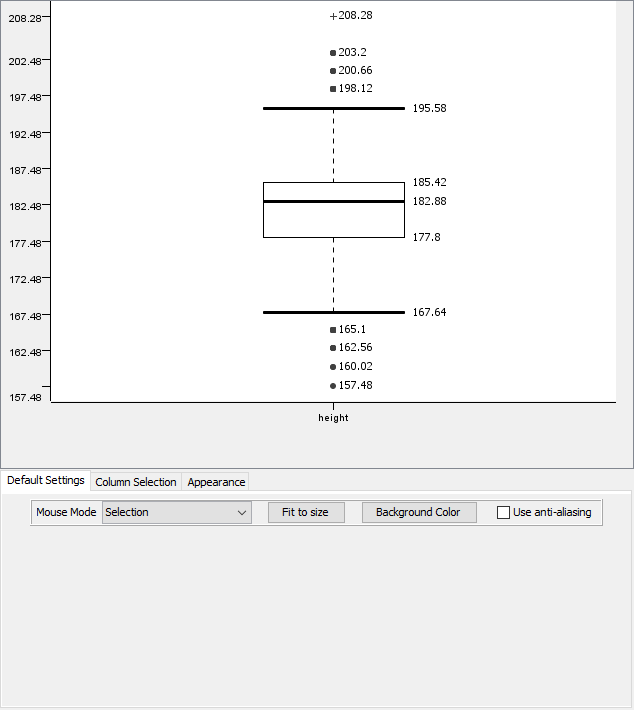
**Rd\_CSV\_PlayerRating:** The CSV file that was generated in the previous approach is Read using this CSV Reader node to perform the further operation.

**Flt\_Height:** The node filters the height of the players in the player's table using column filter.

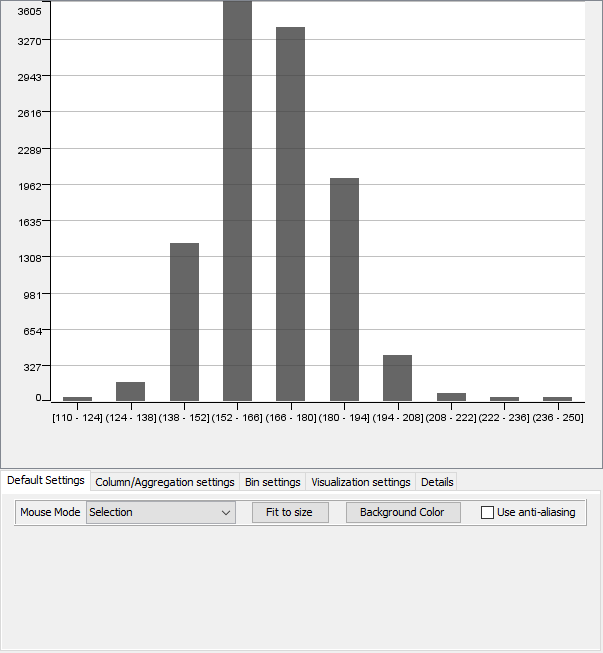
**Flt\_Weight:** The weight of the players is extracted from this node using the column filter.

**Flt\_Stamina\_OverallRating:** The node filters out the stamina and the overall rating of each individual player.

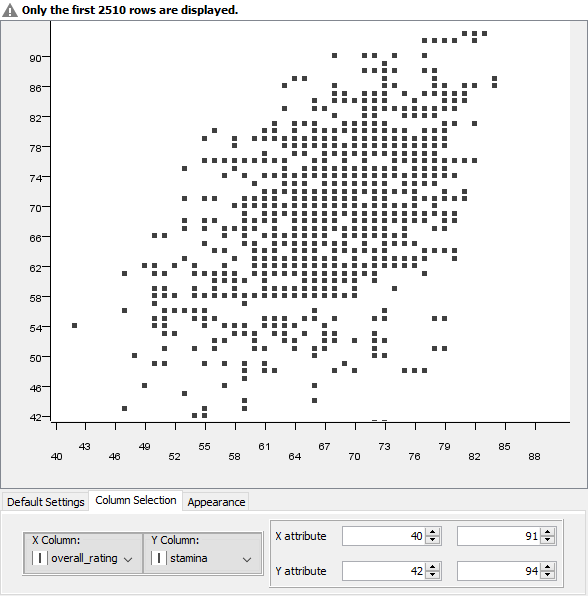
**BP\_Height:** The node helps the user to generate the Box-Plot using the heights of the player from the previous filter. The mean height of the player is 182.88cm whereas the other players lie in the bound of 185.42cm to 177.8cm.



**Hist\_Weight:** The weight of the players is used to create the Histogram. The Histogram using the Histogram Node in Views Repository of KNIME Platform. The weight in the histogram is in pounds. The Histogram created shows the highest number of player has similar weight, there are maximum players in the bin 150-180 pounds.

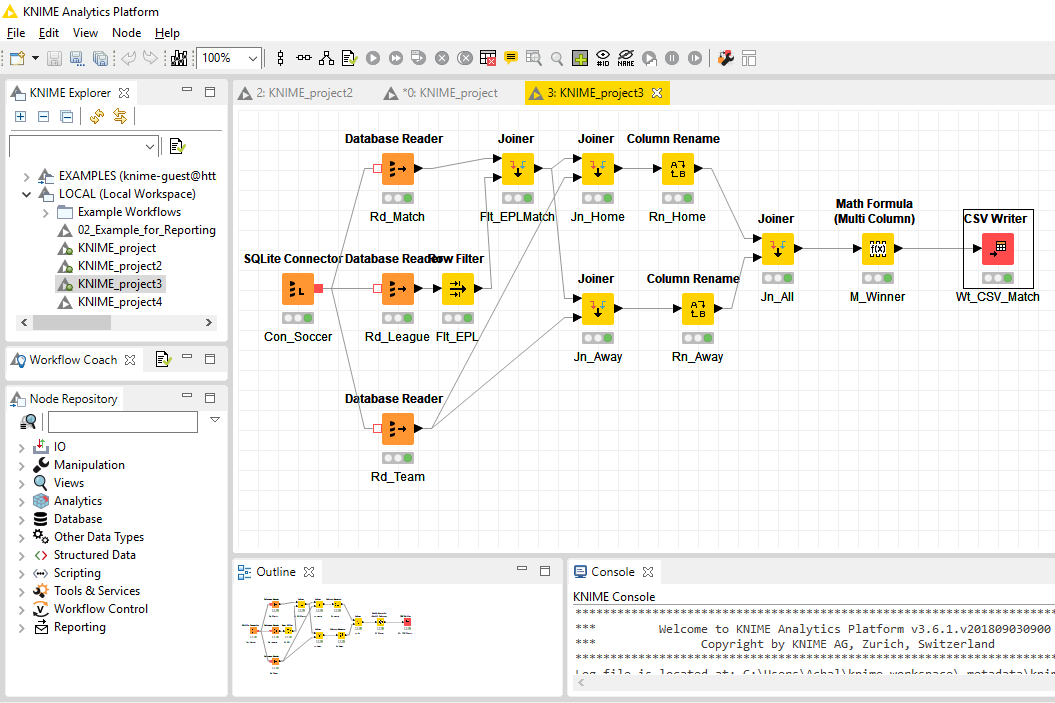


**SP\_Stamina\_Rating:** This is the Scatter plot that is made using the values from the stamina and the overall rating column. From the plot, we can see that there is a positive relation between Stamina and Overall rating and hence we can say that more the stamina of player more will be the overall rating.



## Example 2:

Compare a number of wins by Home team vs Away team and also analyze the win statistics of 6 major EPL teams.



In this approach, the matches played in the respective league between the teams participating in the league are taken to extract the teams whether there is a win or loss of the home or the away team.

**Con\_Soccer:** It is the same connector node that connects the database to the KNIME.

**Rd\_Match:** Through this node, the match table in the database is read by the KNIME.

**Rd\_League:** Here the league table in the database is taken.

**Rd\_Team:** Node reads the teams that are there in the database irrespective of any constraint of the league.

**Flt\_EPL:** This is the node that filters out the English Premier League from the League table present in the database.

**Flt\_EPLMatch:** This is the Joiner node that joins the league that is filtered that is the EPL and then joins that to the matches that are played such that the output shows the matches that are held in the EPL.

**Jn\_Home:** The Joiner node here joins the EPL match data and the Team data to generate the names of teams that play matches in the EPL playing on home ground.

**Jn\_Away:** This node gives the names of teams that play a match in EPL and those which are playing as the Away from the away team.

**Rn\_Home and Rn\_Away:** Here the Columns that are generated from the joiner home node and the joiner away are renamed to avoid the confusion.

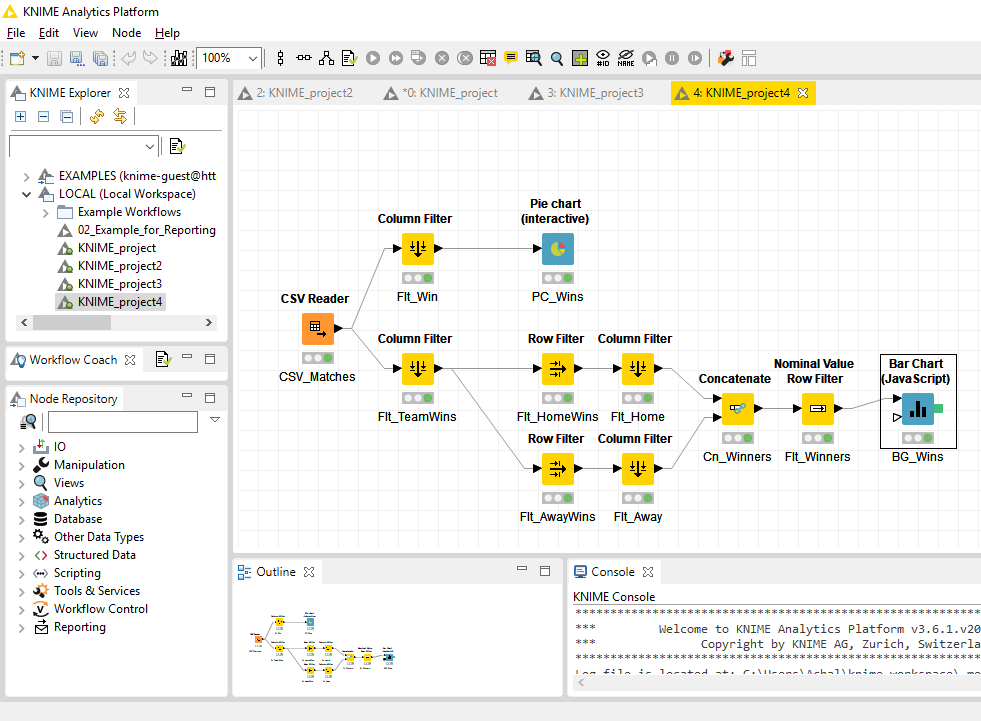
**Jn\_All:** This is the joiner that is used to join the two-renamed column such that there is a proper result in an alphabetic format with teams and there win and loss.

**M\_Winner:** It is the node that consists of the mathematical formula that will result in generating the data of the team that is the winner in the home or away.

**Wt\_CSV\_Match:** This is the final node of this approach where a CSV file is constructed that consist of the result for the user. Few rows of the data are shown below



**Generating Reports:**



**CSV\_Matches:** Read the CSV file generated using the above workflow.

**Flt\_Win:** Get only the column which describes which team won the matching Home or Away.

**PC\_Wins:** Plots a pie chart describing percentage of wins by Home team vs and Away teams and Draws.

**Flt\_TeamWins:** Filter winner and the name of Home and Away team of the match.

**Flt\_HomeWins:** Filter rows where Home team was the winner.

**Flt\_AwayWins:** Filter rows where Away team was the winner.

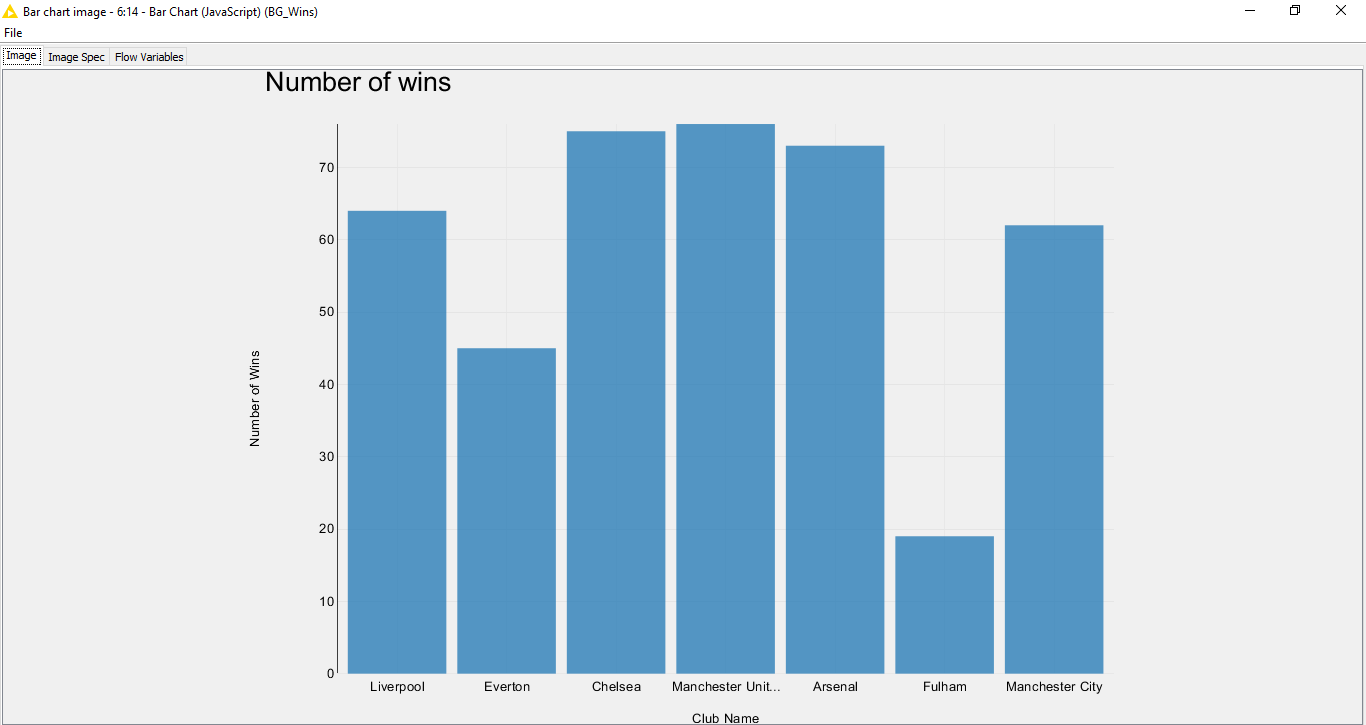
**Flt\_Home:** Filter column with the team name, hence get the names of the winner

**Flt\_Away:** Filter column with the team name, hence get the names of the winner

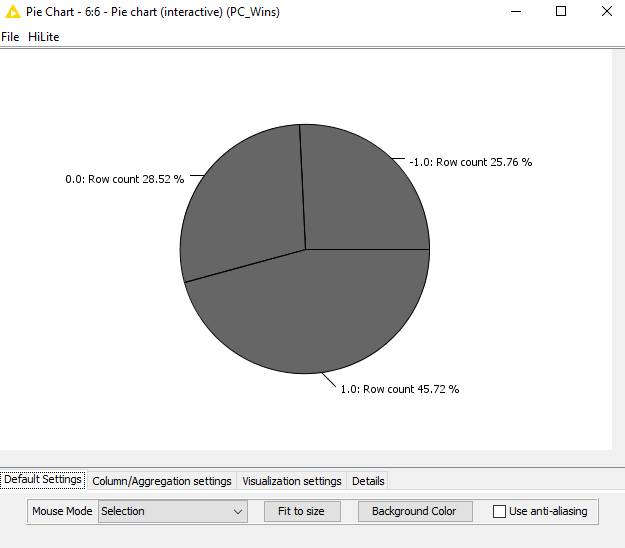
**Cn\_Winners:** Concatenate winners name of every match.

**Flt\_Winners:** Select only rows with 6 major EPL team values. Shown in the graph.

**BG\_Wins:** Plot the bar graph of the count of wins for each team.



The Bar Chart plotted specifies about the number of wins and the varying team have at the home. In the above we can easily visualize the highest rate of a win at home ground of the team Manchester United, comparatively, we can say that Chelsea and Arsenal are competing to win in their home ground. The least wins at home are suffered by Fulham, whereas the Manchester City as the city rival of the Manchester United is doing well with Everton growing casually.



The Pie Chart here is the proof that the win percentage of the home team is way much higher counting about 45.72% whereas the away team win is lesser and outside the league, it is quite similar for away games.

# Opinion on Application

## Experience level required

As per our opinion, a high level of technical expertise is required to work with Knime Analytics. Although not much of coding or programming skills are required but still a sound technical knowledge is required to configure various nodes.

## Data size

Using variously available extensions one can utilize a huge a dataset ranging from very big to medium size datasets. As per our opinion performance of the tool is not as par with some of the other ETL tools like SSIS (SQL Server Integration Services) in processing medium-sized datasets. But KNIME offers a lot of functionalities like big data integration and advanced statistical algorithms which a conventional ETL tool like SSIS can't deliver.

## Visualization capabilities

Knime offers a wide range of visuals starting from basic bar graphs, pie charts to advance visuals like decision trees etc. As per our opinion, KNIME is excellent for basic visuals but lack several capabilities when compared to other available visualization tools like Tableau or Power BI.

# Conclusion

KNIME offers a wide range of functionalities starting from Data Extraction, Data Transformation, Visualization, Data connectivity to various sources and can be utilized as a one-stop shop for a data science project. Although, it lacks various features when compared to a dedicated ETL or Visualization software such as SSIS or Tableau respectively.

It is one of the highest rated Data Analytic tools that provides the facility to make the implementation way much easier. The most convincing part of the platform is the high-end interface that enables us to add extensions of every possible format and the huge repository that make us more dependable on the platform.

As per the Analytical point of view based on the current market and when compared to its competitors, one can conclude that this is one of the most obliging tools that exists in the Analytical word winning the award of the best analytical tool at the spring summit.

# References

Gartner Inc. (2018, February 22). *Magic Quadrant for Data Science and Machine-Learning Platforms*. Retrieved from www.Gartner.com: https://www.gartner.com/doc/3860063?ref=mrktg-srch

KNIME. (2018). *KNIME Analytics Platform*. Retrieved from www.knime.com: https://www.knime.com/textprocessing/documentation/parsing

Wikipedia. (2018, November 11). *Logistic regression*. Retrieved from www.wikipedia.org: https://en.wikipedia.org/wiki/Logistic\_regression