



Open for Innovation

KNIME

[L4-DV] Low Code Data Extraction and Visualization

17 April, 2023



Course objectives

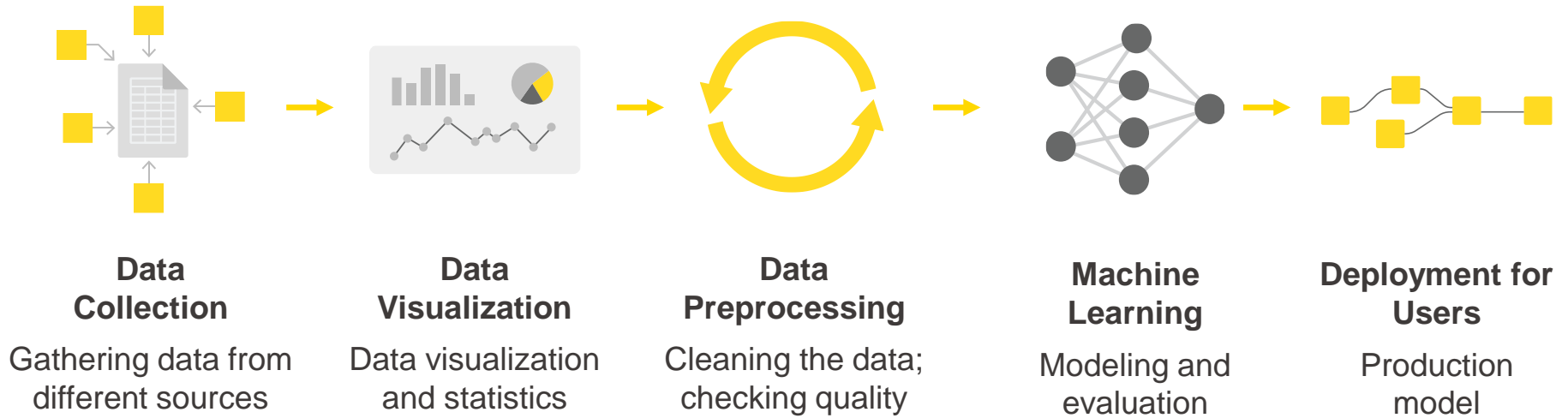
Once you have completed this course, you will be able to:

- Collect data via **REST APIs**, **web text scraping**, and an **interactive** data collection **tool**
- Explore and visualize data
- Extract data and images from **PDF** documents
- Write regular expressions (**regex**)
- Identify and correct errors in data via **outlier detection**
- Build effective and **beautiful visuals**

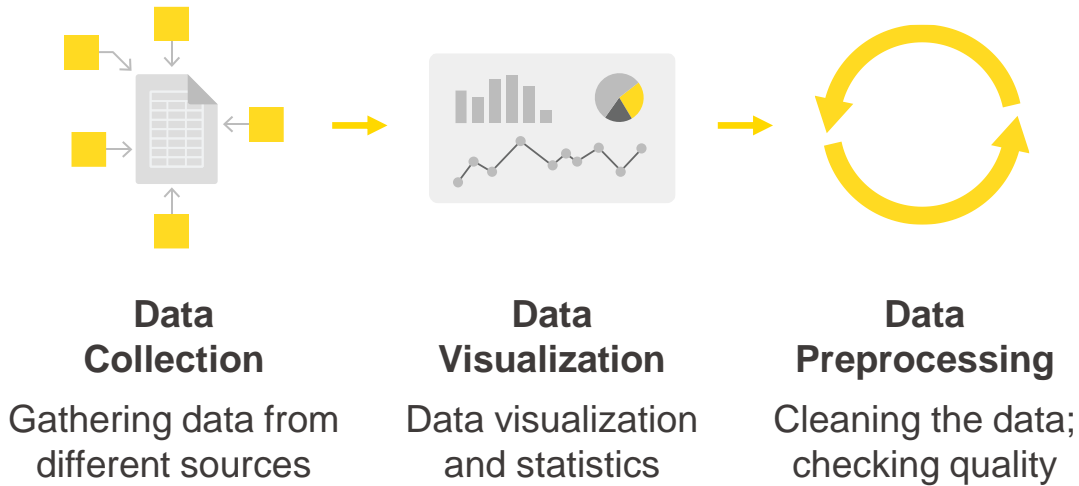
Who is this course for? Data and business analysts

Data Analyst	Data Scientist	Data Engineer
Data acquisition, cleaning, analysis, visualizations, descriptive statistics, reporting, dashboards.	Data pre-processing, training machine learning and statistics algorithms, modeling, predicting.	Integrating various data sources, building data pipelines (ETL, ELT), databases, data lakes, data warehouses, file systems, and/or data mart maintenance, monitoring and testing.

Data analytics within an organization



Data analytics for this course



Structure of the course

This course consists of five sections:

1. Data Collection
2. Data Visualization
3. Data Extraction
4. Data Quality and Visualization Best Practices

Each section offers demo workflows and exercises.

Find all materials at: *tinyurl.com/L4DV-SpringSummit*

How to Download Materials



Hub

Search workflows, nodes and more...

Pricing

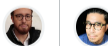
About

Sign in

KNIME Community Hub > hayasaka > Spaces > KNIME Spring Summit Training 2023 > L4-DV Low Code Data Extraction and Visualization

Public space

KNIME Spring Summit Training 2023



Last edited: Apr 5, 2023

3

Copy link

Home > L4-DV Low Code Data Extraction and Visualization



Session_1



Session_2



Session_3



Session_4



Section 1: From links to data

www.knime.com/blog/declutter-four-tips-for-an-efficient-fast-workflow



JavaScript Table View

Open for Innovation
KNIME

Q MENU


KNIME > About > Blog > Declutter - four tips for an efficient, fast workflow

Create

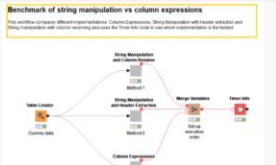
Declutter - four tips for an efficient, fast workflow

September 2, 2019

Recently on social media we asked you for tips on tidying up and improving workflows. Our aim was to find out how you declutter to make your workflows not just superficially neater, but faster, more efficient, and smaller: ultimately an elegant masterpiece! Check out the original posts on [LinkedIn](#) and [Twitter](#).



Before



After

Benchmark of string manipulation vs column expressions
This workflow compares different string manipulation methods (Column Expressions, String Manipulation with regular expressions and String Manipulation with column selection) and shows the time taken to process the data. The results show that Column Expressions are significantly faster than the other methods.

Fig. 1 From confusion to clarity - decluttering your workflow
In this collection of your feedback, we isolate and discuss a few areas worthy of investigation in the post-development phase of your workflow. Inspired by Marie Kondo's approach to tackling things category by category, this article is tidily organized into the following

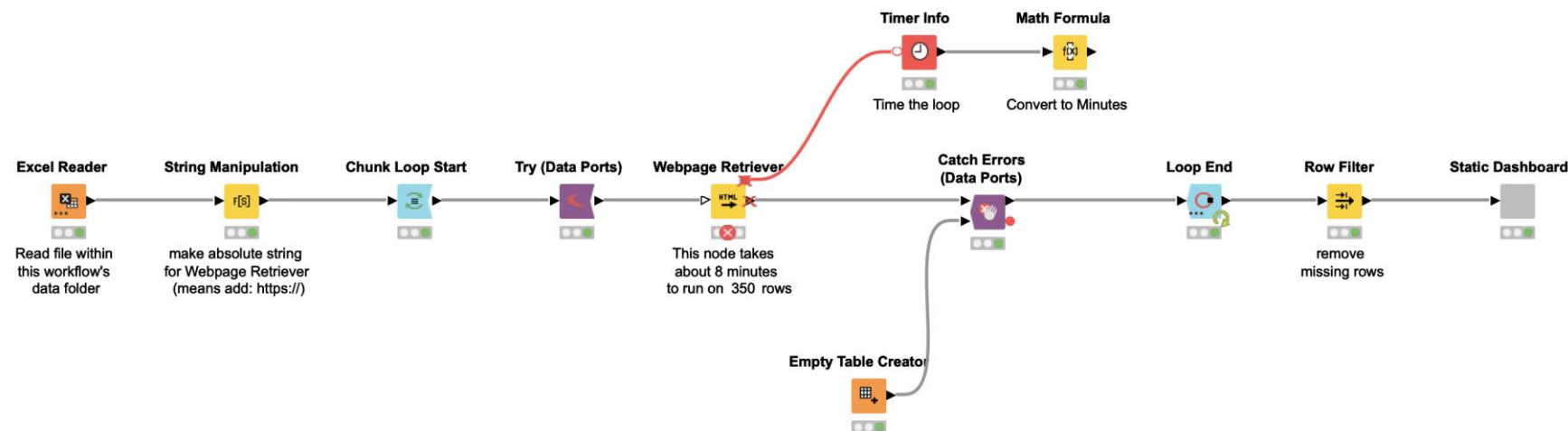
Reset Apply Close

© 2023 KNIME AG. All rights reserved.

8

Open for Innovation
KNIME

Section 1: From links to data



Section 1: Customer satisfaction form

Customer Satisfaction Form (Simple)

Contact Information

First name

Last Name

Email Address

example@example.com

Customer Service

Using a scale of 1 to 5 with 1 being below average and 5 being excellent, please rate how we perform in the following areas by marking the appropriate number

When contacting a member of our staff how would you rate them?

☐ 1

☒ 2

☐ 3

☐ 4

☐ 5

How would you rate the professionalism and courteousness of our staff?

☐ 1

☒ 2

☐ 3

☐ 4

☐ 5

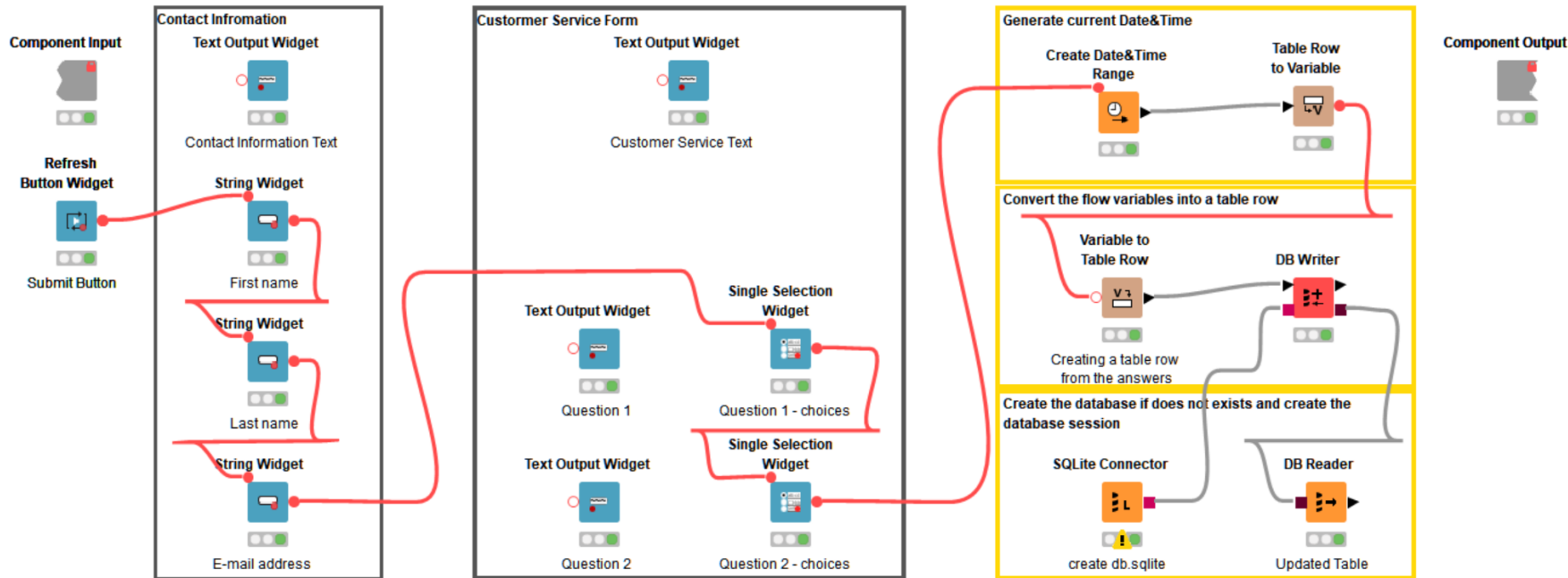
Submit

Reset

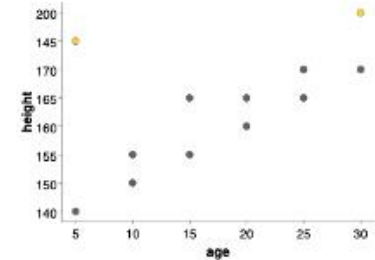
Apply ▾

Close ▾

Section 1: Customer satisfaction form



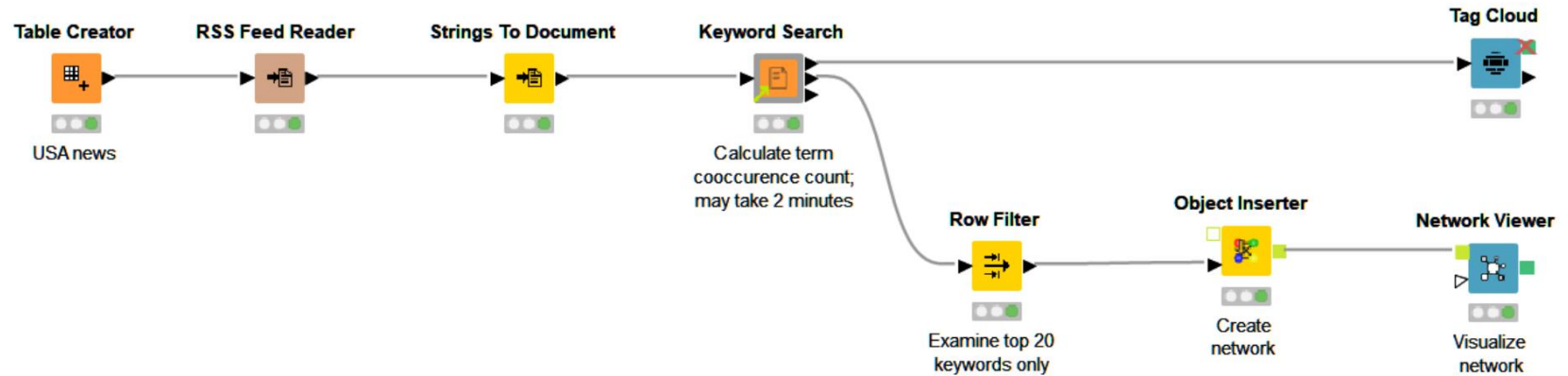
Outliers



Geography



Section 2: Common visualization tasks



Section 3: PDF parsing with Regex

Extracting with Regex

Choose column with name "Text" and any other columns to display.

Excludes

Includes

>

Document

>>

Text

<

<<

Type Regex Below:

[0-9]{4}-[0-9]{1,2}-[0-9]{1,2}

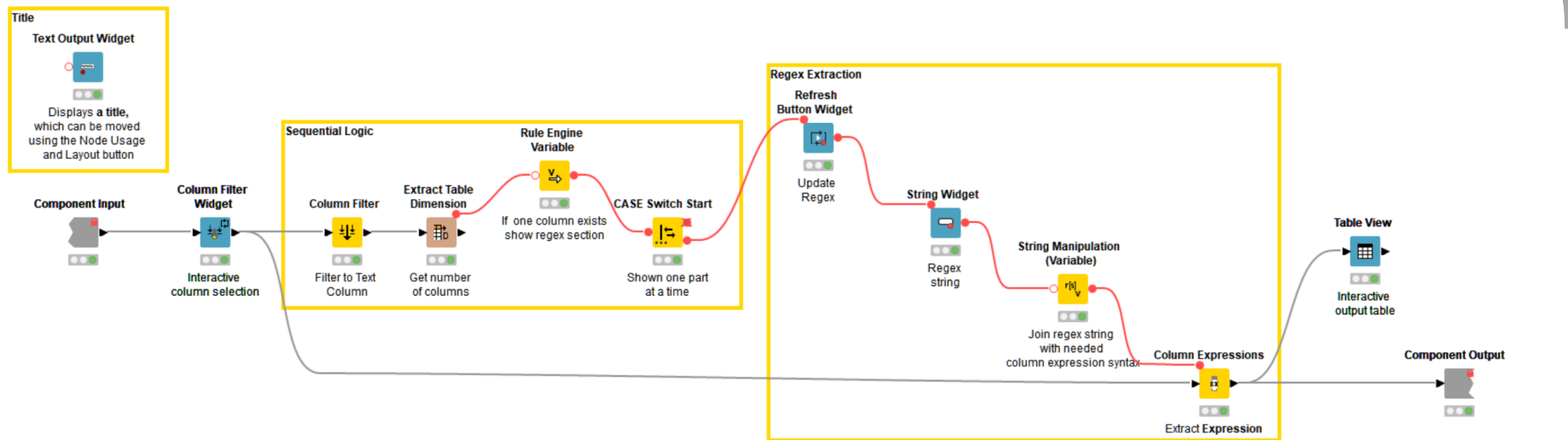
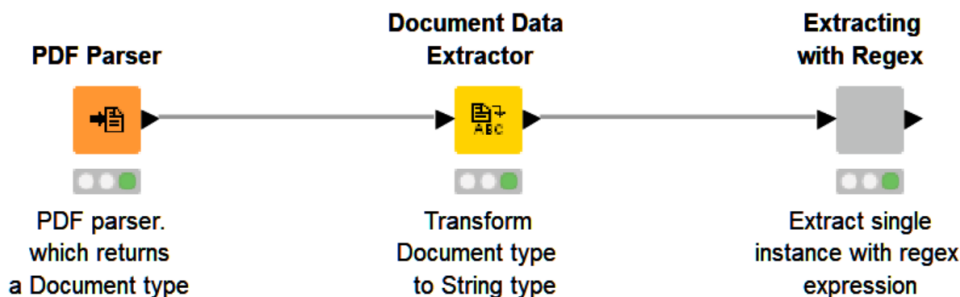
Click to re-execute regex

Show 10 entries

Search:

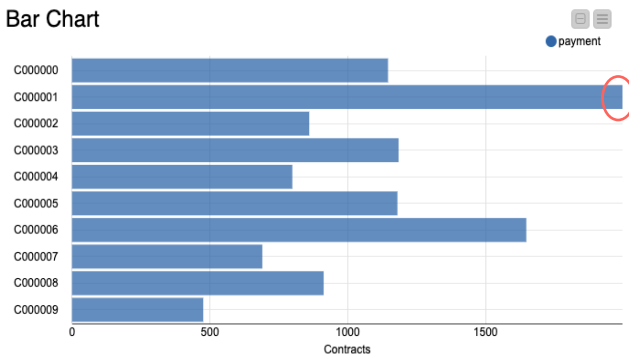
<input type="checkbox"/>	Document	Text	Captured Expression
<input type="checkbox"/>	"C:\Users\daria.liakh\knome-workspace\L4-DV Low Code Data Extraction and Visualization\Session_3\00_Demos\03_Parsing_PDFs_Demo\...\data\pdfs\11012.pdf"	Investment Agreement C000004 KNOW ALL MEN BY THESE PRESENTS : This Contract is entered by and between : Bank A GmbH , whose principal place of address is at Debussysstr . 18,Konstanz , Germany , 78460 , (hereinafter referred to as "	2016-04-05

Section 3: PDF parsing with Regex

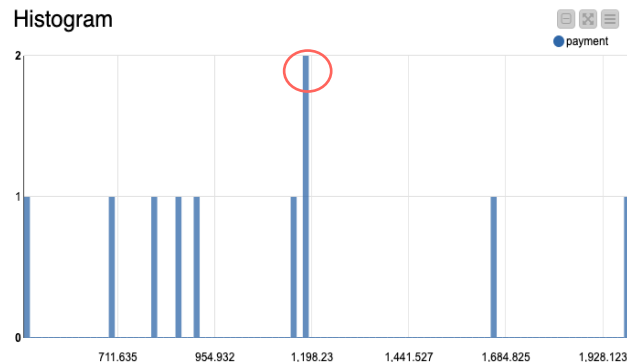


Section 4: Anomaly detection

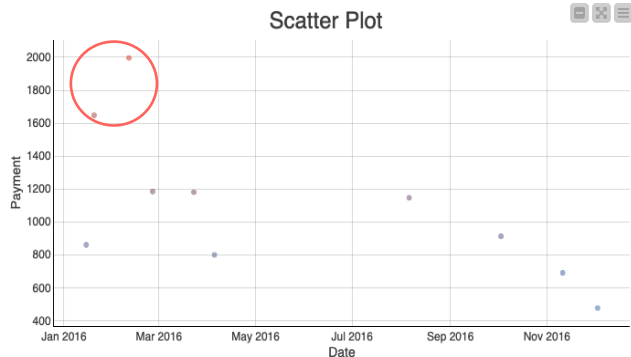
Bar Chart



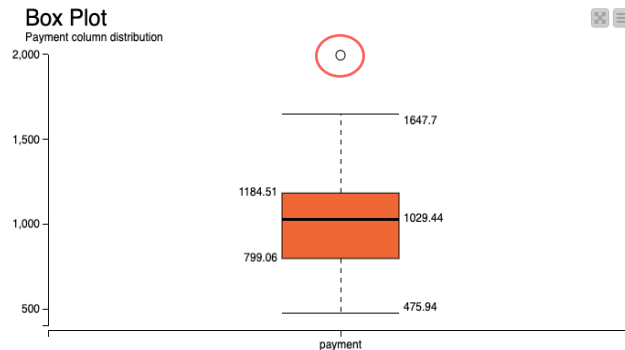
Histogram



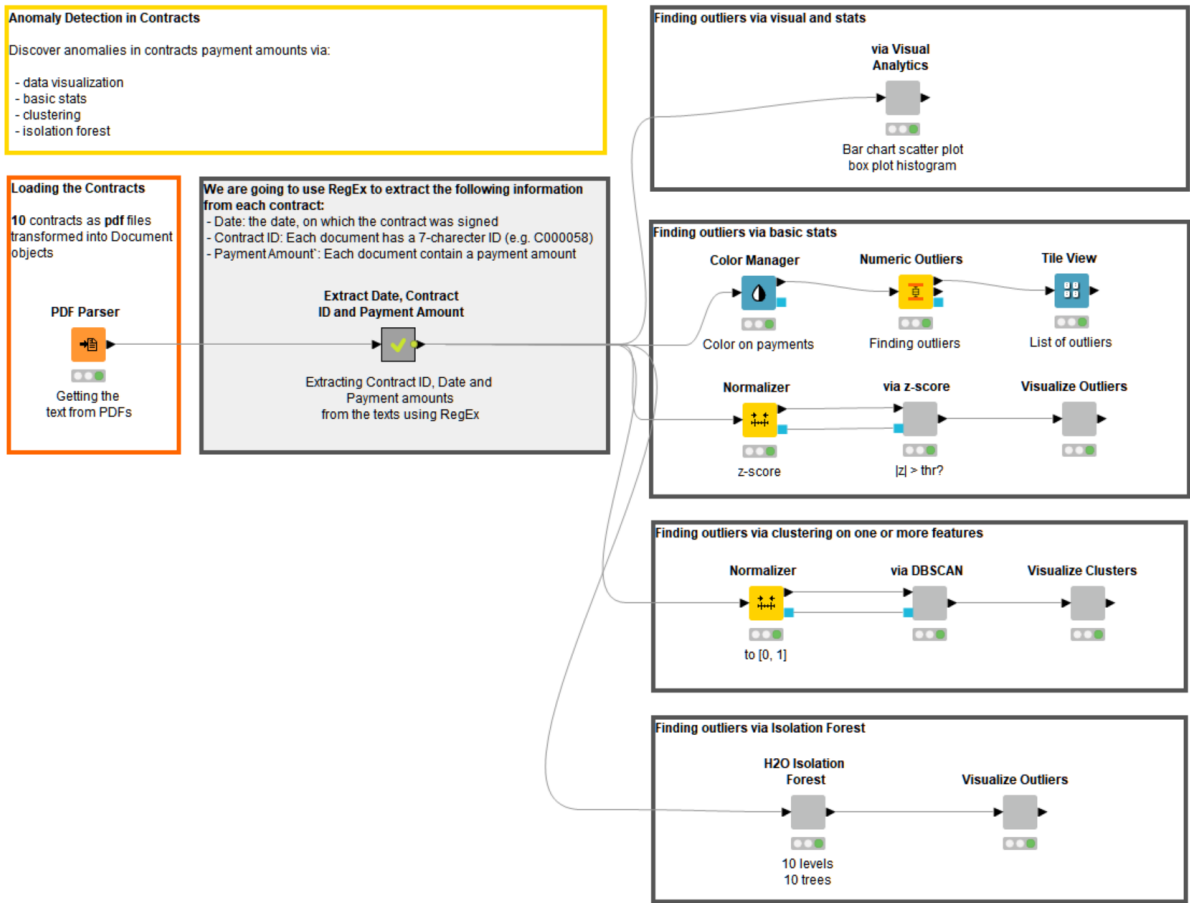
Scatter Plot



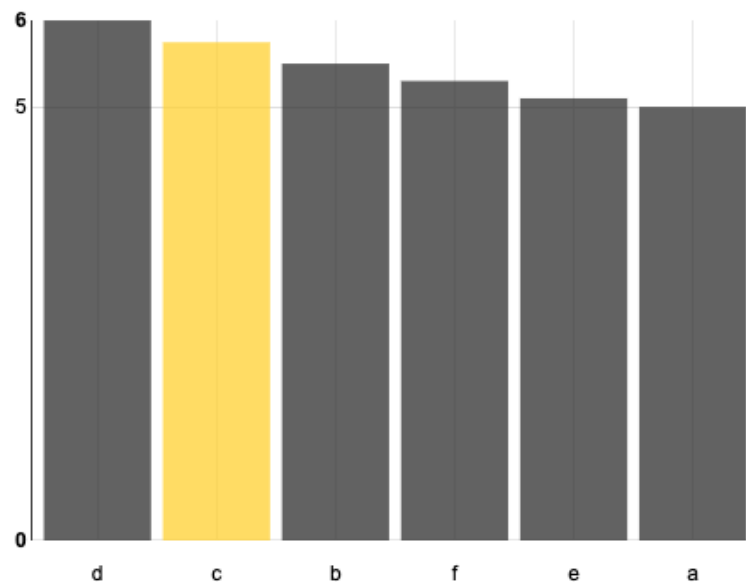
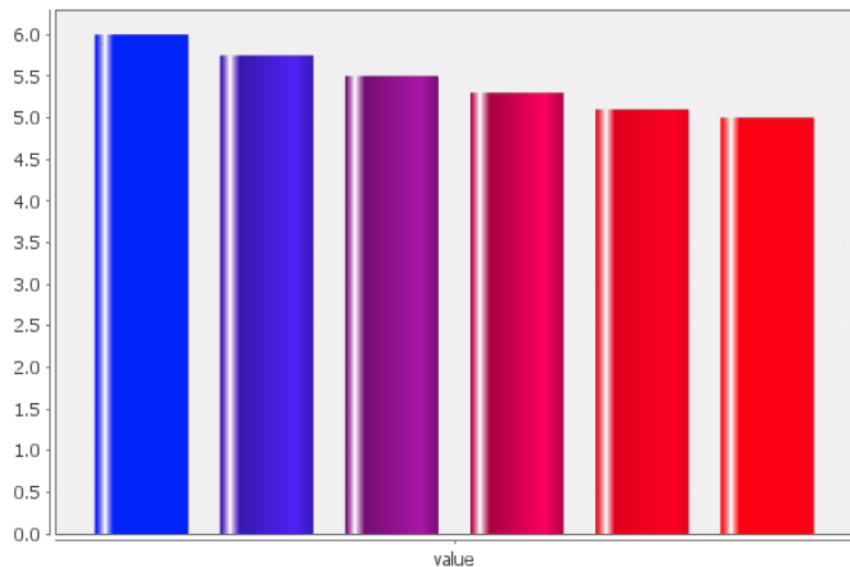
Box Plot

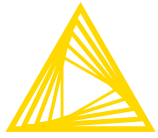


Section 4: Anomaly detection



Section 4: Visualization best practices





Open for Innovation

KNIME

[L4-DV] Low Code Data Extraction and Visualization

Section 1



Data Collection

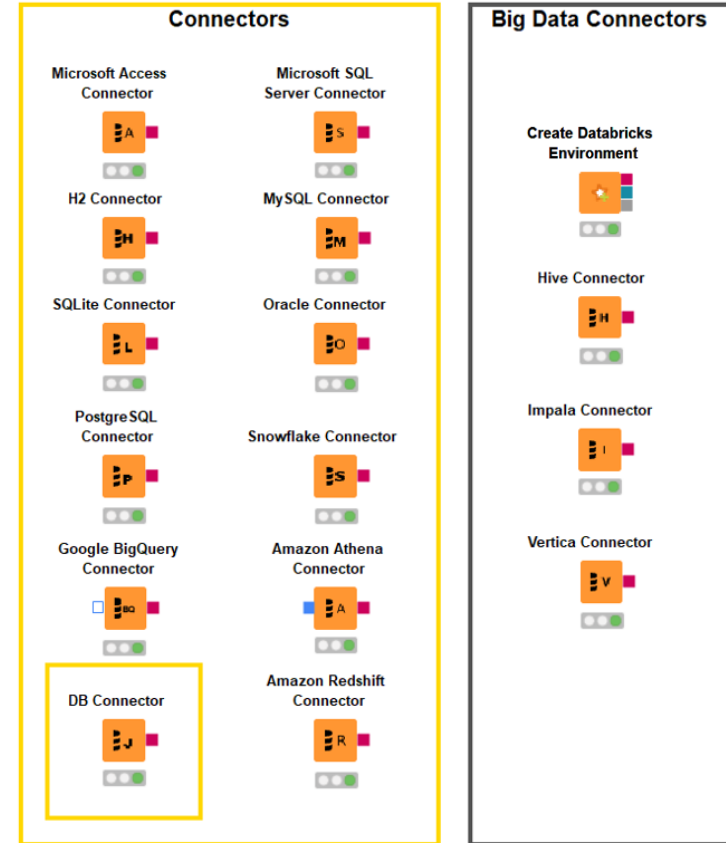
At the end of this section, you will be able to:

1. Recognize data access nodes.
2. Perform webpage retrieval.
3. Differentiate between widgets.
4. Build a data collection tool.



Database nodes

- Dedicated nodes to connect to specific Databases
- Hive and Impala connector part of the KNIME Big Data Connectors extension
- General Database Connector
 - Register new JDBC driver via File -> Preferences -> KNIME -> Databases



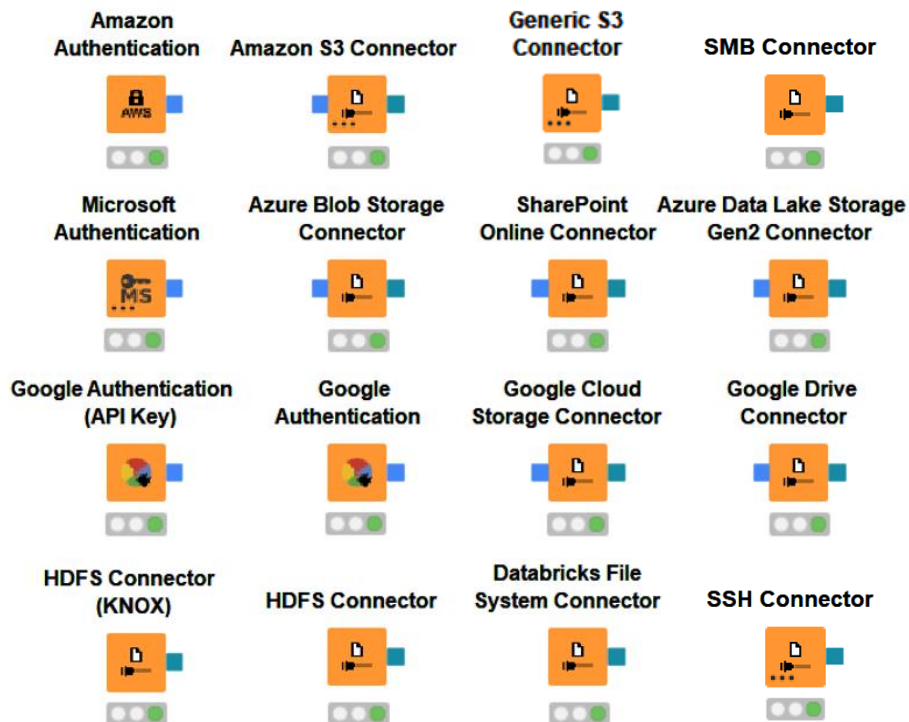
Reader/Writer/Utility nodes

- Reading/writing **tabular, structured, textual**, chemical data, audio, image, and model files
- Reading one or **multiple** files
- Support of integrations: **Python**, R, H2O, **PowerBI**, **Tableau**



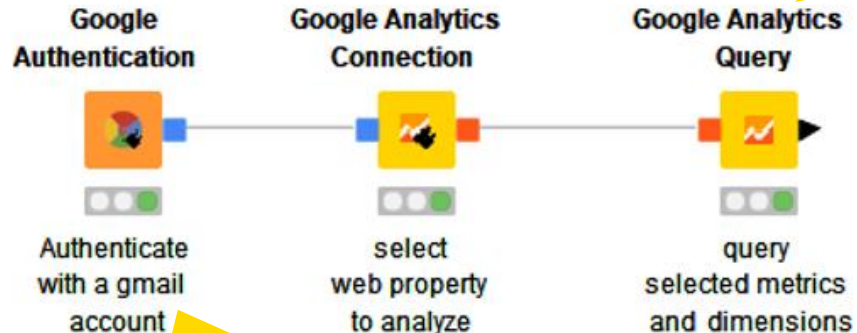
Authentication/Connector nodes

- Connected file systems
 - File systems with external authentication
 - Amazon
 - Microsoft
 - **Google**
 - File systems without external authentication
 - Databricks
 - HDFS, HttpFS
 - SSH, HTTP(S), KNIME Business Hub



Google analytics extension

- Access Google Analytics data
 - Assemble the queries to extract metrics and dimensions of interest
- No need to write API requests



Authenticate via pop-up window (OAuth2)

Select metrics and dimensions, filter, restrict period, sort, etc.

The screenshot shows the "Google Analytics Query" dialog box. The "Dimensions" section includes "pagePath" and "Date". The "Metrics" section includes "uniquePageviews", "pageviews", "bounceRate", "avgTimeOnPage", "users", and "newUsers". The "User Type" dimension is expanded, showing a description: "A boolean, either New Visitor or Returning Visitor, indicating if the users are new or returning." The "Segment" section has "Use predefined segment: All Users" selected. The "Filters" section has "pagePath=~/blog". The "Sort" section has "-uniquePageviews". The "Start date" is "2013-01-01" and the "End date" is "2021-12-31". The "Start index" is "1" and the "Max results" is "10,000". A warning message at the bottom states: "The 'end-date' parameter is controlled by a variable." The "OK", "Apply", "Cancel", and "Help" buttons are at the bottom right.

Google sheets

- Access data stored in Google Services:
 - Read data from Google Sheets
 - Transform in KNIME
 - Modify existing sheets
- No need to write API requests

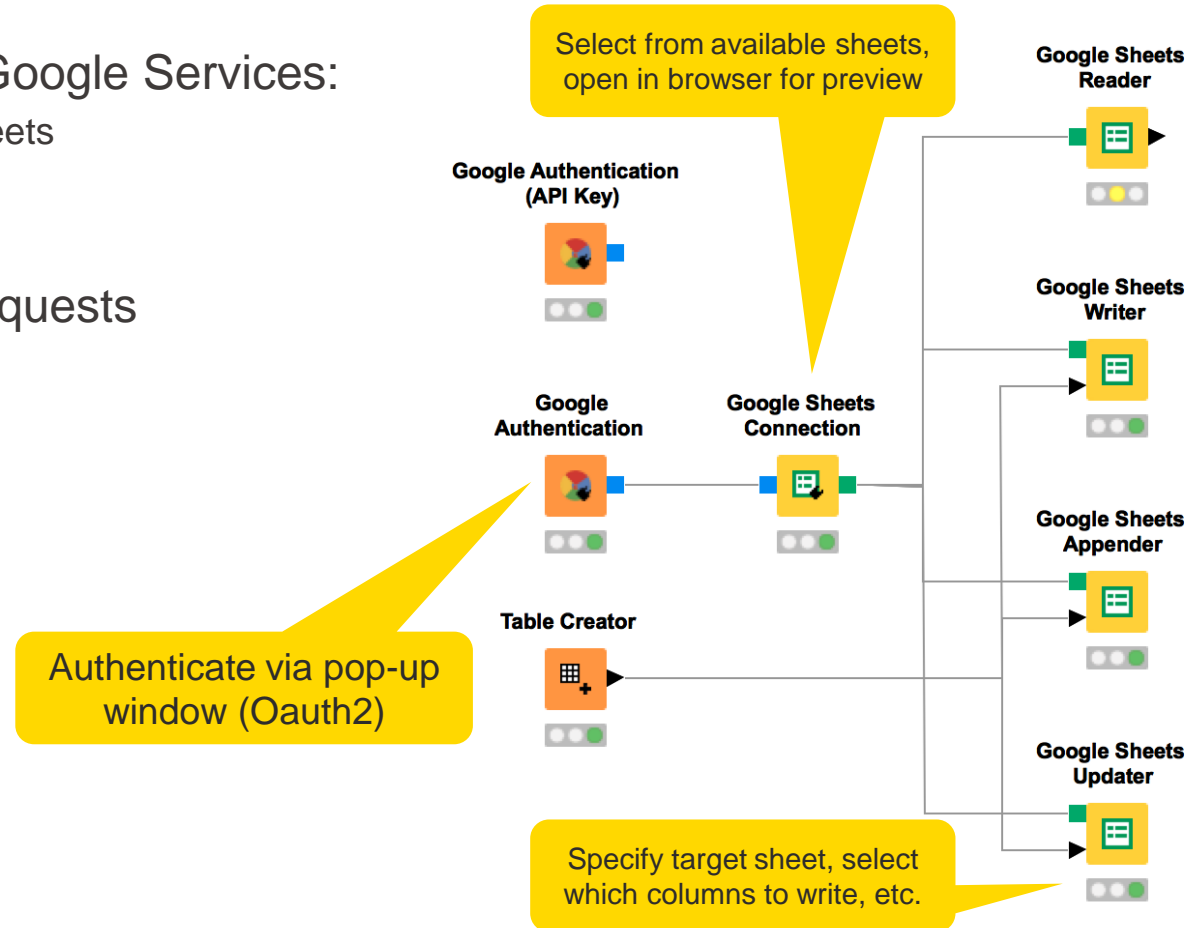
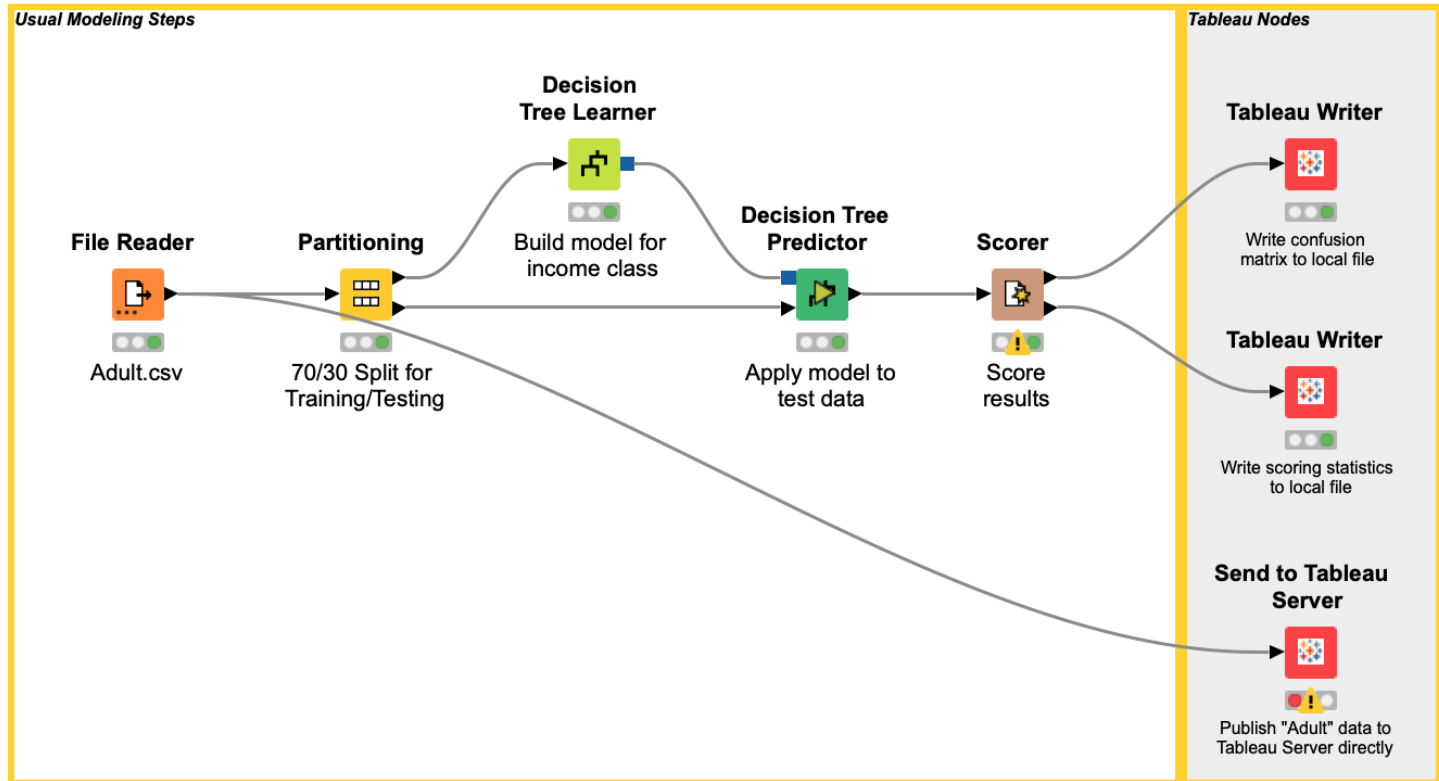


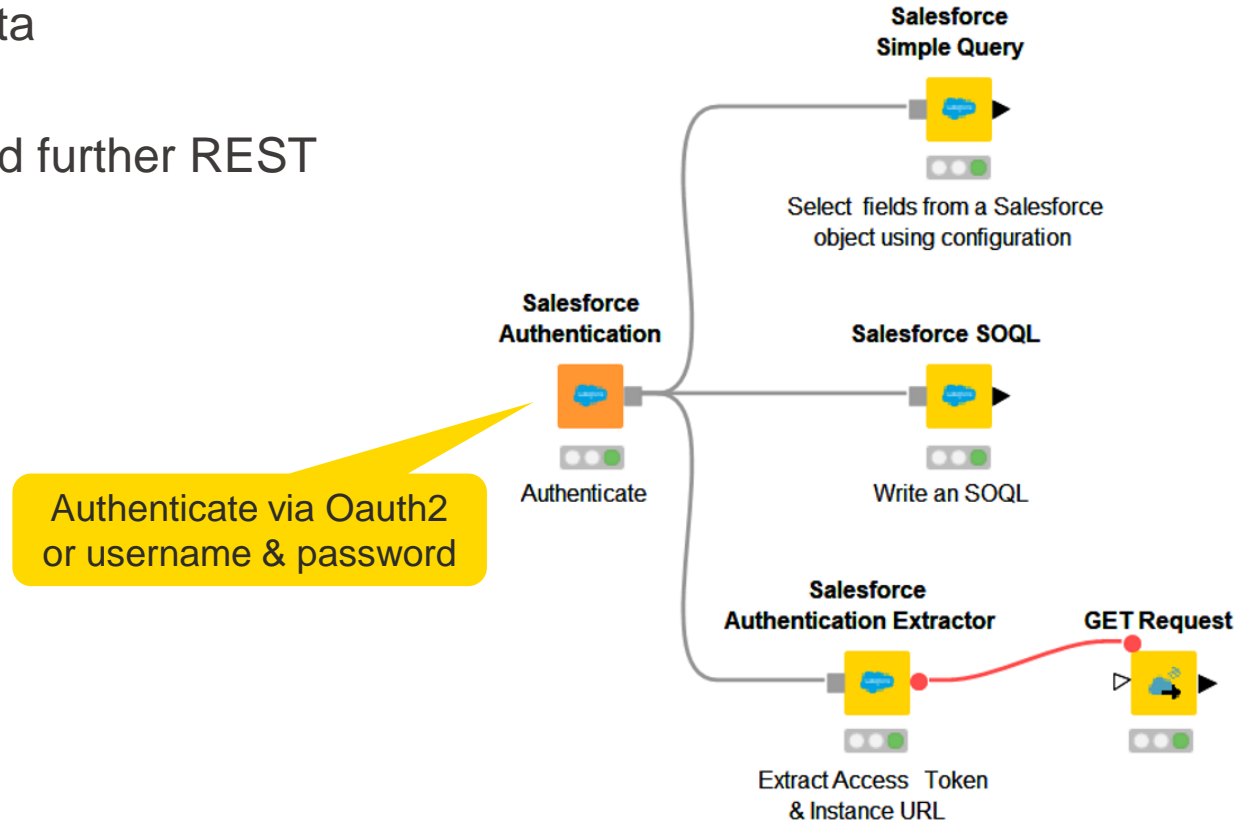
Tableau extension

- Write tableau files/send to Tableau server



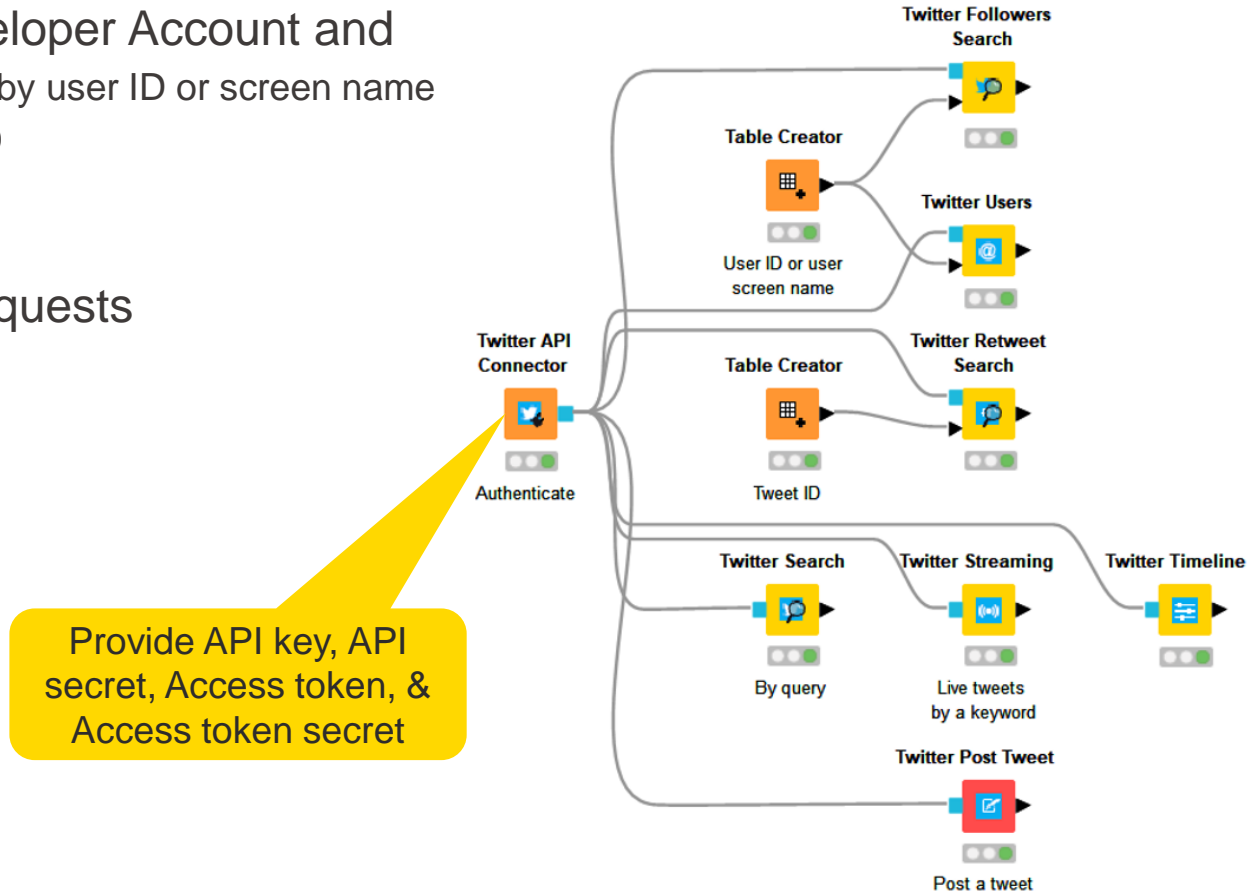
Salesforce extension

- Access Salesforce data
- Write queries and build further REST requests



Twitter extension

- Connect to Twitter Developer Account and
 - Search users and followers by user ID or screen name
 - Search retweets by tweet ID
 - Return the timeline
- No need to write API requests



RESTful API extension

- Execute **R**epresentational **S**tate **T**ransfer commands
 - RESTful APIs are Web Service APIs that adhere to the [REST constraints](#)
 - One the most predominant **architectures for obtaining and managing data** across applications
- Existing KNIME nodes



RESTful web services / API

GET Request



Enter URL, or
use from column

Provide authentication
if necessary

Add delay between
individual requests

Dialog - 0:1 - GET Request

File

Connection Settings Authentication Error Handling Request Headers Response Headers Flow Variables Job Manager Selection Memory Policy

☒ URL:

☐ URL column:

☐ Delay (ms):

Concurrency:

SSL

☐ Ignore hostname mismatches

☐ Trust all certificates

☒ Follow redirects

☒ Send large data in chunks

Timeout (s)

Body column:

OK Apply Cancel ?

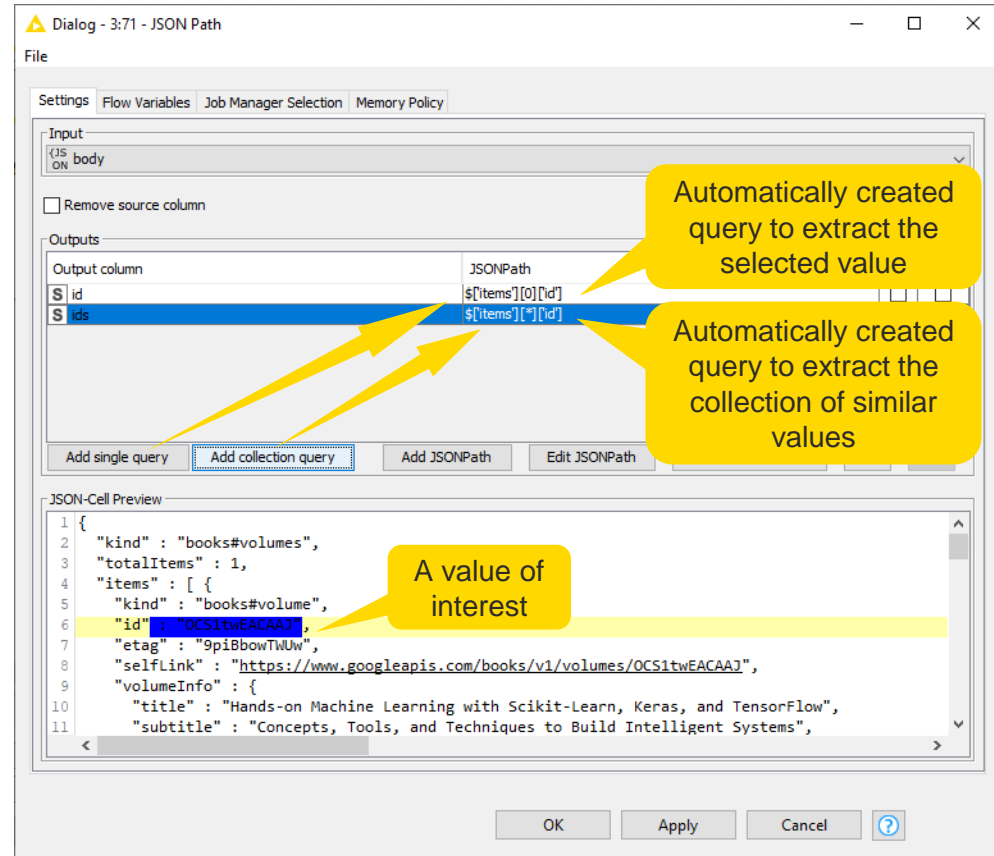
<https://www.youtube.com/watch?v=HeE7tEgUdq0>

<https://www.knime.com/blog/a-restful-way-to-find-and-retrieve-data>

<https://www.knime.com/blog/OSM-meets-CSV-file-and-Google-API>

JSON and XML parsing tips

- Use the JSON Path node to query the JSON file and extract parameters
- Editor window simplifies construction of JSON queries by auto-generating them
 - Select the value of interest in the JSON-Cell Preview and use the buttons to automatically add a query to extract this single value or a collection of similar values
 - OR write a JSONPath query manually
- Analogously with Xpath node for XML



KNIME knowledge check 1

- All the extensions mentioned today (Salesforce, Google, Twitter, etc.) come automatically installed in the KNIME Analytics Platform.
 - True
 - False

01: Get Request node demo

Live
Demo

Course exercises

- All exercises are available [here](https://tinyurl.com/L4DV-SpringSummit): tinyurl.com/L4DV-SpringSummit

Education

Last edited: 25 Jun 2020

Home > Courses

- ←
- 📁 L1-DS KNIME Analytics Platform for Data Scientists - Basics
- 📁 L1-DW KNIME Analytics Platform for Data Wranglers - Basics
- 📁 L1-LS KNIME Analytics Platform for Data Scientists - Life Sciences - Basics
- 📁 L2-DS KNIME Analytics Platform for Data Scientists - Advanced
- 📁 L2-DW KNIME Analytics Platform for Data Wranglers - Advanced
- 📁 L2-LS KNIME Analytics Platform for Data Scientists - Life Sciences - Advanced
- 📁 L3-PC KNIME Server Course - Life Sciences - Productionizing and Collaboration
- 📁 L3-PC KNIME Server Course - Productionizing and Collaboration
- 📁 L4-BD Introduction to Big Data with KNIME Analytics Platform
- 📁 L4-CA Machine Learning for Chemical Applications
- 📁 L4-CH Introduction to Working with Chemical Data
- 📁 L4-DE Best Practices for Data Engineering
- 📁 L4-DL Introduction to Deep Learning
- 📁 L4-DV Low Code Data Extraction and Visualization
- 📁 L4-ML Introduction to Machine Learning Algorithms
- 📁 L4-TP Introduction to Text Processing

Public space

KNIME Spring Summit Training 2023

Last edited: Apr 5, 2023

Home > L4-DV Low Code Data Extraction and Visualization



📁 Session_1

📁 Session_2

📁 Session_3

📁 Session_4

📁 data

📄 Agenda-L4-DV.pdf

📄 Extension_Requiring_Nodes

📄 Slides-L4-DV.pdf

Download material

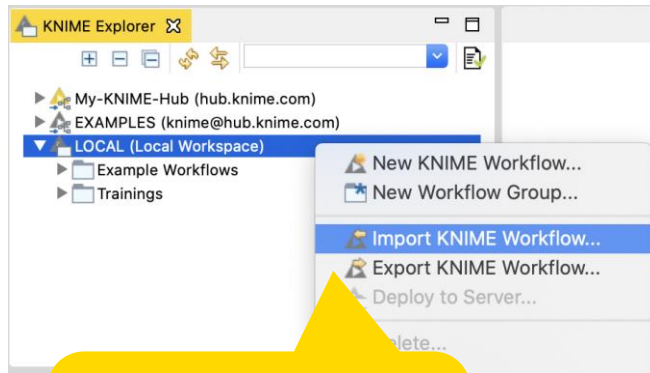
- Download the course material from the KNIME Community Hub [here](https://tinyurl.com/L4DV-SpringSummit):
tinyurl.com/L4DV-SpringSummit



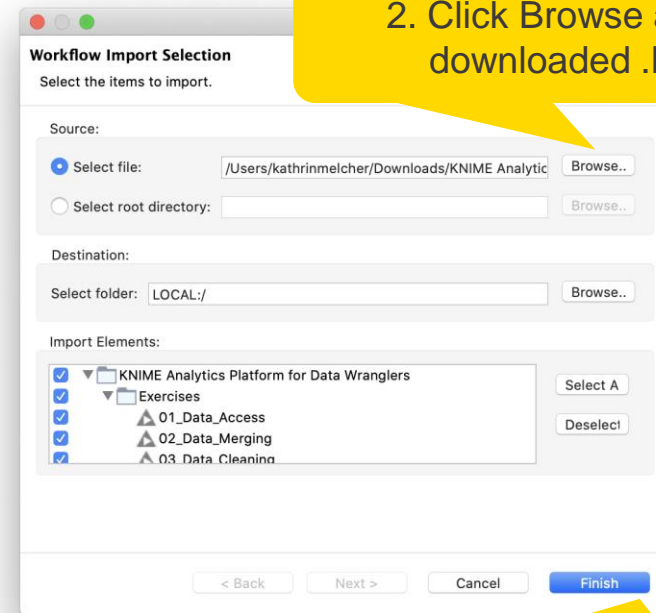
Note: You must be logged into the KNIME Hub to see that download icon for all course material

Import material

- Import the course material to KNIME Analytics Platform



1. Right click LOCAL and select Import KNIME Workflow....



2. Click Browse and select downloaded .knar file

3. Click Finish

Exercises: Section 1

1. Get Request Exercises

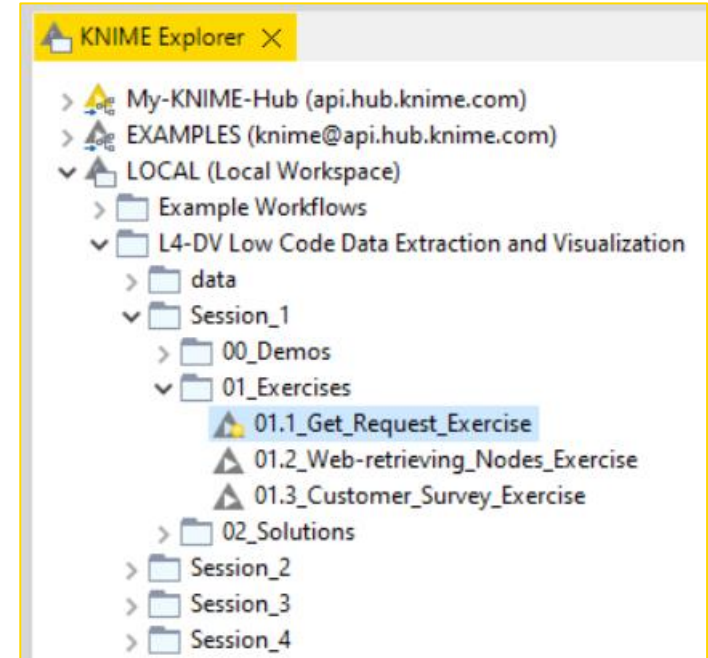
Given a table with retrieve information with the Get Request node.

2. From Links to Data Exercise

Given a list of URLs, acquire the text from each link.

3. Customer Survey Exercise

Create your own customer feedback form.



Data Collection

At the end of this section, you will be able to:

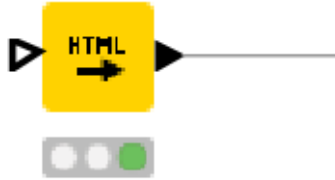
1. Recognize data access nodes.
2. Perform webpage retrieval.
3. Differentiate between widgets.
4. Build a data collection tool.



Webpage Retriever node

- Get *whole* webpage when no (useful) API exists

Webpage Retriever



```
[S] webpage_text
<!doctype html>
<html lang="en" dir="ltr" prefix="content: http://purl.org/rss/1....
<head>
<meta charset="utf-8">
<script type="text/javascript">(window.NREUM||(NREUM={})).in...
</script>
<!doctype html>
<html lang="en" dir="ltr" prefix="content: http://purl.org/rss/1....
```

Dialog - 0:8 - Webpage Retriever

General Settings | Authentication | Error Handling

Connection Settings

☐ URL:

☒ URL column:

☒ Delay (ms):

Concurrency:

SSL

☐ Ignore hostname mismatches

☐ Trust all certificates

☒ Follow redirects

☐ Send large data in chunks

Timeout (s)

Output Settings

Output column name

☐ Output as XML

☒ Replace relative URLs with absolute URLs

☐ Extract cookies

Cookie column name

OK Apply Cancel ?

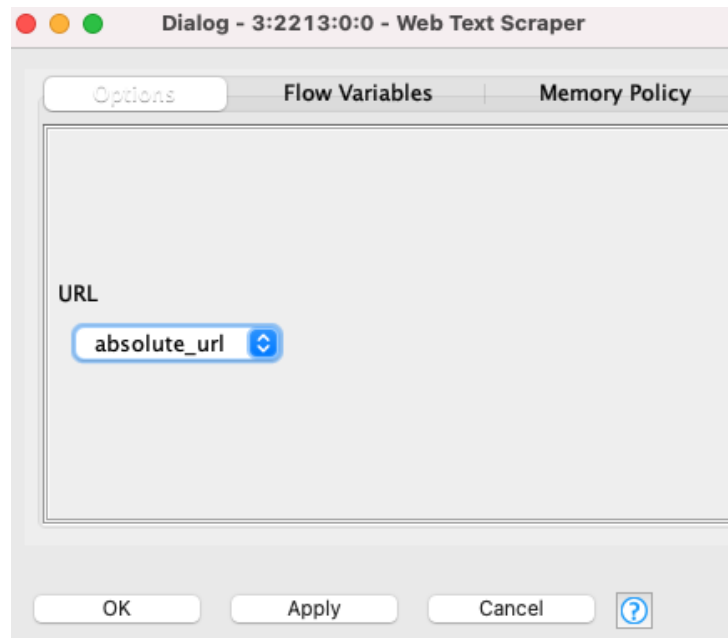
Web Text Scraper Verified Component

- Get *only* text from webpage

Web Text Scraper

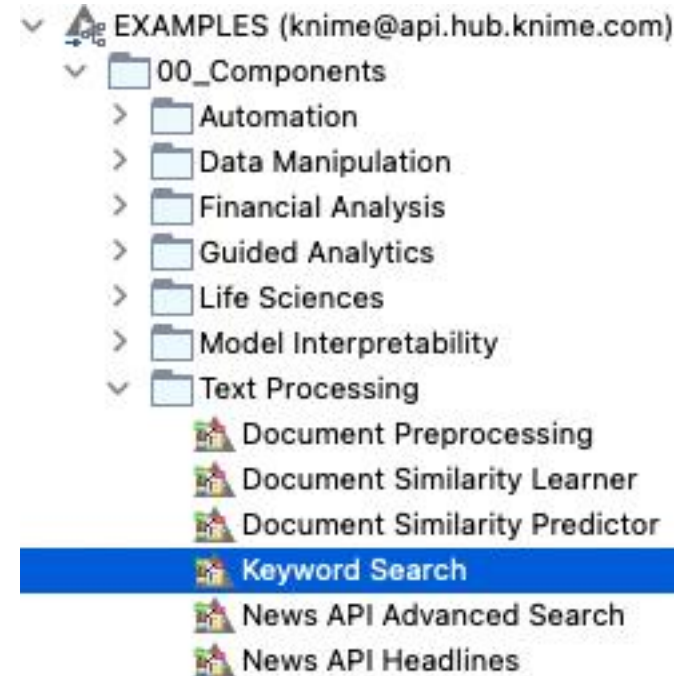
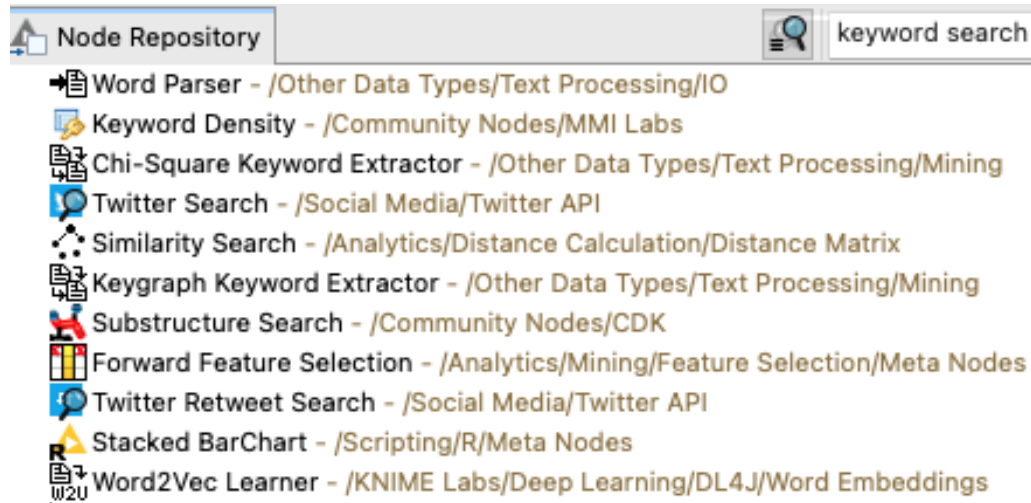


S text
Guided Labeling Blog Series – Episode 6: Comparing Active L...
Create
Guided Labeling Blog Series – Episode 6: Comparing Active L...
July 27, 2020 — by Paolo Tamagnini & ...
Recap: In the last episode we made an analogy with a numbe...
Let's pick up where we left off.
You can blend friends' movies opinions in a single model, but...
Weak Supervision instead of Active Learning
The key feature that differentiates active learning from weak ...
Unique vs Flexible



Where can I find Verified Components?

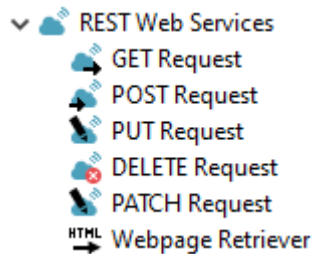
- They are not in the Node Repository, because it's a Verified Component



[See more verified components here](#)

Error handling on REST Web Service Nodes

- Many errors might occur
 - **client**-side errors, **server**-side errors or **rate**-limiting conditions
- Individually configurable
- Works with all the nodes in KNIME REST Client Extension



Dialog - 3:1 - GET Request

File

Response Headers | Flow Variables | Job Manager Selection | Memory Policy
Connection Settings | Authentication | **Error Handling** | Request Headers

Connection problems (timeouts, certificate errors, ...)

☐ Fail node execution
☒ Output missing value

Server-side errors (HTTP 5XX)

☐ Fail node execution
☒ Output missing value

☒ Retry on error

Number of retries: 3

Retry delay [s]: 1

Client-side errors (HTTP 4XX)

☐ Fail node execution
☒ Output missing value

Rate-limiting error (HTTP 429)

☒ Pause execution (and retry)

Pause execution [s]: 60

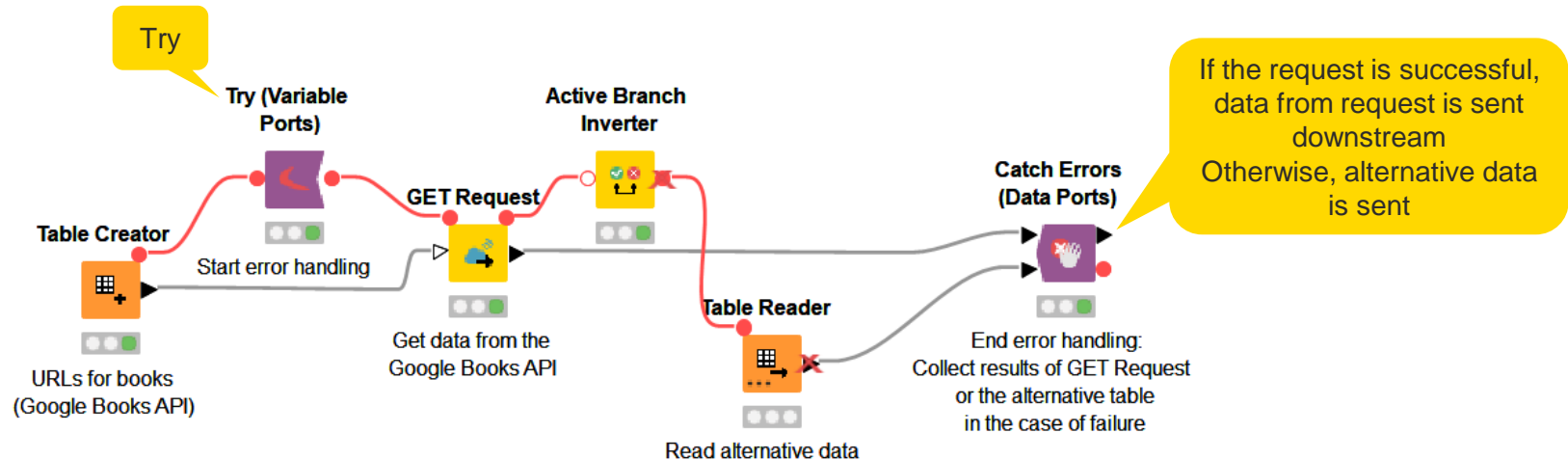
Error reporting

☐ Output additional column with error cause

OK Apply Cancel ?

Custom error handling

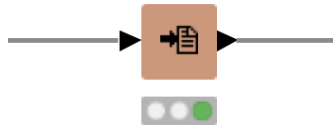
- Try & Catch nodes handle errors
- **Prevent whole workflow failure** even when a single node fails in between



RSS Feed Reader: Another way to read web data

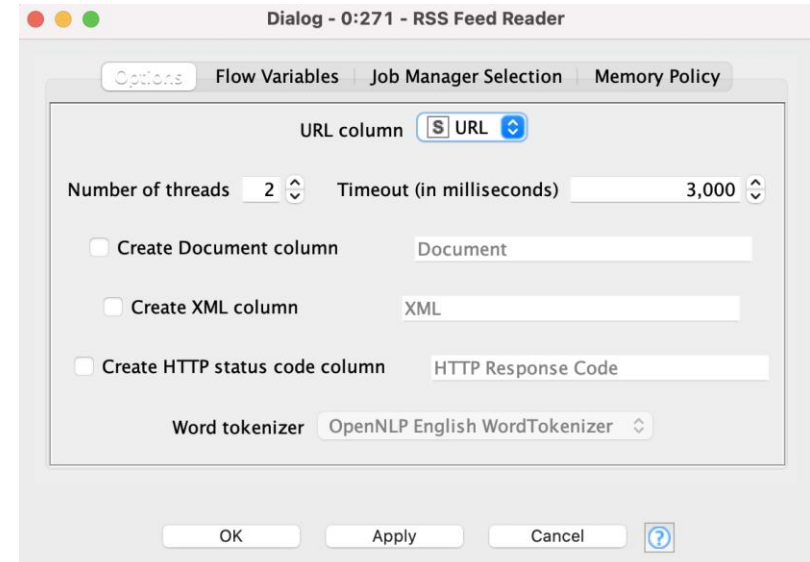
- Download up-to-date information from a particular website such as a news website

RSS Feed Reader



\$ Title	\$ Description
Worry and fear as US faces baby fo...	With stocks running low across the country,
Camille Vasquez: Johnny Depp's la...	The young lawyer has caught the internet's
Oklahoma passes bill banning most...	The state's ban, its third in recent months,

- What's RSS?
 - RSS (Really Simple Syndication) is a **content** distribution method that allows access to **updates** to websites in a standardized format.



02: From links to data demo

Live
Demo

Exercises: Section 1

1. Get Request Exercises

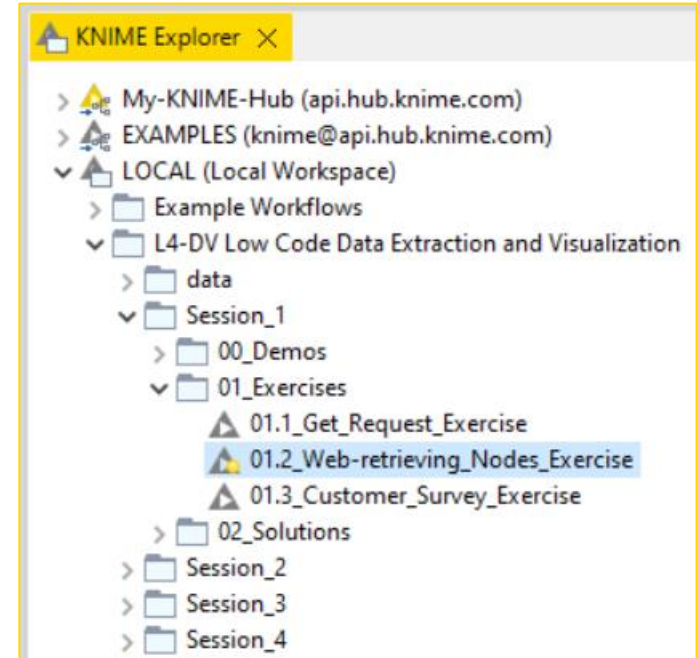
Given a table with retrieve information with the Get Request node.

2. From Links to Data Exercise

Given a list of URLs, acquire the text from each link.

3. Customer Survey Exercise

Create your own customer feedback form.



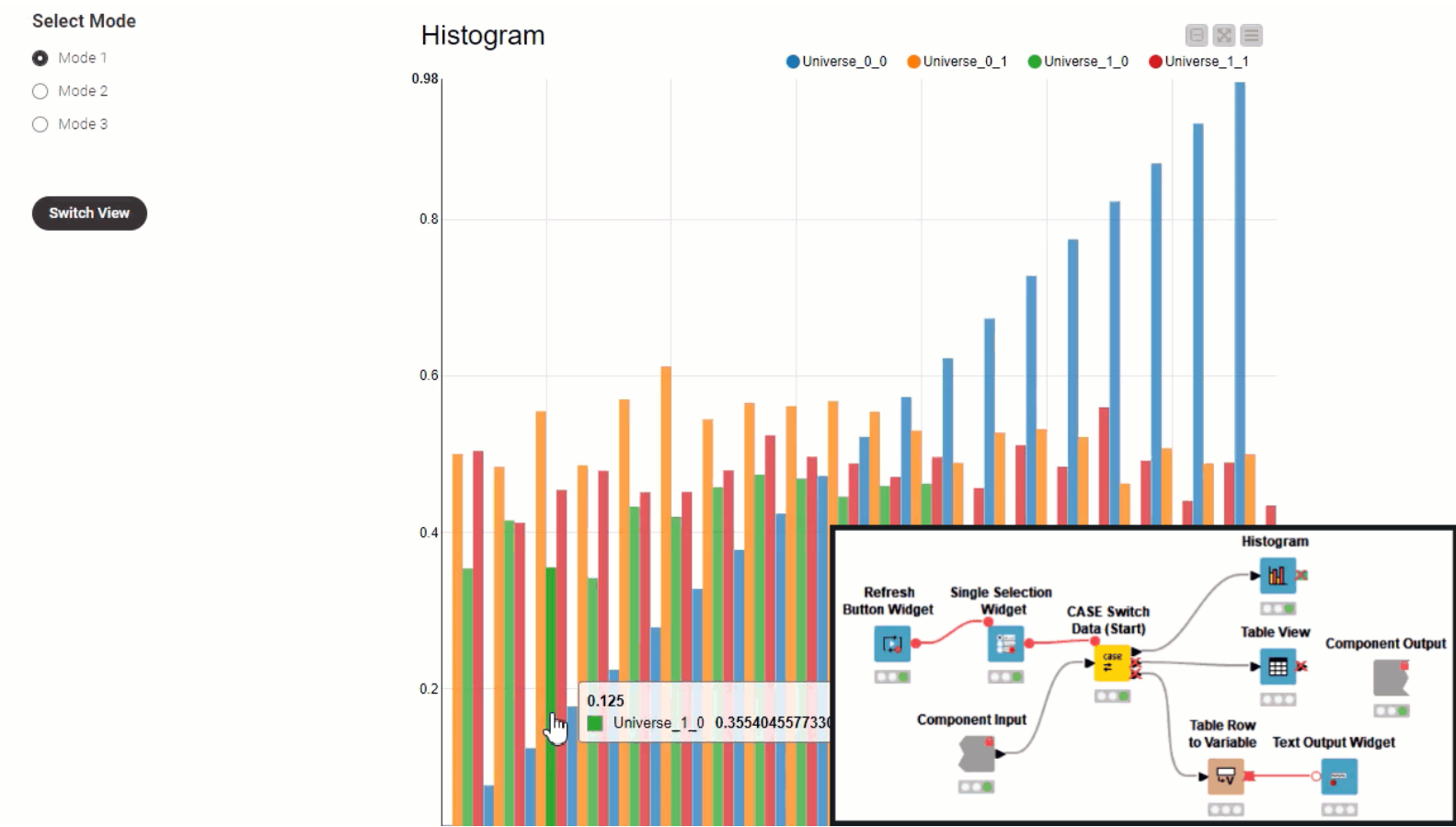
Data Collection

At the end of this section, you will be able to:

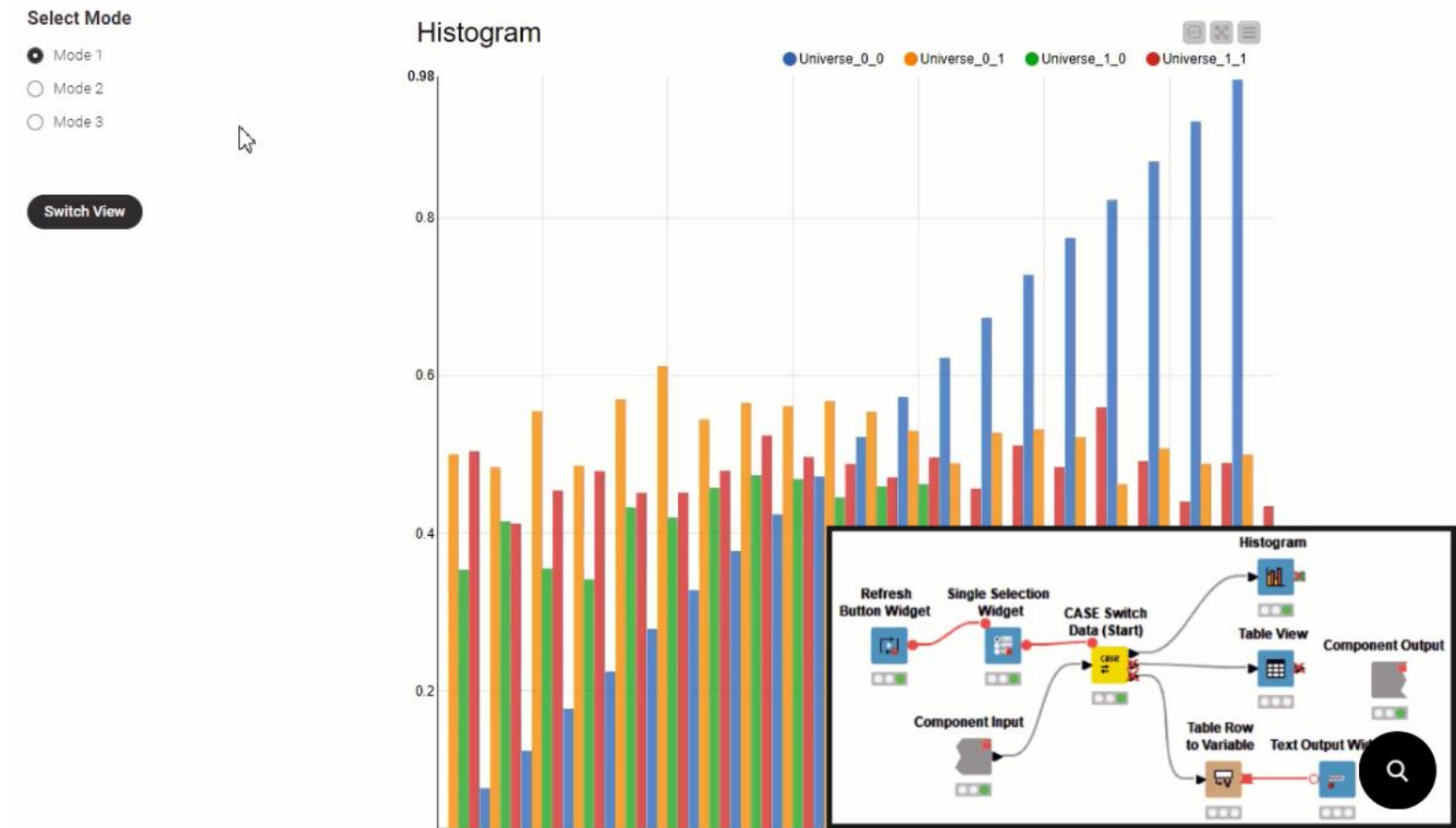
1. Recognize data access nodes.
2. Perform webpage retrieval.
3. Differentiate between widgets.
4. Build a data collection tool.



What's a widget? An interaction enabler!

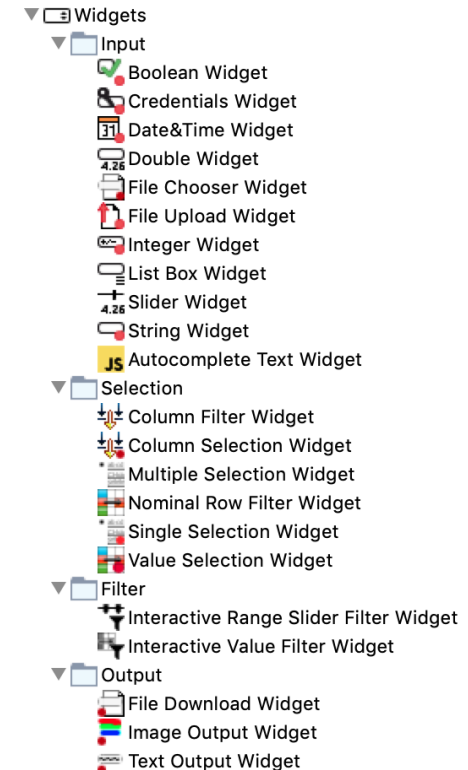
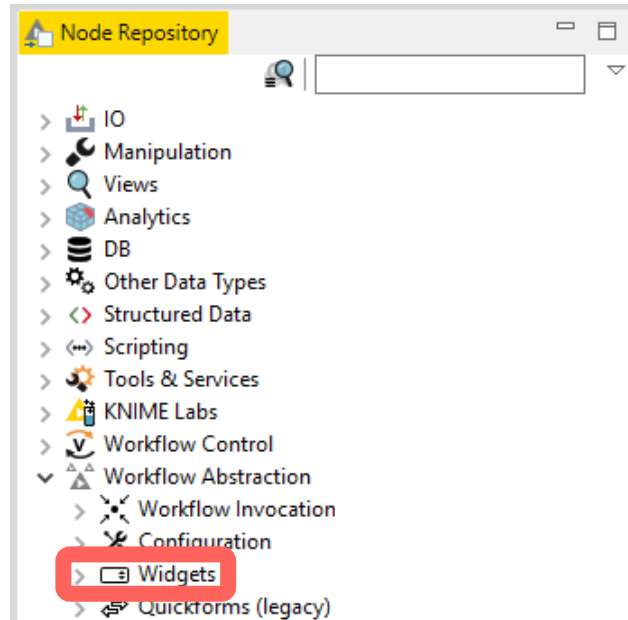


What's a widget? An interaction enabler!



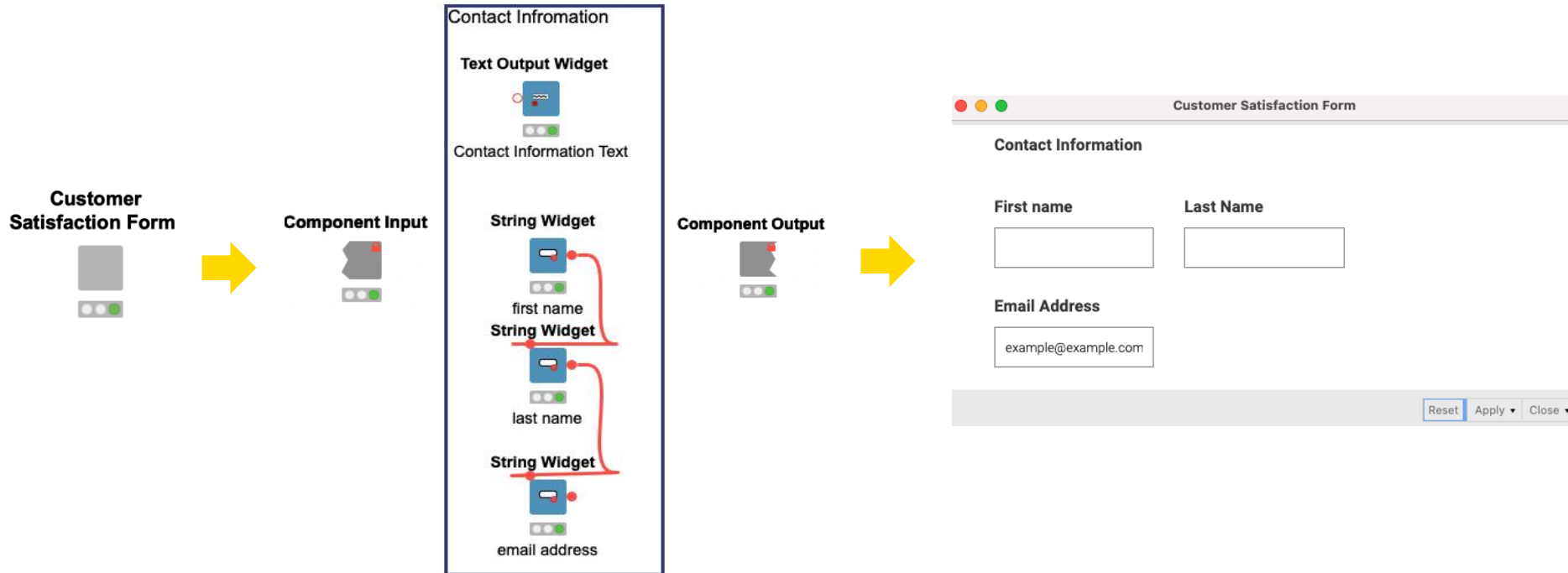
Widget nodes within a component

- Enable interaction by **selecting, filtering, and entering values** in the interactive view.



Widgets within a component create interactivity

- Add custom interactions in the component's interactive view.

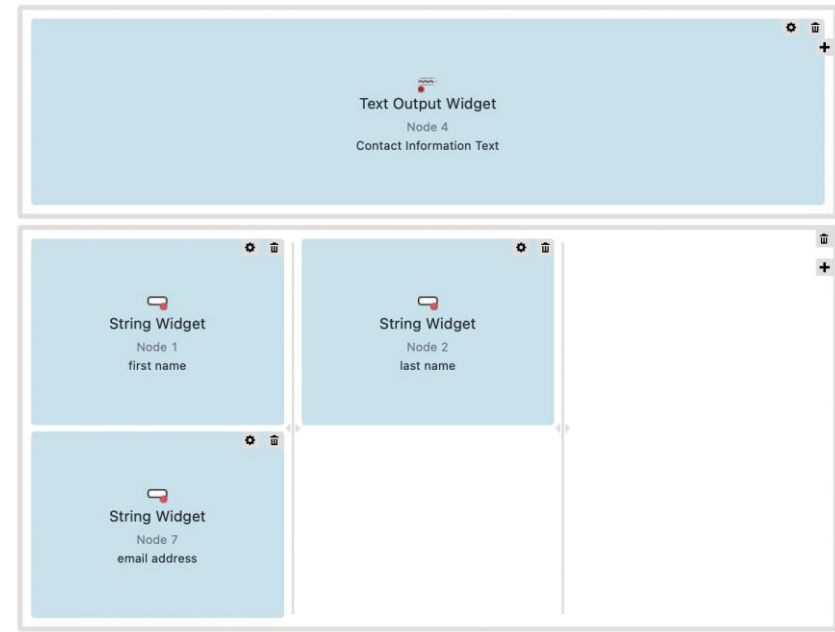


Component Node Usage and Layout button



- Layout editor of the composite view:

1. Node Usage
- 2. Visual Layout**
3. Basic Layout
4. Advanced Layout

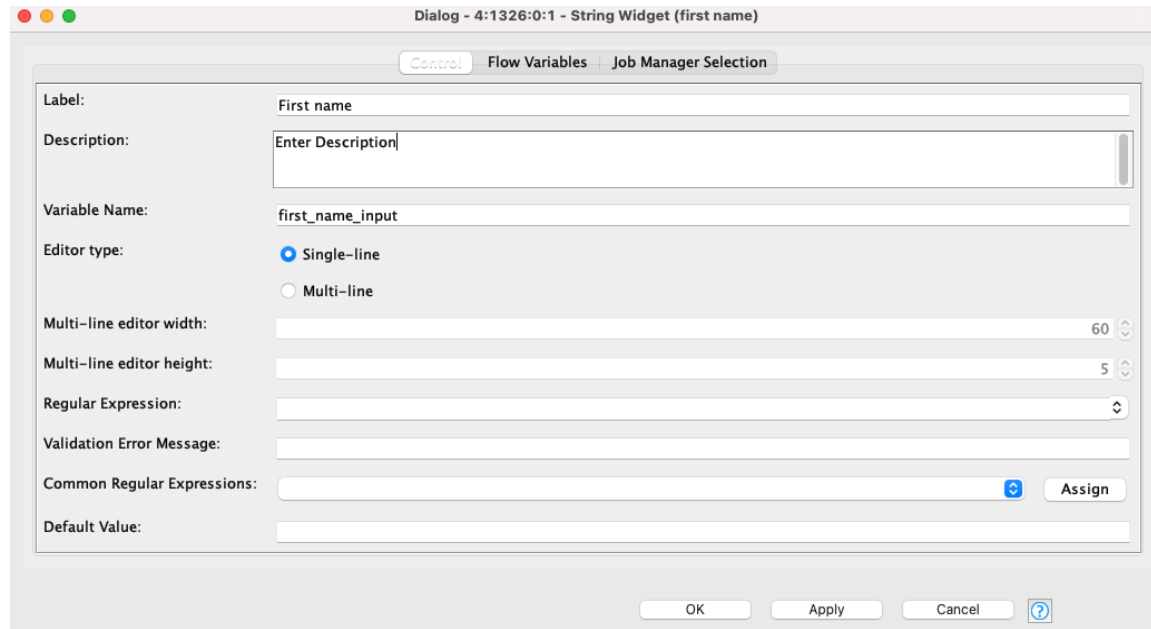
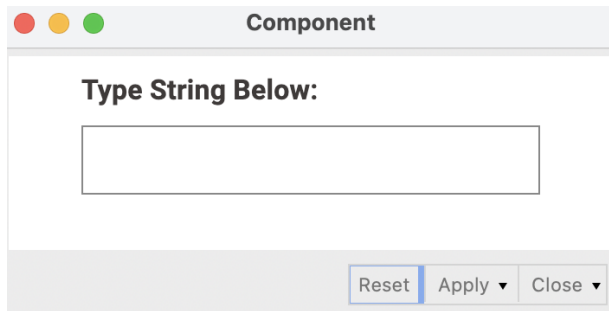
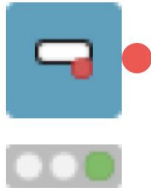


https://docs.knime.com/2022-12/analytics_platform_components_guide/index.html#introduction

Input widget node: String Widget

- Allow user to type.

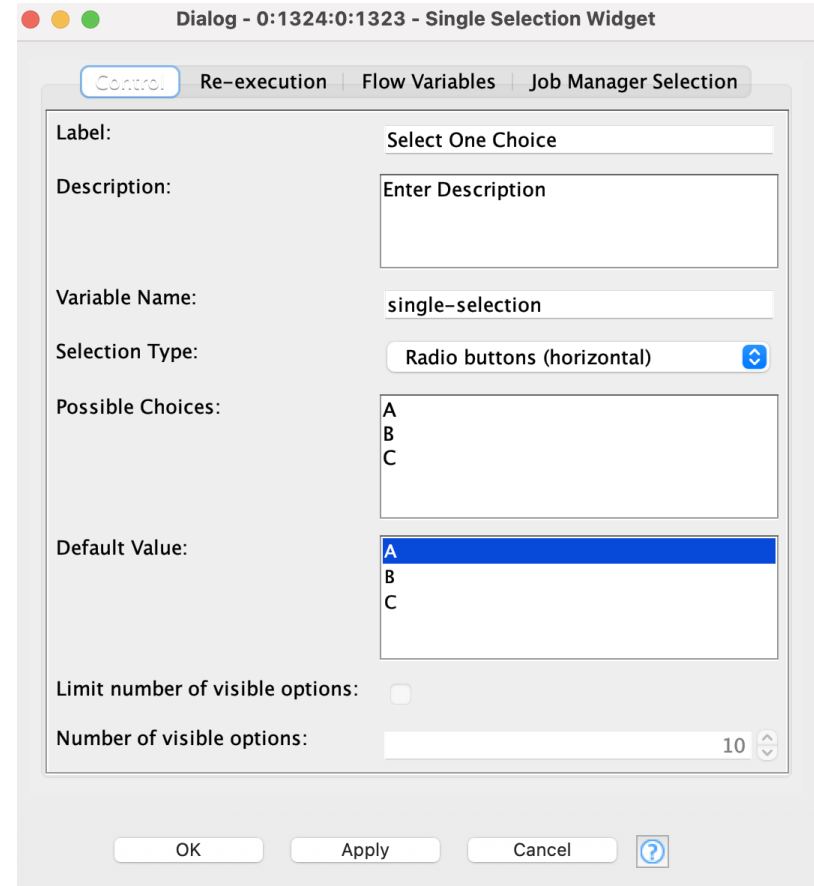
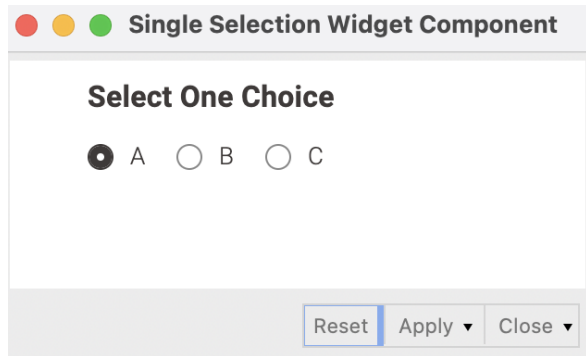
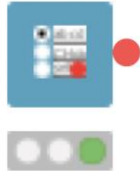
String Widget



Selection widget node: Single Selection Widget

- Allow user to choose an option.

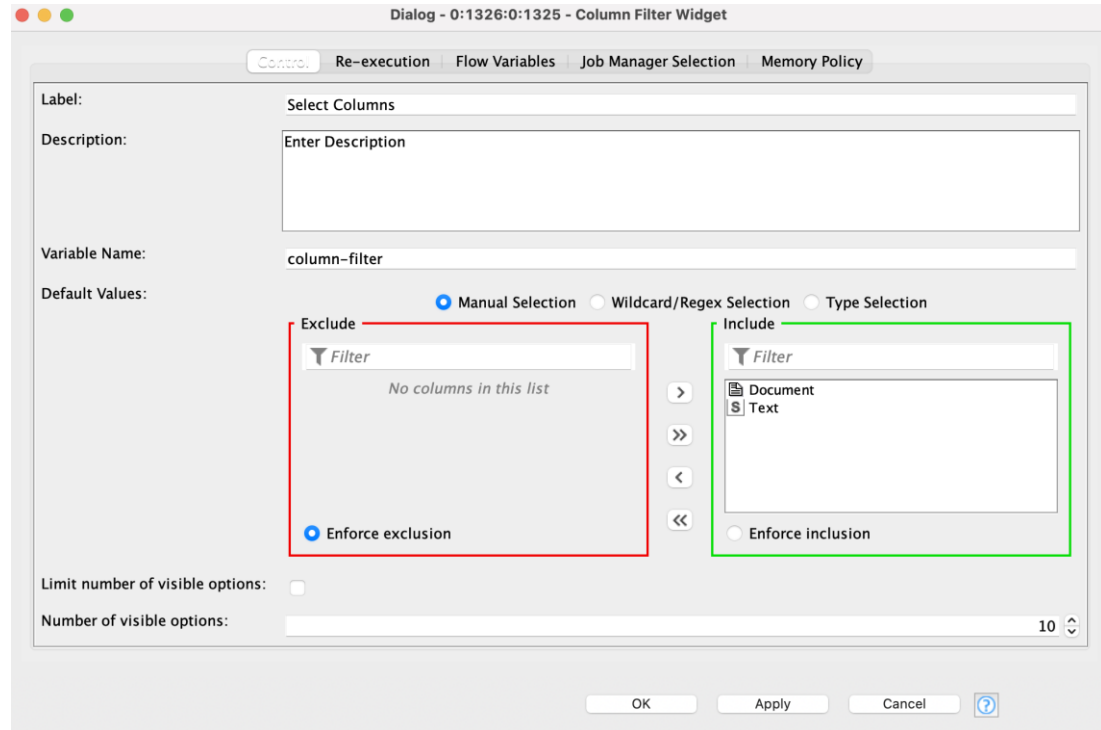
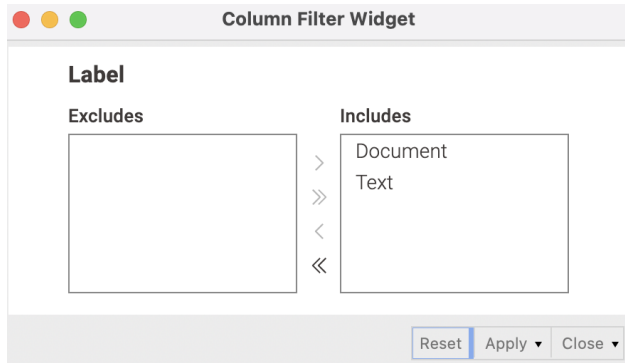
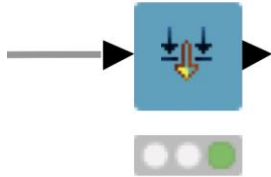
Single Selection Widget



Filter widget node: Column Filter Widget

- Allow user to choose which columns to include and exclude.

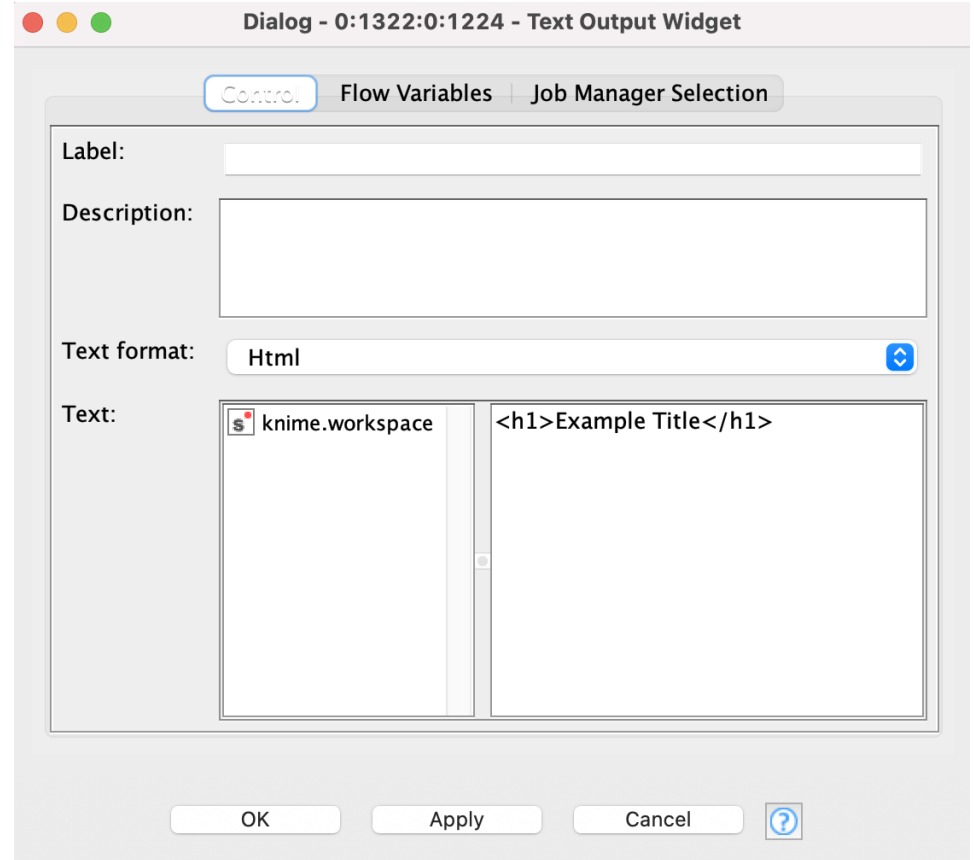
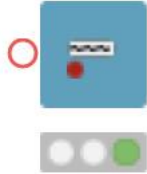
Column Filter Widget



Output widget node: Text Output Widget

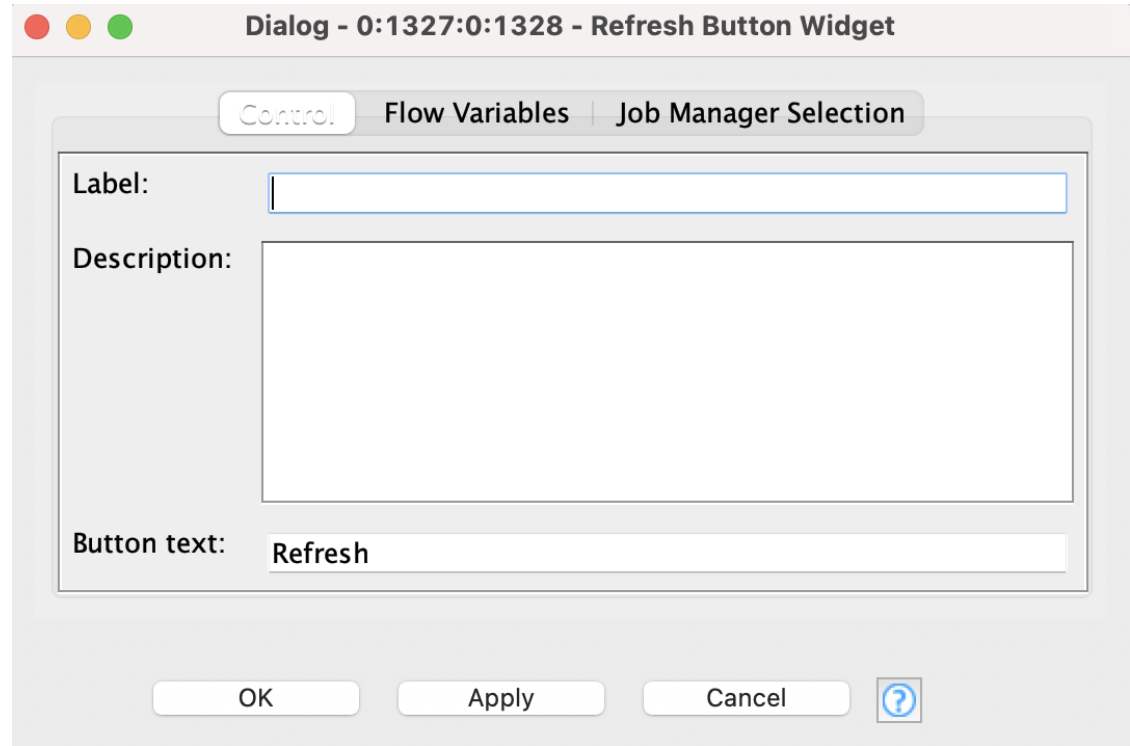
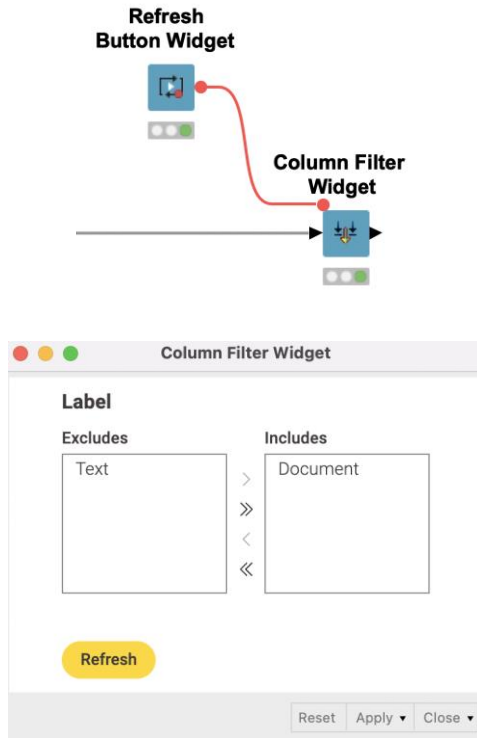
- Generates non-interactive text.

Text Output Widget



Re-execution widget node: Refresh Button Widget

- Re-execute all downstream nodes.



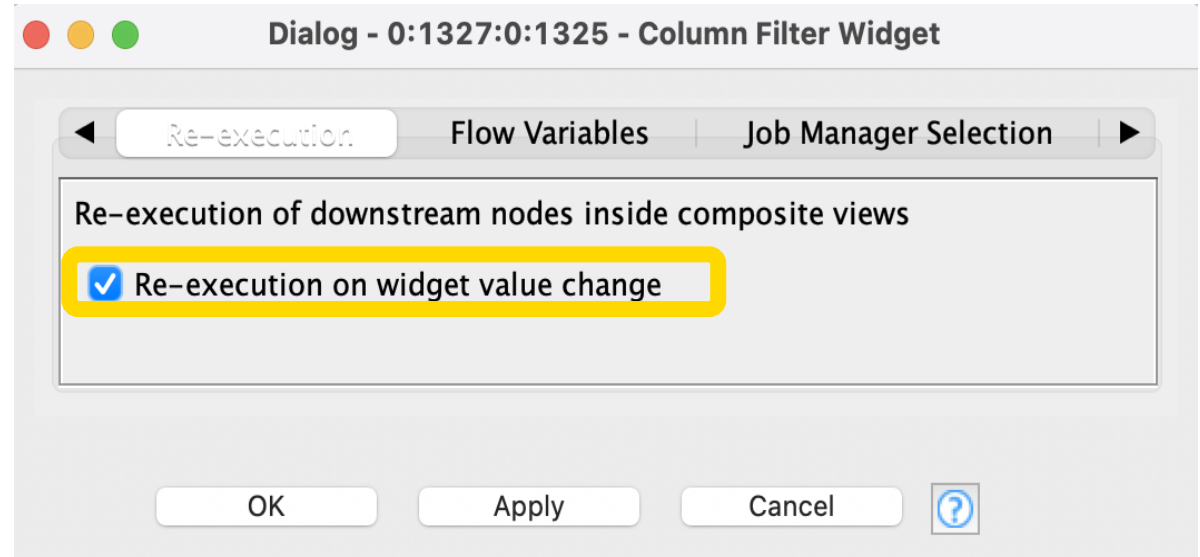
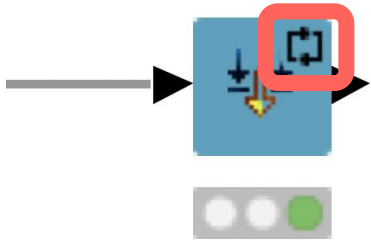
KNIME knowledge check 2

- Which widget allows a user to type text:
 - Text Output Widget
 - String Widget
 - Refresh Button Widget

Re-execution using dialog options

- **Automatically** re-execute all downstream nodes when a selection is made.

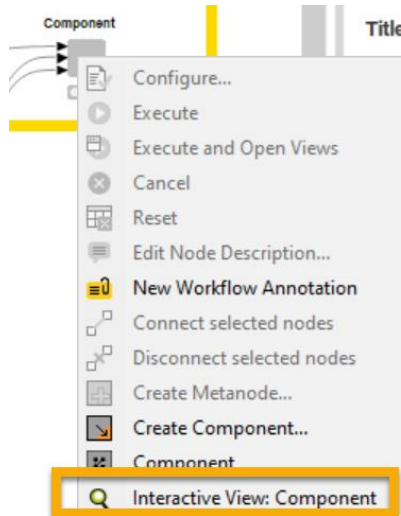
Column Filter Widget



Note: Not all widgets have the Re-execution tab

View your creation

- To View:
 - Right click the component
 - Select Interactive View: Component



Cheat Sheet for components and widgets

Cheat Sheet: Components with KNIME Analytics Platform



COMPONENTS DESCRIPTION

Documents the purpose and usage of the component and defines the appearance of the component. To access and edit the component description, open the Description panel from inside the component and select the empty canvas.

Description: Provides the text describing the use case, requirements, licensing and copyrights, disclaimers, and any other extra information you would like to add.

Icon: Drag and drop a PNG 16x16 px from your local file system. The icon will appear on the component, giving it a unique look.

Color: Usually the color of nodes and components is associated with a precise category. Pick the category that best fits your use case. The selected color appears behind the logo.

In/Out Ports: Describe the ports requirements here. For example: What kind of input columns types are supported? Are there new rows/columns in the output? Does the input/output connect only to a precise other node or component?

Options: The component description also lists all the settings available inside the component dialog. To add text describing the settings open each configuration node dialog and enter it there node by node.

CONFIGURATION DESCRIPTION

Create a list of options of type String for a menu or radio buttons. Define options in the configuration dialog together with the selected default value. This node produces the value of the selected option, which can be used inside the component to configure other node settings.

Create a boolean selection for an enabled/disabled flag (Y/N) in the form of a checkbox. This node produces the value of the selected option in a flow variable at its output port. Usually adopted to configure switch nodes and trigger.

Select the columns of the component input table. Set the node to include or exclude all columns by default. Useful to remove columns that the component should not use in its execution.

Create a string flow variable for configuring other nodes inside the component. Pick a default value so that the user understands how this component settings works. The string flow variable is useful to determine how to resume the component output columns or for text displayed in the component's composite view.

DIALOGUE LAYOUT PANEL

The component dialog shows one setting for each configuration node in the component. From inside the component, select the Node Usage and Layout icon from the top toolbar. In the panel selected in the Configuration Dialog Layout tab, Drag and drop tiles to change the component settings based on sequential steps the user should follow, from top to bottom.

WIDGET NODES

Displays an SVG or PNG image in the composite view of the component. The static image can be displayed in different panel sizes and it usually comes from a Table to Image node connected to this widget.

Creates a slider to filter data to only include rows with values in the selected column within the specified range. The slider can interact with most of the other views inside the component when connected to its output.

Creates a paragraph of either free, preformatted, or HTML text. This widget is useful to show styled text in the component composite view, guiding the user on how to interact with other widget and views.

Creates a list of selectable columns from the input data table in the form of a menu or radio buttons. The node produces the name of the selected columns in a flow variable at its output port.

Offers a button to be placed in the composite view to trigger workflow execution. When clicked, the nodes after this widget re-execute and if any of them are widgets or views they are also updated.

VIEWS NODES

Displays one tile card per row. Useful to browse information row after row exp. when displaying image or text data. It can automatically publish and subscribe to interactive events when it shares the same input with other views and/or interactive widgets in the component.

Displays one curve for every row and one parallel axis for each included column, both numerical and categorical. Useful to explore data points over several dimensions, looking for interesting patterns. It can automatically publish and subscribe to interactive events when it shares the same input with other views and/or interactive widgets in the component.

Displays a curve for each selected column on the x-axis. The x-axis is between the curves is based on another column or the RowID. This view comes from the KNIME Priority Integration, a JavaScript based open source visualization library.

COMPOSITE VIEW LAYOUT

By default KNIME stacks views and widgets from top to bottom in the composite view. Access the Composite View Layout tab via the Node Usage Layout panel from inside the component. Define the layout of the composite view with tables; reset/clear the current layout (button) and drag from the left. Add the columns using "x", drag the views/widgets on the left into the empty structure. Optionally set the size of the panel sizes of individual views/widgets via the gear button.

FLOW VARIABLE FILTER

Filters the flow variables at the input and output of the component. Double-click the Input and Output ports inside the component to access a dialog. By default, flow variables pre-existing upstream of the component input cannot be used inside. Similarly, flow variables generated downstream of the component cannot be used downstream of the component output.

COMPONENT SETUP

You can still change component names and parts after component creation. The panel offers a text field to edit the title, buttons to change the order, remove or add. When adding a new port, a drop down menu appears to define the port type.

VERIFIED COMPONENTS

Verified Components behave like actual nodes and are regularly registered on the KNIME Hub. Browse them all at www.knime.com/verified-components. Each new component is assigned to one of the 14 categories, represented by a color and symbol.

Automation Example: Trains a classification model and outputs the best one in an automated fashion. The component adopts Integrated Deployment to capture the end-to-end process as an optimized workflow.

Visualizations Example: Visualizes a bar chart changing over time via a Generic JavaScript View node. Each visual bar represents a different category, cycling in range with the others in a smooth animation.

Data Manipulation Example: A series of transformations have been implemented via the KNIME Python Integration. Use it to learn how to reliably package your Python scripts for other KNIME users.

Guided Analytics Example: Once a classifier model is trained and deployed, it can be monitored. The component generates a chart with performance over time and controls for retraining of the model.

Finance Analytics Example: Computes the fraction of the year of the difference between two dates, similar to the YEARFRACTION function in Microsoft Excel.

Model Interpretability Example: Computes and visualizes global feature importance for an input model with available model-agnostic XAI techniques.

Time Series Example: Trains in Python a SARIMA (Seasonal ARIMA) model. This model is a powerful option when designing forecasts on data with seasonal or cyclic patterns.

Text Processing Example: Uses a Java based library to extract the main textual content from a web page to analyze online trends for Search Engine Optimization (SEO).

Life Sciences Example: Selects a subset of molecules based on molecular properties via RORX nodes. The Interactive View depicts the properties and structural formula of the selected molecules.

SHARED COMPONENTS

Components can be reused as your personal customized KNIME nodes or they can also be shared with others via KNIME Hub and KNIME Server.

To share a component choose the destination for the shared component in the window that opens. How the component is shared and it can only be edited in its new location. A green arrow appears on top of the component to represent the link between the instance and the shared component in the KNIME Explorer. The link can be of relative type when the component is shared locally.

Manual update is useful to make sure your component is still up to date. A check for updates can be automated or prompted based on your KNIME Preferences. If an update is found but you opt out, the green arrow becomes red. If the link between the instance and the shared component is no longer valid the green arrow will become a red cross.

The owner of the shared component can edit its new location on the KNIME Explorer. After editing, KNIME can update the shared component instances in other workflows. If you are using someone else's shared component you can still edit it, but first you need to break the link, note that then you won't be able to get updates anymore.

SCRIPTED COMPONENTS

Component builders can optionally implement component functionalities through coding. KNIME supports a number of coding frameworks as well as ways to ship dependencies with the component.

Offers a code editor for JavaScript to implement a custom view. Optionally feed in data to visualize it based on your implementation. The node offers checkboxes for a few dependencies (e.g., js, ...) as well as a CSS editor.

Offers a code editor for R to process a KNIME table. KNIME executes the R installation configured either in the node settings and/or in KNIME Preferences. More nodes with different ports are available from the KNIME Interactive R Statistics Integration.

Offers a code editor for Python to process any number and type of inputs into outputs: three dots on the outside of the node means you can add ports. KNIME executes the Python installation configured either in the node settings and/or in KNIME Preferences.

Automatically installs the Conda environment necessary for your component to execute the downstream R/Python nodes. The environment usually includes the R/Python installation plus precise versions of the libraries. Useful to share scripted components with the necessary dependencies. Requires installation of Anaconda or Miniconda and minimal configuration in KNIME Preferences.

ERROR HANDLING

Fails on purpose when a certain condition is met. A custom error message appears on the node and on the outside of the component. Use it to detect whether the input or configurations of the component satisfy minimum requirements and provide the user with an intuitive message on what should be fixed after making the component fail on purpose.

Resources

E-Books: KNIME Advanced Lock covers advanced features and more. Practicing Data Science is a collection of data science case studies from past projects. Both available at www.knime.com/knimepress.

KNIME Blog: Engaging topics, challenges, industry news, and knowledge nuggets at www.knime.com/blog.

E-Learning Courses: Take our free online self-paced courses to learn about the different steps in a data science project (with exercises & solutions to test your knowledge) at www.knime.com/knime-self-paced-courses.

KNIME Hub: Browse and share workflows, nodes, and components. Add ratings, or comments to other workflows at hub.knime.com.

KNIME Forum: Join our global community & engage in conversations at forum.knime.com.

KNIME Server: For team-based collaboration, automation, management, & deployment check out www.knime.com/knime-server.

Data Collection

At the end of this section, you will be able to:

1. Recognize data access nodes.
2. Perform webpage retrieval.
3. Differentiate between widgets.
4. Build a data collection tool.



03: Customer survey demo

■ Goal:

- Create an interactive survey

■ Method:

- Add four unique widgets
- Build component
- Define layout

The screenshot shows a web browser window titled "Customer Satisfaction Form". The form is divided into three main sections: "Contact Information", "Email Address", and "Customer Service".

Contact Information

First name:

Last Name:

Email Address

Customer Service

Using a scale of 1 to 5 with 1 being below average and 5 being excellent, please rate how we perform in the following areas by marking the appropriate number

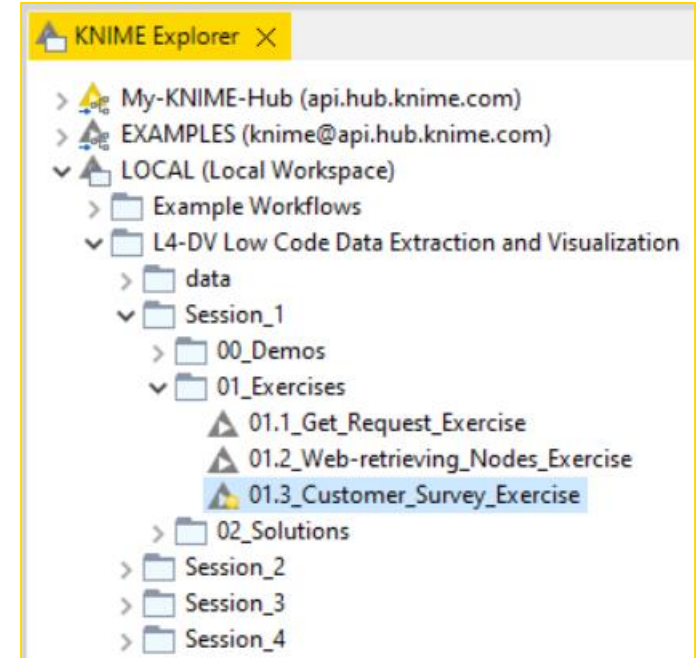
When contacting a member of our staff how would you rate them? ☐ 1 ☐ 2 ☒ 3 ☐ 4 ☐ 5

How would you rate the professionalism and courteousness of our staff? ☒ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

At the bottom right, there are buttons for "Reset", "Apply", and "Close".

Exercises: Section 1

1. Get Request Exercises
Given a table with retrieve information with the Get Request node.
2. From Links to Data Exercise
Given a list of URLs, acquire the text from each link.
3. Customer Survey Exercise
Create your own customer feedback form.



KNIME knowledge check 3

- Which widget was used to create the question in the red box?
 - String Widget
 - Text Output Widget
 - Column Filter Widget
 - Single Selection Widget

Customer Satisfaction Form

Contact Information

First name

Last Name

Email Address

Customer Service

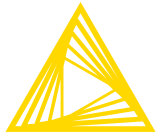
Using a scale of 1 to 5 with 1 being below average and 5 being excellent, please rate how we perform in the following areas by marking the appropriate number

When contacting a member of our staff how would you rate them? ☐ 1 ☐ 2 ☒ 3 ☐ 4 ☐ 5

Summary of section 1

Now you should be able to:

1. Recognize data access nodes.
2. Perform webpage retrieval.
3. Differentiate between widgets.
4. Build a data collection tool.



Open for Innovation

KNIME

[L4-DV] Low Code Data Extraction and Visualization

Section 2



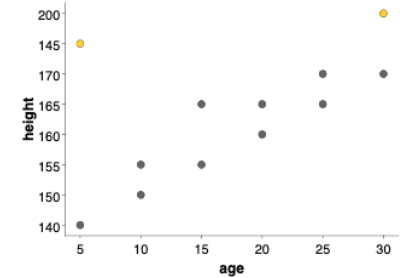
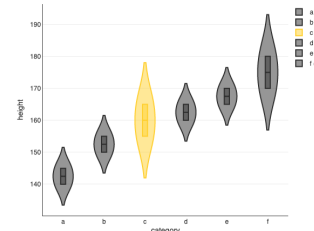
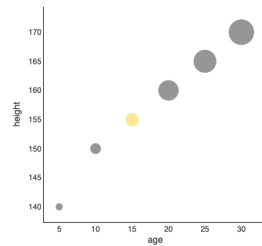
Data Visualization

At the end of this section, you will be able to:

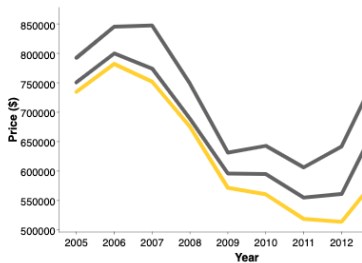
1. Match correct visualization to a specific task.
2. Apply visualizations to common tasks.



Outliers



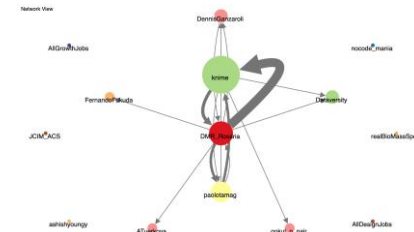
Time



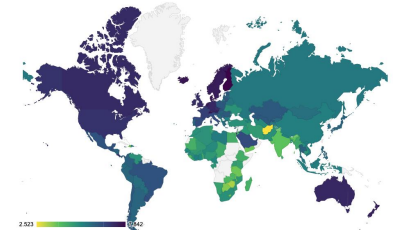
Text



Networks



Geography



The visualizations discussed today

- **Comparison**

- Bar chart, pie chart

- **Correlation**

- Scatter plot, bubble chart

- **Distribution**

- Histogram, violin plot

- **Outliers**

- Box plot, scatter plot

- **Time**

- Line plot

- **Text**

- Tag cloud

- **Networks**

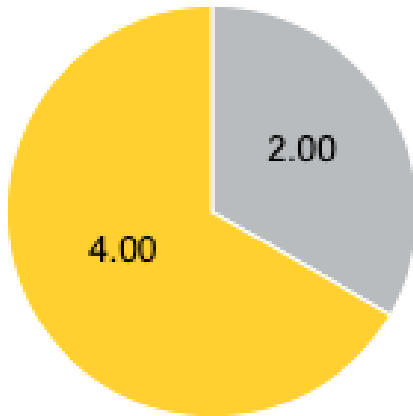
- Network viewer

- **Geography**

- Choropleth map

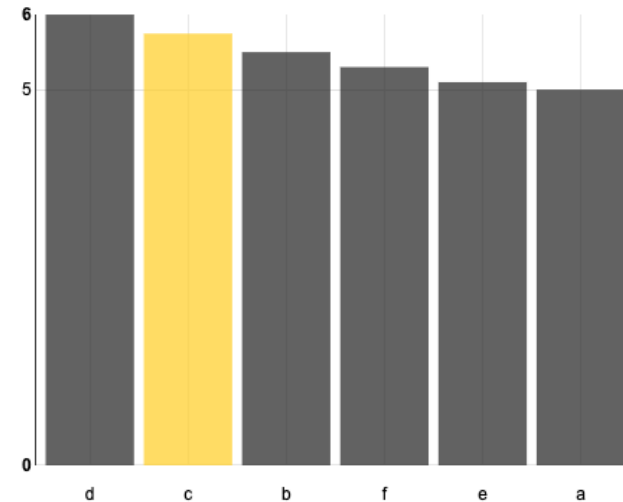
Comparison: Pie chart vs bar chart

- Pie chart
 - differences between **2 categories**



Example: compare revenue

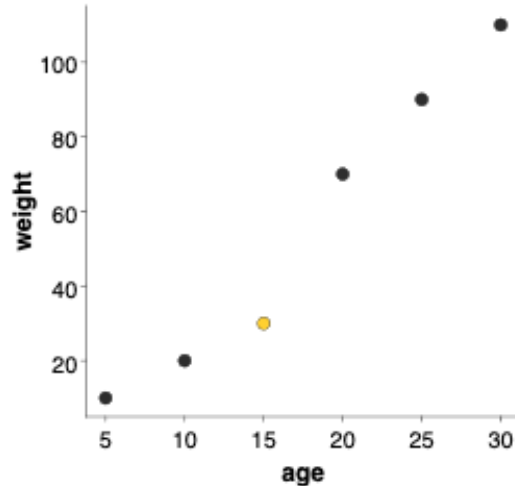
- Bar chart
 - Differences between **2 or more categories**



Example: compare growth

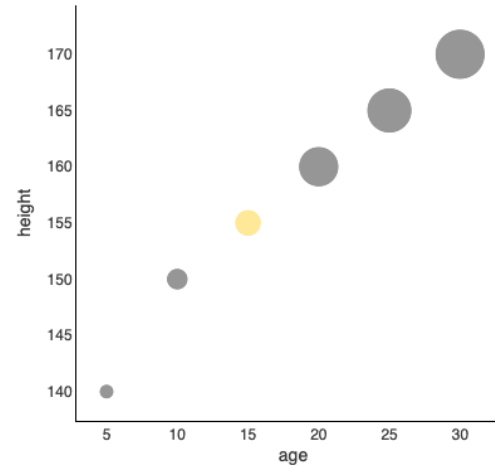
Correlation: Scatter plot vs bubble chart

- Scatter plot
 - correlation among **2 numeric columns**



Example: comparing
age vs weight

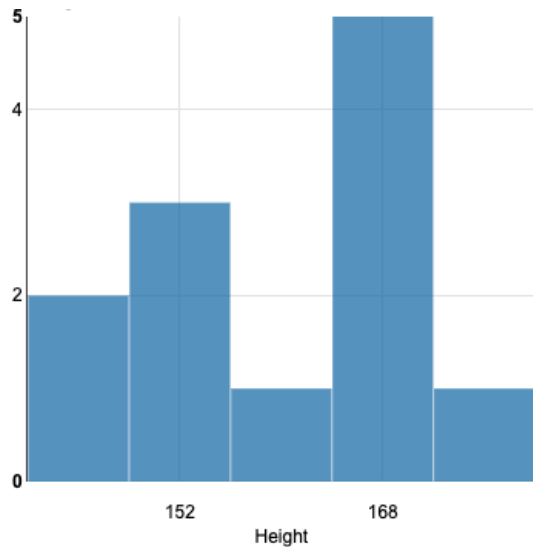
- Bubble chart
 - correlation among **3 numeric columns**



Example: comparing
height vs age vs weight

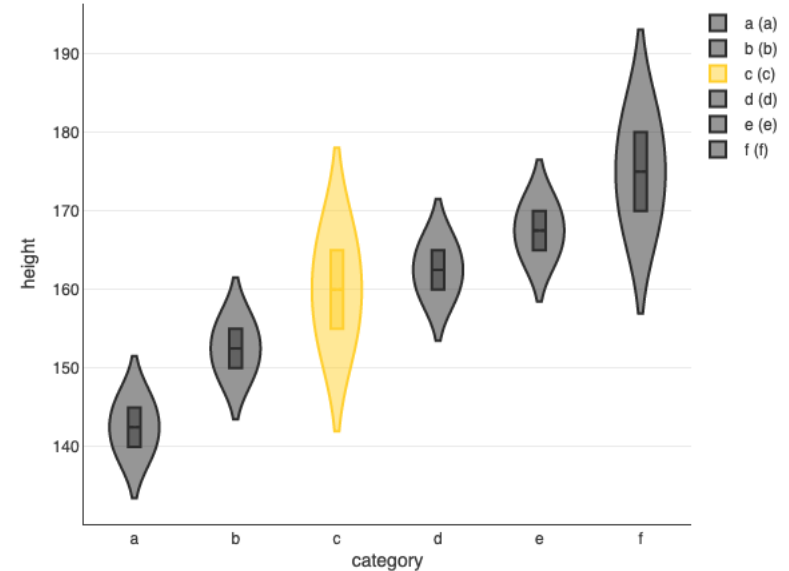
Distribution: Histogram vs violin plot

- Histogram
 - the distribution of a **single numeric column**



Example: distribution of height

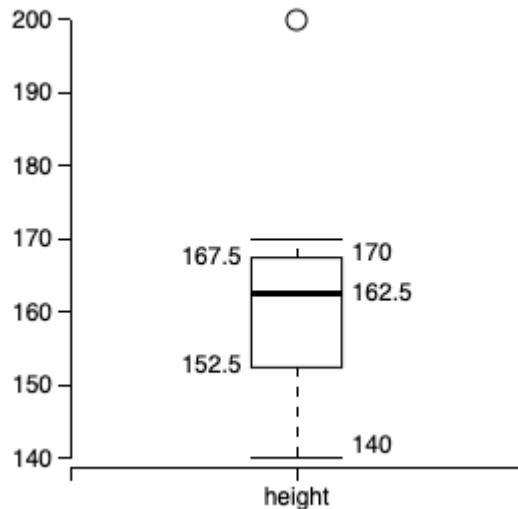
- Violin plot
 - the distribution of a **numeric column per category**



Example: distribution of height per category

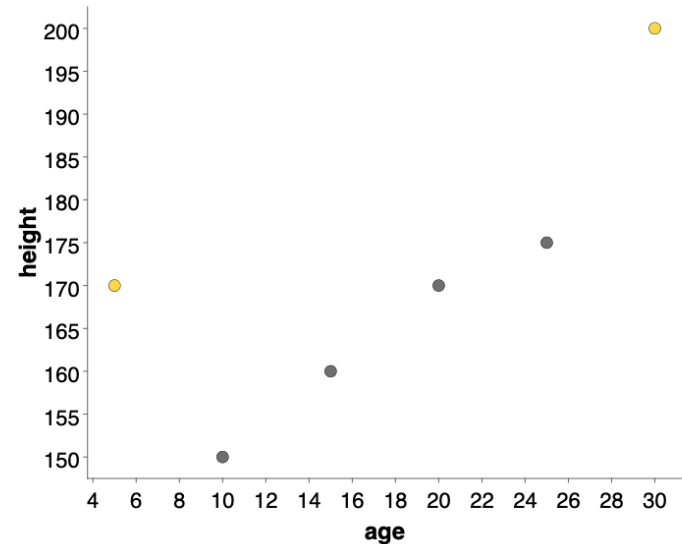
Outliers: Box plot vs scatter plot

- Box plot
 - spot outliers for **1 numeric column**



Example: spot unusually tall or short people

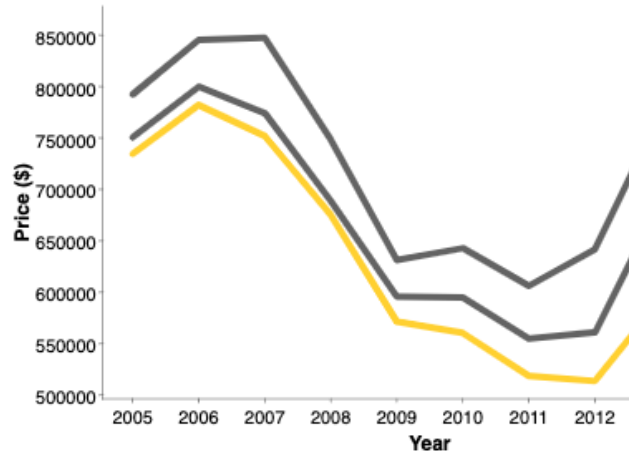
- Scatter plot
 - spot outliers among **2 numeric columns**



Example: spot unusually tall or short people for their age

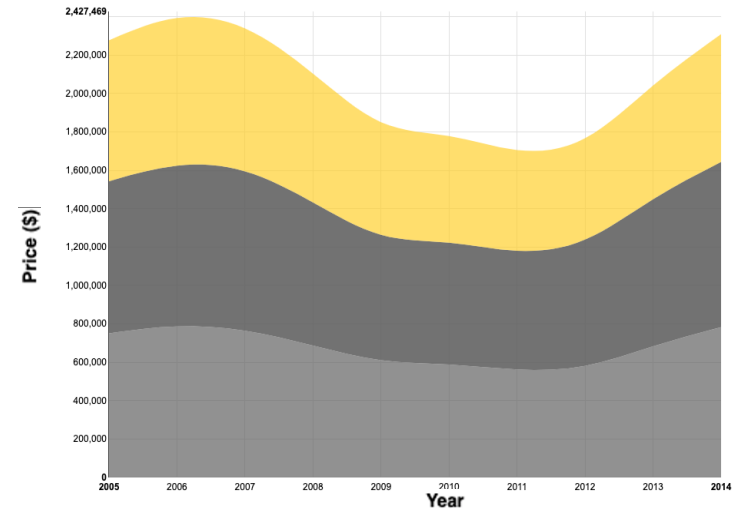
Time: Line plot vs stacked area chart

- Line plot
 - 1 or more **numeric columns** with a **time dependency**



Example: prices of American houses in the early 2000s

- Stacked area chart
 - 2 or more **aggregated numeric columns** with a **time dependency**



Example: prices of American houses in the early 2000s

KNIME Knowledge Check 01

- For each visualization node below, state the type of data visualized such as categorical or numerical data and the number of columns required.
 - For instance,
 - the Bar Chart node would be categorical with one or two columns required.
- Nodes to label: Line Plot, Box Plot, Histogram, Bubble Chart, Violin Plot

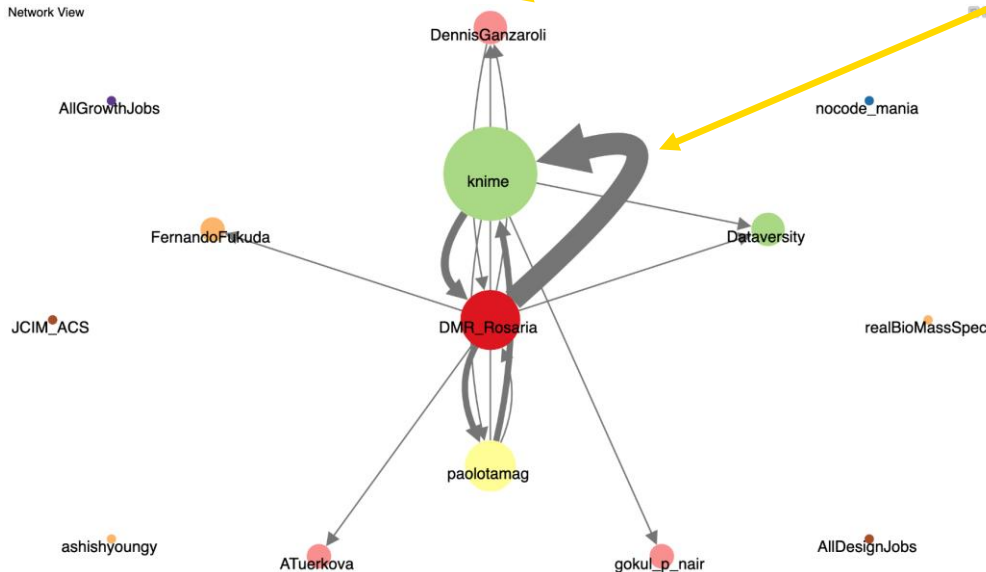
Networks: Network viewer

- Networks

- analyze relationship strength among entities

- Key features:

- Bubble size represents one column
- Arrow size indicates strength of relationship
- Pointing arrows indicate a one-way relationship



- Use case:

- Locating trend setters on Twitter, marketing groups for TikTok

Geography: Choropleth map

- Choropleth map

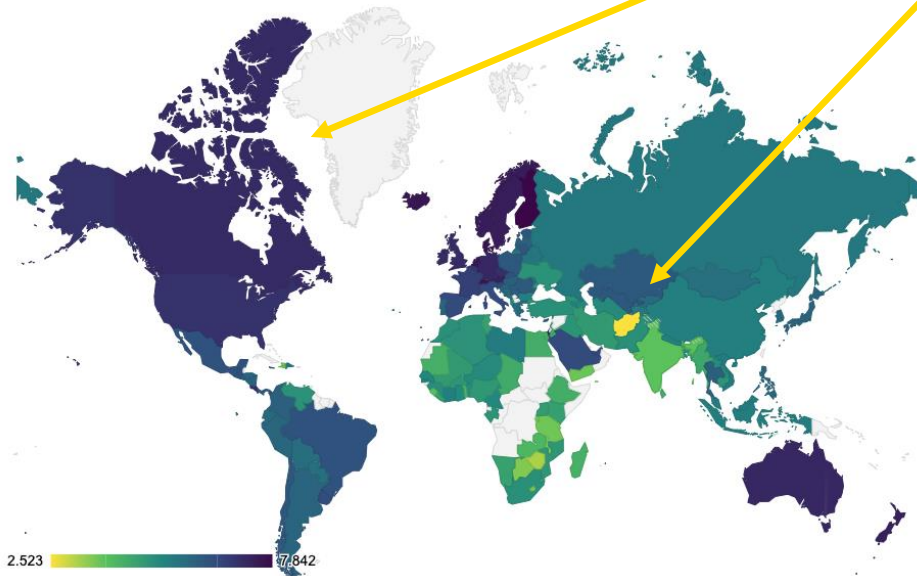
- analyze numeric metrics by regions

- Key features:

- Gradient colors indicate the progression of the numeric values

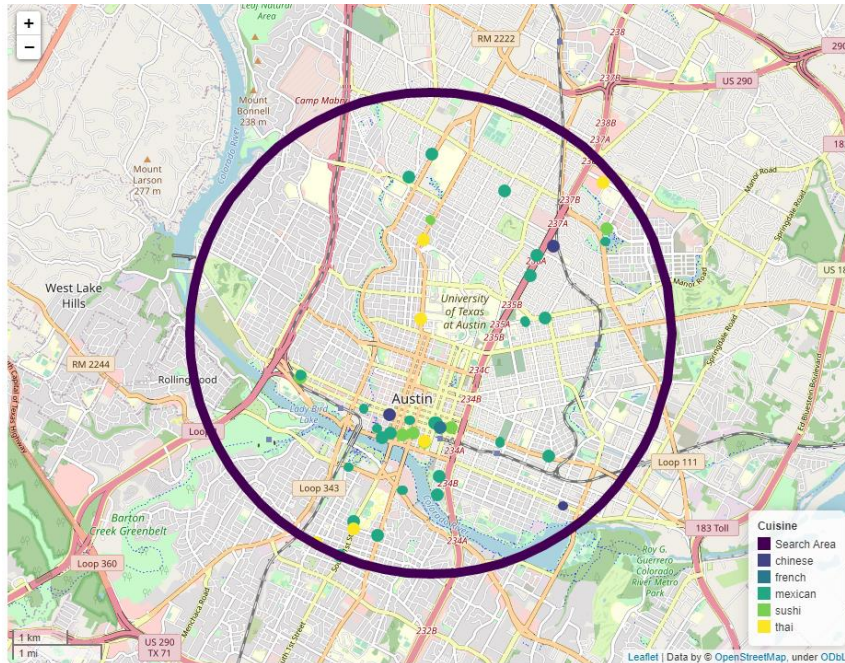
- Use cases:

- Regional sales report, customer satisfaction, hot-spot tracking



Geography: Geospatial view

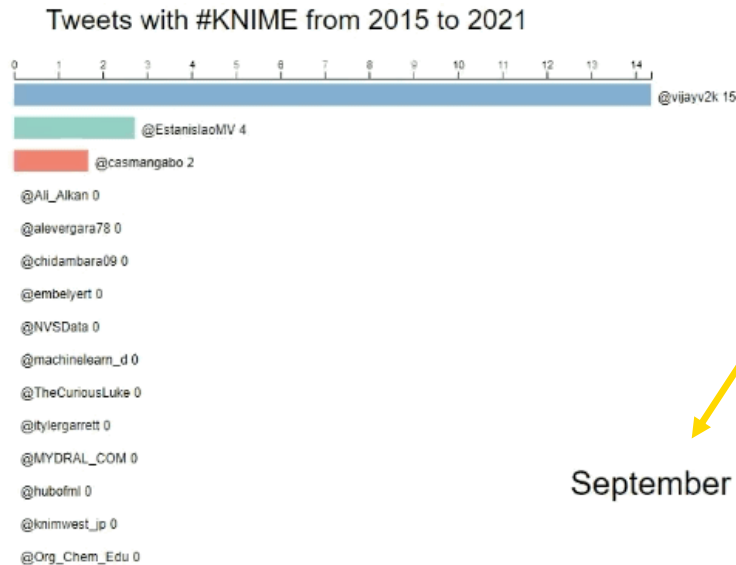
- Analyze data with location information (e.g., latitude and longitude) in an interactive map



- Key features
 - A part of the [Geospatial Analytics Extension for KNIME](#) by the Center for Geographic Analysis from Harvard and KNIME
- Use cases
 - Impact analysis
 - Points of interest search
 - Location optimization

Bonus: Animated bar chart

- Animated bar chart
 - visualize different entities competing or changing over time

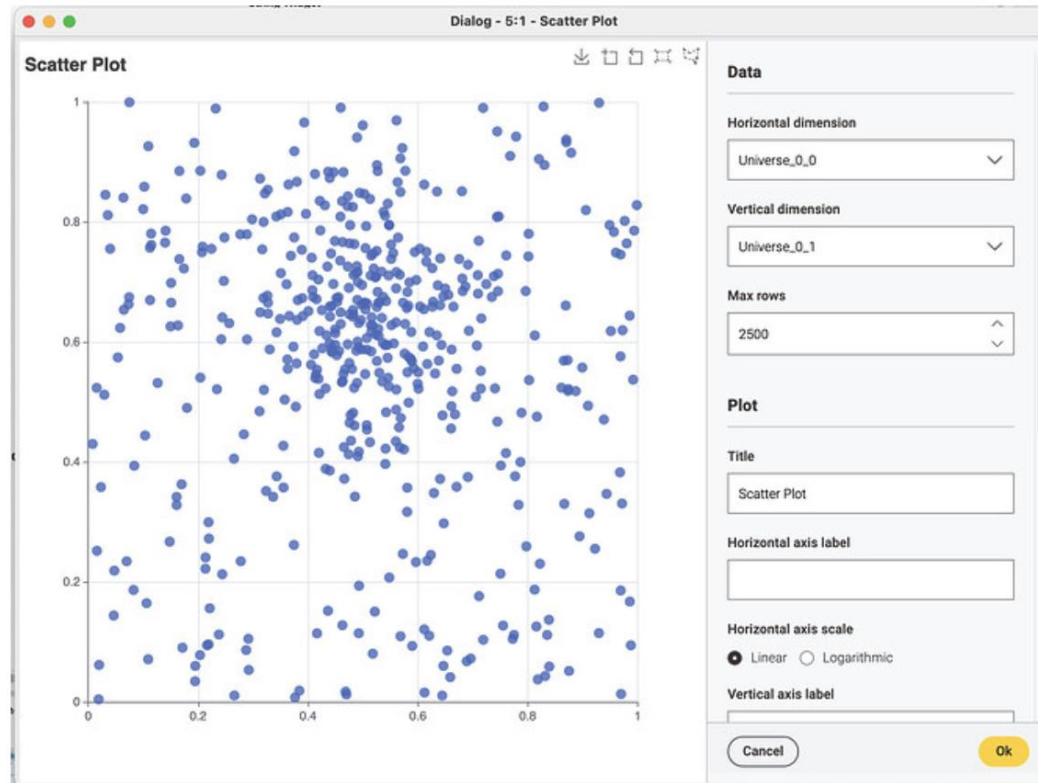


September 2015

- Key features:
 - Time change
- Use case:
 - Attention-grabbing, trend development analysis

What's next for KNIME visualizations?

- To preview the next generation of KNIME visualizations, check out the [labs extension](#) called “KNIME Views (Labs)”



KNIME Knowledge Check 02

- Which visualization is best for examining relationships among entities?
 - Tag Cloud
 - Choropleth map
 - Histogram
 - Network

Data Visualization

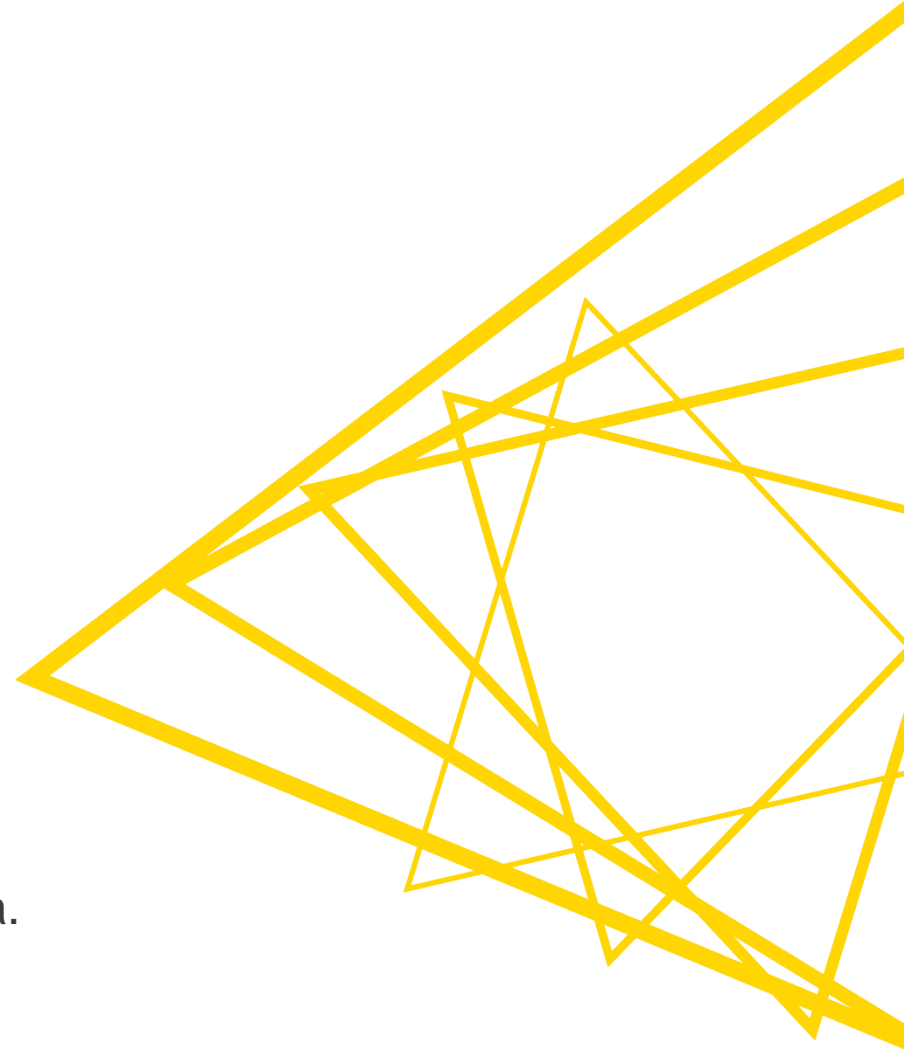
At the end of this section, you will be able to:

1. Match correct visualization for a task.
2. Apply visualizations to common tasks.



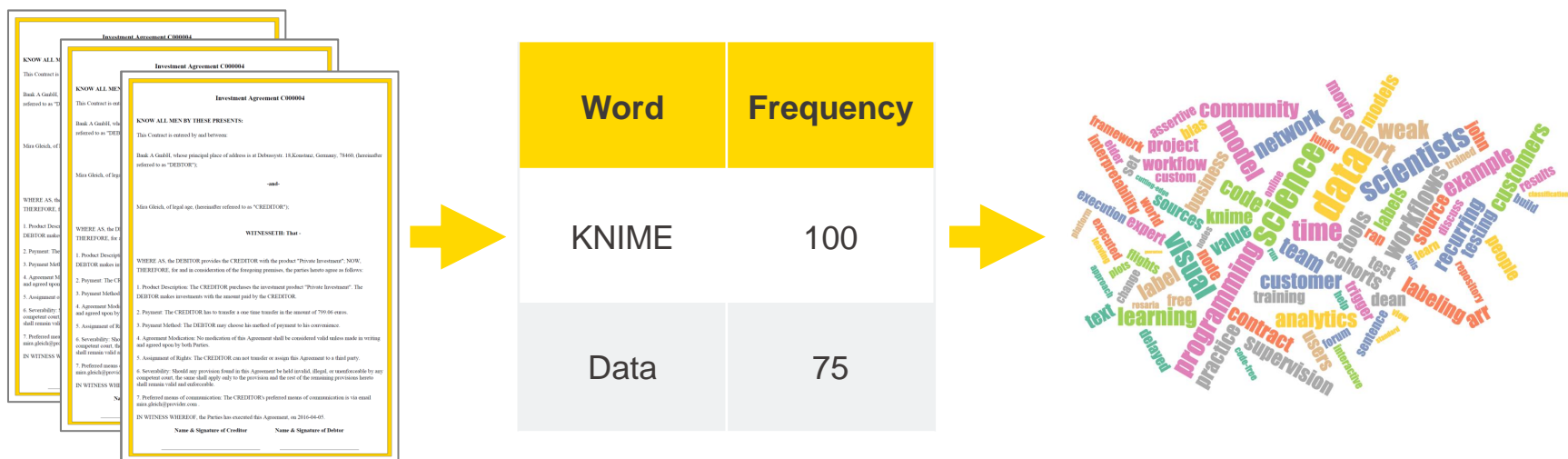
Common Tasks

1. Visualizing text topics.
2. Visualizing relationship strength.
3. Add multiple lines on a line plot.
4. Color bars on a 2-D bar chart.
5. Adding filters on plots.
6. Accessing and visualizing geospatial data.



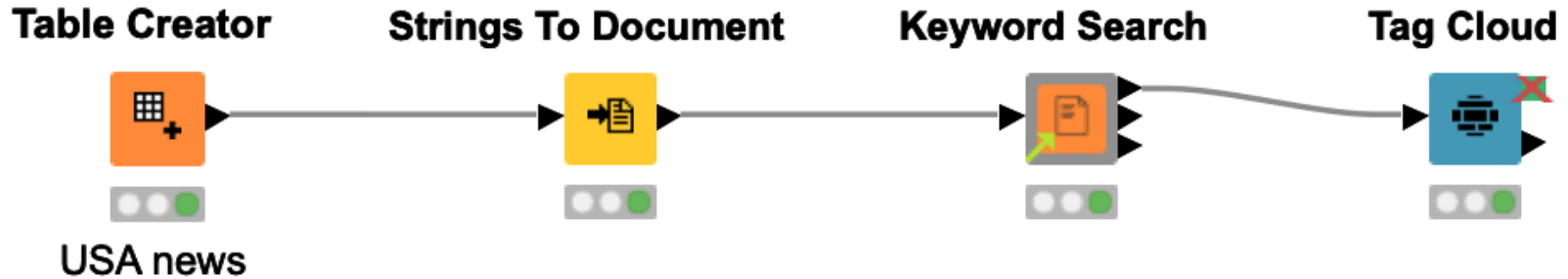
Visualizing text topics

- A 3-stage process: acquire text, rank text, and visualize



Visualizing text topics in KNIME

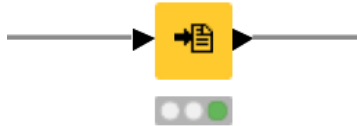
- Making tag clouds is a simple process with 4 nodes.



Strings to Document node

- Converts “String” to “Document” type for text processing tasks.

Strings To Document



Document
"Great food , interesting service"
"Excellent Lunch Destination"
"Hidden treasure near KaDaWe"
"Excellent Food Very Reasonable !"
"Good food , great prices !"
"Nice food at a reasonable price"

Must set these

Warning: The Keyword Search will use words from both the Title column and Full Text.

Dialog - 0:2212 - Strings To Document

Options | Flow Variables | Job Manager Selection | Memory Policy

Title: ☒ Column ☐ Row ID ☐ Empty string
Title column: [S] Title

Text: Full text: [S] Description

Meta Information

Document source:
☐ Use sources from column Document source column: [S] Item Url

Document category:
☐ Use categories from column Document category column: [S] Item Url

☐ Use authors from column Authors column: [S] Item Url

Author names separator: ,
Default author first name: Default author last name:

Type and Date

Document type: UNKNOWN
Date: 2022-05-24
☐ Use publication date from column Publication date column:

Column

Document column: Document

Processes

Number of maximal parallel processes: 3


Tokenization

Word tokenizer: OpenNLP English WordTokenizer

OK Apply Cancel ?

What's a Document type?

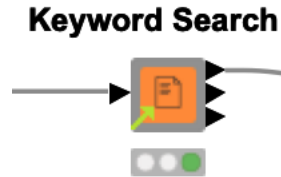
- A composite/aggregate data type for textual content
- Fields include:
 - Title
 - Text
 - Source
 - Category
 - Author(s)
 - Date, ...
 - Generic Meta Data

 Document
"Great food , interesting service"
"Excellent Lunch Destination"
"Hidden treasure near KaDaWe"
"Excellent Food Very Reasonable !"
"Good food , great prices !"
"Nice food at a reasonable price"

Warning: The text seen in a KNIME table for a document type only displays the title.

Keyword Search component

- Extracts keywords from a document for tag clouds, networks, etc.



S ▼	Term	D ▲	Weight
	women	2	
	winter	1	
	winners	1	
	weather	1	
	visit	2	

Must set these

Warning: The number of terms found will be: Topics x Words

Dialog - 0:2210 - Keyword Search

Options | Flow Variables | Memory Policy | Job Manager Selection

Document Column
Document

Parallel LDA: No. of Topics
10

Parallel LDA: Number of Words per Topic
10

Parallel LDA: Alpha
0.1

Parallel LDA: Beta
0.01

IDF Measure
smooth

Deactivate Text Pre-Processing
☐

OK Apply Cancel ?

Keyword Search outputs 1 and 2

- Outputs useful for tag clouds and networks

Output 1:
Word and
importance
(for tag clouds)

<input type="text" value="S"/> ▼ Term	<input type="text" value="D"/> ▲ Weight
women	2
winter	1
winners	1
weather	1
visit	2

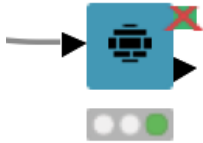
Output 2:
Word pair
frequency
(for networks)

<input type="text" value="S"/> UniqueID	<input type="text" value="S"/> Term1	<input type="text" value="S"/> ▼ Term2	<input type="text" value="I"/> ▼ Document cooccurrence
york-university	university	york	1
york-bestowed	bestowed	york	1
york-renowned	renowned	york	1
york-pop	pop	york	1
york-singer	singer	york	1

Tag Cloud node

- Generates a tag cloud showing words and their frequencies/importance


Tag Cloud



Must set these

A note on text / natural language processing

- The Keyword Search component performs document preprocessing automatically.
- The document preprocessing techniques are introduced in our free, self-paced text processing course on LearnUpon:



[L4-TP] Introduction to Text Processing
★★★★★ · Difficulty Advanced · Length 8 Hours

This course is about text mining, its theory, concepts, and applications. Specifically, the course focuses on the acquisition and processing of textual data with KNIME Analytics Platform. You wil... [Read More](#)

☰ 27 Modules

★ 52 Reviews

➡ Start

[Click here for KNIME Text Processing Educational Material](#)

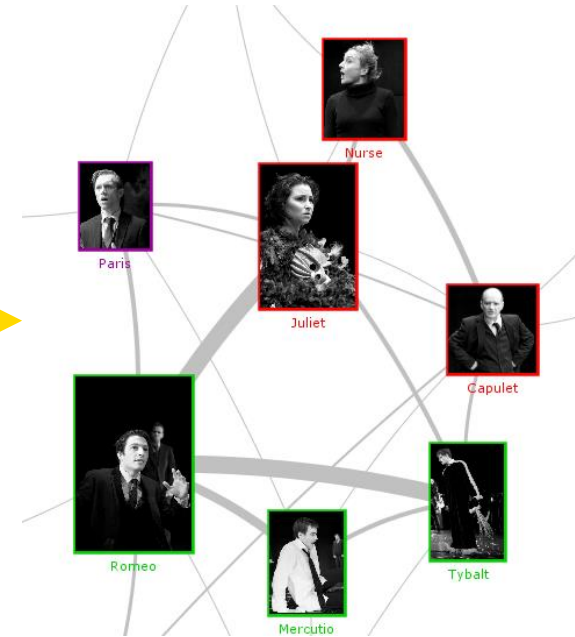
Visualizing relationship strength

- A 3-stage process: acquire text, build a network, and visualize the graph

Romeo, art thou mad?

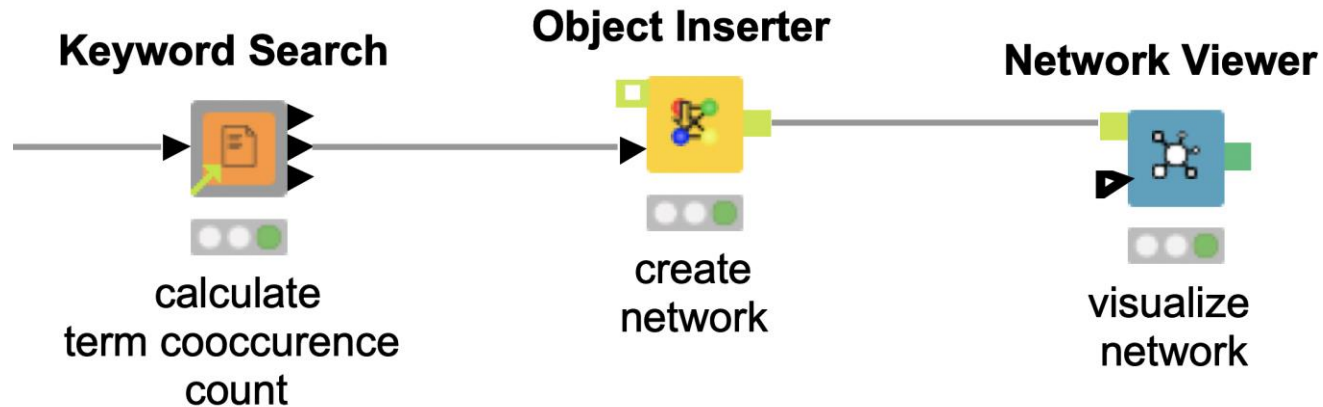


Network id: objectInserter uri: objectInserter
Directed: false weighted: true
No of nodes: 14 No of edges: 30



Visualizing relationship strength in KNIME

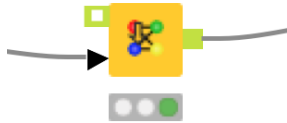
- Example workflow for making a network graph (with 2 *more* nodes)



Object Inserter node

- Generates a network from a table.
- Provides input for the Network Viewer node.

Object Inserter



Dialog - 0:2213 - Object Inserter (creates network)

Options | Advanced Options | Flow Variables | Job Manager Selection

Node settings

Node id column: (opt.) Node label column: (opt.)

Second node id column: (opt.) Second node label column: (opt.)

Edge settings

Edge id column: (opt.) Edge label column: (opt.) ☐ Create directed edges

Weight settings

☐ None ☐ Default ☒ Column Default weight: Weight column: ☐ All nodes have same weight

OK Apply Cancel ?

Updated Network - 0:2213 - Object Inserter (creates network)

File | Network: objectinsserter | Flow Variables

Network id: objectinsserter **uri:** objectinsserter
Directed: false **weighted:** true

No of nodes: 13 **No of edges:** 20

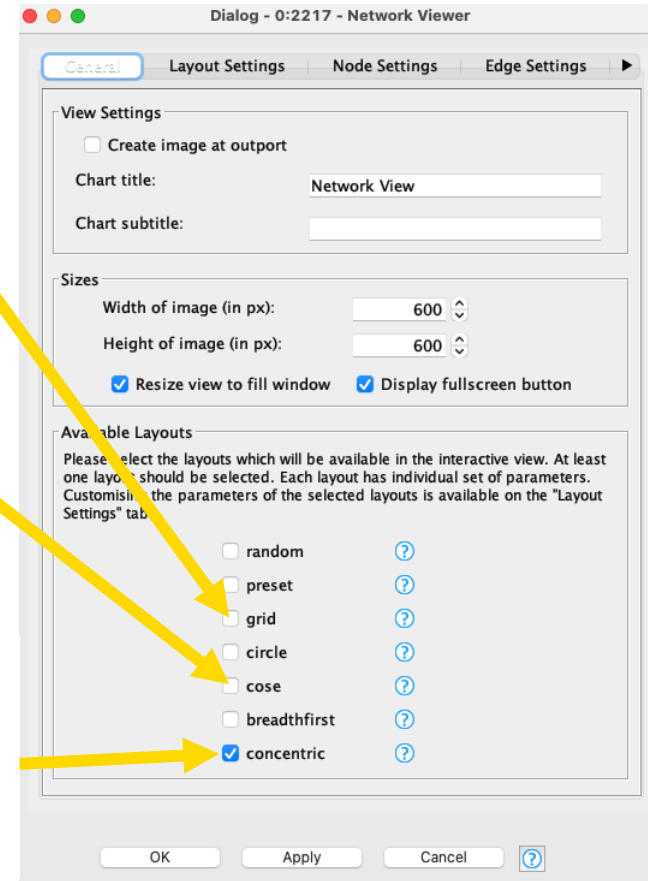
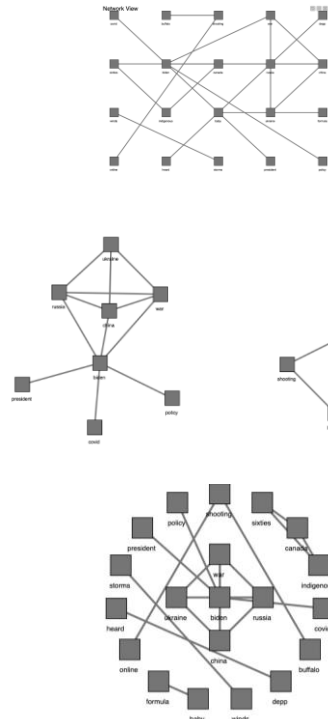
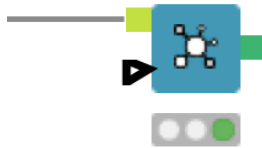
Partition:
No of partitions: 2 by type: EDGE(1), NODE(1)
Type: EDGE
1. edges
Type: NODE
1. nodes

Feature:
No of features: 1 No of value types: 1
1. weight
Type: double

Network Viewer node

- Allows us to view a network

Network Viewer

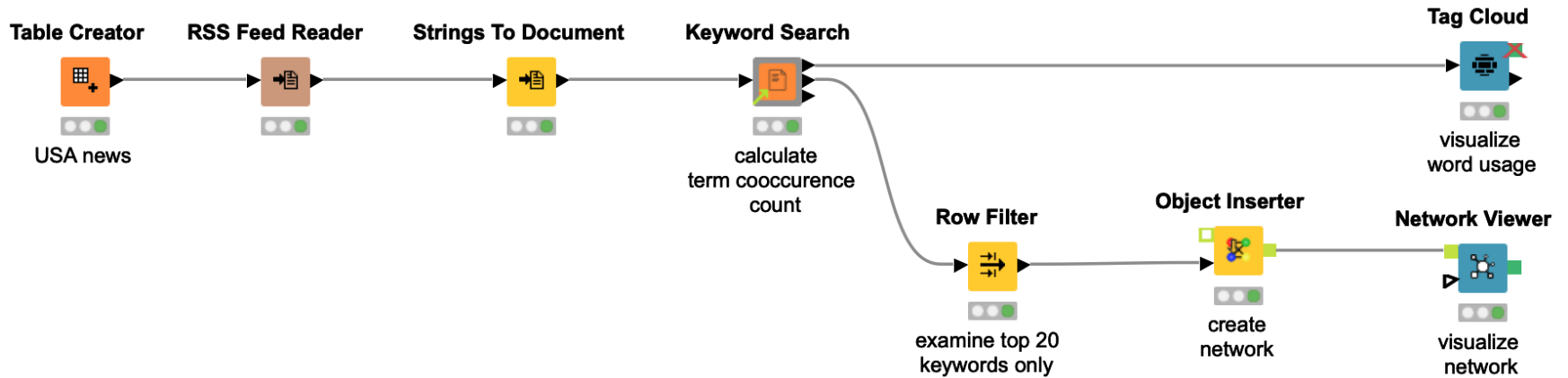


[Click here to learn more about network analysis](#)

KNIME Knowledge Check 03

- Word-Pair Frequency is useful for which visualization?
 - Tag Clouds
 - Networks
 - Choropleth maps

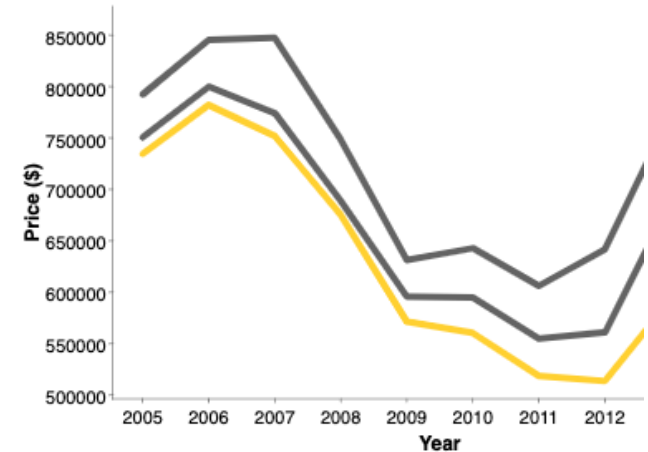
01: Simple tag cloud and network demo



Add multiple lines on a line plot

- How do we visualize each region on a line plot?

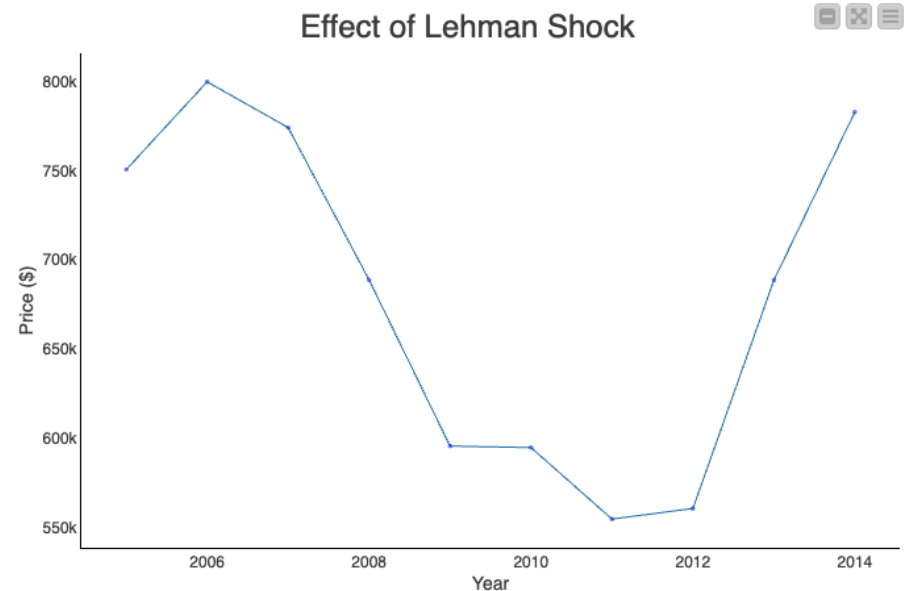
S	RegionName	I	Year	D	Mean(ColumnValues)
	San Francisco, CA		2004		627,953.417
	San Francisco, CA		2005		750,609
	San Francisco, CA		2014		782,904.583
	San Jose, CA		2003		601,880.917
	San Jose, CA		2006		845,468.25
	San Jose, CA		2015		958,072.667
	Santa Cruz, CA		2004		614,160.167
	Santa Cruz, CA		2008		675,264.333
	Santa Cruz, CA		2021		1,078,152.833



Background

- Typically, we plot one line using data in this format:

I	Year	D	San Francisco.
	2005		750,609
	2006		799,768.75
	2007		774,172.25
	2008		688,893.25
	2009		595,844
	2010		595,020.833
	2011		554,982.25
	2012		560,916.167
	2013		688,696.25
	2014		782,904.583



Problem

- We cannot plot multiple lines from a column with mixed data
- We'll need to reformat

Distinct region data only

I Year	D San Francisco
2005	750,609
2006	799,768.75
2007	774,172.25
2008	688,893.25
2009	595,844
2010	595,020.833
2011	554,982.25
2012	560,916.167
2013	688,696.25
2014	782,904.583

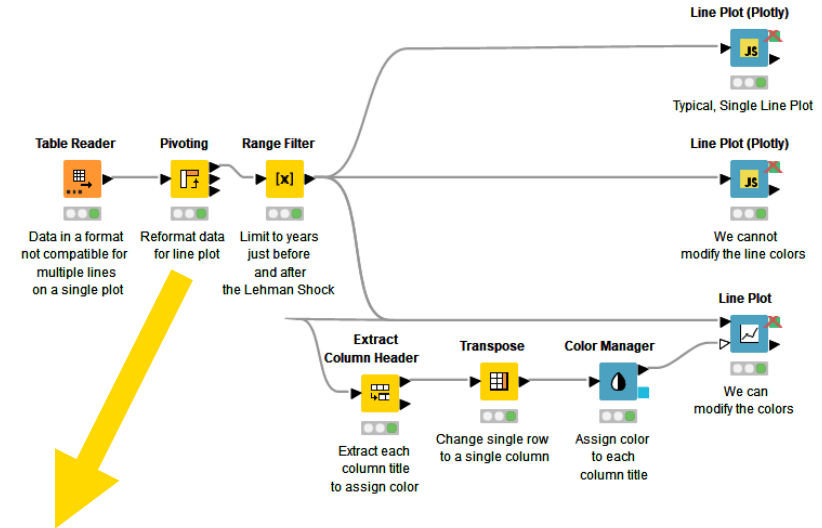
Mixed region data

S RegionName	I Year	D Mean(ColumnValues)
San Francisco, CA	2004	627,953.417
San Francisco, CA	2005	750,609
San Francisco, CA	2014	782,904.583
San Jose, CA	2003	601,880.917
San Jose, CA	2006	845,468.25
San Jose, CA	2015	958,072.667
Santa Cruz, CA	2004	614,160.167
Santa Cruz, CA	2008	675,264.333
Santa Cruz, CA	2021	1,078,152.833

Solution

- Key Points

1. For each line we want, we need a **distinct column**
2. Therefore, we **pivot** the data
3. Adding colors to the columns requires **exact column name matching**



I Year	D San Francisco, CA+First*(Mean(ColumnValues))	D San Jose, C...	D Santa Cruz, ...
2000	394,841.833	489,902.75	382,738.25
2001	468,226.417	590,062.417	475,892
2002	500,029.75	585,127.25	505,906.917
2003	545,562.75	601,880.917	546,822.75
2004	627,953.417	663,949.417	614,160.167
2005	750,609	792,311	734,655
2006	799,768.75	845,468.25	782,231.917
2007	774,172.25	847,380.417	751,948.75
2008	688,893.25	748,603.583	675,264.333

Exercises: Section 2

1. Line Plot

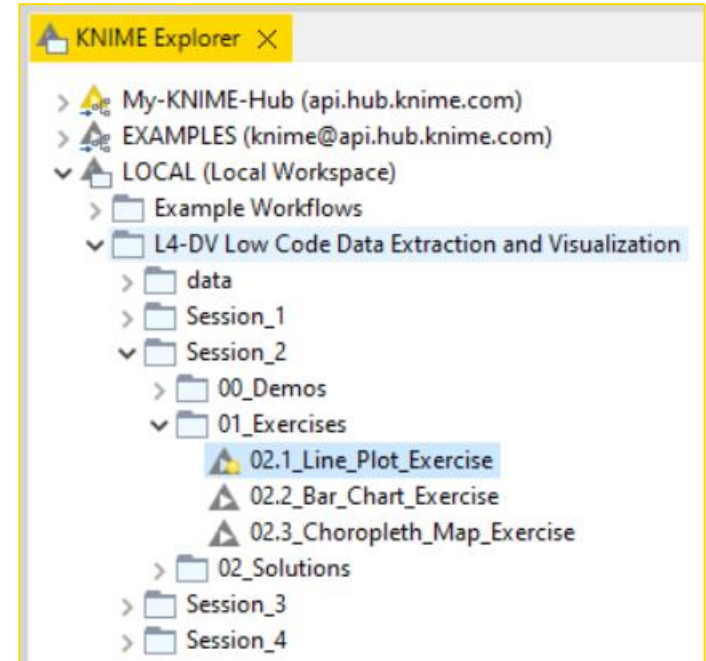
Using the data provided, transform it so that you can display multiple lines at once.

2. Bar Chart

Using the data provided, transform it to display the data using custom colors.

3. Choropleth Map

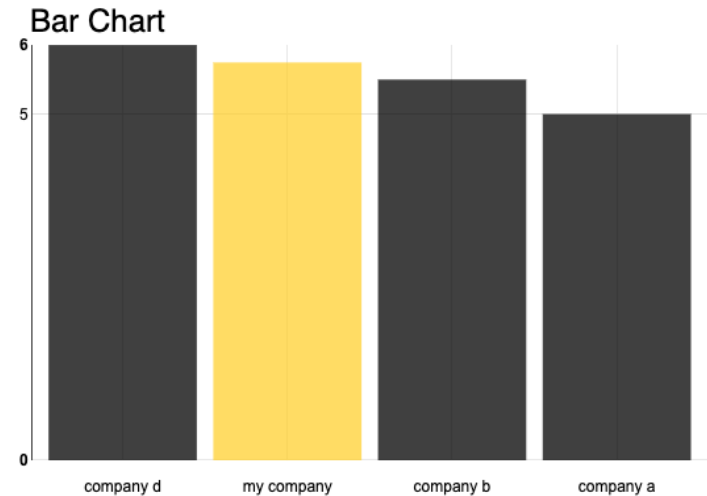
Using the data provided, transform it to create a map highlighting key assets.



Color bars on a 2-D bar chart

- How do we go from two columns (categorical and numeric data) to a colored bar chart?

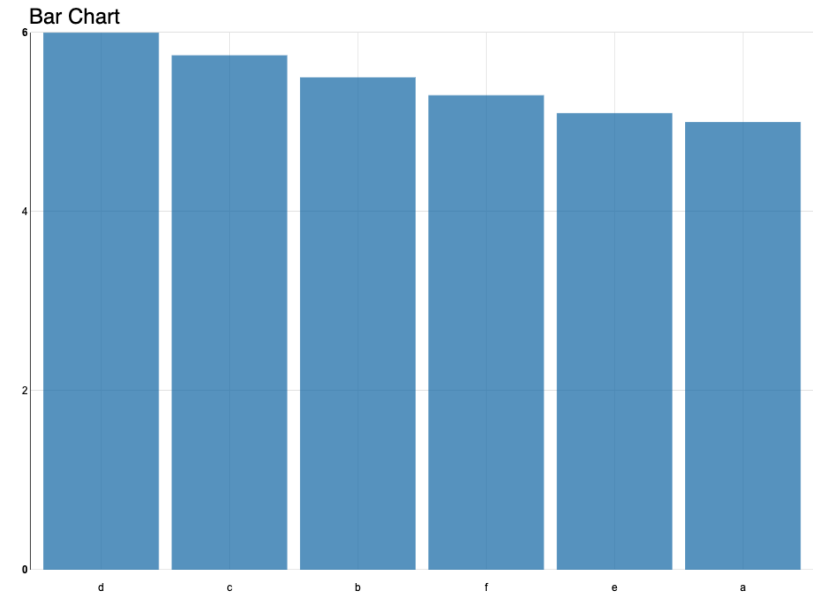
Row ID	S category	D value
Row0	a	5.001
Row1	b	5.5
Row2	c	5.747
Row3	d	5.999
Row4	e	5.1
Row5	f	5.3



Background

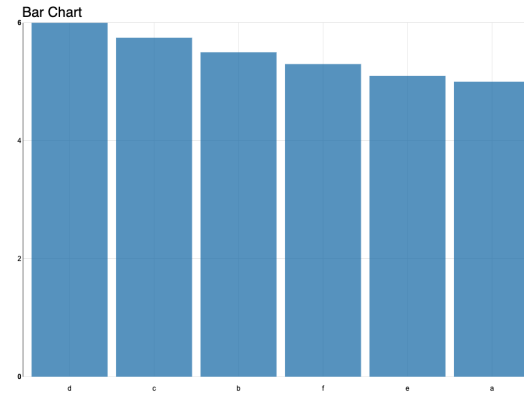
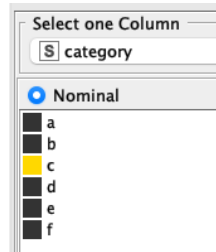
- Typically, we plot multiple bars using data in this format:

S	category	D	value
d			5.999
c			5.747
b			5.5
f			5.3
e			5.1
a			5.001



Problem

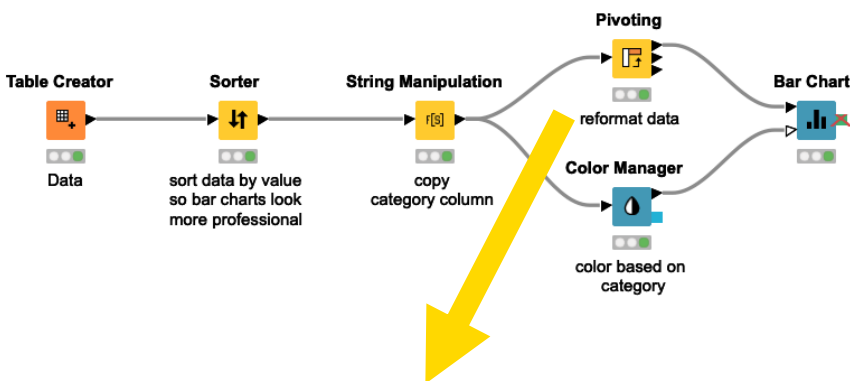
- The Color Manager node doesn't seem to affect the bar colors.



Solution

- Key Points

1. We need to have **one column per value**.
2. Therefore, we make a **copy** of our column we want to color.
3. Then we **pivot** the data.

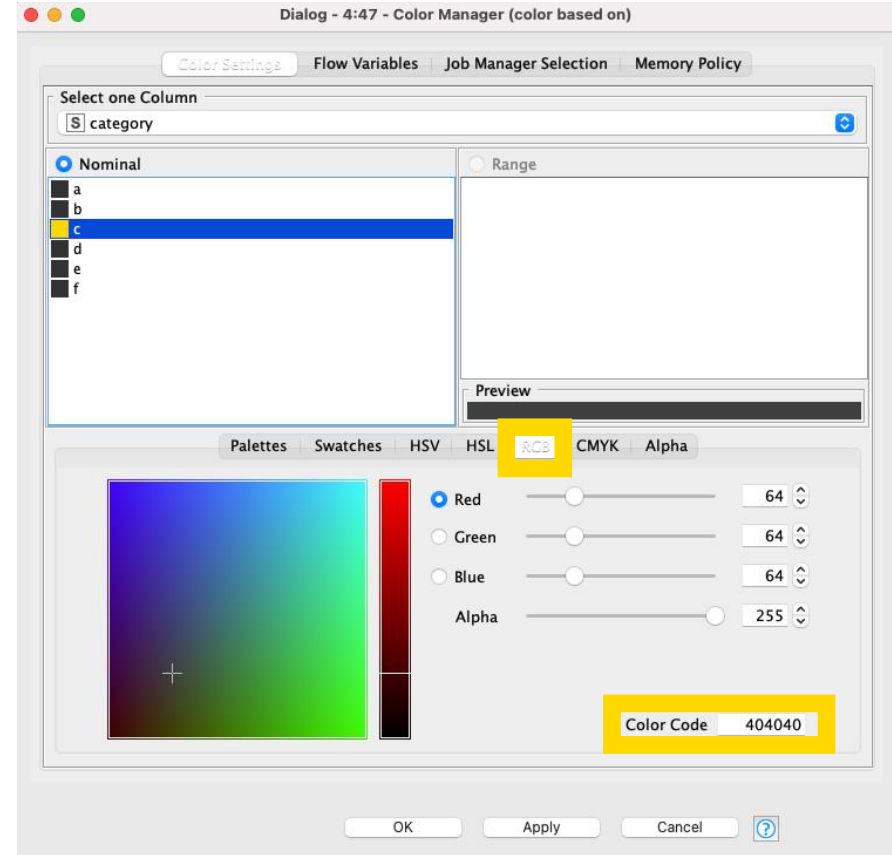


Row ID	S category	D f	D e	D d	D c	D b	D a
Row0	d	?	?	5.999	?	?	?
Row1	c	?	?	?	5.747	?	?
Row2	b	?	?	?	?	5.5	?
Row3	f	5.3	?	?	?	?	?
Row4	e	?	5.1	?	?	?	?
Row5	a	?	?	?	?	?	5.001

Adding distinct color palettes

- Use custom colors instead of the default colors or given palettes.
- For distinct colors use color codes.
 1. In the Color Manager node, select the RGB tab
 2. Then input the color code you would like

Note: Find color palettes and/or hex codes (color codes) on the web.



Exercises: Section 2

1. Line Plot

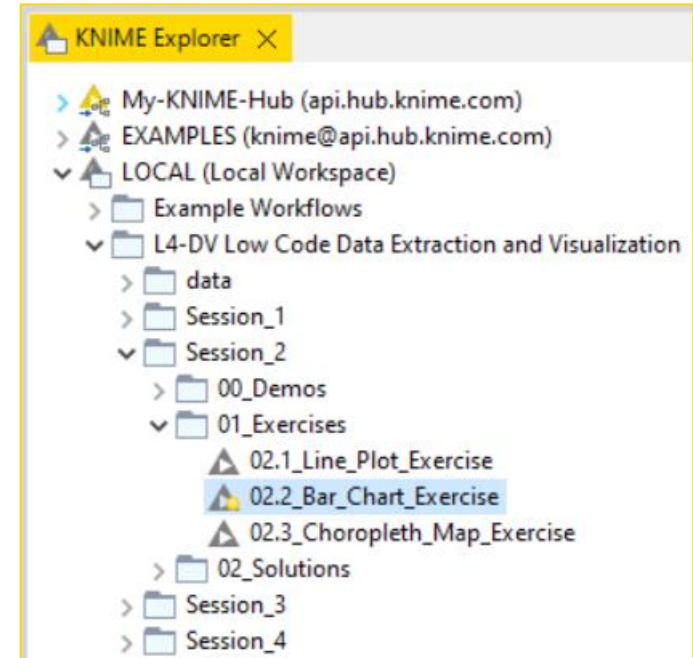
Using the data provided, transform it so that you can display multiple lines at once.

2. Bar Chart

Using the data provided, transform it to display the data using custom colors.

3. Choropleth Map

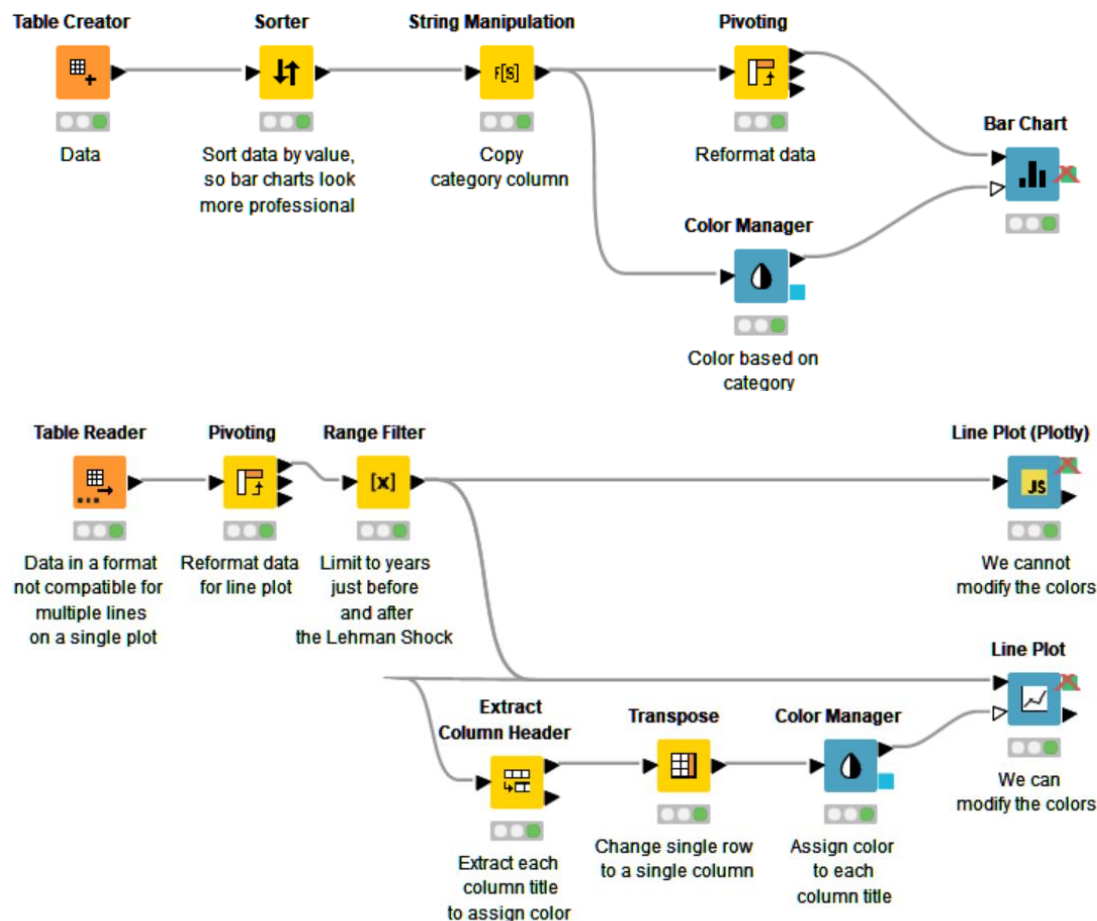
Using the data provided, transform it to create a map highlighting key assets.



KNIME Knowledge Check 04

- Coloring bars on a 2D bar chart and adding lines to a line plot had a similar issue. What was it?
 - They required distinct rows with categorical data
 - They required distinct columns with numerical data
 - They required more rows with numerical data
 - They required more columns with categorical data

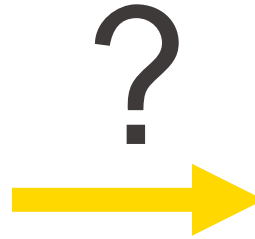
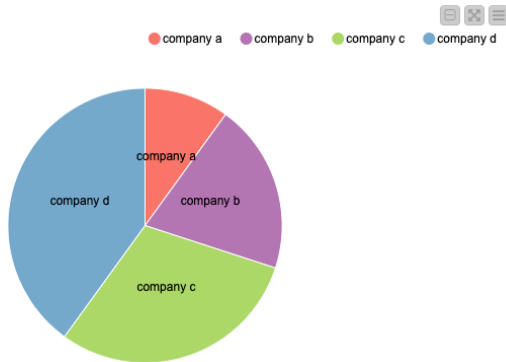
02: Coloring lines and bars on a chart demo



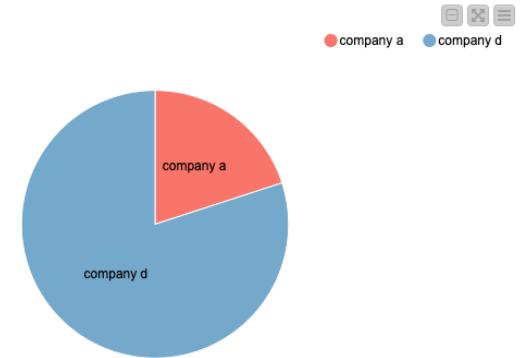
Adding filters on plots

- How do you add filtering in the interactive view to update the visuals?

Pie Chart



Pie Chart



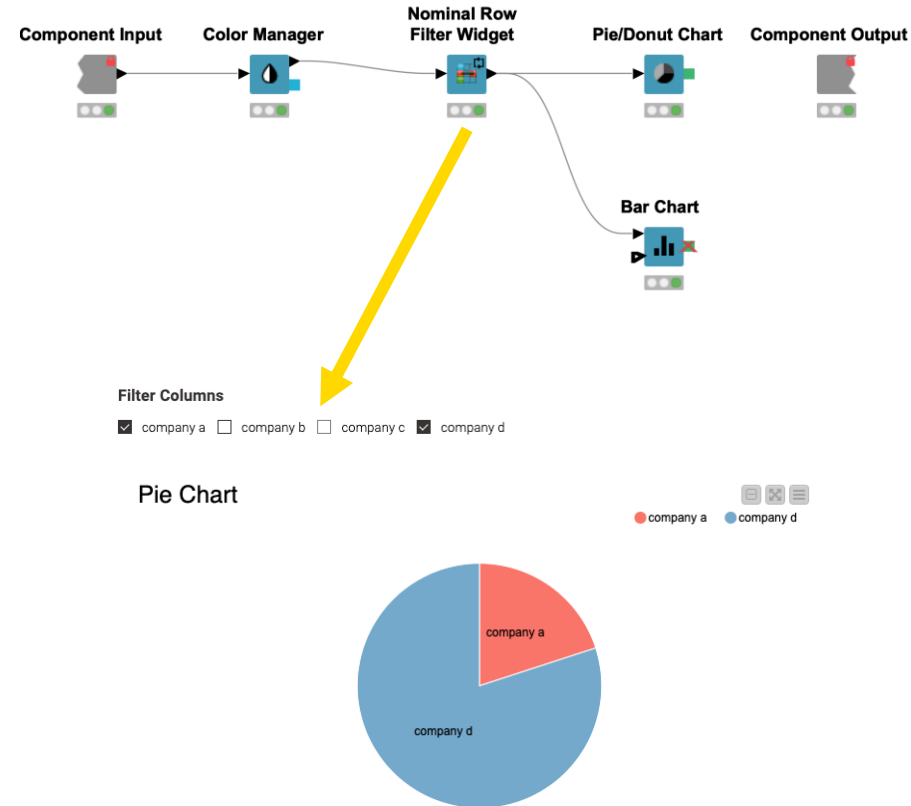
Filtering plots

- Approaches:

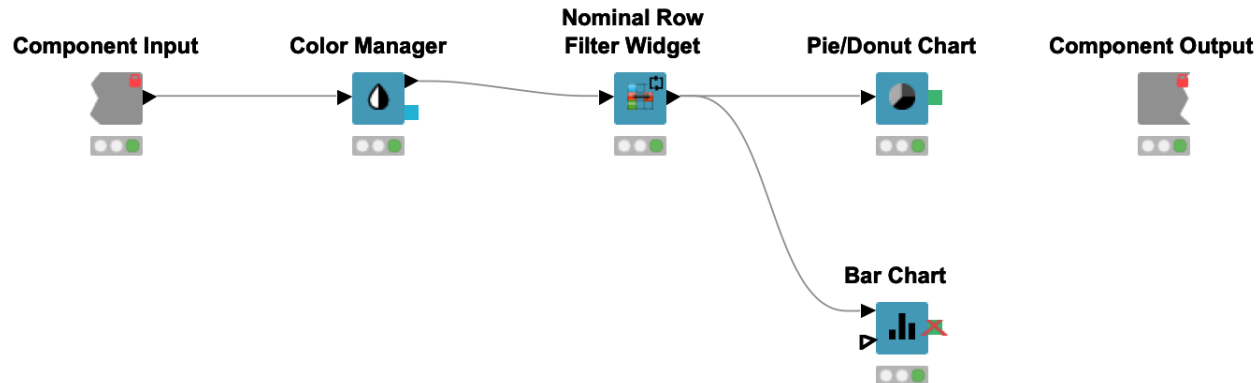
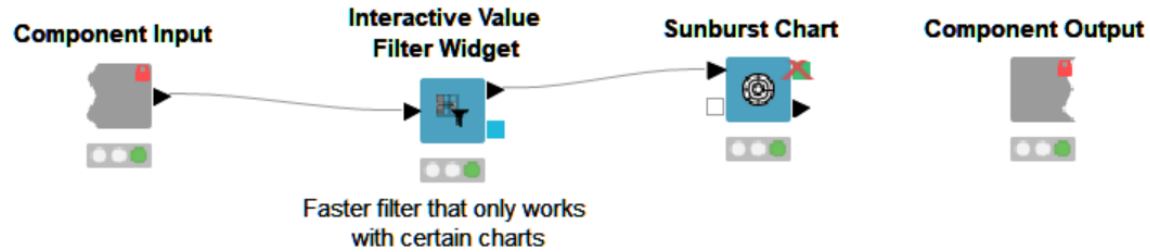
1. Try to use the **Interactive Value Filter Widget** node.

Not all visualizations in KNIME can make use of the Interactive Value Filter Widget node, so...

2. To filter for bar charts, pie charts, etc., use the **Nominal Row Filter Widget** or **Column Filter Widget** with Re-execution enabled.

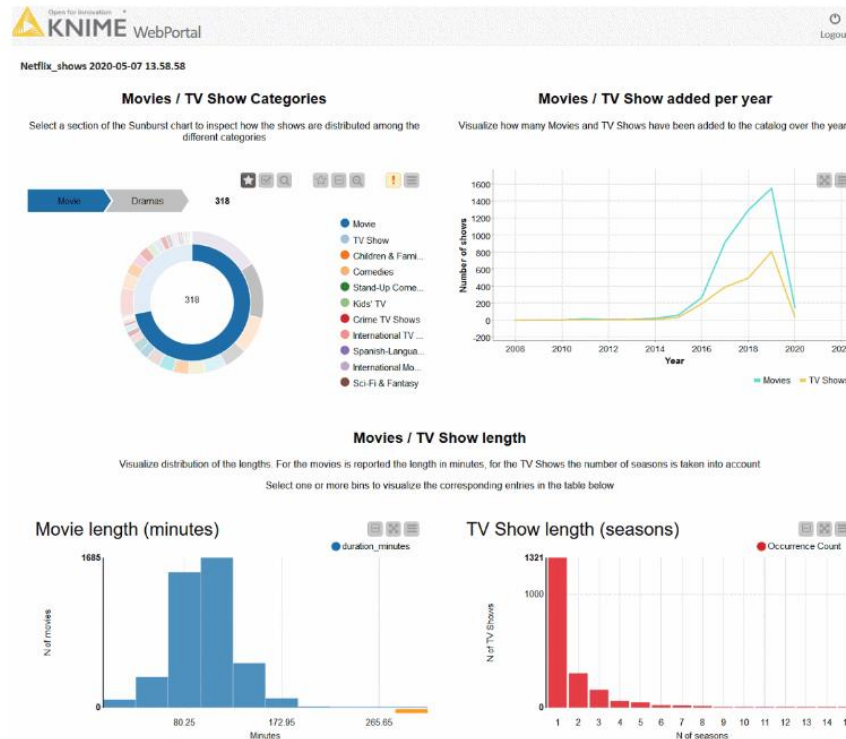


03: Adding interactive filtering demo



More resources for interactive dashboards

- <https://www.knime.com/blog/how-to-create-an-interactive-dashboard-in-three-steps-with-knime>



Accessing and visualizing geospatial data

- Enter and access location information and visualize it in an interactive map



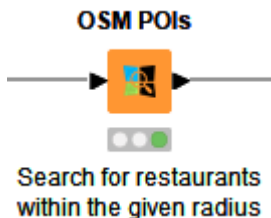
Geospatial Analytics Extension for KNIME

- KNIME Community Extension (Trusted) jointly developed by the Center for Geographic Analysis from Harvard and KNIME for
 - Reading and writing geospatial files (e.g., shapefiles, GeoJSON)
 - Performing spatial calculations (e.g., computing distances, joining)
 - Viewing data on an interactive map
- Access a collection of [example workflows](#) on the KNIME Community Hub
- View the KNIME Geospatial Analytics [playlist](#) on YouTube



OSM POIs node

- Returns the geolocations of selected entities (e.g., restaurant, parking) within a geometric boundary



Dialog - 4:8 - OSM POIs (Search for restaurants)

Geometry column

geometry(#1)

Input places tags

amenity

Input tags value

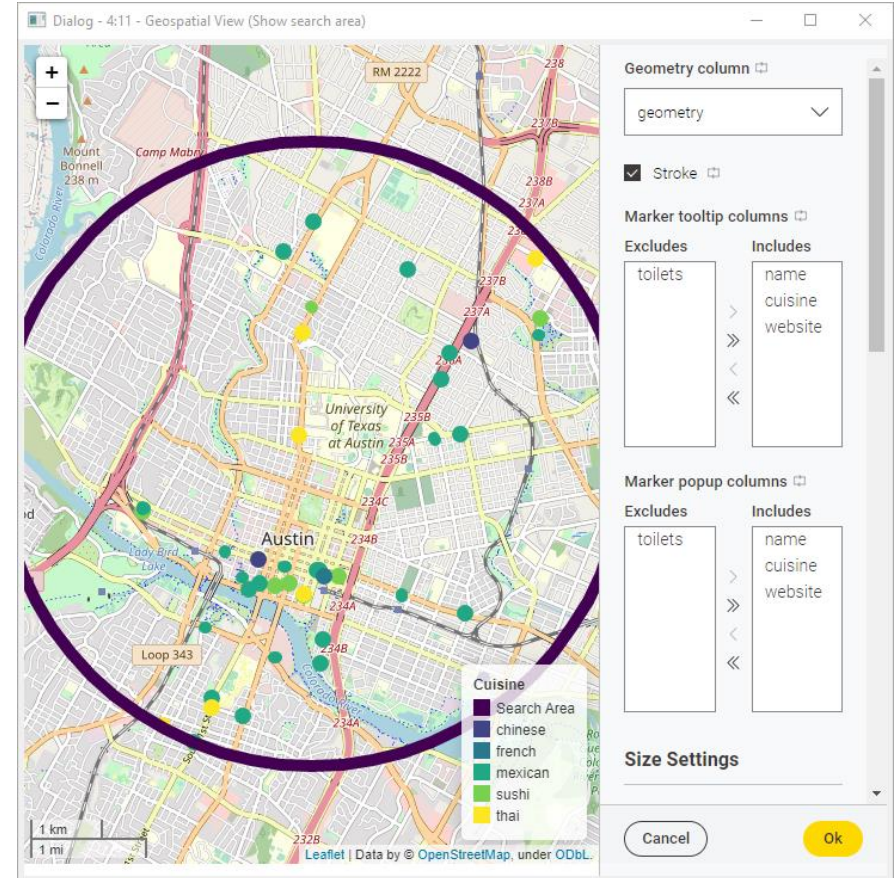
restaurant

Cancel Ok

amenity	name	geometry	addr:city	addr:h...	addr:po...	addr:str...	cuisine
restaurant	Mangia Pizza	POINT - EPSG:4326	?	?	?	?	?
restaurant	Maudie's Te...	POINT - EPSG:4326	Austin	2608	78703	West 7th S...	tex-mex
restaurant	Galaxy Cafe	POINT - EPSG:4326	?	1000	78703	West Lynn St	american
restaurant	Blue Star Ca...	POINT - EPSG:4326	Austin	?	78756	?	?
restaurant	Gusto Italia...	POINT - EPSG:4326	Austin	?	78756	?	italian

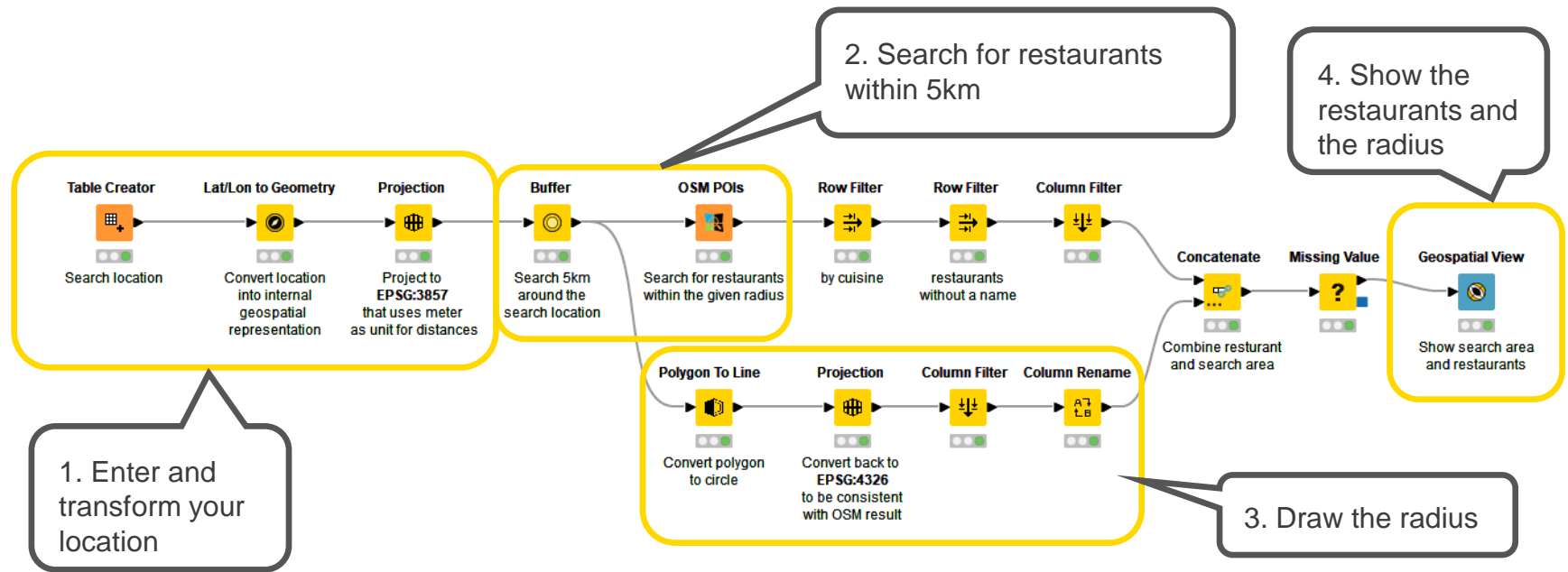
Geospatial View node

- Visualize geometric elements (e.g., points, line strings and polygons) in an interactive map and display additional information via tooltips, and shape/size/color of the markers

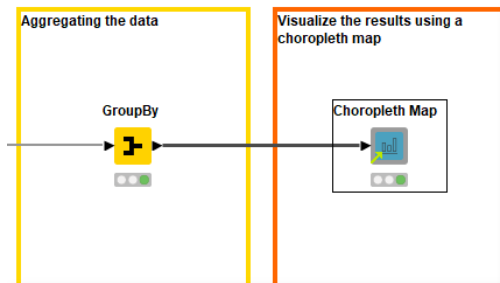
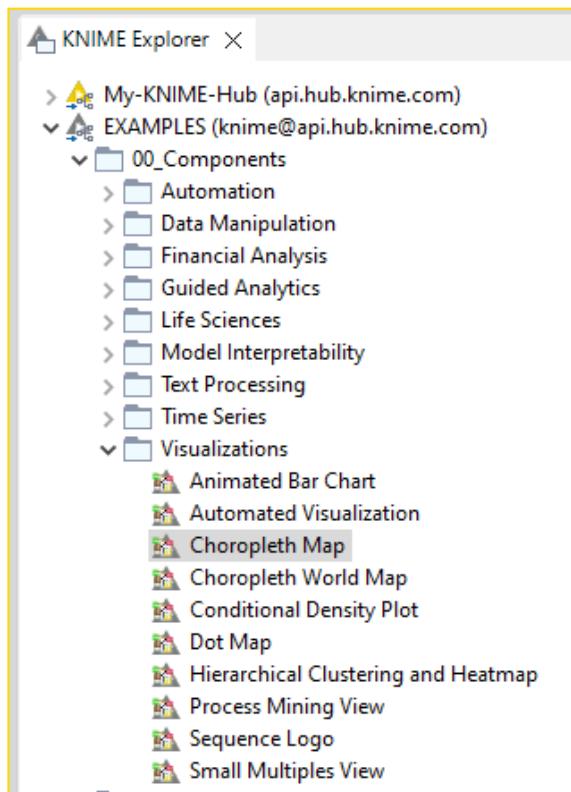


Finding restaurants nearby

- Example workflow for preprocessing geospatial data, retrieving restaurants nearby, and visualizing their proximity in an interactive map
- View and download the [POI Search](#) workflow from the KNIME Community Hub



Verified Component: Choropleth Map



Group table - 4:113 - GroupBy

File Edit Hilite Navigation View

Table "default" - Rows: 49 Spec - Columns: 2 Properties Flow Variables

Row ID	S State	D Profit
Row0	Alabama	5,786.825
Row1	Arizona	-3,427.925
Row2	Arkansas	4,008.687
Row3	California	76,381.387
Row4	Colorado	-6,527.858

Dialog - 4:112 - Choropleth Map

File

Options Flow Variables Memory Policy Job Manager Selection

Chart Title

Sales profit in US

Chart Subtitle

Theme

Viridis

Location Information Column

State

Color Axis Column

Profit

Region

United States of America (the)

Resolution

Province

OK Apply Cancel ?

Exercises: Section 2

1. Line Plot

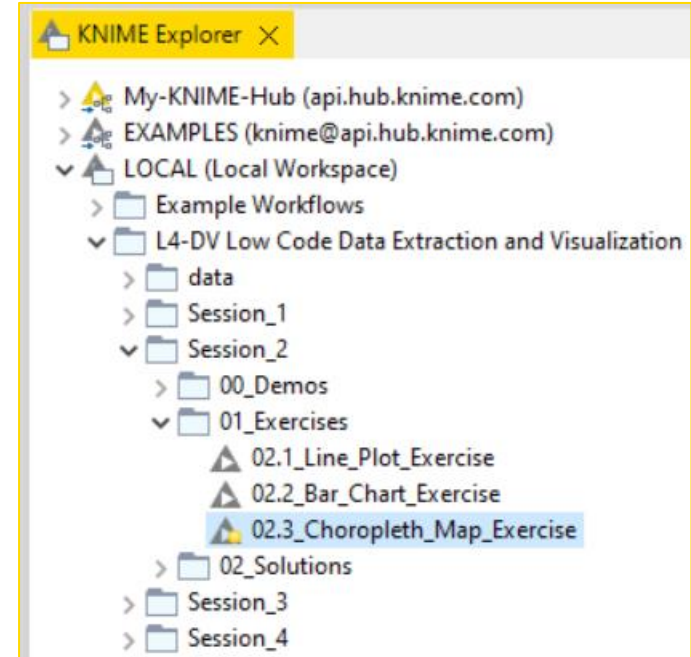
Using the data provided, transform it so that you can display multiple lines at once.

2. Bar Chart

Using the data provided, transform it to display the data using custom colors.

3. Choropleth Map

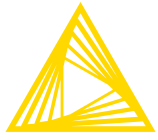
Using the data provided, transform it to create a map highlighting key assets.



Summary of Section 2

Now you should be able to:

1. Match correct visualization for a task.
2. Apply visualizations to common tasks.

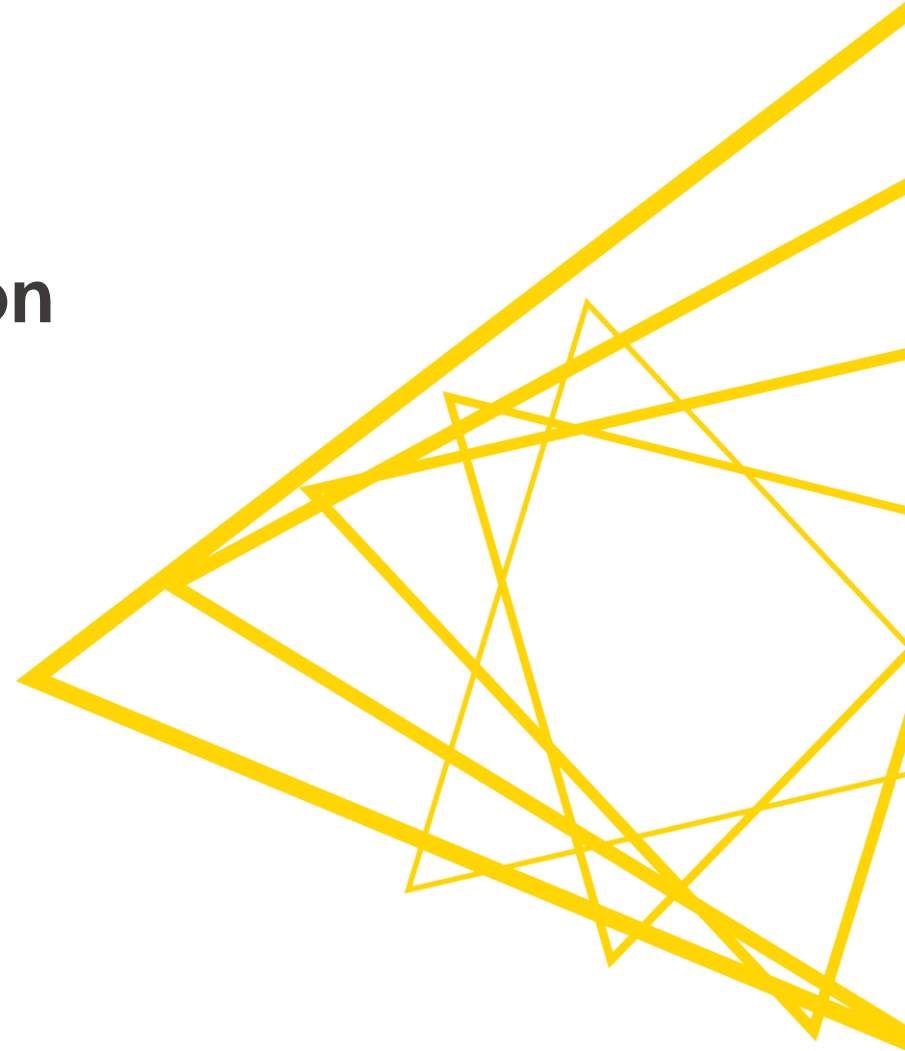


Open for Innovation

KNIME

[L4-DV] Low Code Data Extraction and Visualization

Section 3



Data Extraction

At the end of this section, you will be able to:

1. Parse PDFs.
2. Recognize and reproduce Regex.
3. Manipulate a data extraction tool.



What's your type (of PDF)?

Text-based
(Can click on text in pdf)

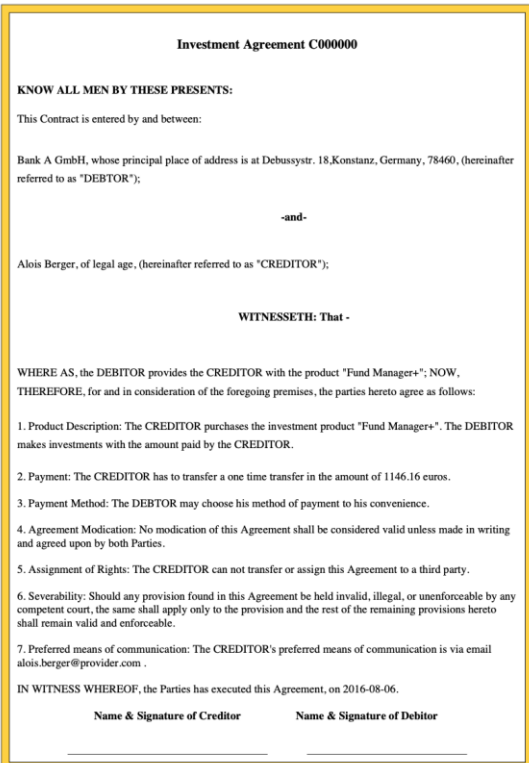
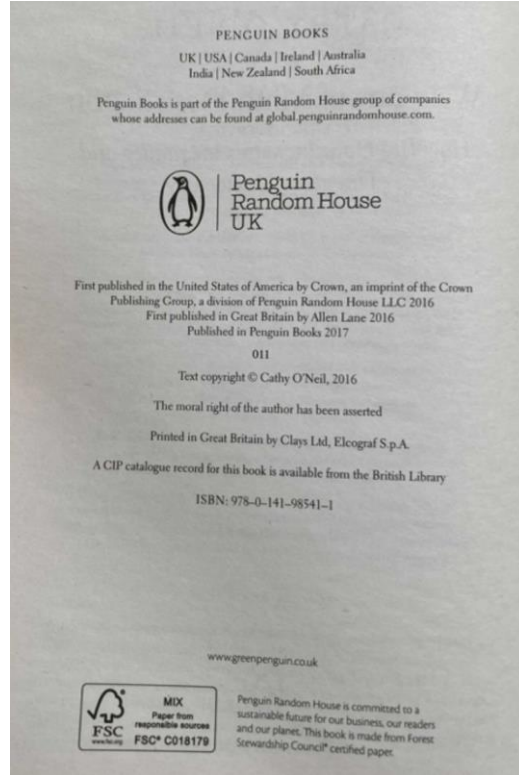


Image-based
(Cannot click on text in pdf)



Text-based PDF extraction: Tika Parser

- Tika Parser extracts text **with formatting** from text-based **sources**

Tika Parser



Metadata output table - 0:102 - Tika Parser (Read new)

Row ID	Content
Row0	Investment Agreement № C005775 KNOW ALL MEN BY THESE PRESENTS: This Contract is entered by and between: Bank A GmbH, whose principal place of address is at Debussystr. 18, Konstanz, Germany, 78460, (hereinafter referred to as "DEBTOR"); - and -
Row1	Investment Agreement № C005776

Data Location

Data Type

Metadata Desired

Dialog - 3:5 - Tika Parser (data from:)

Options | Flow Variables | Job Manager Selection | Memory Policy

Directory settings
Selected Directory: knime://knime.workflow/data
Search recursively ☐ Ignore hidden files ☒

File type settings
Choose which type to parse
☒ File Extension ☐ MIME-Type

Exclude
Filter
3g2
3gp
7z
a
aart
ac
acfm
afm
aif
aifc
aiff
Enforce exclusion ☒

Include
Filter
pdf
Enforce inclusion ☐

Output settings
Create error column ☐ New error output column Error Output
Extract embedded files to a directory
☐ Extract attachments and embedded files
☐ Extract inline images from PDFs
Selected Directory: /Users/victorpalacios
Encrypted files settings
☐ Parse encrypted files Enter password

Metadata
Coverage
Creator Tool
Comment
Metadata Date
Content

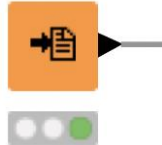
OK Apply Cancel ?

Example of how to use the [Tika Parser on the KNIME Hub](#)

Text-based PDF extraction: PDF Parser

- PDF Parser extracts text **without formatting** from text-based PDFs as a **document type**

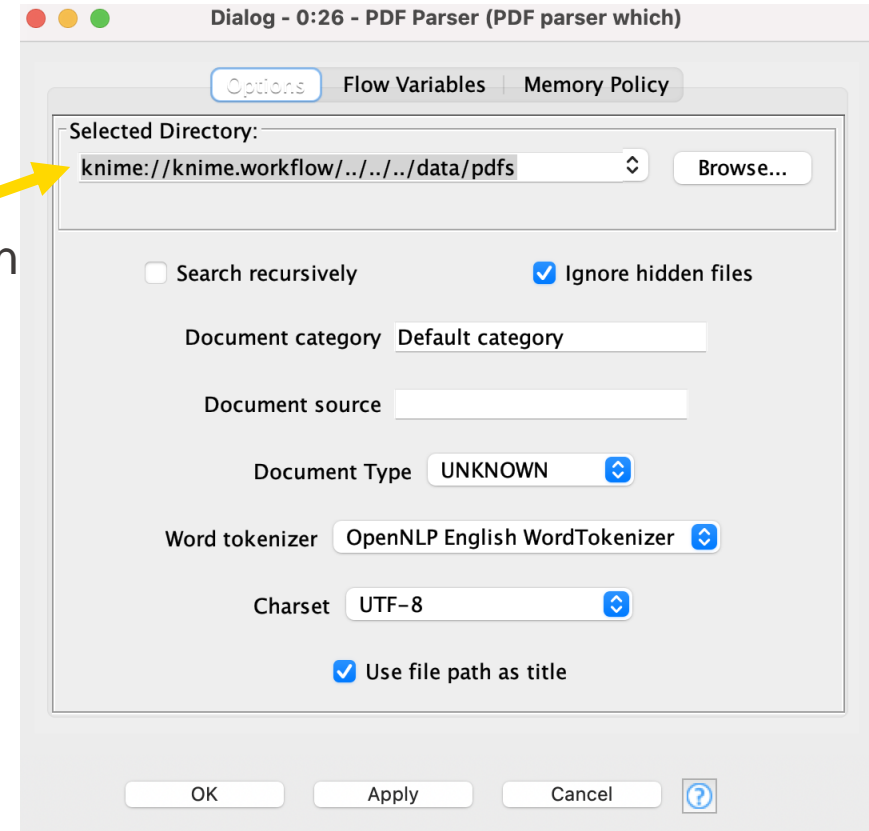
PDF Parser



Document

"/Users/victorpalacios/knime-workspace/L4-DA KNIME Analytics Platform for

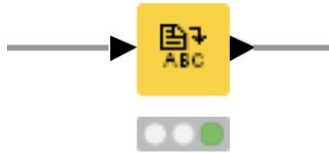
Data
Location



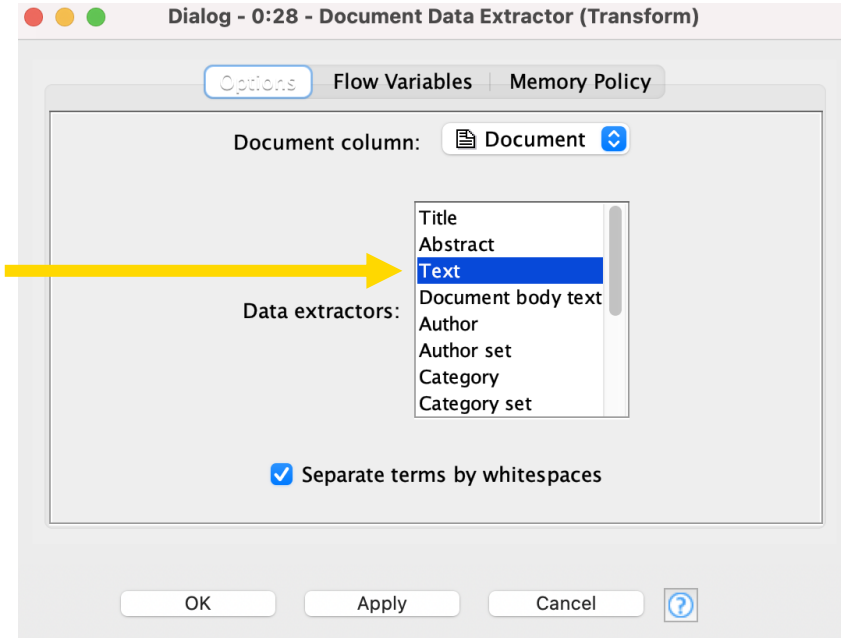
Text-based PDF extraction: Document Data Extractor

- Document Data Extractor extracts text from a document type.

Document Data Extractor



Metadata
Desired



A screenshot of the 'Enhanced data table - 0:28 - Document Data Extractor (Transform)' window. The table has 2 columns: 'Row ID' and 'Text'. The first row, 'Row0', contains the text: 'Investment Agreement C000006 KNOW ALL MEN BY THESE PRESENTS : This Contract is entered by and between : Bank A GmbH'.

Row ID	Text
Row0	Investment Agreement C000006 KNOW ALL MEN BY THESE PRESENTS : This Contract is entered by and between : Bank A GmbH

Tika Parser vs PDF Parser

	Tika Parser	PDF Parser
Versatility	reads many file types	reads only PDFs
Output Type	string	document
Output Text	text only	text and path
Formatting	kept	removed

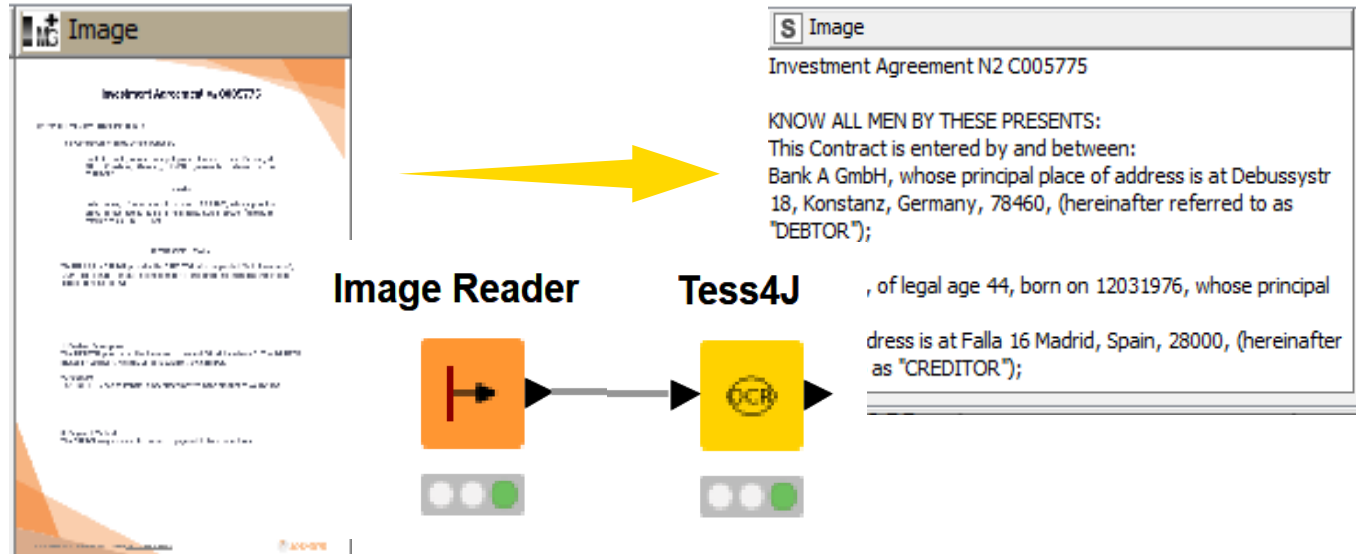
Tika Parser or PDF Parser?

- Use the Tika Parser node when...
 - You need the formatting preserved
 - You need to extract data from multiple sources including PDFs
- Use the PDF Parser node when...
 - You want the formatting removed
 - You are doing Natural Language Processing or Text Processing tasks

Image-based PDF extraction

Tess4J node

- Tess4J for **OCR** (Optical Character Recognition) only on **Windows**



Example of how to use the [Tess4J node on the KNIME Hub](#)

Image quality and structure determine output

High Quality Image
maximum differentiation between
background and target text
(i.e., white and black color scheme)

Low Quality Image
poor differentiation between
background and target text and tabular
(i.e., various shades of grey)

Image Input

Besides tweaking the configurations of the Tess4J node to the use case at hand, it is a good practice to preprocess input images thoroughly, if needed. In particular, Tesseract works best when images are sufficiently scaled up such that the pixel count of the x-height of characters is at least 20 pixels; images are correctly aligned and have a sufficiently high resolution; and any dark borders is removed, or they might be misinterpreted as characters [3]. The [KNIME Image Processing](#) extension includes several nodes for image cleaning, manipulation and transformation, and many [example workflows](#) can be found on the [KNIME Hub](#).

Lab	Result Content	Entered By	Received Time	Released
Anti-Müllerian Hormone		eIVF Connect	24/03/2022 10:34 AM	✓
Prolactin	8.1 ug/L	eIVF Connect	23/03/2022 10:16 PM	✓
TSH		eIVF Connect	23/03/2022 11:02 PM	✓
Progesterone	1.99 nmol/L	eIVF connect	24/03/2022 11:09 AM	✓
BETA hcg - Quantitative	0.34 IU/L	eIVF connect	24/03/2022 10:27 AM	✓
Progesterone	1.20 nmol/L	eIVF connect	24/03/2022 11:12 AM	✓
BETA hcg - Quantitative	0.10 IU/L	eIVF connect	24/03/2022 10:42 AM	✓
Progesterone	0.93 nmol/L	eIVF connect	24/03/2022 11:10 AM	✓
BETA hcg - Quantitative	0.30 IU/L	eIVF connect	24/03/2022 10:27 AM	✓
E2*	556.7 pmol/L	eIVF connect	24/03/2022 09:35 AM	✓
LH*	7.7 IU/L	eIVF connect	24/03/2022 09:38 AM	✓
PROG*	1.3 nmol/L	eIVF connect	24/03/2022 09:11 AM	✓

KNIME Output

Besides tweaking the configurations of the Tess4J node to the use case at hand, it is a good practice to preprocess input images thoroughly, if needed. In particular, Tesseract works best when images are sufficiently scaled up such that the pixel count of the x-height of characters is at least 20 pixels; images are correctly aligned and have a sufficiently high resolution; and any dark borders is removed, or they might be misinterpreted as characters [3]. The [KNIME Image Processing](#) extension includes several nodes for image cleaning, manipulation and transformation, and many [mple workflows](#) can be found on the [KNIME Hub](#).

Arm MuHenan Hum Pm'achn TSH Manama: am he: -
mam-v. Pvagexlsvuns BETA hcg mama-v: ngeslemne am
hcg mam»: E2* M" van:- 2' ug/L ' as nmuVL u 93 mum/L u
an m/L 555 7 max/L eWF Emma-2A 2un3/2u22 m 3o AM 2
1

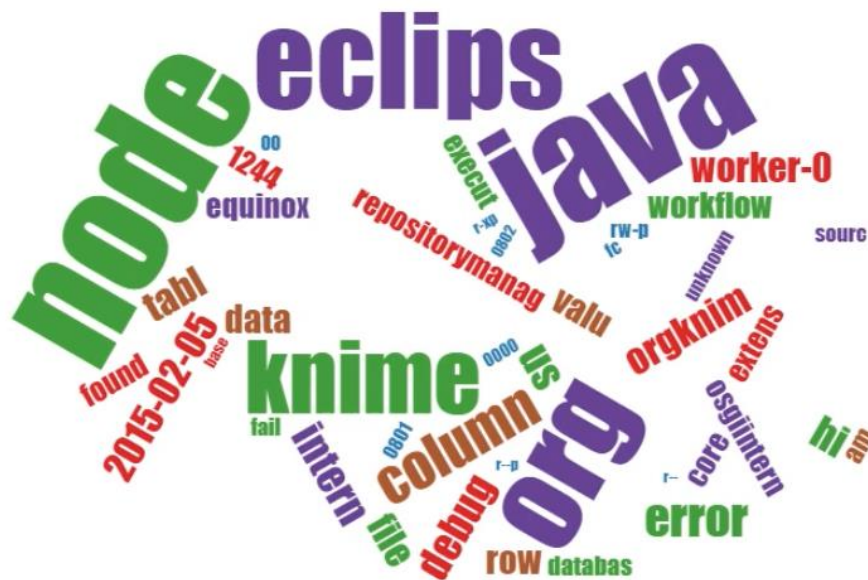
[See KNIME Forum discussion](#)

After successful extraction... What's next?

Common tasks after extraction:

1. Table Creation
2. Topics Labeling or Modeling
3. Target Text Analysis

WS FORM F-6							
PRELIMINARY LOCAL CLIMATOLOGICAL DATA							
LATITUDE 44 DEGREES 16 MINUTES NORTH						LONGITUDE	
TEMPERATURE (°F)							
						DEGREE DAYS	
DAY	MAX	MIN	AVG	NORM	DEPART	HEAT	COOL
1	32	27	30	37	-7	35	0
2	42	31	37	37	0	28	0
3	45	41	43	37	6	22	0
4	48	41	45	36	9	20	0
5	50	43	47	36	11	18	0
6	52	41	47	35	12	18	0
7	53	41	47	35	12	18	0
8	52	38	45	34	11	20	0
9	51	38	45	34	11	20	0
10	48	39	44	34	10	21	0



IN WITNESS WHEREOF, the Parties has executed this Agreement, on 2016-08-06.

Name & Signature of Creditor

Name & Signature of Debtor

Table creation

- Analyze weather data from a PDF table

Text-based PDF
(with table)

WS FORM F-6

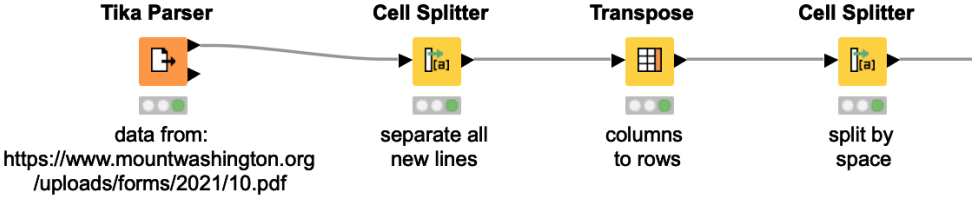
PRELIMINARY LOCAL CLIMATOLOGICAL DATA								
LATITUDE					LONGITUDE			
44 DEGREES 16 MINUTES NORTH					71 DEGREES 18 MINUTES WEST			
TEMPERATURE (°F)								
						DEGREE DAYS		
DAY	MAX	MIN	AVG	NORM	DEPART	HEAT	COOL	
1	32	27	30	37	-7	35	0	
2	42	31	37	37	0	28	0	
3	45	41	43	37	6	22	0	
4	48	41	45	36	9	20	0	
5	50	43	47	36	11	18	0	
6	52	41	47	35	12	18	0	
7	53	41	47	35	12	18	0	
8	52	38	45	34	11	20	0	
9	51	38	45	34	11	20	0	
10	48	39	44	34	10	21	0	

Tika Parser Output
(1 row)

Metadata output table - 3:5 - Tika Parser (data from:)	
File	Edit Hilite Navigation View
Table "default" - Rows: 1 Spec - Column: 1 Properties Flow Variables	
Row ID	S Content
Row0	WS FORM F-6 STATION MOUNT WASHINGTON OBSERVATORY PRELIMINARY LOCAL CLIMATOLOGICAL DATA MONTH YEAR OCTOBER 2021 LATITUDE LONGITUDE GROUND ELEVATION (H) STANDARD TIME 44 DEGREES 16 MINUTES NORTH 71 DEGREES 18 MINUTES WEST 6280 FT EASTER TEMPERATURE (°F) PRECIPITATION (IN) WIND (MPH) SUNSHINE SKY DEGREE DAYS TOTAL SNOW & SNOW/ICE ON AVG FASTEST MILE (MINUTES) COVER DAY MAX MIN AVG NORM DEPART HEAT COOL (EQUV) ICE GROUND-7AM SPEED : 1 32 27 30 37 -7 35 0 0 10 1246 2 42 31 37 37 0 28 0 0 10 1246 3 45 41 43 37 6 22 0 0 10 12 4 48 41 45 36 9 20 0 0 10 12 5 50 43 47 36 11 18 0 0 10 12 6 52 41 47 35 12 18 0 0 10 12 7 53 41 47 35 12 18 0 0 10 12 8 52 38 45 34 11 20 0 0 10 12 9 51 38 45 34 11 20 0 0 10 12 10 48 39 44 34 10 21 0 0 10 12

Workflow Output
(1 table)

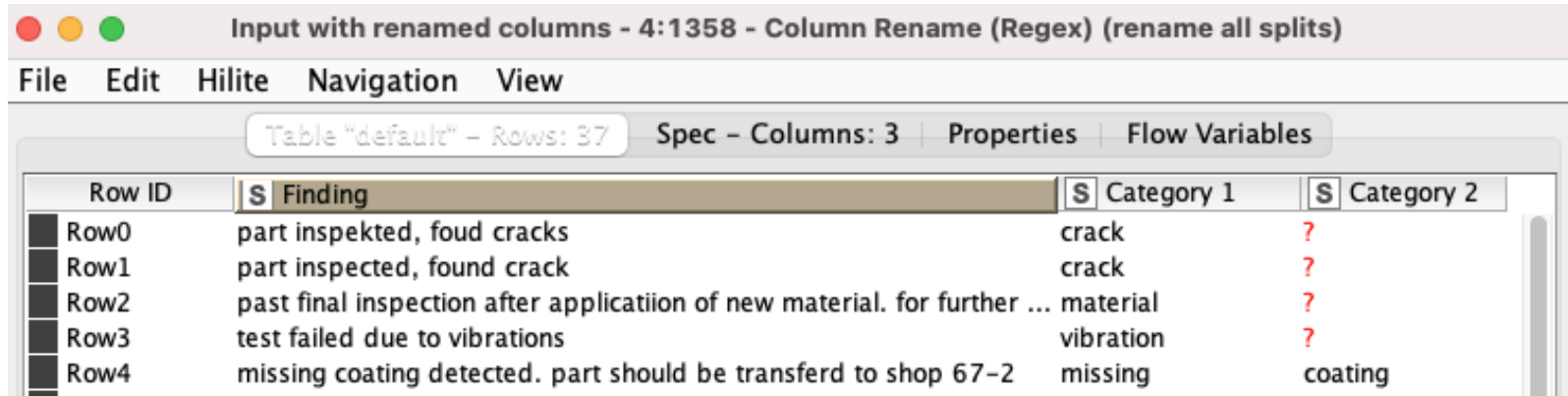
Filtered table - 3:1366 - Column Filter (remove)									
Table "default" - Rows: 31 Spec - Columns: 8 Properties Flow Variables									
Row ID	S Day	S Max	S Min	S Avg	S Norm	S Depart	S Heat	S Cool	
Row14	1	32	27	30	37	-7	35	0	
Row15	2	42	31	37	37	0	28	0	
Row16	3	45	41	43	37	6	22	0	
Row17	4	48	41	45	36	9	20	0	
Row18	5	50	43	47	36	11	18	0	
Row19	6	52	41	47	35	12	18	0	
Row20	7	53	41	47	35	12	18	0	
Row21	8	52	38	45	34	11	20	0	
Row22	9	51	38	45	34	11	20	0	
Row23	10	48	39	44	34	10	21	0	



[See KNIME Forum discussion](#)

Topic labeling

- Categorize each row with one or more **known** labels
 - Example: Look for “crack, material, vibration, missing, and coating” in the text:



Input with renamed columns - 4:1358 - Column Rename (Regex) (rename all splits)

File Edit Hilite Navigation View

Table "default" - Rows: 37 Spec - Columns: 3 Properties Flow Variables

Row ID	Finding	Category 1	Category 2
Row0	part inspekted, foud cracks	crack	?
Row1	part inspected, found crack	crack	?
Row2	past final inspection after applicatiion of new material. for further ...	material	?
Row3	test failed due to vibrations	vibration	?
Row4	missing coating detected. part should be transferd to shop 67-2	missing	coating

[See KNIME Forum discussion](#)

Topic modeling

- Discover **unknown** topics in your text
 - Example: Parse news feeds and try to automatically detect topics.

Document	S ▲ Assi...
"york mets apologize mascot met fan finger"	topic_0
"stanley cup final penguin 2-0 series lead l...	topic_0
"forget story pushball game giant bewitch b...	topic_0

Probably
“sports” is
the topic

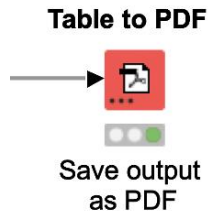
Document	S ▲ Assi...
"brine turkey mad hunky meat poultry bri...	topic_3
"motor city bbq sauces"	topic_3
"smoked aubergine recipe eggplant miso ...	topic_3

Probably
“food” is
the topic

[See KNIME Hub example](#)

Bonus: Table to PDF node

- Create very basic PDF report
 - With report title, author, & date



Dialog - 0:2389 - Table to PDF (Create the)

File

Settings Flow Variables Job Manager Selection

Number of Rows

No. of rows to report: 1.000

Output location

Write to: Relative to Current workflow data area

File: ./Testflows_Report_2021-11-18.pdf Browse...

Write options ☐ Create missing folders If exists: ☒ overwrite ☐ fail

⚠ There exists a file with the specified path './Testflows_Report_2021-11-18.pdf' that will be overwritten.

Design File

☒ Delete rptdesign file

Basic Settings

Report Title: Testflows Report

Report Author: Name.Surname

Image Settings

☒ Create images with default size

☐ Create images with specified size

Image width: 100

Image height: 100

Image Scaling: ☒ Scale to size ☐ Paint to size

⚠ The "path" parameter is controlled by a variable.

OK Apply Cancel ?

Data Extraction

At the end of this section, you will be able to:

1. Parse PDFs.
2. Recognize and reproduce Regex.
3. Manipulate a data extraction tool.



Regex (Regular Expressions)

- What is it?
 - A special set of characters that can help with...
 - Searching through text
 - Replacing text
 - Filtering / Splitting columns
- Why use it?
 - You know there is a certain **pattern** in your documents (such as date) but it is time-consuming to find it manually.
- Let's see some pattern examples.

Regex possible targets

Dates with various formats

2022-2-22

7/9/2010

12.25.2010

Regex example

2022-2-22

`[0-9]{4}-[0-9]{1,2}-[0-9]{1,2}`

Regex example

2022-2-22

Any
Number

$[0-9]\{4\}-[0-9]\{1,2\}-[0-9]\{1,2\}$

Occurring 4
times

Occurring 1
or 2 times

Mini regex cheat sheet

Category	Expression	Will find	Example	Example extraction:
Wildcard	.	any character	F.r	For, Fur, Far, etc.
(Unknown) Quantity	+ or *	repeated pattern	[0-9]+	77, 9, 123, etc.
Letters	[A-Z]	any uppercase letter	[a-zA-Z]	A, a, B, b, etc.
Or		Pattern on left or pattern on right of	[0-9]{4} 0-9]{2}	4 digit sequence or 2 digit sequence
Capture Group	()	a group to capture	(.*)([0-9]{4})(.*)	Captures 3 groups: 1. (any sequence) 2. (4 digits) 3. (any sequence)

[See full regex documentation](#)

KNIME Knowledge Check 01

- What regex can capture the dates or IDs in these sentences:

Date

12-4-2002 is the date

Today's date is: 12/4/2002

IDs

The id is D0000

D001 is the ID

Hint: You can use a combination of [0-9], period wildcard, and the repeated pattern expression +

KNIME Knowledge Check 01: Solution

Date

12-4-2002 is the date

Today's date is: 12/4/2002

[0-9]{2}.[0-9]{1}.[0-9]{4}

or

[0-9]+.[0-9]+.[0-9]+

IDs

The id is D0000

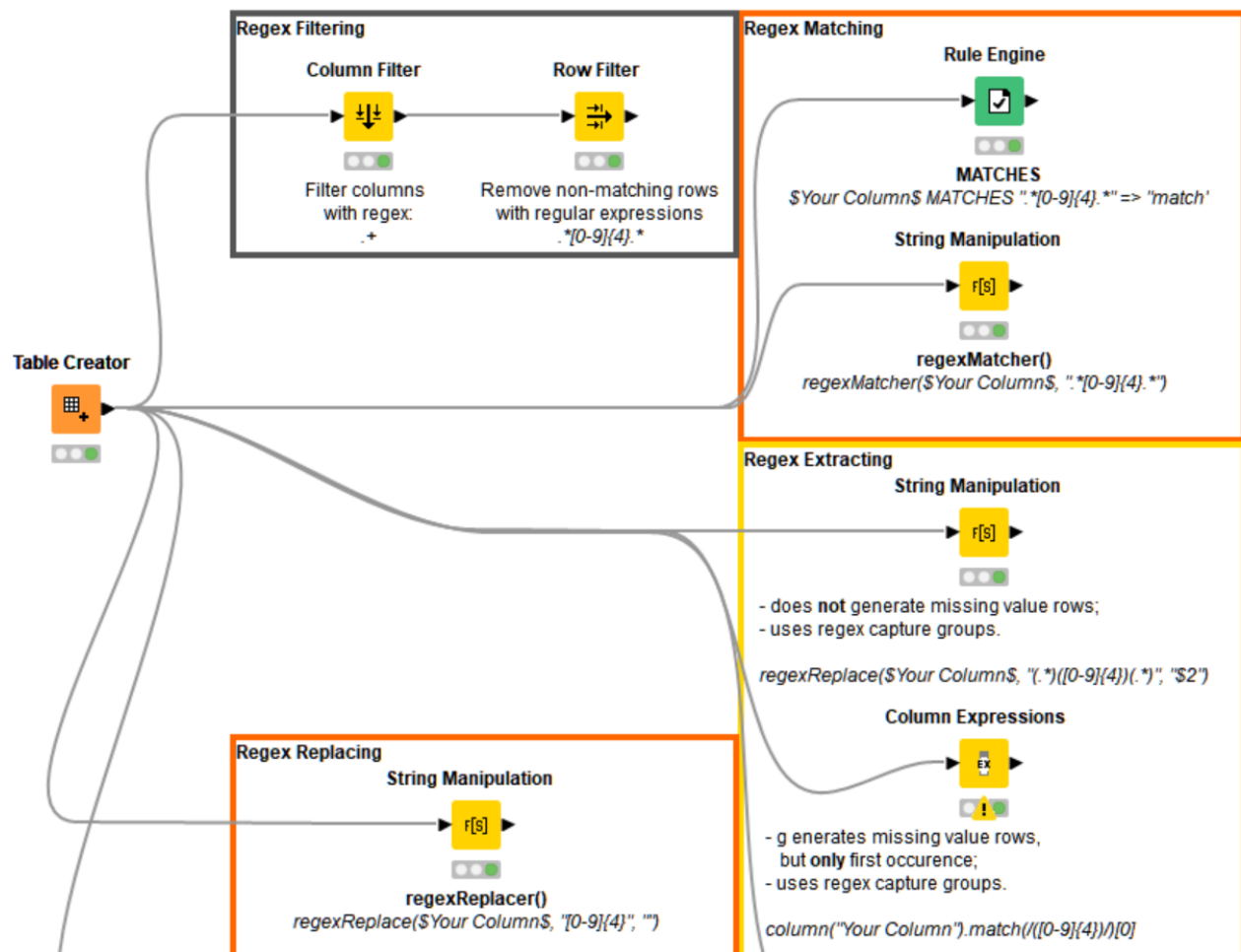
D001 is the ID

D[0-9]{3,4}

or

D[0-9]+

01: Various Examples of Regex Demo



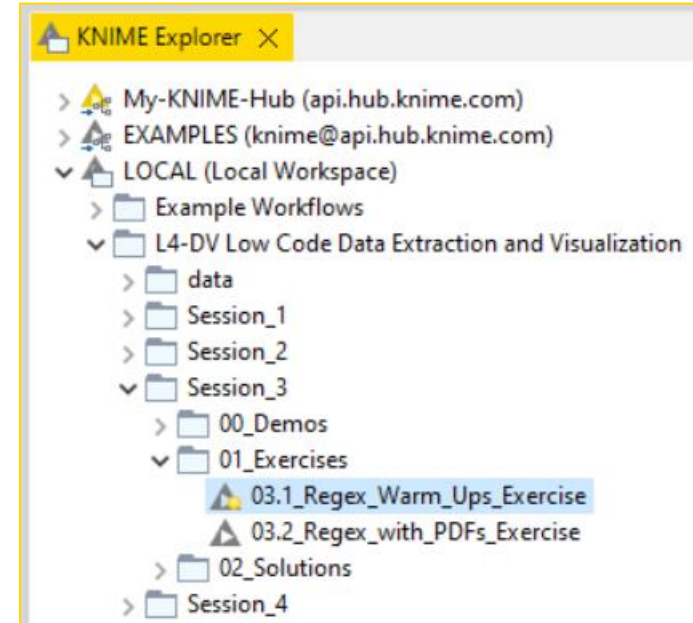
Exercises: Section 3

1. Regex Warm Ups

Using the data provided, extract information using regex.

2. Regex with PDFs

Using the PDFs provided, parse each PDF and then extract the relevant information using regex.



KNIME Knowledge Check 02

- Write regex that can capture the date-like and true date entities in these sentences:

Date-Like

12-4-2002 is the date.

Today's month and year is:
12-2005.

Date

12-4-2002 is the date.

Today's date is:
December-14-2004

Hint: Use the “or” expression |

KNIME Knowledge Check 02: Solution

Date-Like

12-4-2002 is the date.

Today's month and year is:
12-2005.

`[0-9]+\.[0-9]+\.[0-9]+|[[0-9]+-[0-9]]{4}`

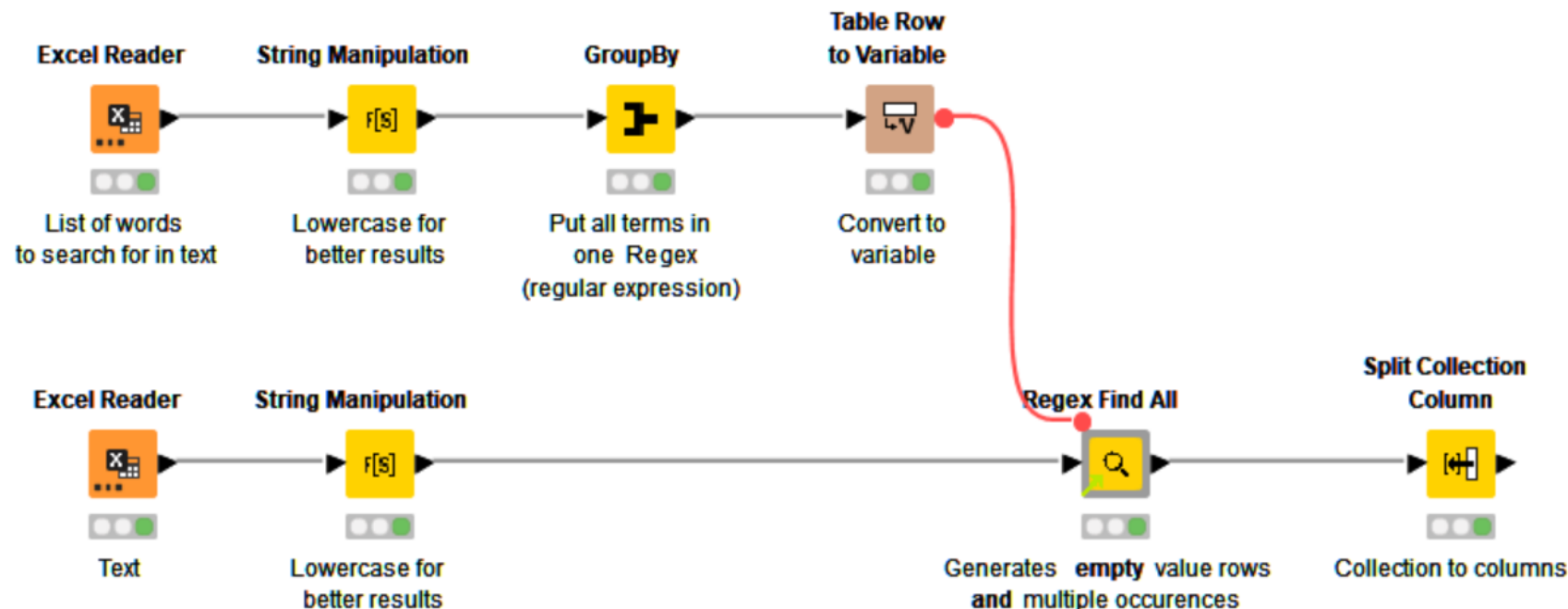
Date

12-4-2002 is the date.

Today's date is:
December-14-2004

`[a-zA-Z0-9]+\.[0-9]+\.[0-9]+`

02: Topic Labeling Regex demo



[Check out solutions from the KNIME community as well](#)

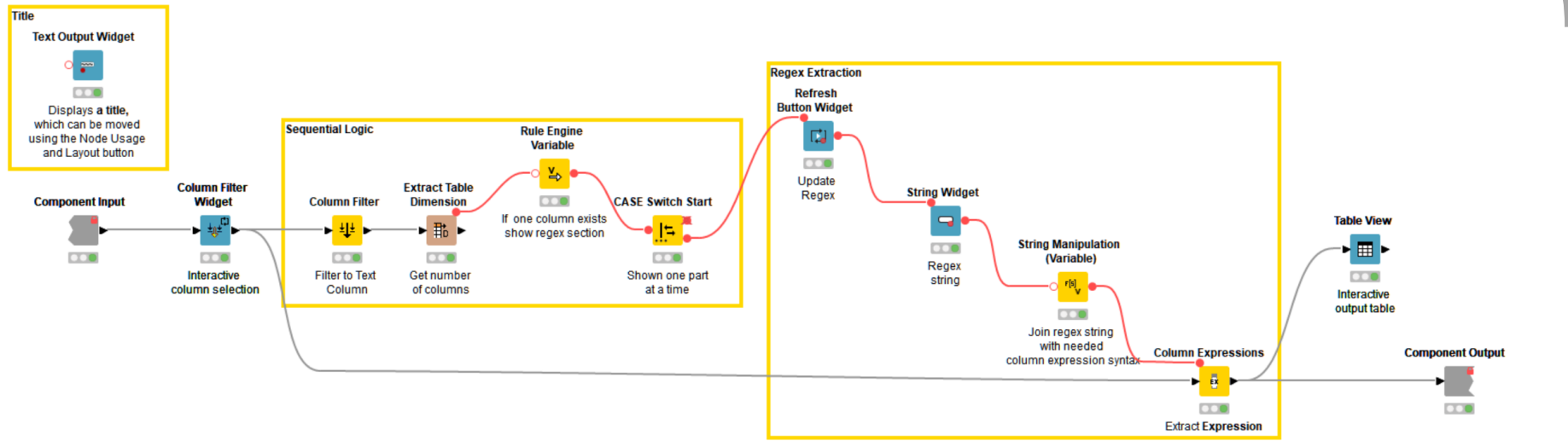
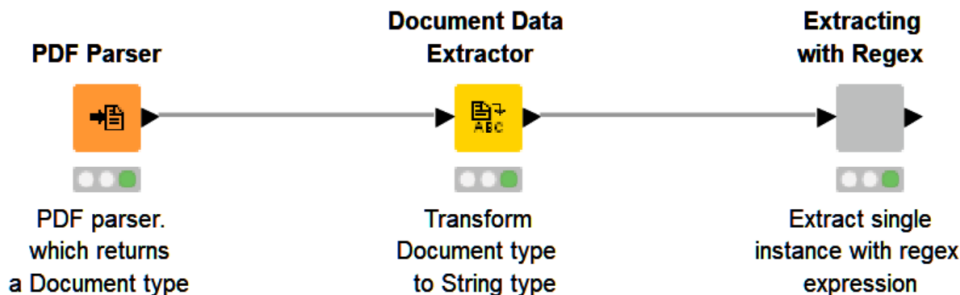
Data Extraction

At the end of this section, you will be able to:

1. Parse PDFs.
2. Recognize and reproduce Regex.
3. Manipulate a data extraction tool.



03: Parsing PDFs demo



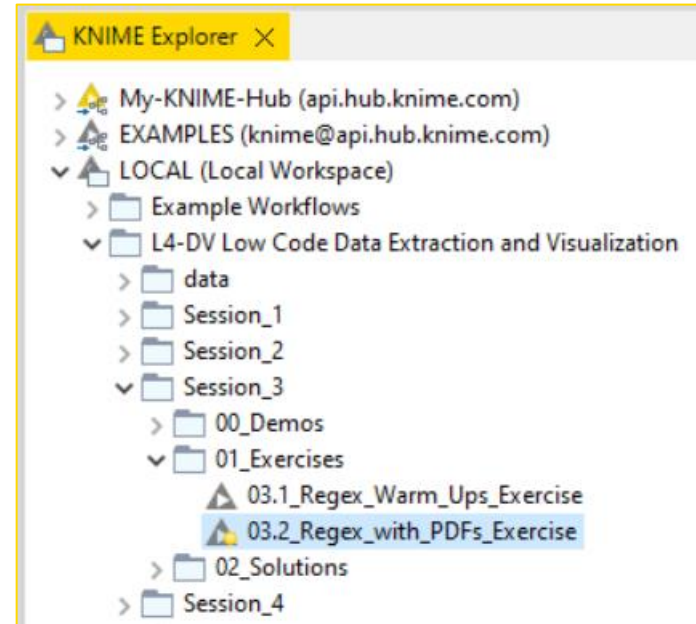
Exercises: Section 3

1. Regex Warm Ups

Using the data provided, extract information using regex.

2. Regex with PDFs

Using the PDFs provided, parse each PDF and then extract the relevant information using regex.



Why create a parsing tool?

- Convenience
 - Consolidate PDF extraction and common Regex techniques in 1 place
- Hide complex steps such as:
 - Extracting data from the PDF
 - Using the Column Expression node without the extra syntax
 - Making the process reusable
- Go from PDF input to target text analysis quickly and efficiently

Summary of Section 3

Now you should be able to:

1. Parse PDFs.
2. Recognize and reproduce Regex.
3. Manipulate a data extraction tool.



Open for Innovation

KNIME

[L4-DV] Low Code Data Extraction and Visualization

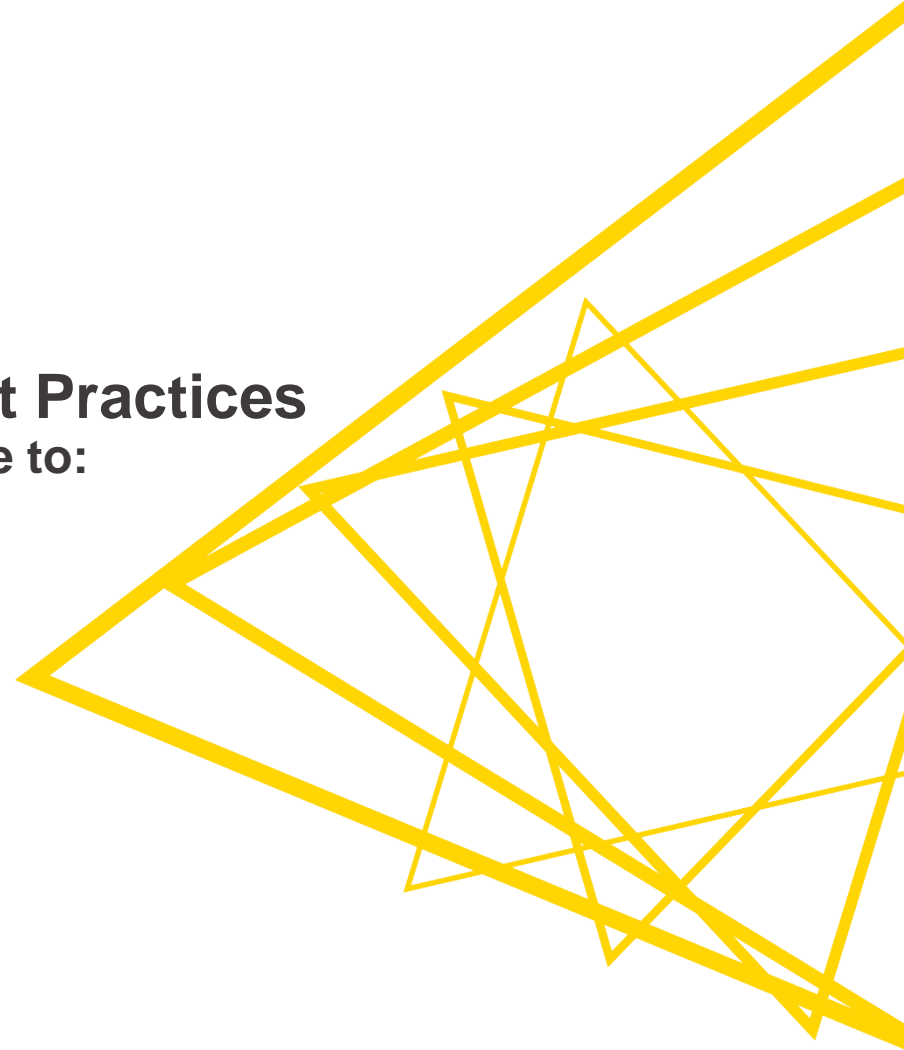
Section 4



Data Quality & Visualization Best Practices

At the end of this section, you will be able to:

1. Assess data quality via outlier detection
2. Apply best practices for data visualization



Data quality via outlier detection



Detecting wrong (fake?) contracts

- Detect exceptional IDs, prices, or dates in large databases

Investment Agreement C000004

KNOW ALL MEN BY THESE PRESENTS:

This Contract is entered by and between:

Bank A GmbH, whose principal place of address is at Debusstr. 18, Konstanz, Germany, 78460, (hereinafter referred to as "DEBTOR");

-and-

Mira Gleich, of legal age, (hereinafter referred to as "CREDITOR");

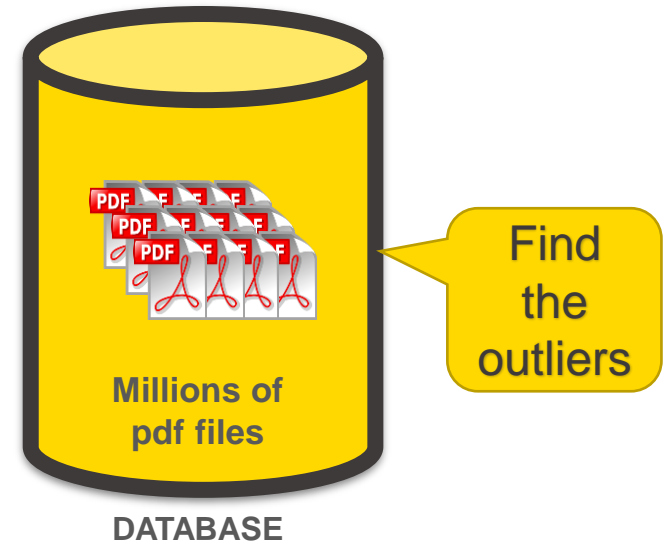
WITNESSETH: That -

WHERE AS, the DEBTOR provides the CREDITOR with the product "Private Investment", NOW, THEREFORE, for and in consideration of the foregoing premises, the parties hereto agree as follows:

1. Product Description: The CREDITOR purchases the investment product "Private Investment". The DEBTOR makes investments with the amount paid by the CREDITOR.
2. Payment: The CREDITOR has to transfer a one time transfer in the amount of 799.06 euros.
3. Payment Method: The DEBTOR may choose his method of payment to his convenience.
4. Agreement Modification: No modification of this Agreement shall be considered valid unless made in writing and agreed upon by both Parties.
5. Assignment of Rights: The CREDITOR can not transfer or assign this Agreement to a third party.
6. Severability: Should any provision found in this Agreement be held invalid, illegal, or unenforceable by any competent court, the same shall apply only to the provision and the rest of the remaining provisions hereto shall remain valid and enforceable.
7. Preferred means of communication: The CREDITOR's preferred means of communication is via email mira.gleich@provider.com.

IN WITNESS WHEREOF, the Parties has executed this Agreement on 2016-04-05.

Name & Signature of Creditor Name & Signature of Debtor



The Data

- From contracts extract all relevant data into a table



Row ID	S contract_id	31 date	D payment
Row1	C000002	2016-01-15	860.4
Row9	C000006	2016-01-20	1,647.7
Row8	C000001	2016-02-11	1,996.55
Row0	C000003	2016-02-26	1,184.51
Row3	C000005	2016-03-23	1,180.31
Row4	C000004	2016-04-05	799.06
Row7	C000000	2016-08-06	1,146.16
Row2	C000008	2016-10-03	912.72
Row5	C000007	2016-11-11	690.21
Row6	C000009	2016-12-03	475.94

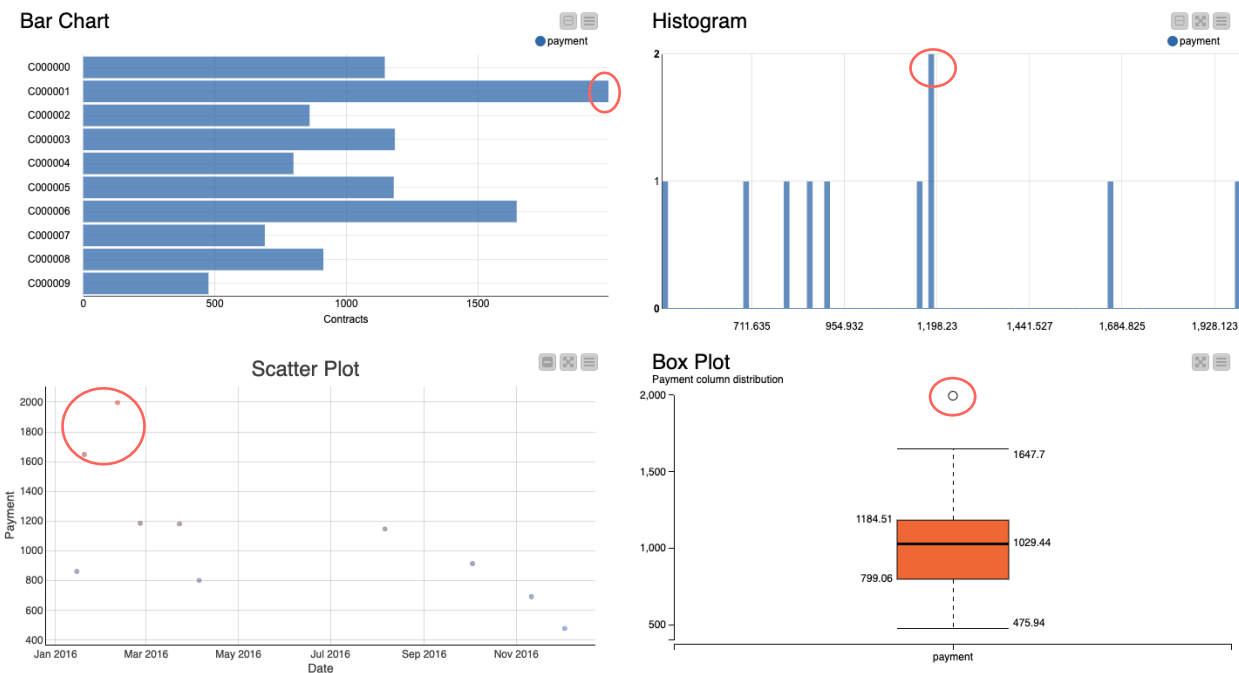
3 Techniques for Outlier Detection

- Visualizations
- Statistics
- Machine Learning

Visual outlier detection - comparative dashboard

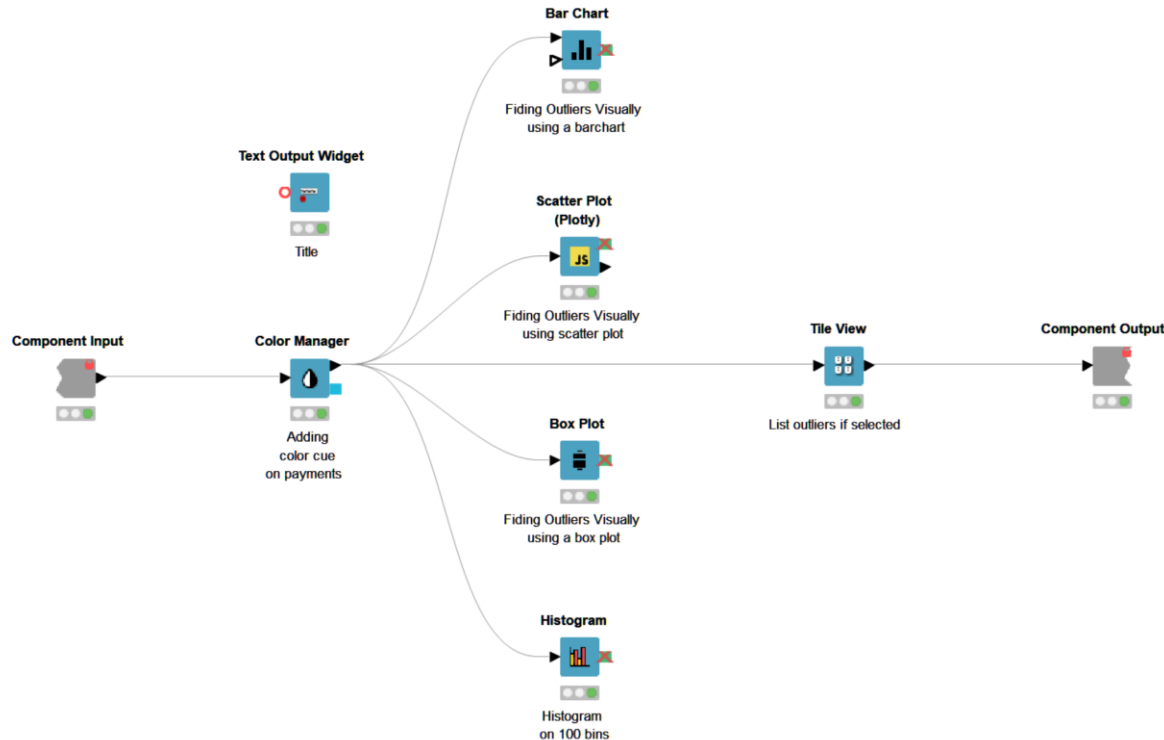
Bar chart, histogram, scatter plot, & box plot

Outlier Detection via Visual Exploration



Creating the visual dashboard

- Simply placing view nodes into a component lets us visualize them as a dashboard



Should you use visualization techniques?

Pros:

- Aesthetically pleasing
- Outliers are easy to spot and interpret

Cons:

- Can only investigate 1~2 dimensions at a time
- Manual checking
- Manual selection for further exploration

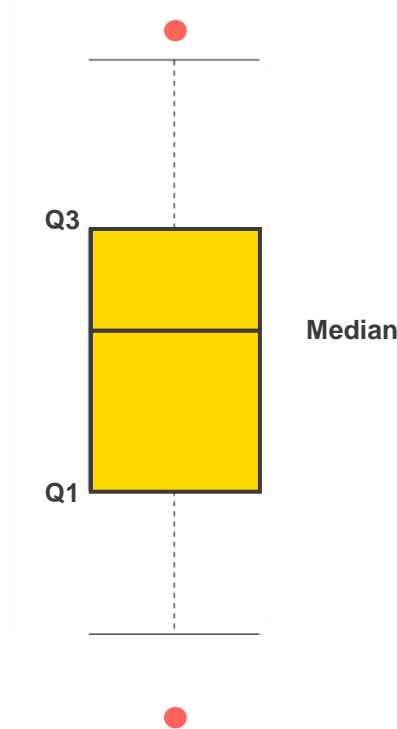
Is it possible to automate this process?

3 Techniques for Outlier Detection

- Visualizations
- Statistics
- Machine Learning

Finding Outliers: Statistics

IQR Technique



1. Find median of data to split data into a lower and upper half.
2. Find median of upper half of data (Q3)
3. Find median of lower half of data (Q1)
4. Calculate IQR and choose k.
$$\text{IQR} = Q3 - Q1$$
$$k = 1.5 \text{ (usually)}$$
5. Point is an outlier if:
$$\text{Point} > Q3 + k * \text{IQR}$$

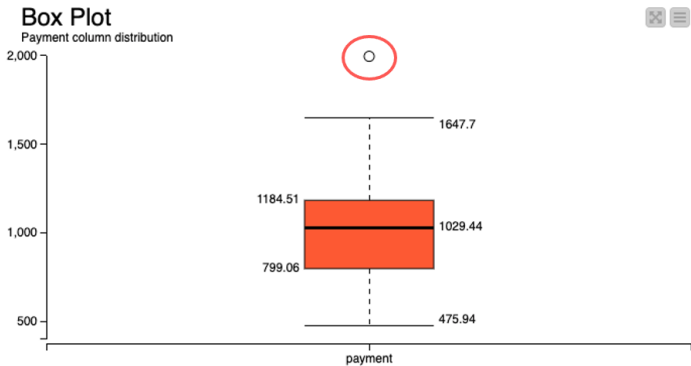
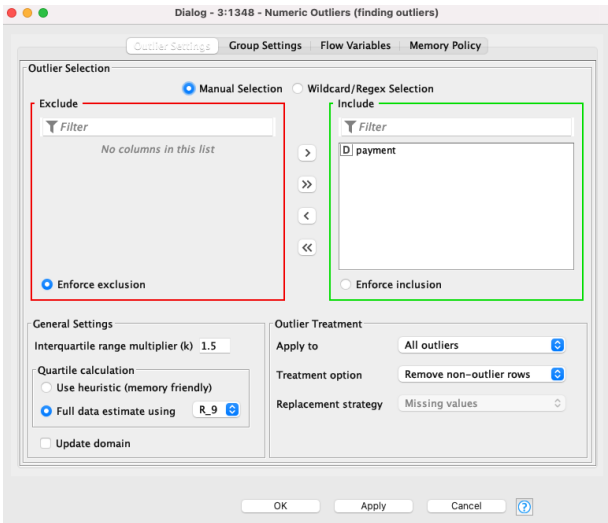
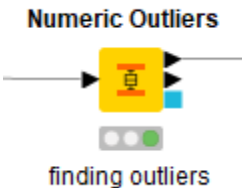
or

$$\text{Point} < Q1 - k * \text{IQR}$$

Finding Outliers: Statistics

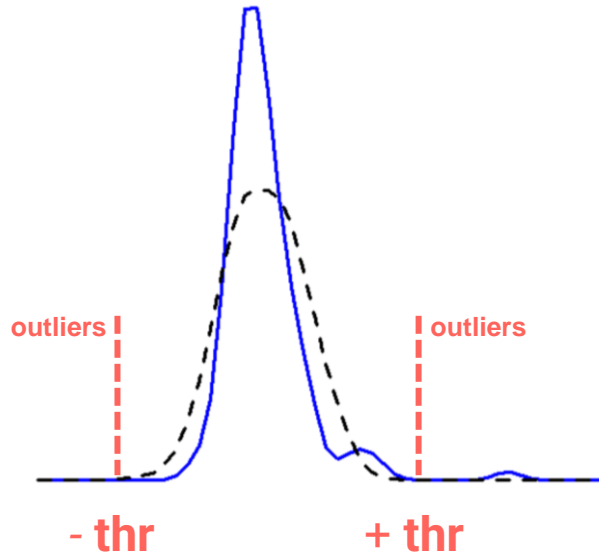
IQR Technique

The Numeric Outliers node implements (and the Box Plot node visualizes) the IQR technique.



Finding Outliers: Statistics

Z-score Technique



1. Convert points to their z-scores:

$$z = \frac{x - \mu}{\sigma}$$

2. Define a threshold (**thr**).
Often $thr = 2.5$ or greater

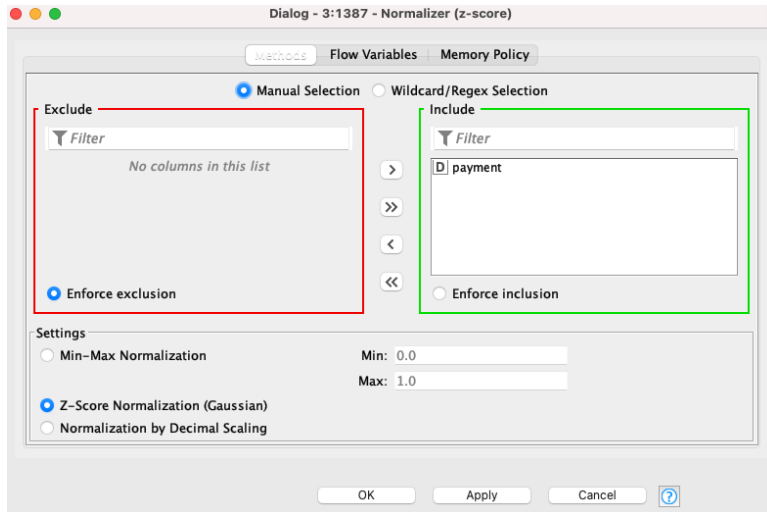
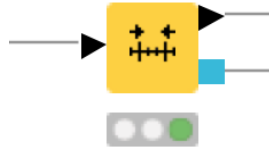
3. z is an outlier if:
 $|z| > \mathbf{thr}$

Finding Outliers: Statistics

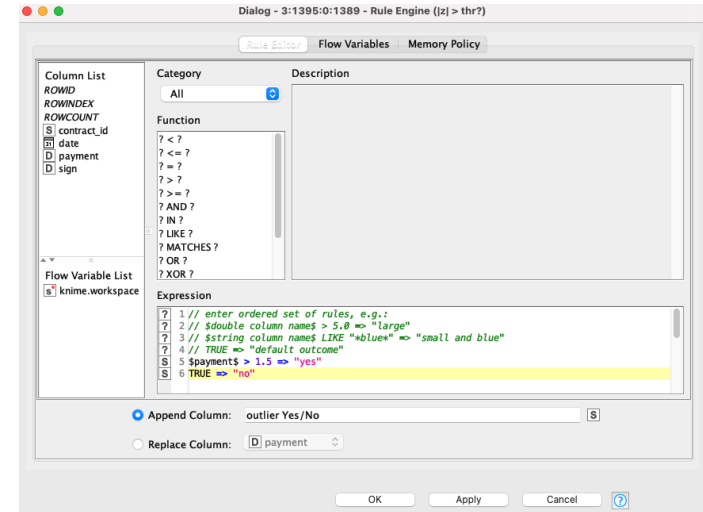
Z-score Technique

Z-score normalization and Rule Engine can implement the z-score technique

Normalizer



Rule Engine



KNIME Knowledge Check 01

- True or False.

For the statistics methods we showed, it is possible to find outliers across 2 dimensions.

For example, a data point that is an outlier because of a height and age combination.

Should you use statistical techniques?

Pros:

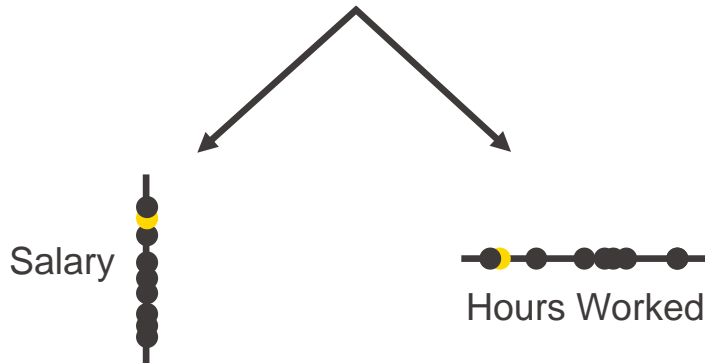
- Extraction automation
- Easy to understand
- Just 1 or 2 nodes

Cons:

- You might need to know some statistics
- Unidimensional analysis
- Z-score technique: the data is forced into a Gaussian distribution

What about multidimensional analysis?

One feature or many dimensions?



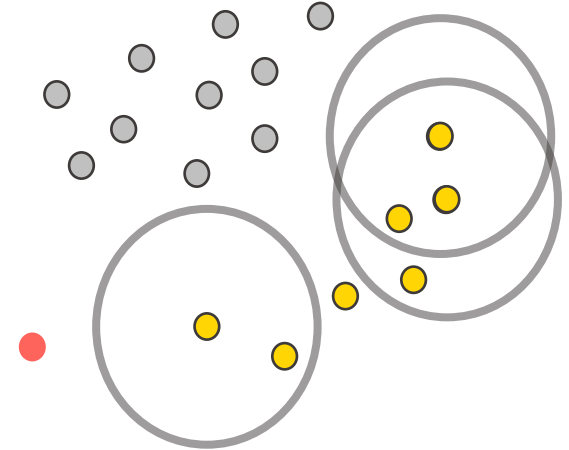
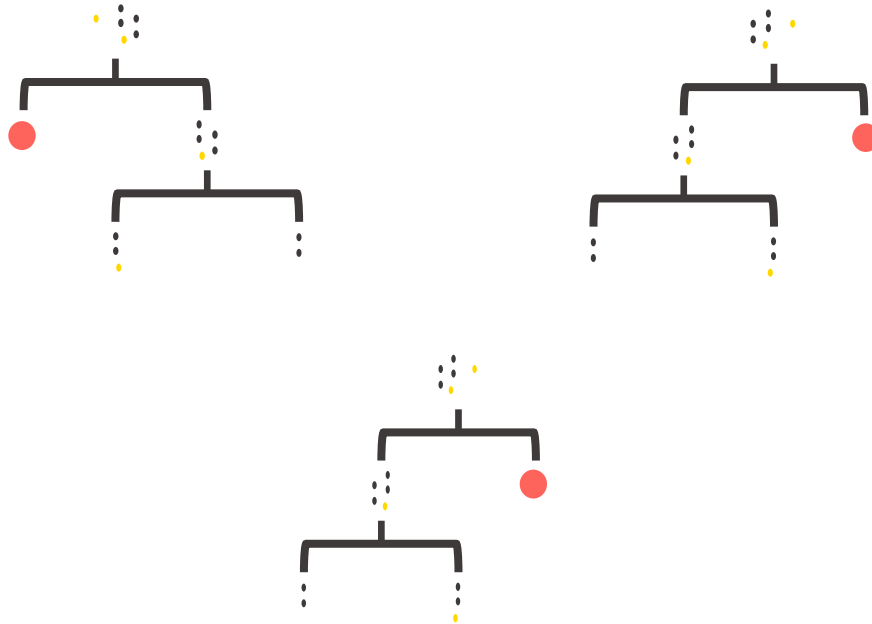
- In unidimensional space, spotting outliers may not be possible.
- We recommend beginning with multidimensional outlier detection first, and then run outlier detection procedures for each dimension.

3 Techniques for Anomaly Detection

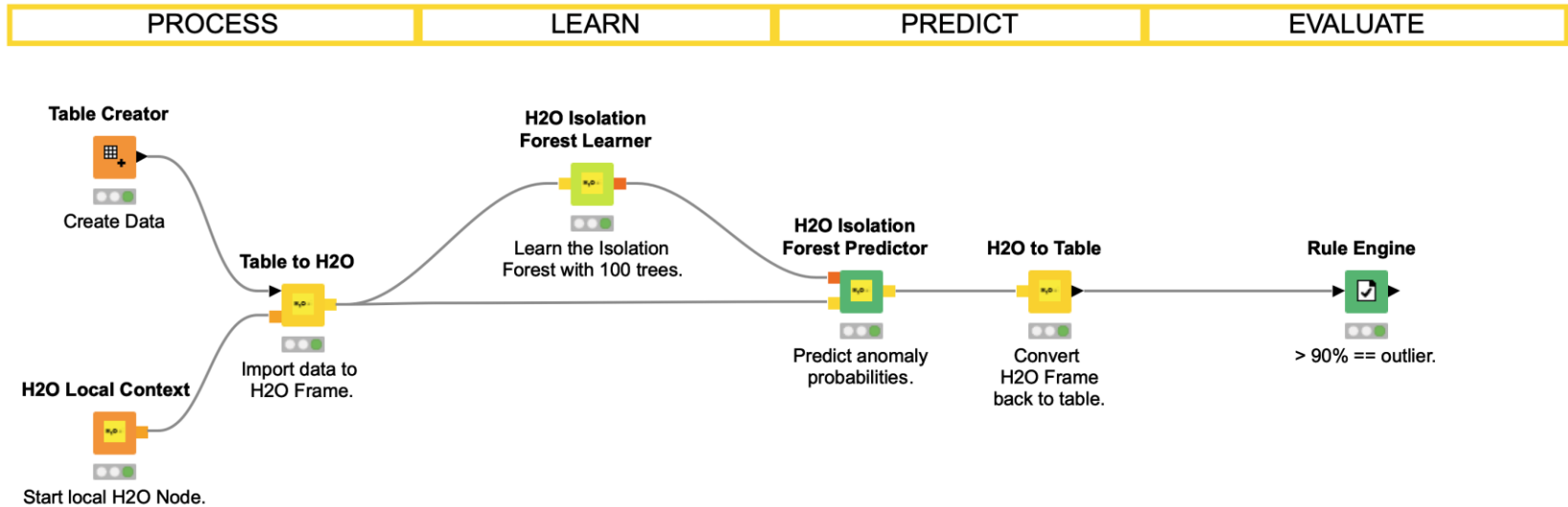
- Visualizations
- Statistics
- Machine Learning

Two Machine Learning Algorithms for Outlier Detection

- Isolation Forest – a tree-based algorithm
- DBSCAN – a clustering-based algorithm



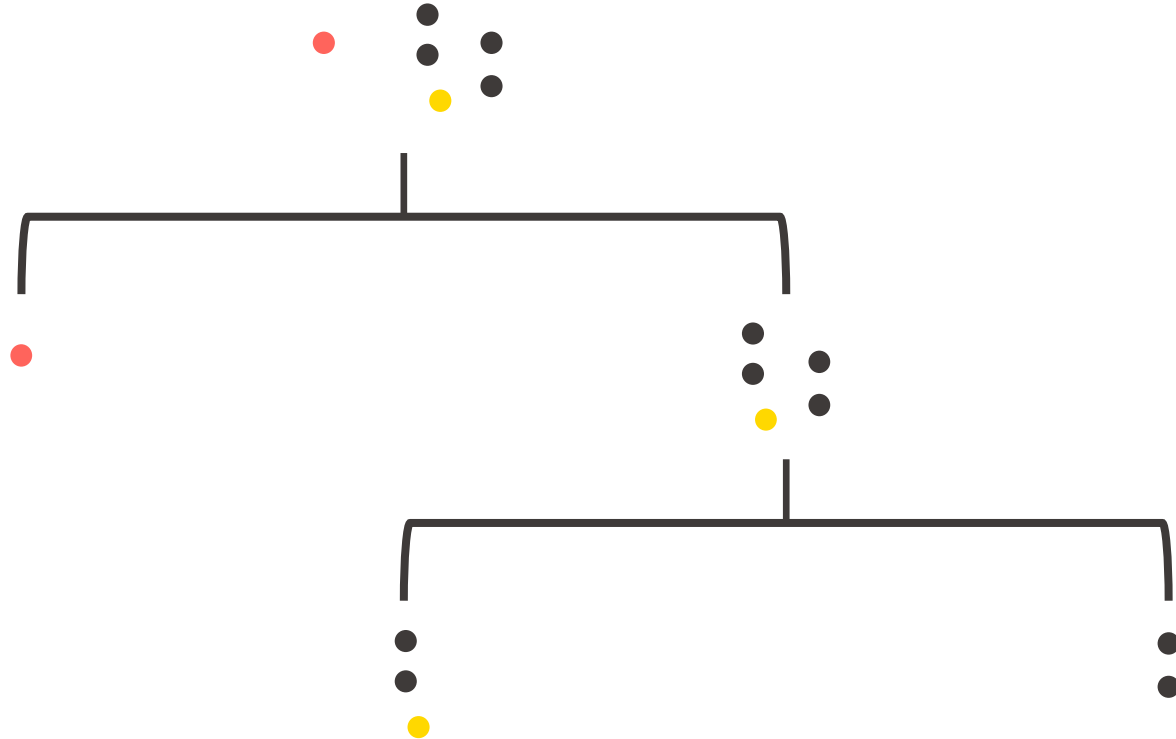
Isolation Forest in KNIME



Isolation Tree

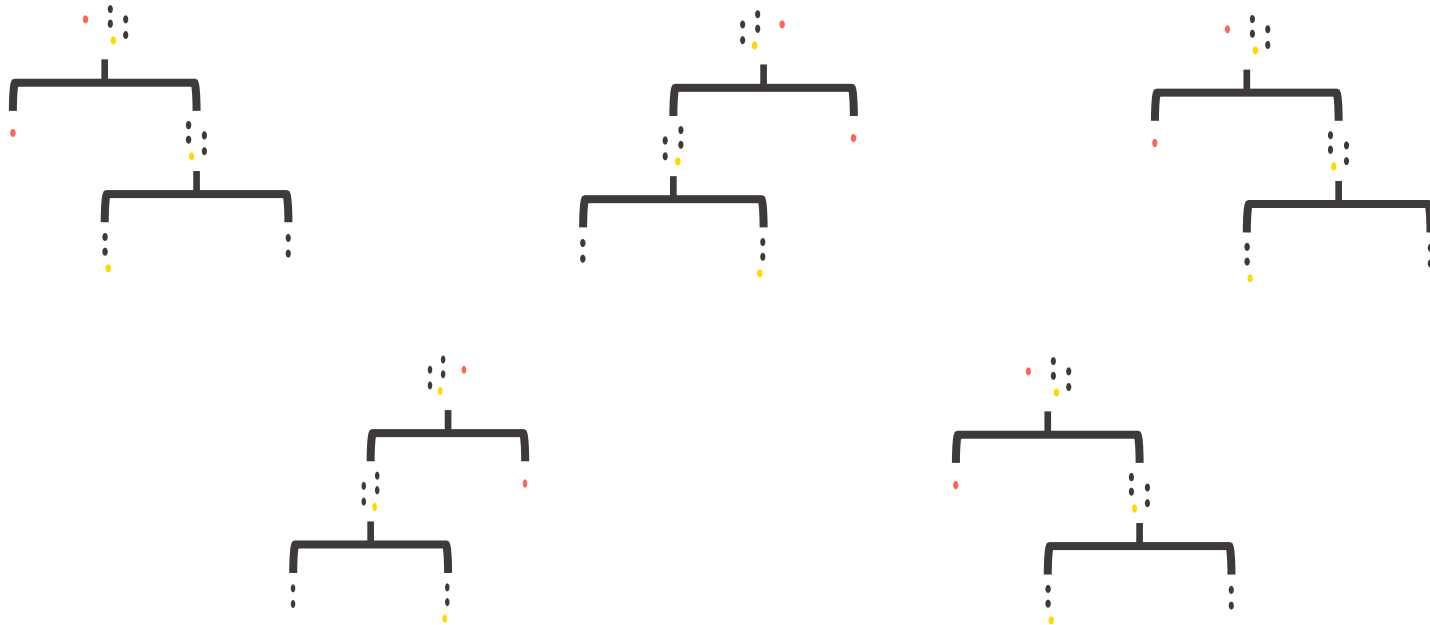
How many splits did it take for a point to be isolated?

Fewer implies outlier.



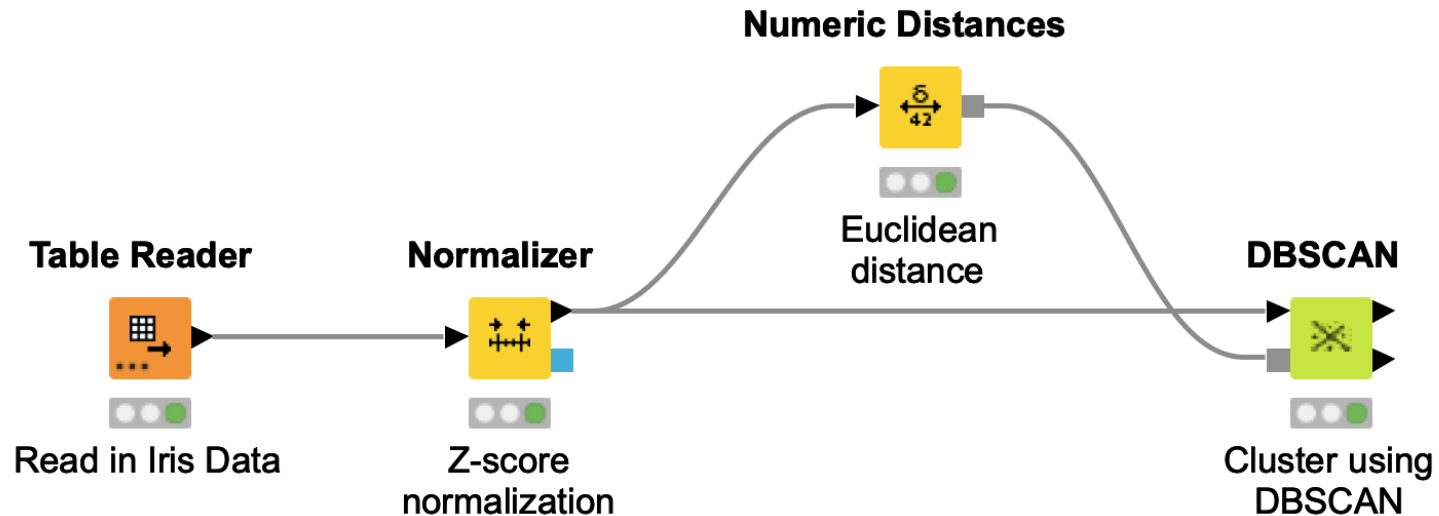
Isolation Forest

In a forest, use many trees and a voting mechanism to detect outliers.



DBSCAN in KNIME

- DBSCAN is a 4-node process in KNIME.

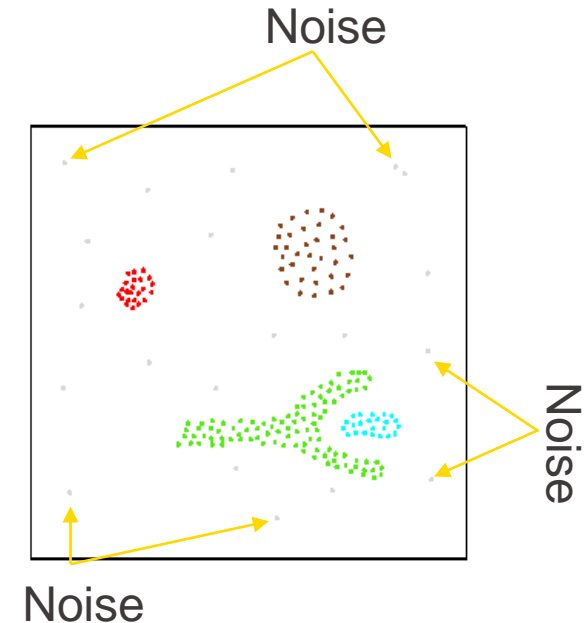


DBSCAN

DBSCAN - defines 5 types of points in a dataset but we'll talk about them as 2.

- **Cluster Points** within a specified distance (ϵ).
- **Noise Points** farther than ϵ from other points

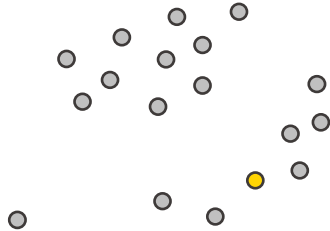
Clusters are built by joining points to one another.



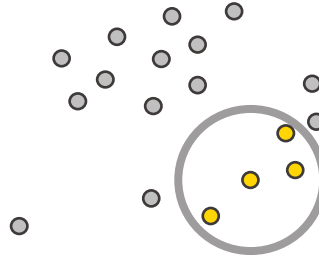
DBSCAN

- For each point is it in an ε -environment? If not, **then it's noise**.

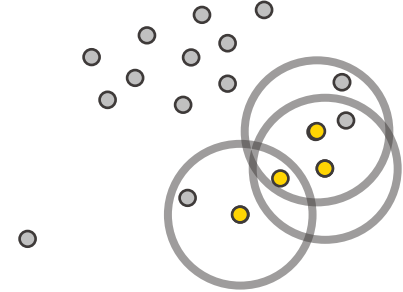
1 Pick random point.



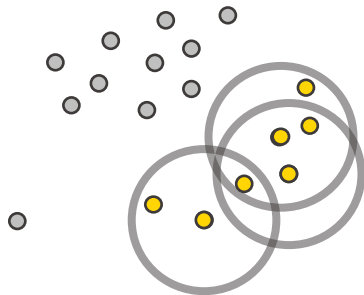
2 Emit ε -environment.



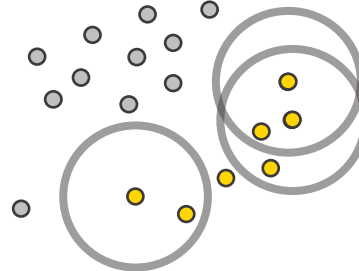
3 Emit ε -environment.



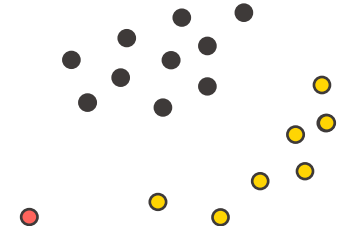
4 Continue...



5 ...until no neighbors.



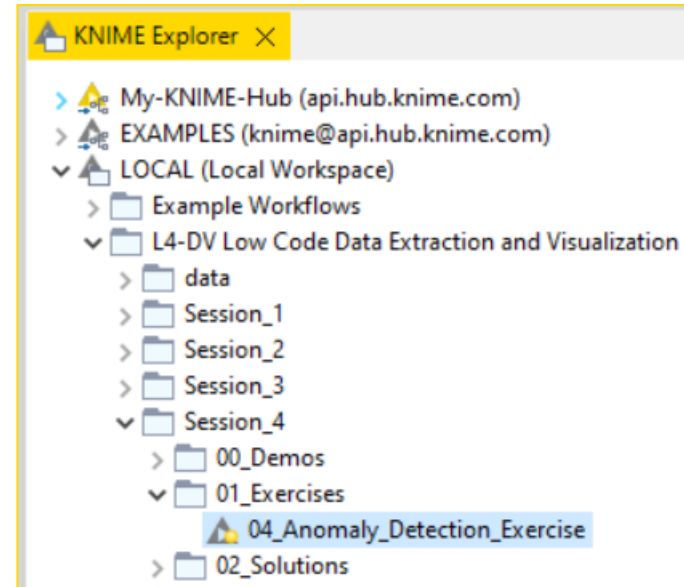
6 Rinse and repeat.



Exercises: Section 4

1. Anomaly Detection

Using the data provided, use various methods to detect outliers and compare the methods as well as output.



KNIME Knowledge Check 02

- True or False.

For the machine learning methods we showed, it is possible to find outliers across 2 or more dimensions.

For example, a data point that is an outlier because of a height and age and weight combination.

Should you use machine learning?

Pros:

- Extraction of outliers is automated
- Multidimensional analysis possible
- Isolation Forest: Non-parametric (no distribution required) & Fast
- DBSCAN: Easy to implement (4 nodes)

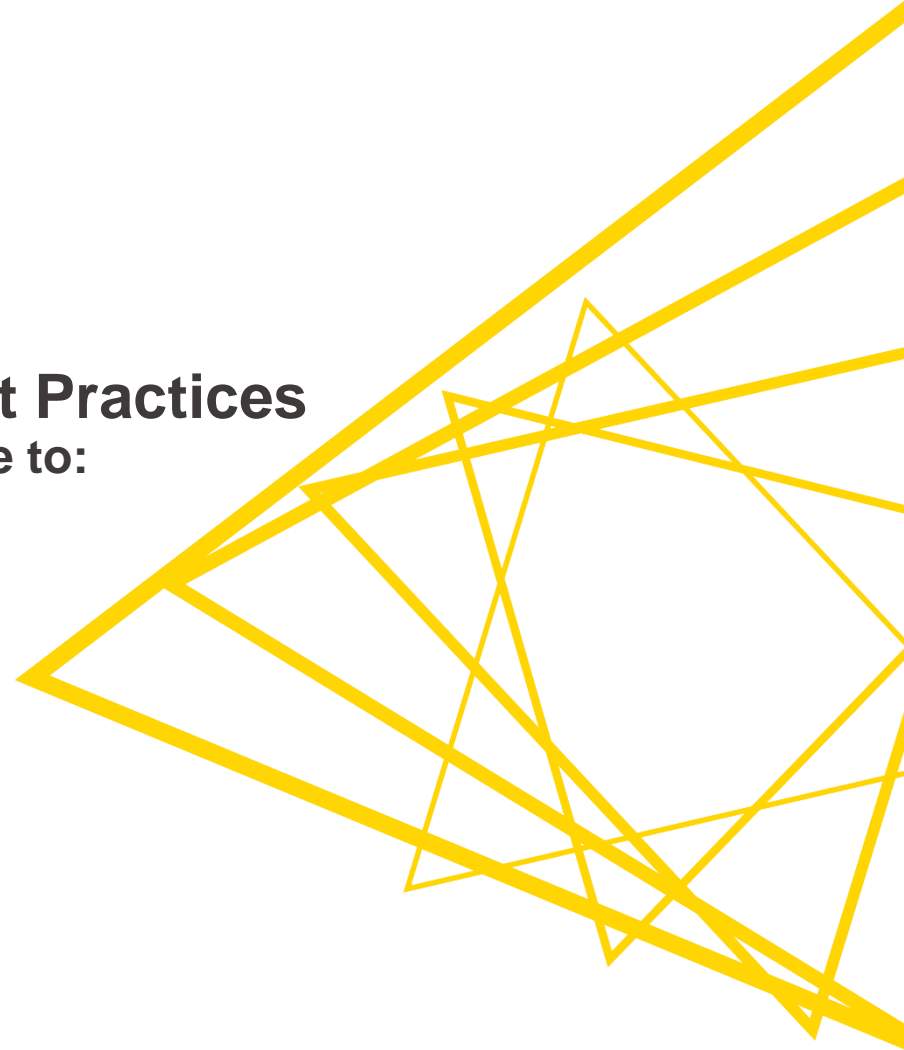
Cons:

- DBSCAN: Can be slow
- Hyperparameter setting required

Data Quality & Visualization Best Practices

At the end of this section, you will be able to:

1. Assess data quality via outlier detection
2. Apply best practices for data visualization



Visualization Principles

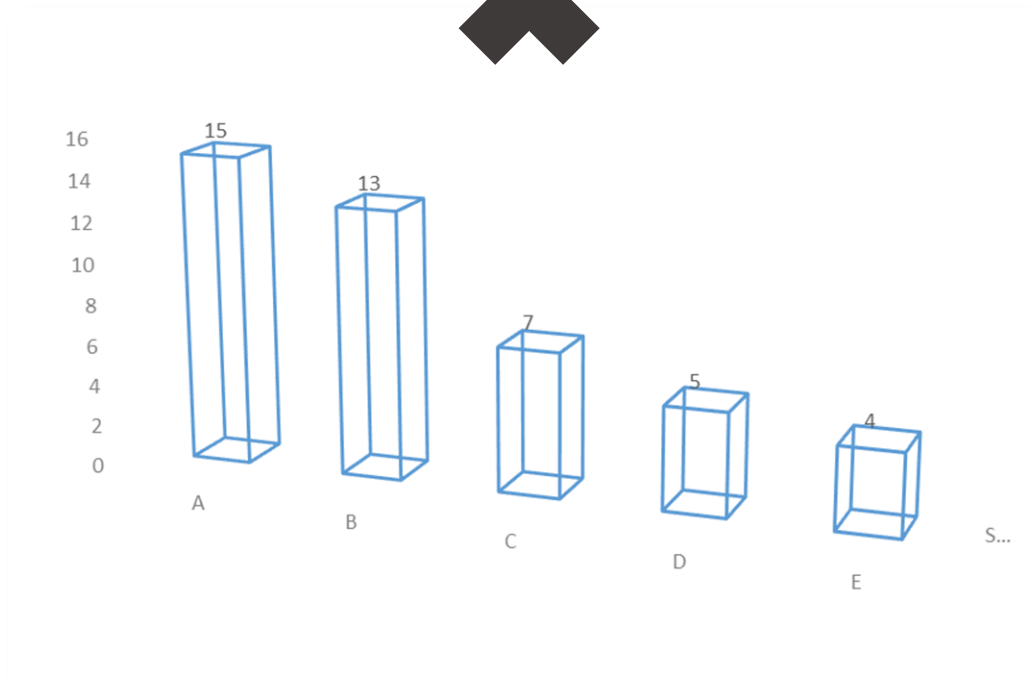
1. Simplification

2. Color

3. Purpose

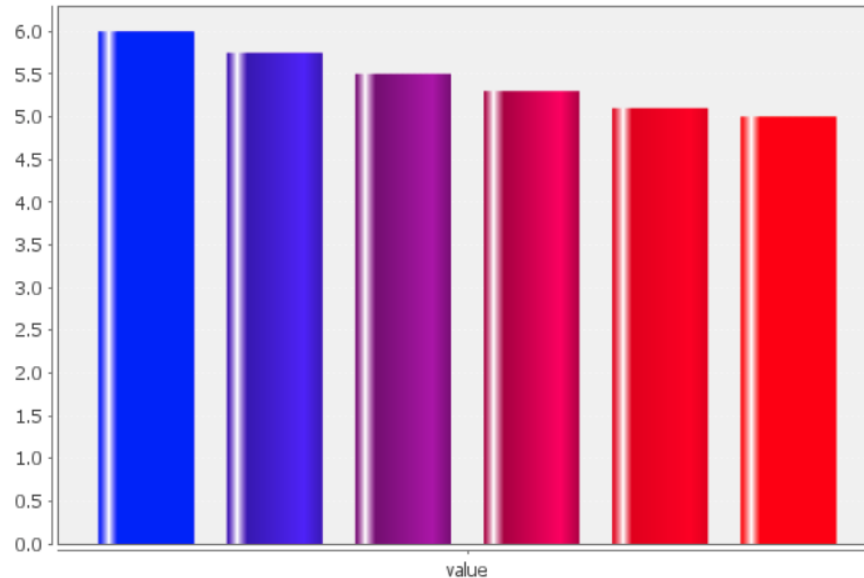
Simplification – be mindful of the cognitive load

Avoid 3D charts; Use 2D charts instead.



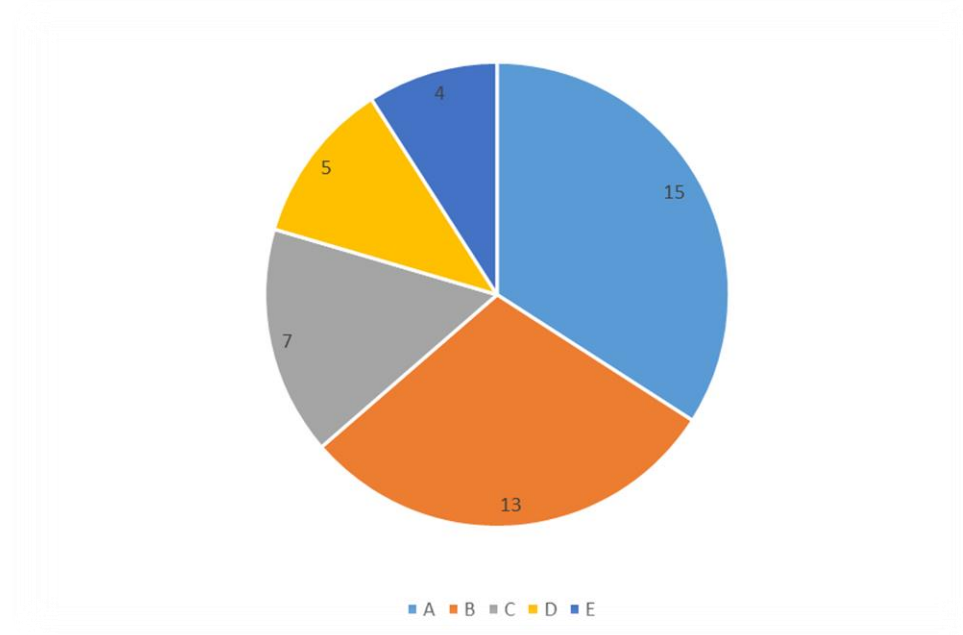
Simplification – remove extra effects

Remove extra effects like the shading on each of these bars.



Simplification – choose charts with minimal processing

A pie chart with more than 2 categories is usually harder to understand immediately than a bar chart.

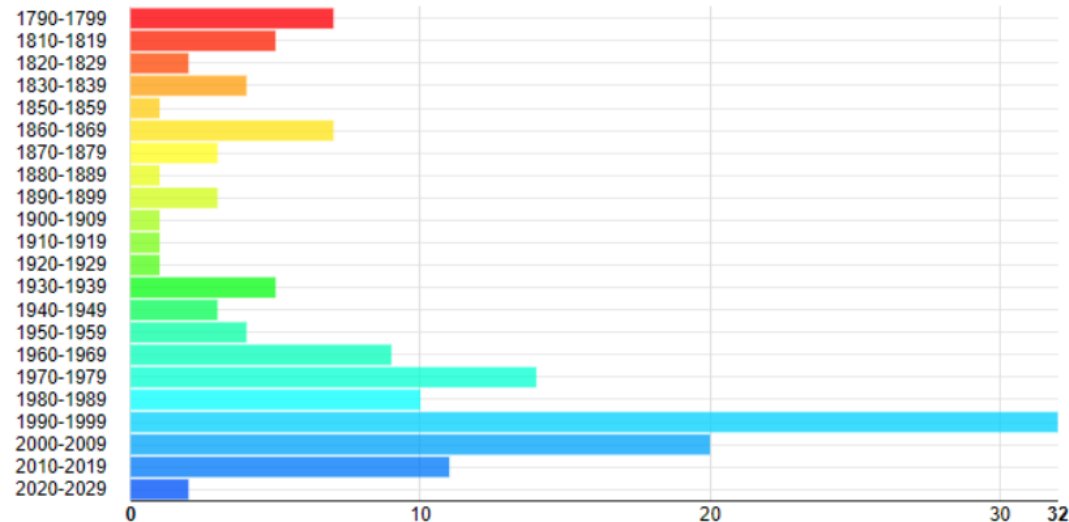


Color – make sure every color is needed

- Use color with intention; be able to explain each color choice or remove it

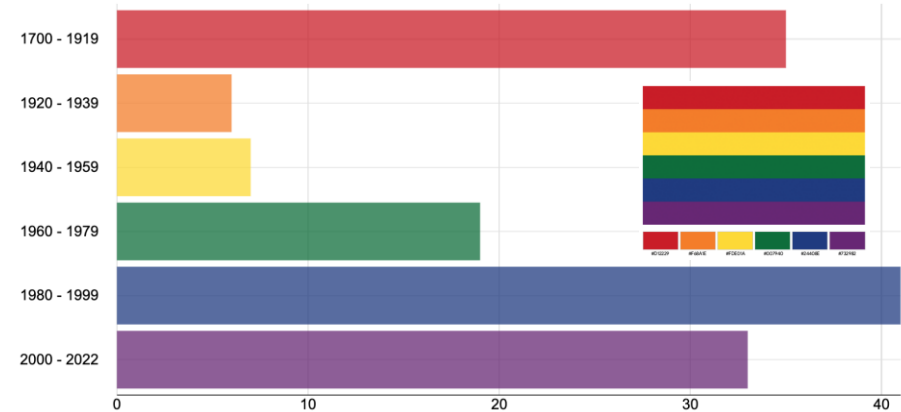
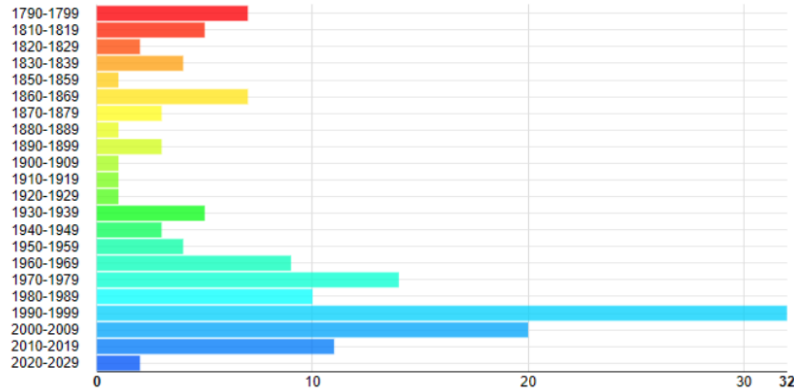


Bar Chart



Color – reduce total colors used

- Image on left has too many colors

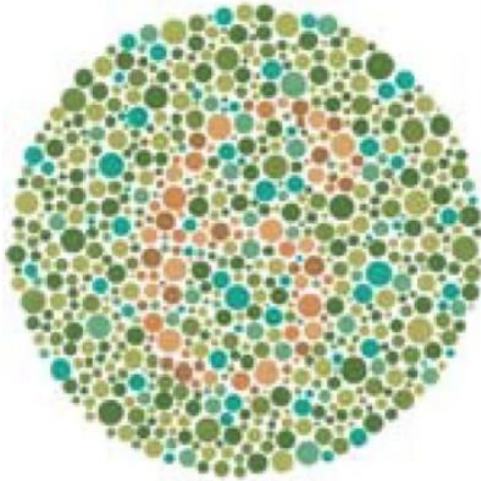


- Bin the years in a different way to show less but have more impact.

Color – be mindful of the colorblind

- Avoid Christmas colors (i.e., red and green) because the colorblind may not be able to distinguish them.

Typical Vision
(can see the red 6)



Colorblind Vision
(red and green are not separable)

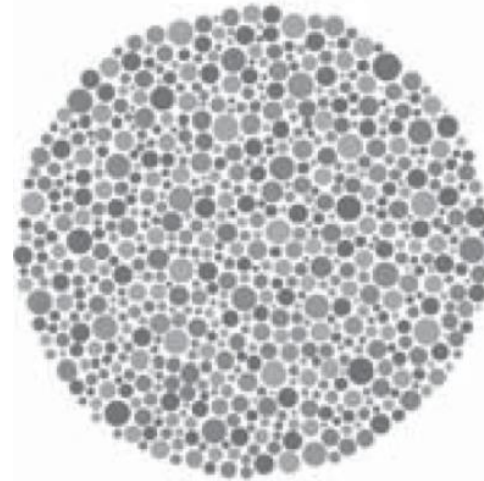
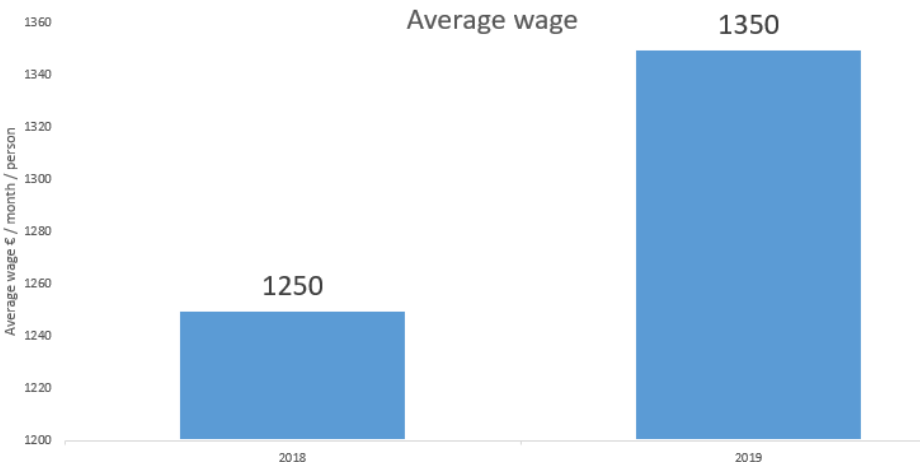


Image source: [article from Nature](#)

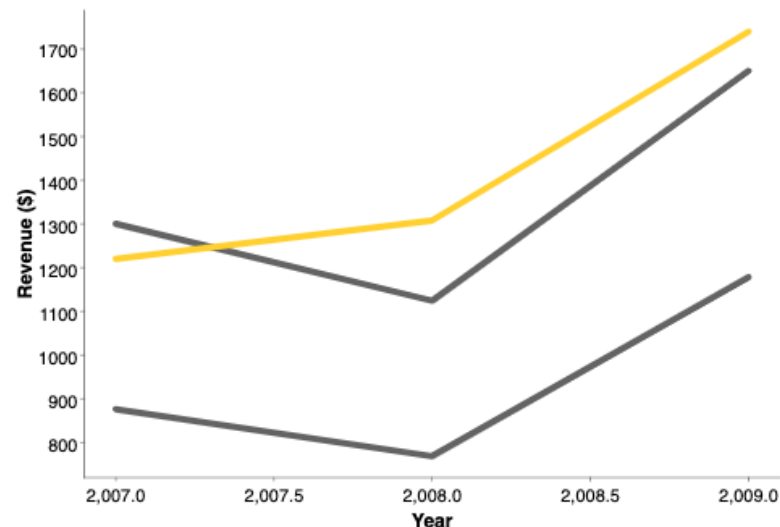
Purpose – ask for a call to action

- Good visualizations usually ask us to do something.



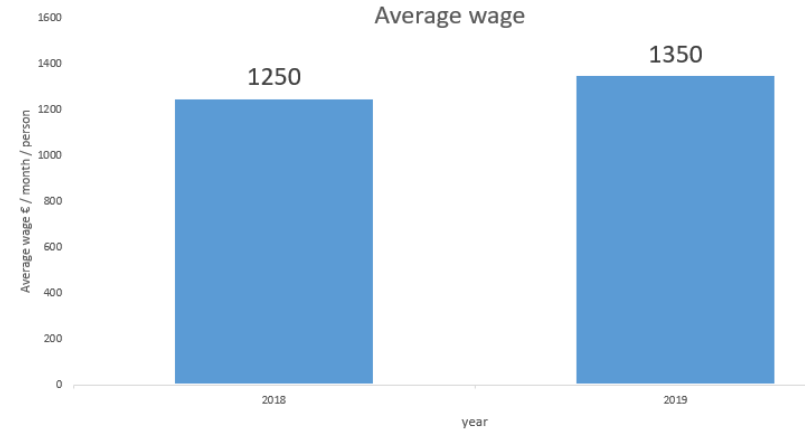
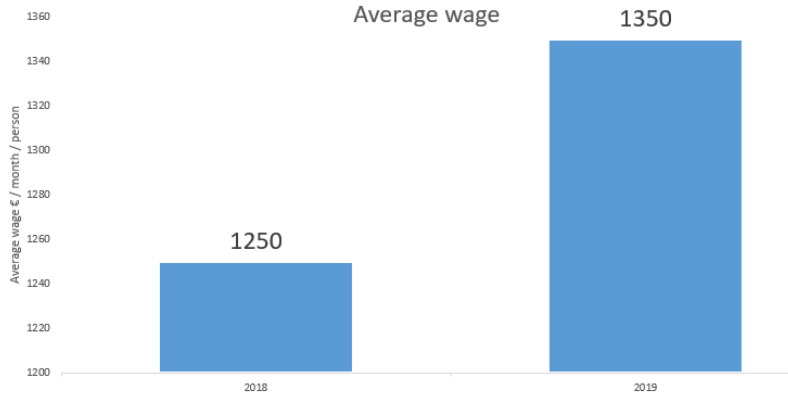
Top Earning Projects Past 3 Years

Give these teams a raise!



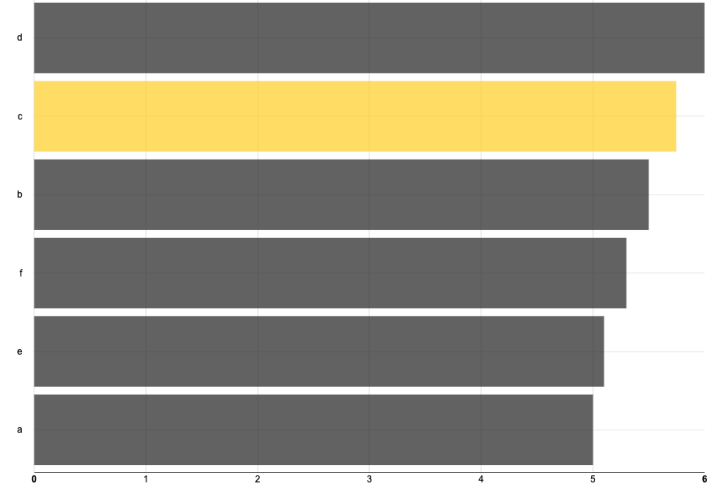
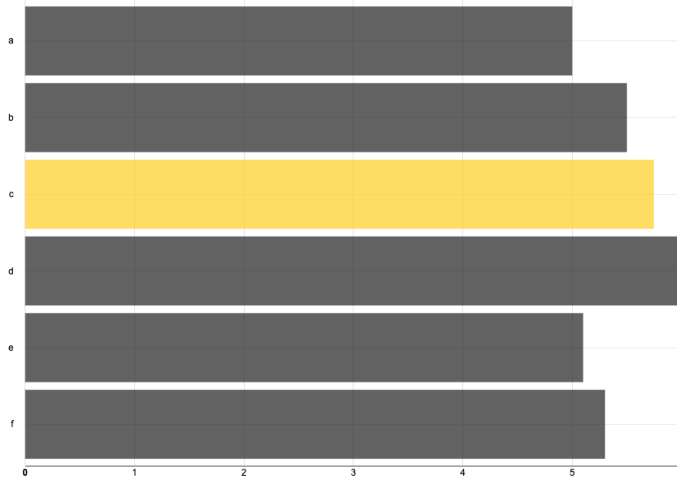
Purpose – do not purposefully mislead

- Non-zero beginning on axis is deceptive



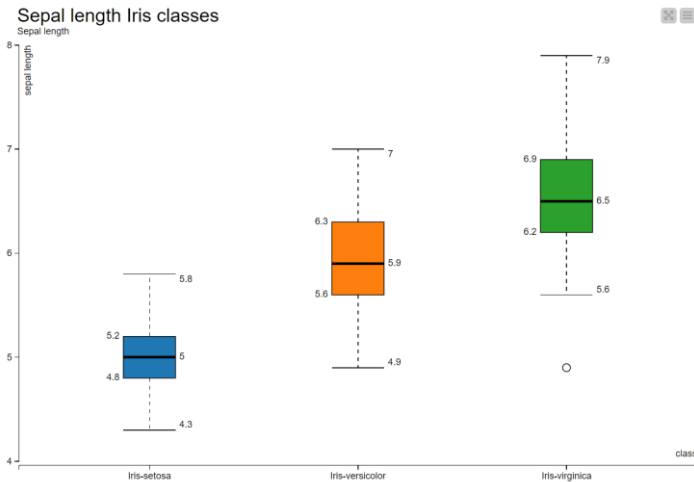
Purpose – true purpose is quick transfer of information

- Order your data to lower the cognitive load on the viewer



KNIME Knowledge Check 03

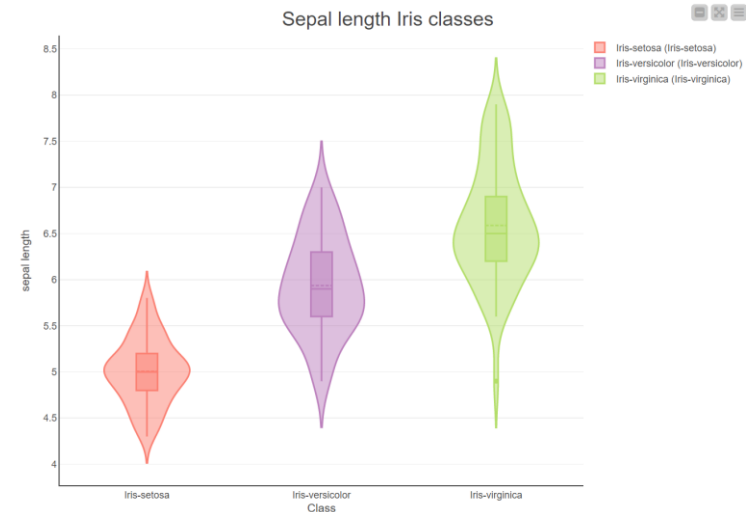
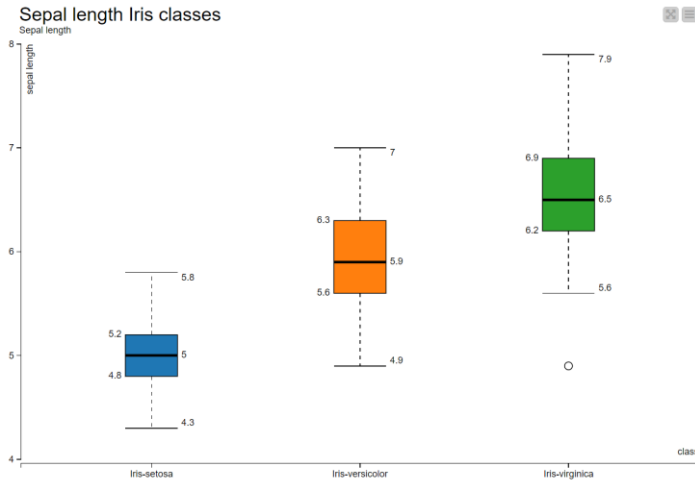
- From a visualization perspective, what advantage does the violin plot have over the box plot and vice-versa?



KNIME Knowledge Check 03: Solution

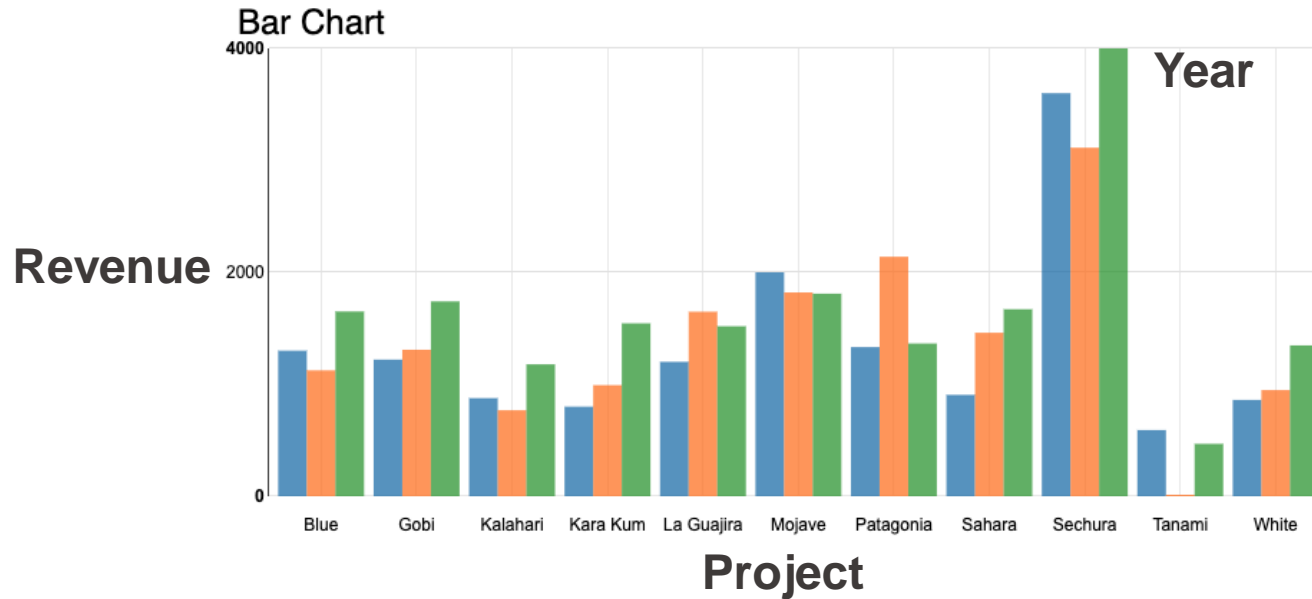
- Advantage of the Box Plot
 - Hides distribution (causing less cognitive overload)

- Advantages of Violin Plot
 - Displays box plots and
 - Displays data distribution



KNIME Knowledge Check 04

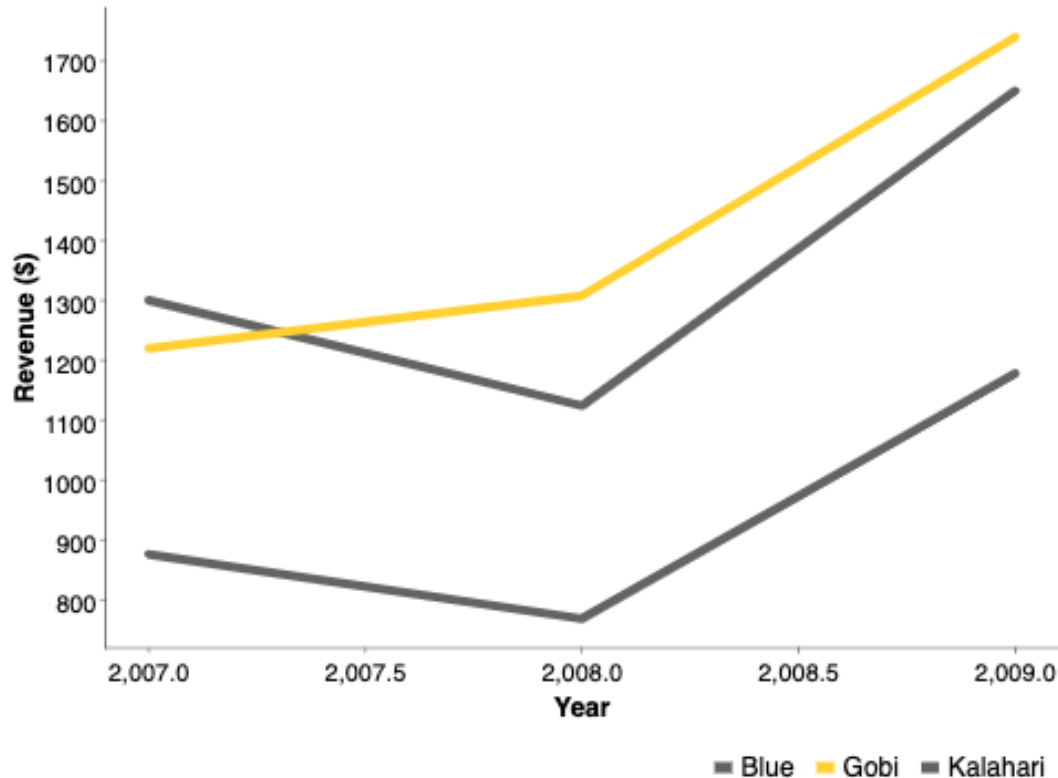
- How can this visualization be improved?



KNIME Knowledge Check 04: Solution

Top Earning Projects Past 3 Years

Give these teams a raise!



Why is this better?

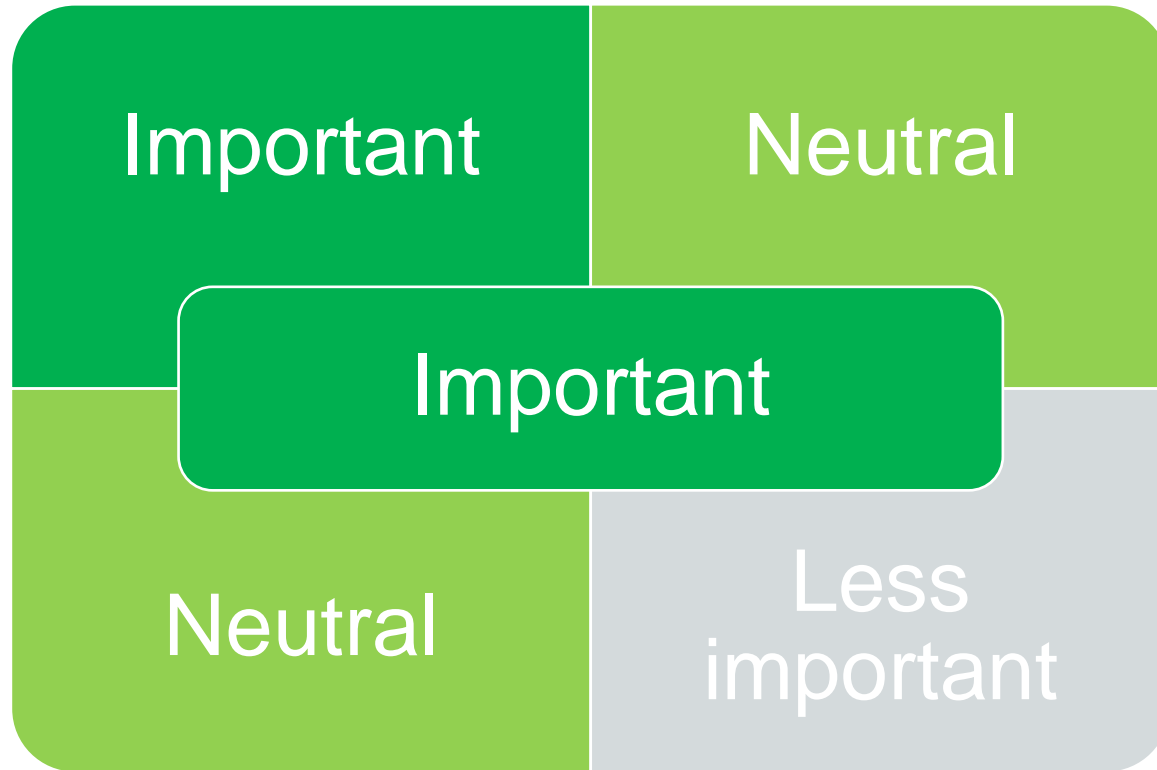
1. Simplified (less irrelevant data)
2. Color (highlighted one project for discussion)
3. Purpose (Advocating for raises)

Dashboard rules

1. Avoid excess detail
2. Make your dashboard fit on one page
3. Follow the standard dashboard distribution

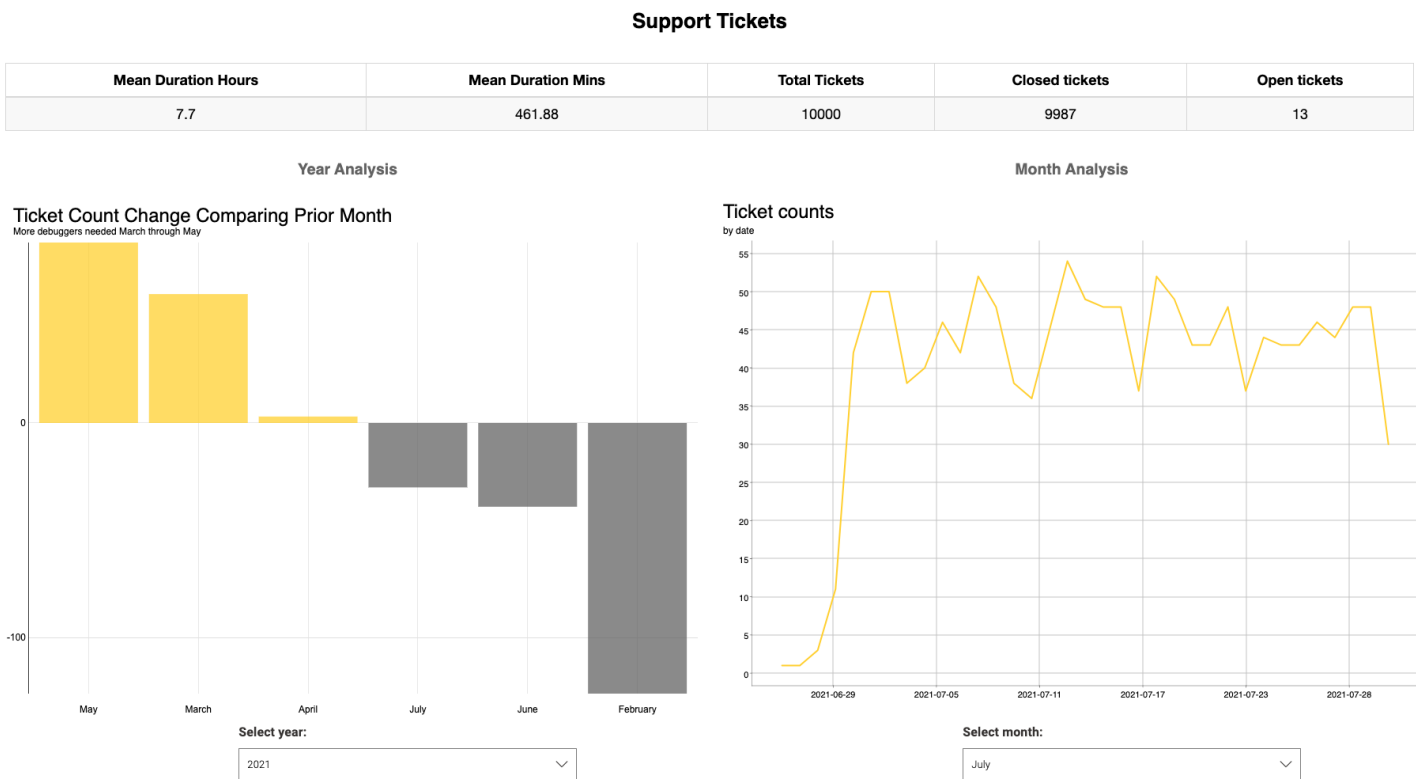
Dashboard distribution

Placement of items in a dashboard is critical for rapid, natural comprehension.
Notice: key points on left-top and center.



Example 1: Support Tickets Dashboard

Exemplar dashboard distribution: Statistics on top, key plot on left with purpose, details on right

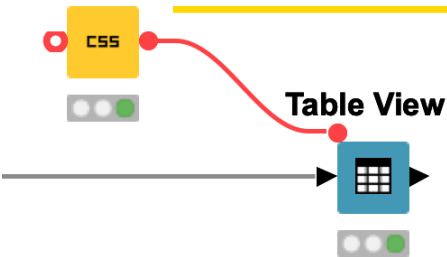


How was the topmost table made?

Support Tickets

Mean Duration Hours	Mean Duration Mins	Total Tickets	Closed tickets	Open tickets
7.7	461.88	10000	9987	13

CSS Editor



Dialog - 5:55:0:45 - CSS Editor

CSS View Flow Variables

☐ Prepend existing stylesheet:

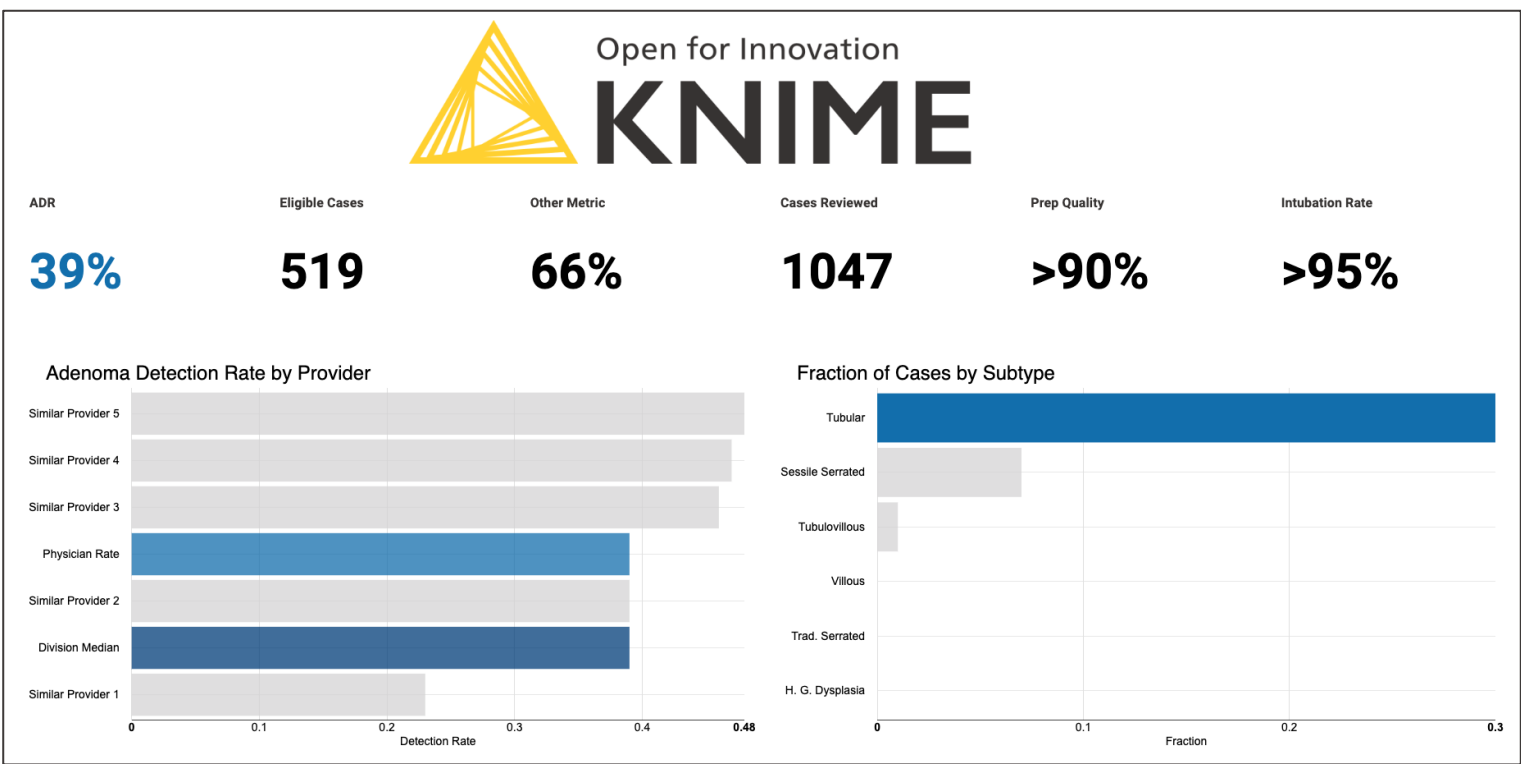
```
1 /* example style rule */
2 .knime-table-header{
3     text-align: center;
4 }
5
6 .knime-table-info{
7     display: none;
8 }
9
10 .knime-table-cell{
11     font-size: 18px;
12     text-align: center;
13 }
14
15
16 .knime-title {
17     font-size: 15px;
18     font-weight: bold;
19     color: #34495E;
20     fill: #34495E;
21 }
```


Additional dashboard customization

1. Add images
 - For instance, integrating a company logo
2. Modify HTML
 - To add more certain text bigger or add color
3. Change CSS
 - To change the size of font for certain views

Example 2: Medical Dashboard

Additional customization: Adding images, modifying HTML, changing CSS

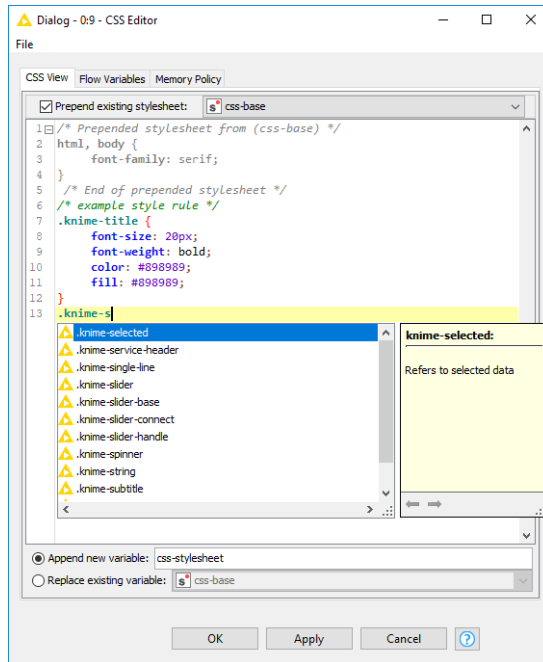
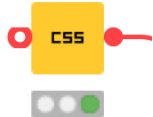


What is CSS?

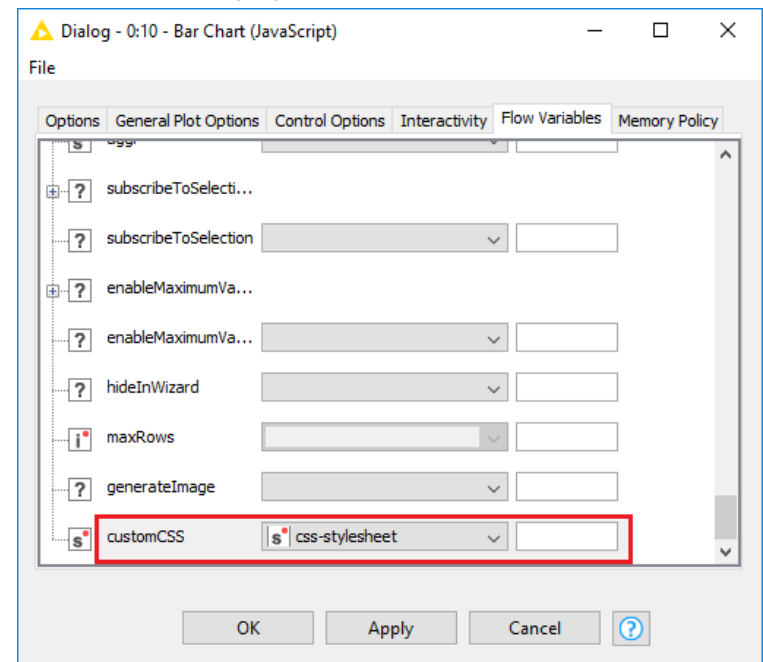
- CSS, Cascading Style Sheets, is a language for presentation in HTML or XML.

Define font size, etc.

CSS Editor



Apply your definitions



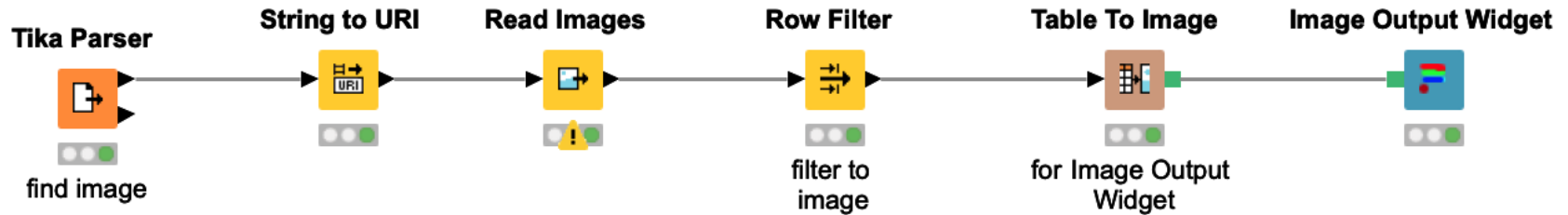
[See the CSS guide for more tips](#)

Additional customization – Adding an image

To show a logo like:



We can use this workflow (found in the demos folder):

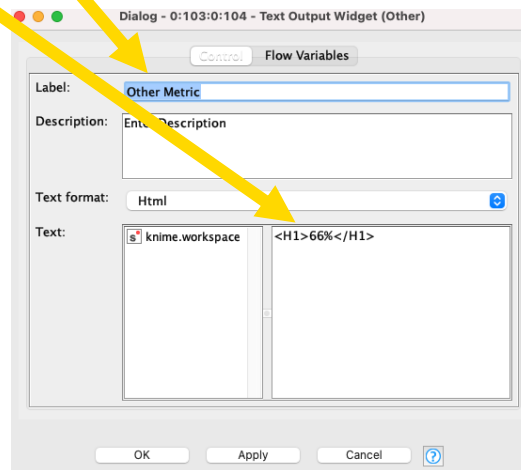
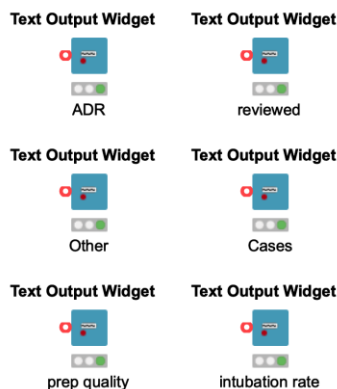


Additional customization – Modifying the HTML

Change the HTML to affect size and color of text in Text Output Widget

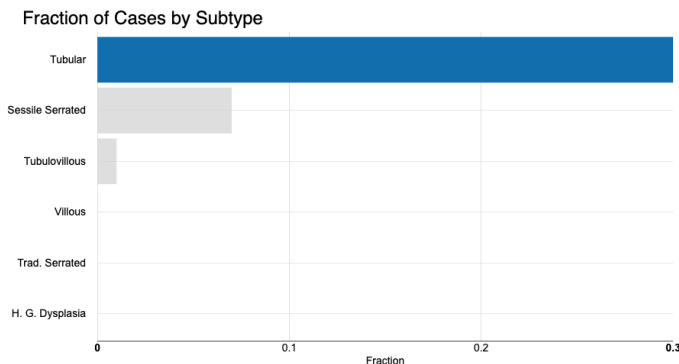
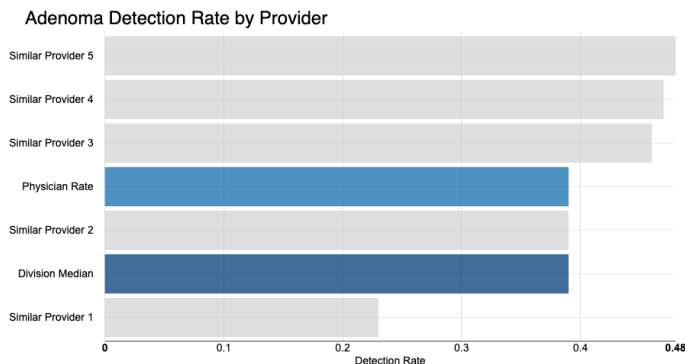
ADR	Eligible Cases	Other Metric	Cases Reviewed	Prep Quality	Intubation Rate
39%	519	66%	1047	>90%	>95%

How to build

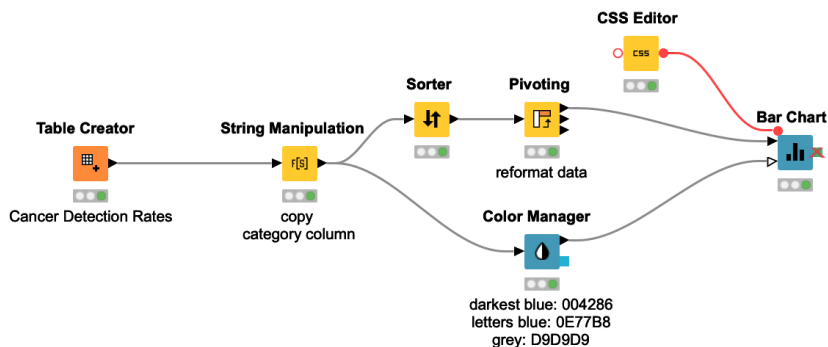


Additional customization – Changing the CSS

Change the CSS of the bar chart to have larger font



How to build



Data Visualization Summary

- Simplify
 - Show only relevant information
 - Avoid a high cognitive load
- Emphasize
 - Significant elements only
 - Use color with intent
- Select the right visualization type
 - Based on the type (category, number, etc.)
 - And on the number of variables (1, 2, or 3)

Summary of Section 4

Now you should be able to:

1. Assess data quality via outlier detection
2. Implement best practices for data visualization

Course objectives

With the completion of this course, you should now be able to:

- Collect data via **REST APIs**, **web text scraping**, and an **interactive** data collection **tool**
- Explore and visualize data
- Extract data and images from **PDF** documents
- Write regular expressions (**regex**)
- Identify and correct errors in data via **outlier detection**
- Build effective and **beautiful visuals**

Confirmation of Attendance and Survey

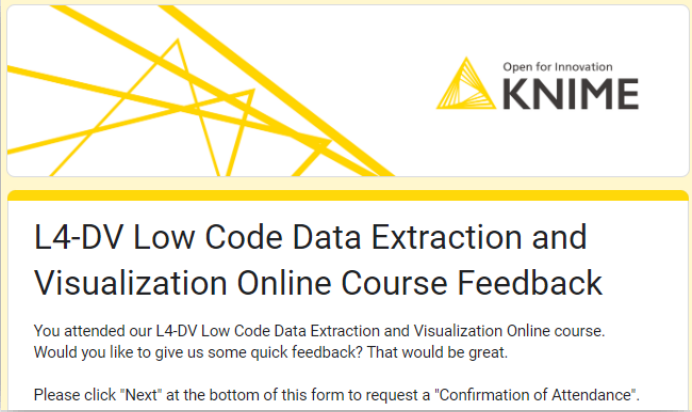
- If you would like to get a “Confirmation of Attendance” please click on the link below*

[Confirmation of Attendance and Survey](#)

- The link also takes you to our course feedback survey. Filling it in is optional but highly appreciated!

Thank you!

*Please send your request within the next 3 days



The image shows a screenshot of a survey form. At the top left, there is a yellow geometric logo consisting of several overlapping triangles. At the top right, the text 'Open for Innovation' is above the 'KNIME' logo. The main title of the form is 'L4-DV Low Code Data Extraction and Visualization Online Course Feedback'. Below the title, the text reads: 'You attended our L4-DV Low Code Data Extraction and Visualization Online course. Would you like to give us some quick feedback? That would be great.' At the bottom, it says: 'Please click "Next" at the bottom of this form to request a "Confirmation of Attendance".'

Thank you!

