

Assignment 2

- 1) Read the adult.csv file available in the **data** folder on the KNIME Hub. The data are provided by the **UCI Machine Learning Repository**.
- 2) Calculate the average age and count for each one of the 4 groups defined by sex and income values
- 3) Join the two aggregated values to the original table

Step 1: Read the adult.csv file

The screenshot displays the KNIME workspace with a 'CSV Reader' node connected to a 'Joiner' node. The 'CSV Reader' node is configured with 'Add comment' and 'GroupBy' options. The 'Joiner' node is connected to the 'CSV Reader' node. The 'CSV Reader' node's dialog box is open, showing the file path and the 'GroupBy' options. The 'CSV Reader' node's dialog box is open, showing the file path and the 'GroupBy' options.

CSV Reader

Reads CSV files. To auto-guess the structure of the file click the Autodetect format button. If you encounter problems with incorrect guessed data types disable the Limit data rows scanned option in the Advanced Settings tab. If the input file structure changes between different invocations, enable the Support changing file schemas option in the Advanced Settings tab. For further details see the KNIME File Handling Guide [File Handling Guide](#).

Note: If you find that this node can't read your file, try the **File Reader** node. It offers more options for reading complex files.

This node can access a variety of different [file systems](#). More information about file handling in KNIME can be found in the official [File Handling Guide](#).

Parallel reading: Individual files can be read in parallel if

- They are located on the machine that is running this node.
- They don't contain any quotes that contain row delimiters.
- They are not gzip compressed.
- No lines or rows are limited or skipped.
- The file index is not prepended to the RowID.
- They are not encoded with UTF-16 (UTF-16LE and UTF-16BE are fine).

Ports Options Views

Output ports

Joiner

This node dialog is not supported here.

Open dialog

1: File Table

Rows: 32561 | Columns: 15

#	RowID	age	workclass	fnlwgt	education	education--	marital-st--	occupation	relations--	race	st
1	Row0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family	White	Male
2	Row1	50	Self-emp-not-in	83311	Bachelors	13	Married-civ-spo	Exec-manageriz	Husband	White	Male
3	Row2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleani	Not-in-family	White	Male
4	Row3	53	Private	234721	11th	7	Married-civ-spo	Handlers-cleani	Husband	Black	Male
5	Row4	28	Private	338409	Bachelors	13	Married-civ-spo	Prof-specialty	Wife	Black	Female
6	Row5	37	Private	284582	Masters	14	Married-civ-spo	Exec-manageriz	Wife	White	Female
7	Row6	49	Private	160187	9th	5	Married-spouse	Other-service	Not-in-family	Black	Female
8	Row7	52	Self-emp-not-in	209642	HS-grad	9	Married-civ-spo	Exec-manageriz	Husband	White	Male
9	Row8	31	Private	45781	Masters	14	Never-married	Prof-specialty	Not-in-family	White	Female
10	Row9	42	Private	159449	Bachelors	13	Married-civ-spo	Exec-manageriz	Husband	White	Male

Koushal Thapliyal
2501940077
MCA (AI & ML)

Step 2: Calculate the average age and count for each one of the 4 groups defined by sex and income values

The screenshot shows the KNIME interface with the **GroupBy** node selected. The left sidebar contains the **Info** panel for the **GroupBy** node, which explains its function: grouping rows by unique values in selected columns and aggregating the remaining columns. The main workspace shows a workflow starting with a **CSV Reader** node, followed by a **GroupBy** node, and then a **Joiner** node. The **GroupBy** node's configuration dialog is open, showing the **Manual Aggregation** tab. The output table, titled "1: Group table", displays the results of the aggregation:

#	RowID	sex	income	Mean(age)	Count(age)
1	Row0	Female	<=50K	36.211	9592
2	Row1	Female	>50K	42.126	1179
3	Row2	Male	<=50K	37.147	15128
4	Row3	Male	>50K	44.626	6662

Step 3: Join the two aggregated values to the original value

The screenshot shows the KNIME interface with the **Joiner** node selected. The left sidebar contains the **Info** panel for the **Joiner** node, which explains its function: combining two tables similar to a join in a database. The main workspace shows a workflow starting with a **CSV Reader** node, followed by a **GroupBy** node, and then a **Joiner** node. The **Joiner** node's configuration dialog is open, showing the **Matching Criteria** tab. The output table, titled "1: Join result", displays the results of the join operation:

sex	capital-g...	capital-lo...	hours-per...	native-co...	income	sex (Right)	income (...)	Mean(age)	Count(a...
Male	2174	0	40	United-States	<=50K	Female	<=50K	36.211	9592
Male	0	0	13	United-States	<=50K	Female	>50K	42.126	1179
Male	0	0	40	United-States	<=50K	Male	<=50K	37.147	15128
Male	0	0	40	United-States	<=50K	Male	>50K	44.626	6662