

Watson Studio SPSS Modeler Overview

Overview

In this lab you will learn how to implement analytics in **SPSS Modeler**, a well-known visual data mining workbench which is part of **Watson Studio**. The lab will introduce the SPSS Modeler capability using the Titanic dataset. The lab will guide the development of an SPSS Modeler stream that will prepare the input data to train and evaluate a machine learning model for predicting survivability of a passenger on the Titanic.







Introduction

SPSS Modeler is a visual data mining workbench. Modeler can be used to complete all tasks in analytic application development

- Data understanding
- Data preparation
- Model building
- Model evaluation

Assets developed in Modeler are called “flows”. Another frequently used term in Modeler documentation is “streams” (used in Modeler desktop documentation). A flow starts with one or several data sources. Using visual nodes, a user can apply different operations to data. Data “flows” from one node to another in the direction of the arrows.

Visual nodes in modeler are color-coded and organized by type of operation: **Record Operations**, **Field Operations**, **Graphs**, **Modeling**, **Output**, and **Export** (data sources). Most operations are well-known functions in data preparation and analytics, such as sampling, filtering, binning, etc.

The data sources are purple	
Data preparation operations are blue	
Algorithms are green	
The models that are created based on algorithms are orange	
Different types of output (graphs, tables, external files) are black	
The nodes with a star icon are called “supernodes” because they contain several	

nodes. Supernodes are used for visual organization of the flow.

If a user needs more information about a particular node, it can be looked up in Modeler documentation. SPSS also publishes the **Algorithms Guide** that explains how machine learning algorithms are implemented in Modeler.

Lab Steps

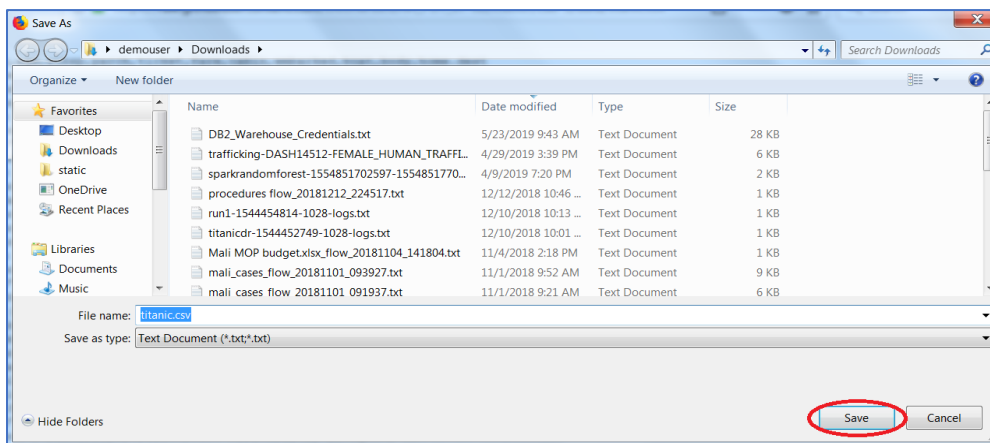
Step 1: Adding a Data Asset to the Watson Studio Labs project

This step can be skipped if the titanic.csv file was already downloaded in a previous lab.

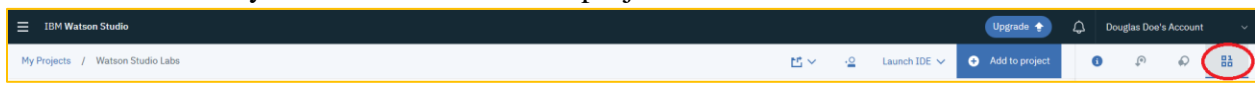
1. Download the Titanic data file from the following location by clicking [here](#).
2. Right-click on the screen and click on Save Page As ...



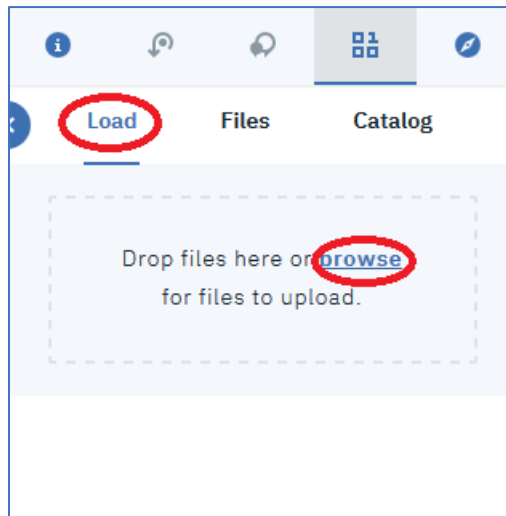
3. Click on **Save** to save the titanic.csv file.



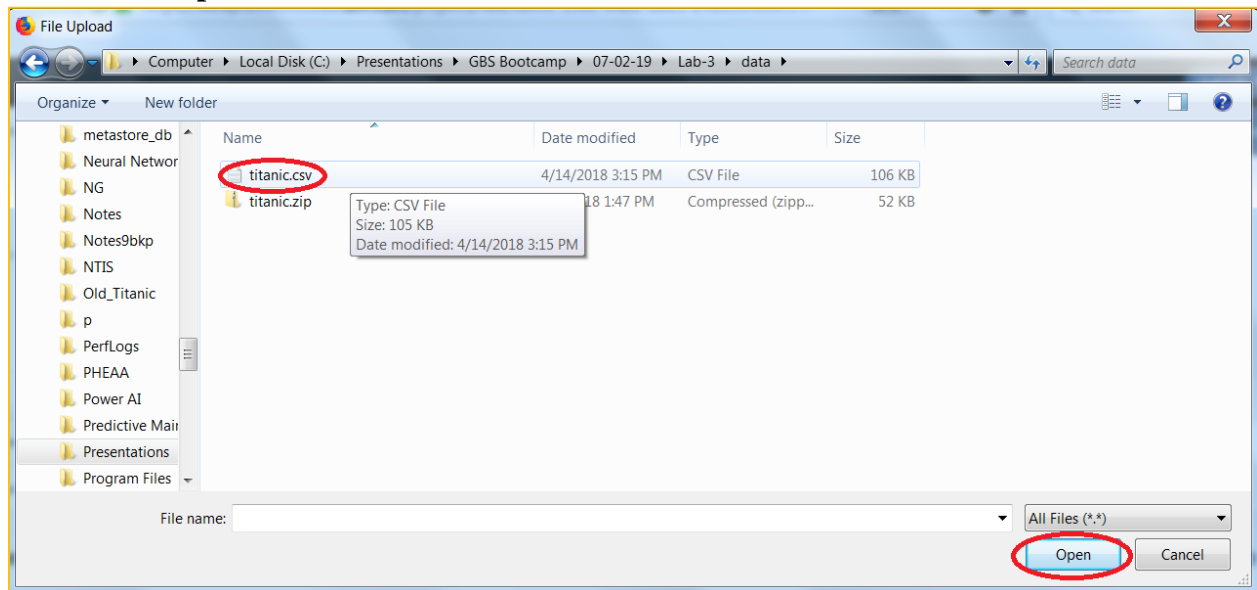
4. Go back to your Watson Studio Labs project. Click on the  icon.



5. Click on the **Load** tab and then click on **browse**. If you don't see the **Load** tab, click on the  icon again.



6. Go to the folder where the titanic_csv file is stored. Select the titanic.csv file and then click **Open**.



7. The file is now added as a Data Asset.

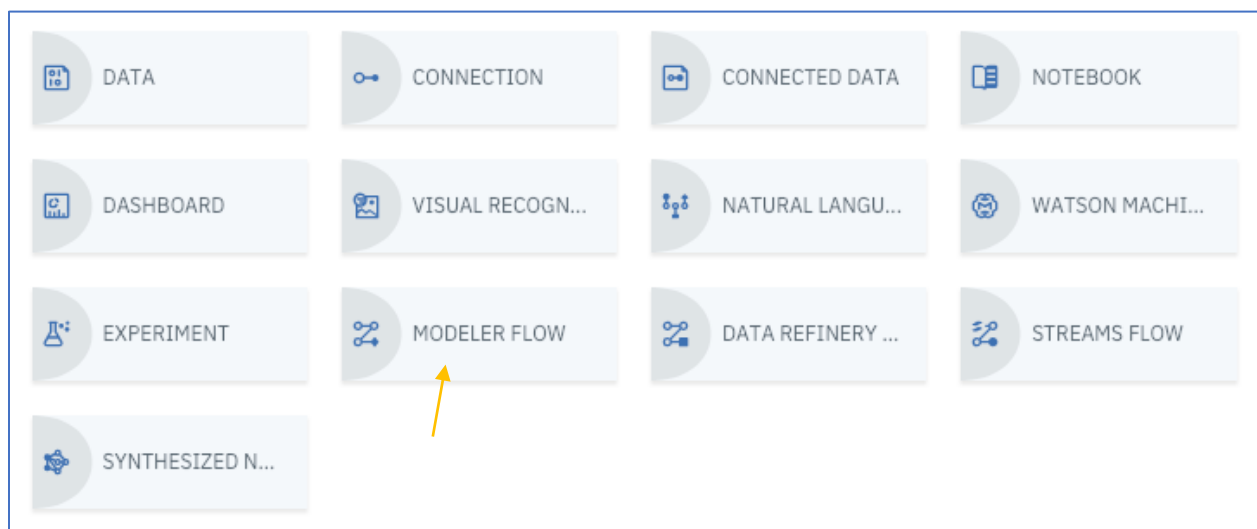
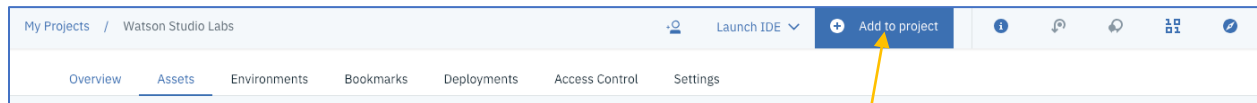
Data assets New data asset					
0 asset selected.					
<input type="checkbox"/>	NAME	TYPE	CREATED BY	LAST MODIFIED	ACTIONS
<input checked="" type="checkbox"/>	 titanic.csv	Data Asset	John Doe	4 Nov 2018, 2:45:59 pm	

Step 2: Create a Model to predict survival

In this section, we will create a Machine Learning flow using SPSS nodes.

Step 2.1 Create a New Flow and Load the Data

1. In the Watson Studio project, click on **Add to project** and select **Modeler flow** section.



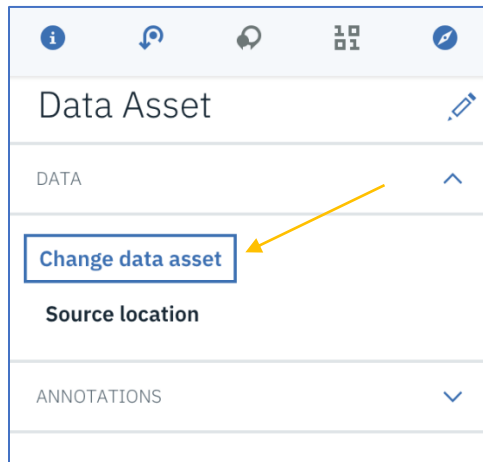
2. Enter a **Name** for the flow, optionally enter a **Description**, click on Modeler Flow for the **flow type** (should be the default), click on IBM SPSS Modeler for the **Runtime** (should be the default), and click on **Create**.

The screenshot shows the 'Modeler' 'New' dialog box. It has tabs for 'New', 'From file', and 'From example'. The 'Name' field contains 'Titanic-SPSS'. The 'Description' field is empty. Under 'Select flow type', 'Modeler Flow' is selected. Under 'Runtime', 'IBM SPSS Modeler' is selected. The 'Create' button is highlighted with a yellow arrow.

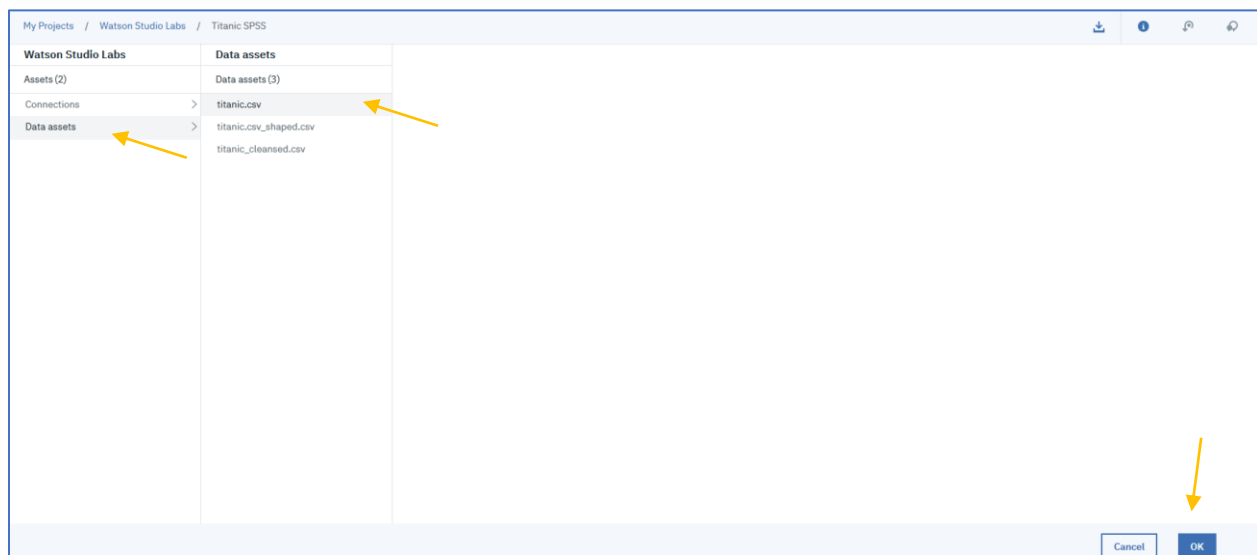
3. This opens the Flow Editor. Click on **Import** and then **Data Asset** and hold the left mouse key on the Data Asset icon and **drag it onto the left side of the canvas**. Release the left mouse key.



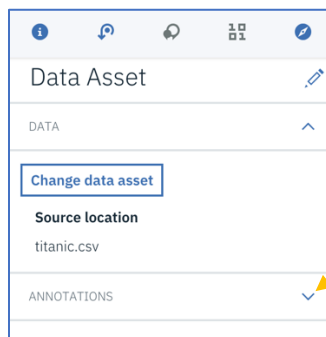
4. Double click on the **Data Asset**. In the window pane on the right-hand-side click on **Change data asset**.



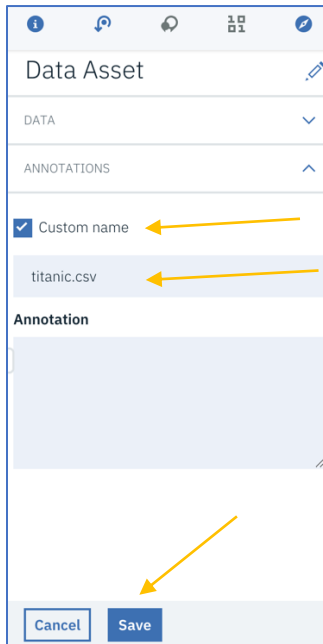
5. Click on **Data assets**, **titanic.csv** and click **OK**.




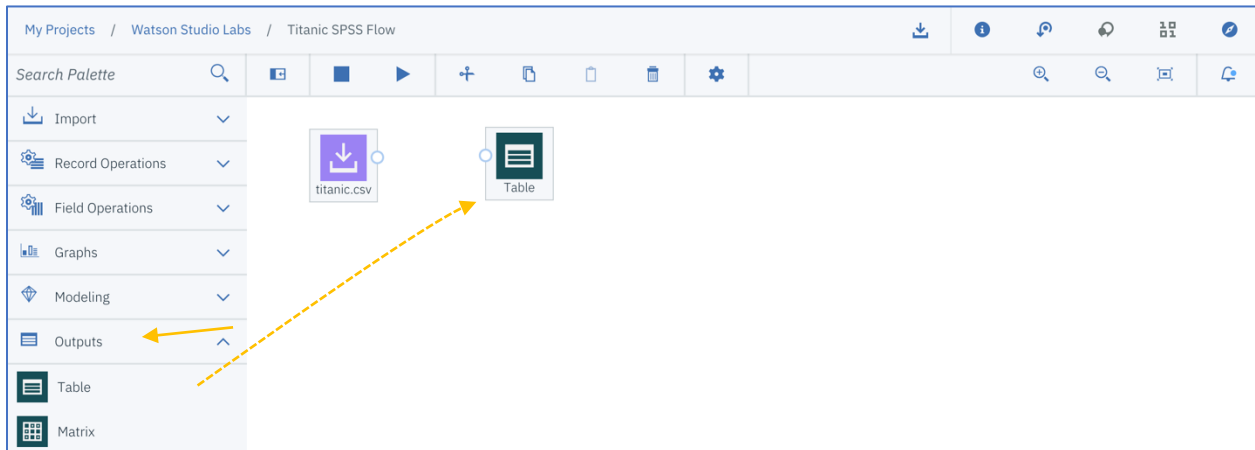
6. Click on **Annotation**.



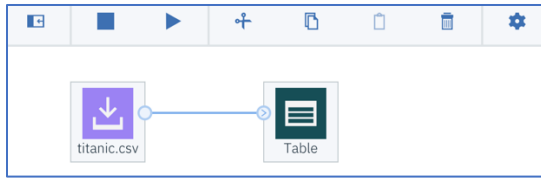
7. Click on **Custom name**, and type **titanic.csv**, and click on **Save**.



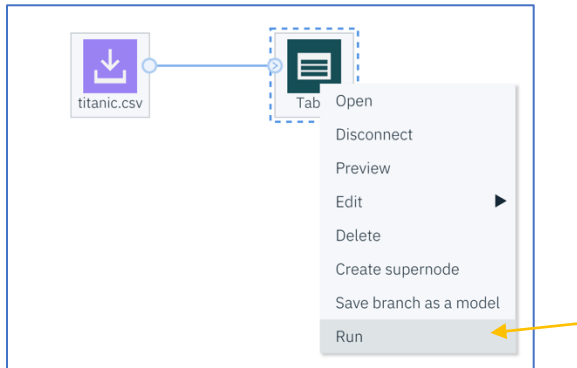
8. Click on the **Outputs** menu item in the Node Palette on the left and then click on the **Table** icon and drag the icon to the right of the titanic.csv icon. The SPSS Table node will display the contents of the csv file. If the Node Palette is not visible, click on the Node Palette icon .




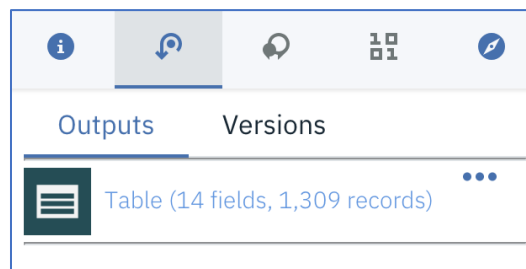
9. Connect the right side of the titanic.csv icon to the left side of the Table icon. This is accomplished by clicking on the little circle at the right side of the titanic.csv icon holding the left mouse key and dragging the mouse to the little circle on the left side of the Table icon, and then releasing the left mouse key.



10. Right click on the **Table** icon and select **Run**.



11. The “Running Flow” prompt will appear and then when completed a Table output selection will appear on the right side of the screen under the **Outputs** tab. If the Table output selection does not appear, select the  icon.




12. Double click on the Table selection and the contents of the titanic.csv will be displayed. Each row contains information on a passenger on the Titanic. We will use this data to make predictions on survivability.

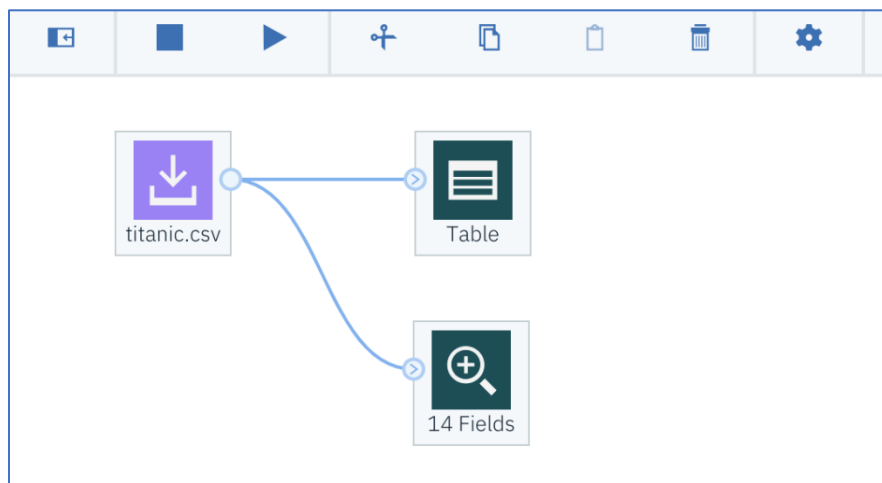
PCLASS	SURVIVED	NAME	SEX	AGE	SIBSP	PARCH	TICKET	FARE	CABIN	EMBARKED	BOAT
1	1	Allen, Miss. Elisabeth	female	29	0	0	24100	211.3375	B5	S	2
1	1	Allison, Master. Huds	male	0.9167	1	2	113781	151.55	C22 C26	S	11
1	0	Allison, Miss. Helen L	female	2	1	2	113781	151.55	C22 C26	S	
1	0	Allison, Mr. Hudson J	male	30	1	2	113781	151.55	C22 C26	S	
1	0	Allison, Mrs. Hudson	female	25	1	2	113781	151.55	C22 C26	S	
1	1	Anderson, Mr. Harry	male	48	0	0	19952	26.55	E12	S	3
1	1	Andrews, Miss. Korn	female	63	1	0	13502	77.9583	D7	S	10
1	0	Andrews, Mr. Thoma	male	39	0	0	112050	0	A36	S	
1	1	Appleton, Mrs. Edwa	female	53	2	0	11769	51.4792	C101	S	0
1	0	Artagaveytia, Mr. Rar	male	71	0	0	PC 17609	49.5042		C	
1	0	Astor, Col. John Jacc	male	47	1	0	PC 17757	227.525	C82 C84	C	
1	1	Astor, Mrs. John Jacc	female	18	1	0	PC 17757	227.525	C82 C84	C	4
1	1	Aubart, Mme. Leonie	female	24	0	0	PC 17477	69.3	B35	C	9
1	1	Barber, Miss. Ellen ?	female	26	0	0	19677	78.85		S	6
1	1	Barkworth, Mr. Alger	male	60	0	0	27042	30	A23	S	8

Page 1 / 7

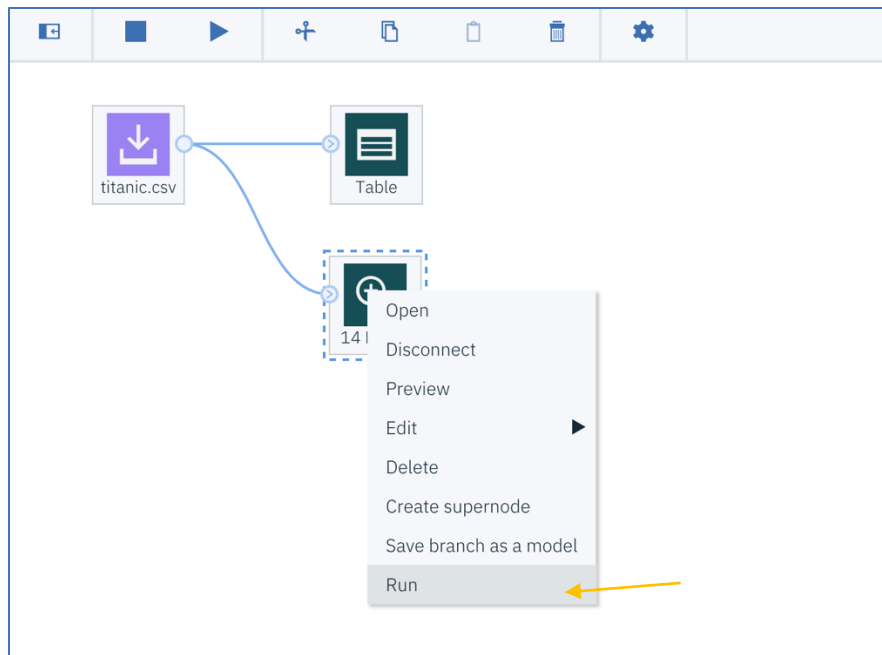
Step 2.2 Explore the Data using the Data Audit Node


Perusing through the data in the table, we can see that there are missing values. The SPSS Modeler has a Data Audit node that provides profiling information on the input data that is useful for cleansing the data. It provides a comprehensive first look at the data, including summary statistics, as well as information about outliers, missing values, and extremes.

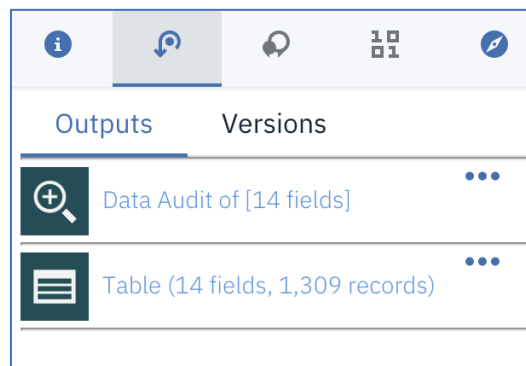
1. Add a **Data Audit** node to the flow clicking on the **Outputs** menu item in the Node Palette, and then dragging the **Data Audit** node to underneath the titanic.csv node. If the Node Palette is not visible, click on the Node Palette icon . Connect the titanic.csv node to the Data Audit node. The canvas should appear as below.



2. Right click on the **Data Audit** node and click **Run**.



3. The “Running Flow” prompt will appear and then when completed a Data Audit output selection will appear on the right side of the screen under the **Outputs** tab. If the **Outputs** tab doesn’t display, click on the  icon.




4. Double click on the **Data Audit of [14 fields]** to view the Data Audit output. We can see that several fields have many missing values (cabin, boat,body,home_dest). These fields will be removed using a **Filter** node below. Other fields have only a few missing values (fare, embarked, age). The rows containing the missing values will be removed using a **Select** node below.

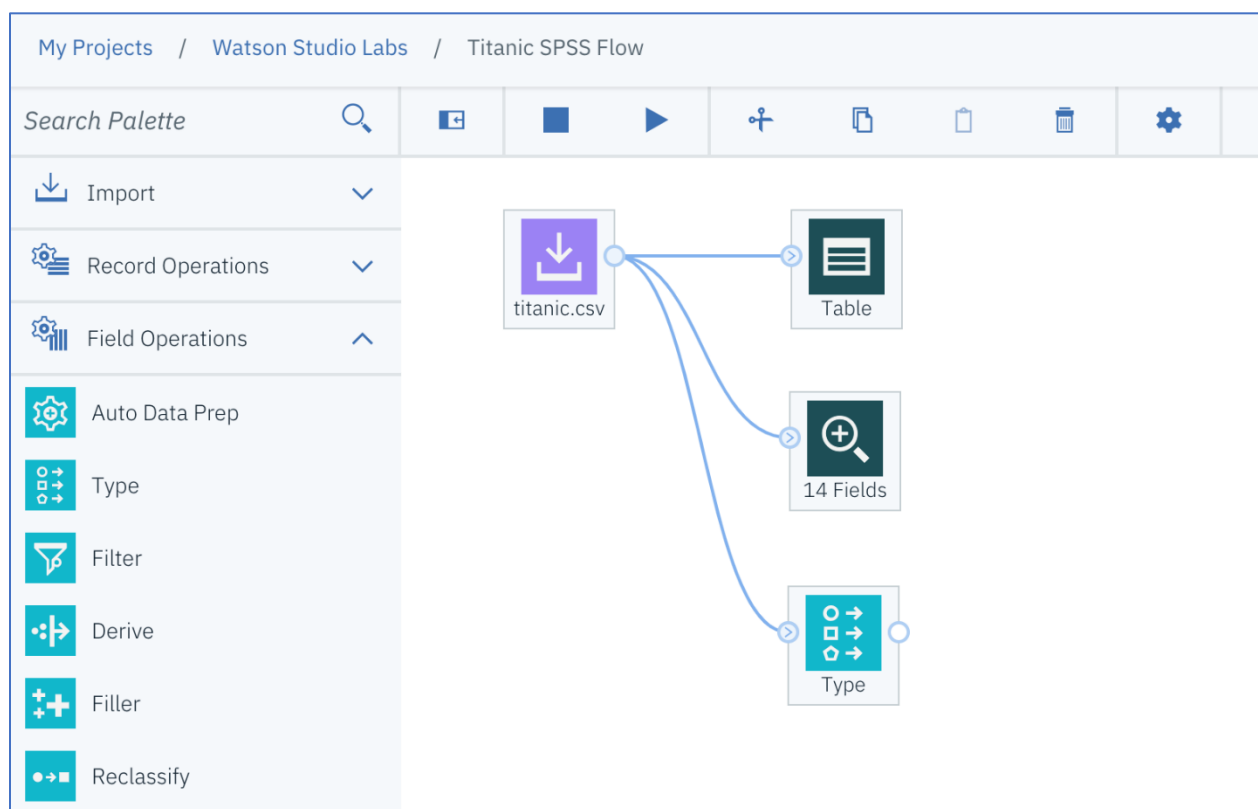
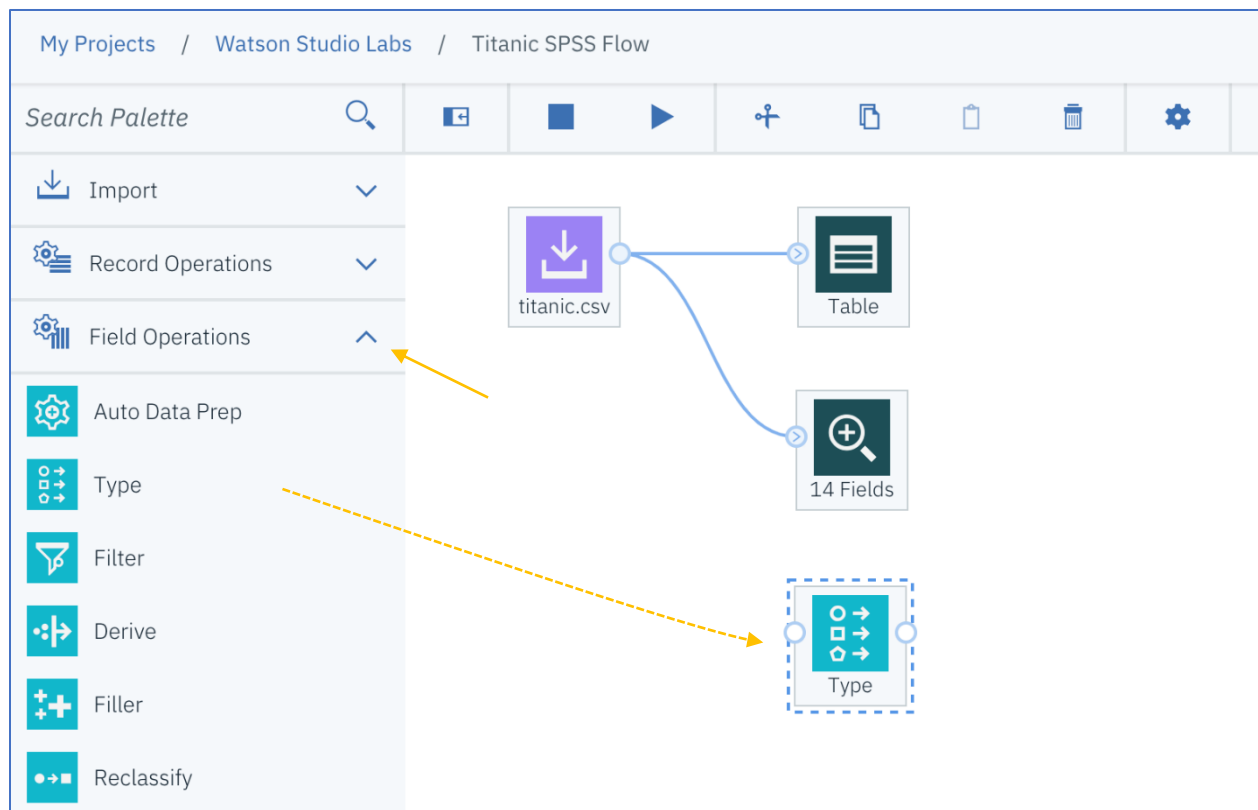
My Projects / Watson Studio Labs / Titanic SPSS Flow / Data Audit of [14 fields]

Data Audit of [14 fields]										
	Field	Graph	Measurement	Min	Max	Mean	Std. Dev	Skewness	Unique	Valid
1	pclass		Continuous	1	3	2.295	0.838	-0.599	--	1309
2	survived		Continuous	0	1	0.382	0.486	0.486	--	1309
3	name		Categorical	--	--	--	--	--	--	1309
4	sex		Categorical	--	--	--	--	--	2	1309
5	age		Continuous	0.167	80.000	29.881	14.413	0.408	--	1046
6	sibsp		Continuous	0	8	0.499	1.042	3.844	--	1309
7	parch		Continuous	0	9	0.385	0.866	3.669	--	1309

Step 2.3 Explore the Data using Graph Nodes.

Let's explore the data using Graph Nodes. The Distribution node, and the Histogram node will be used to explore some of the characteristics of the Titanic Data Set. First, we will add a Type node to the canvas. The Type node specifies field metadata and properties. We will change the measurement property for the "pclass" and "survived" fields that was derived as "Continuous" (by scanning the data values) to "Ordinal" and "Flag" respectively.

1. Add a **Type** node to the flow by clicking on the **Field Operations** menu item in the Node Palette and then drag the **Type** node underneath the **Data Audit** node. If the Node Palette is not visible, click on the Node Palette icon . Connect the titanic.csv node to the **Type** node. The canvas should appear as below.



2. Double click on the **Type** node. This will open a **Type** menu pallet on the right side of the screen.
3. Click on **Read Values**.

Type

SETTINGS

Default Mode

☒ Read metadata ☐ Pass (do not scan)

> Type Operations

Read Values **Clear All Values**

Field ^	Measure ^	Role ^	Value mode	Values ^	Check
---------	-----------	--------	------------	----------	-------

+ [Configure Missing Values](#)

FORMAT

ANNOTATIONS

Cancel **Save**

4. Select the dropdown in the Measure column next to **Survived**. Change the Measure from Continuous to Flag.

Read Values		Clear All Values			
Field ^	Measure ^	Role ^	Value modeValues ^	Check	
fare	Continuous	Input	Specify	0.0, 512.3292	None
ticket	Typeless	None	Specify		None
sex	Flag	Input	Specify	female, male	None
cabin	Nominal	Input	Specify	A10, A11, A14, A16...	None
survived	Continuous	Input	Specify	0, 1	None
body	Continuous	Input	Specify	1, 328	None
pclass	Continuous	Input	Specify	1, 3	None
sibsp	Continuous	Input	Specify	0, 8	None

[Configure Missing Values](#)

Read Values		Clear All Values			
Field ^	Measure ^	Role ^	Value modeValues ^	Check	
fare	Continuous	Input	Specify	0.0, 512.3292	None
ticket	Typeless	None	Specify		None
sex	Flag	Input	Specify	female, male	None
cabin	Nominal	Input	Specify	A10, A11, A14, A16...	None
survived	Continuous	Input	Specify	0, 1	None
body	Continuous	Input	Specify	1, 328	None
pclass	Continuous	Input	Specify	1, 3	None
sibsp	Continuous	Input	Specify	0, 8	None

Read Values		Clear All Values			
Field ^	Measure ^	Role ^	Value modeValues ^	Check	
fare	Continuous	Input	Specify	0.0, 512.3292	None ⚙
ticket	Text	None	Specify		None ⚙
sex	Flag	Input	Specify	female, male	None ⚙
cabin	Nominal	Input	Specify	A10, A11, A14, A16...	None ⚙
survived	Flag	Input	Specify	0, 1	None ⚙
body	Continuous	Input	Specify	1, 328	None ⚙
pclass	Continuous	Input	Specify	1, 3	None ⚙
sibsp	Continuous	Input	Specify	0, 8	None ⚙

5. Using the same process, change the Measure of **pclass** to **ordinal**.

Read Values		Clear All Values			
Field ^	Measure ^	Role ^	Value modeValues ^	Check	
fare	Continuous	Input	Specify	0.0, 512.3292	None ⚙
ticket	Text	None	Specify		None ⚙
sex	Flag	Input	Specify	female, male	None ⚙
cabin	Nominal	Input	Specify	A10, A11, A14, A16...	None ⚙
survived	Flag	Input	Specify	0, 1	None ⚙
body	Continuous	Input	Specify	1, 328	None ⚙
pclass	Ordinal	Input	Specify	1, 3	None ⚙
sibsp	Continuous	Input	Specify	0, 8	None ⚙

6. Click **Save**.

Type

SETTINGS

Default Mode

☒ Read metadata ☐ Pass (do not scan)

> Type Operations

Read Values

Clear All Values

Field ^	Measure ^	Role ^	Value modeValues ^	Check
fare	Continuous	Input	Specify 0.0, 512.3292	None
ticket	Typeless	None	Specify	None
sex	Flag	Input	Specify female, male	None
cabin	Nominal	Input	Specify A10, A11, A14, A16...	None
survived	Flag	Input	Specify 0, 1	None
body	Continuous	Input	Specify 1, 328	None
pclass	Ordinal	Input	Specify 1, 3	None
sibsp	Continuous	Input	Specify 0, 8	None


+ Configure Missing Values

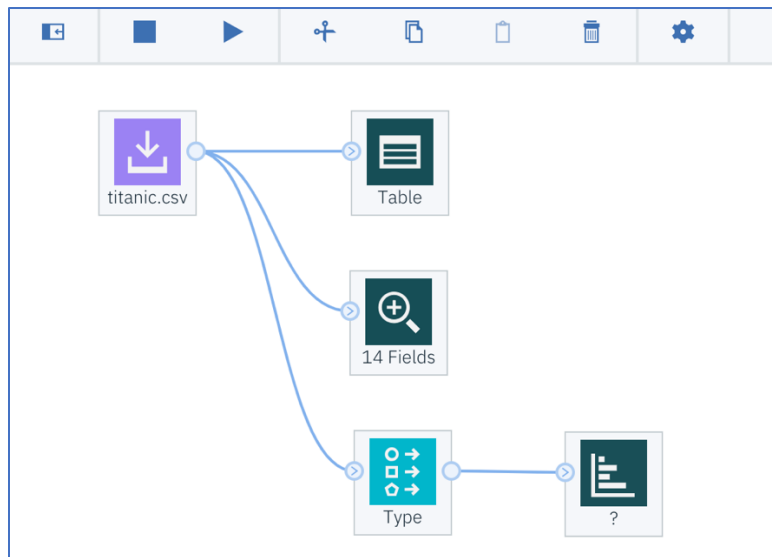
Missing Values

More than ten fields...

Cancel

Save

7. Add a **Distribution** node to the flow by clicking on the **Graph** menu item and then dragging the **Distribution** node to the canvas to the right of the **Type** node. If the Node Palette is not visible, click on the Node Palette icon . Connect the **Type** node to the **Distribution** node. The canvas should appear as below. The ? indicates that the fields to be plotted have not been identified.



8. Double click on the Distribution Node. Click on the **Plot** dropdown. In the **Field (discrete)** dropdown, select **pclass**. In the Color (discrete) dropdown, select **survived**. Click on the **normalize by color** checkbox, and then click **Save**.

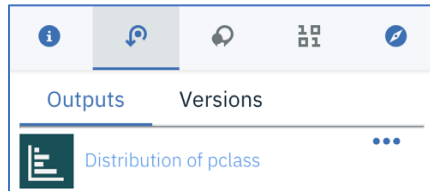
The configuration panel for the Distribution node is open, showing the following settings:

- PLOT**
 - Plot**
 - ☒ Selected fields
 - ☐ All flags (true values)
 - Field (discrete)**
 - pclass
 - Color (discrete)**
 - survived
 - ☒ Normalize by color
 - Sort**
 - ☐ Alphabetic
 - ☒ By count
 - ☐ Proportional scale
- APPEARANCE**
- ANNOTATIONS**
- Buttons:** Cancel, Save

9. Right click on the Distribution node and select **Run**.

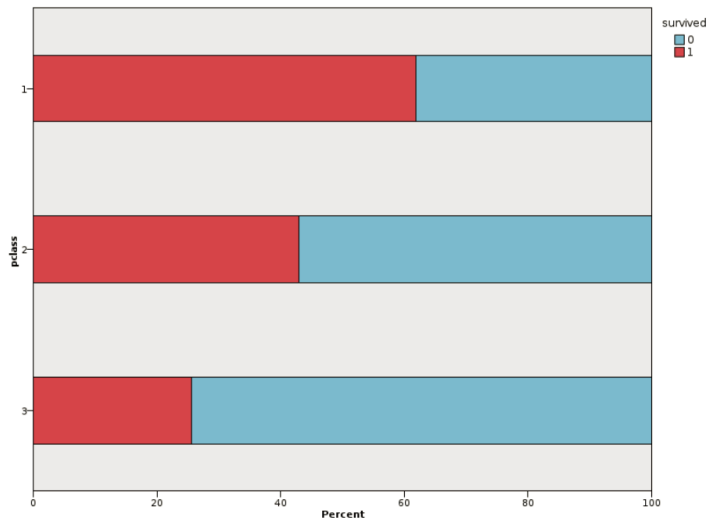


10. The Distribution of pclass output will appear under the **Outputs** tab.

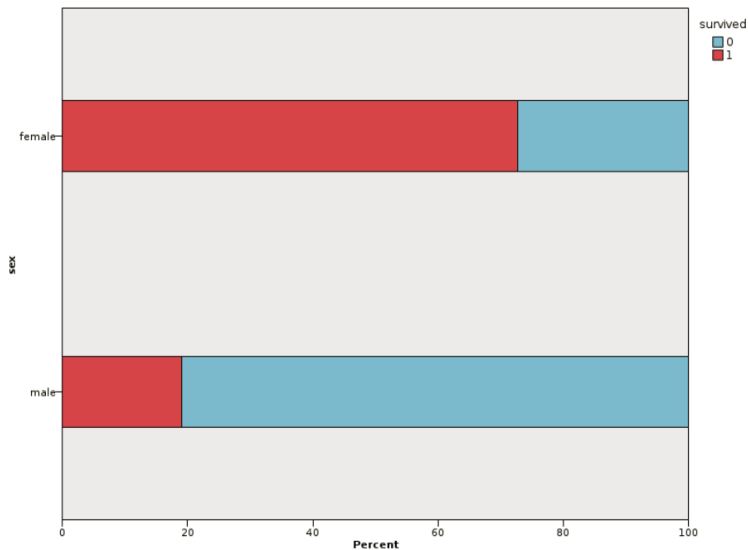



11. Double click on the **Distribution of pclass** to view the graph. We can see from the graph that the likelihood of surviving is correlated to the passenger class. The first-class passengers have the highest rate of survivability. **Note if you see a graph with green**

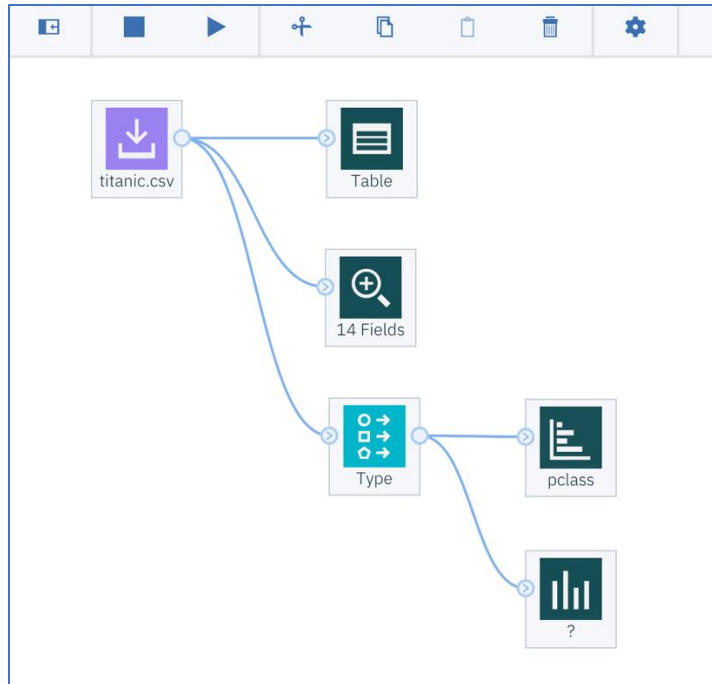
bars, instead of the one below, redo Steps 9-11.



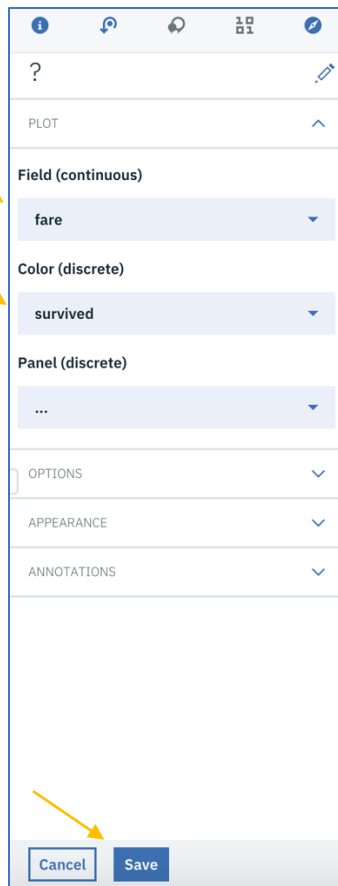
12. You can change the distribution graph to show the survivability by gender by double clicking on the Distribution node and replacing **pclass** with **sex** and clicking Save. Re-run the graph by right clicking on the Distribution node and selecting Run. Double click on the **Distribution of sex** to display the graph.



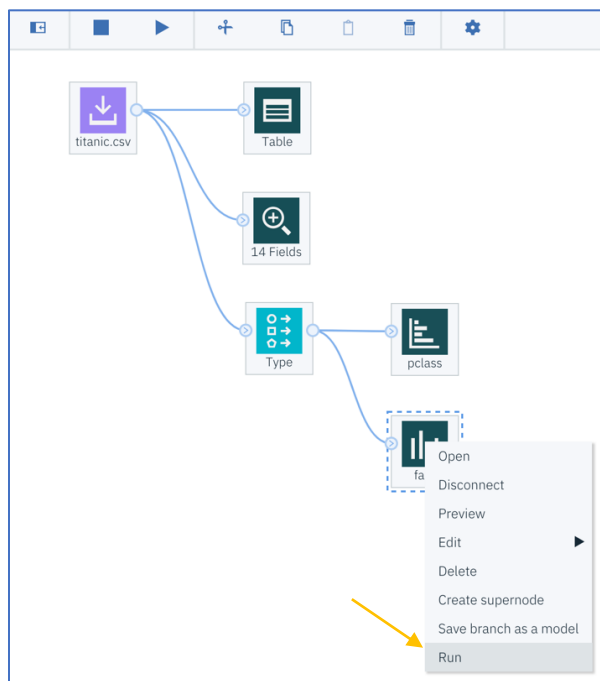
13. Add a **Histogram** node to the flow by clicking on the **Graphs** menu item and then dragging the **Histogram** node to the canvas underneath the **Distribution** node. If the Node Palette is not visible, click on the Node Palette icon . Connect the **Type** node to the **Histogram** node. The canvas should appear as below. The ? indicates that the fields to be plotted have not been identified.



14. Double click on the **Histogram** node. Click on the **Plot** dropdown. Select **fare** from the Field (continuous) dropdown. Select **survived** from the Color (discrete) dropdown. Click on **Save**.



15. Right click on the **Histogram** node and select **Run**.

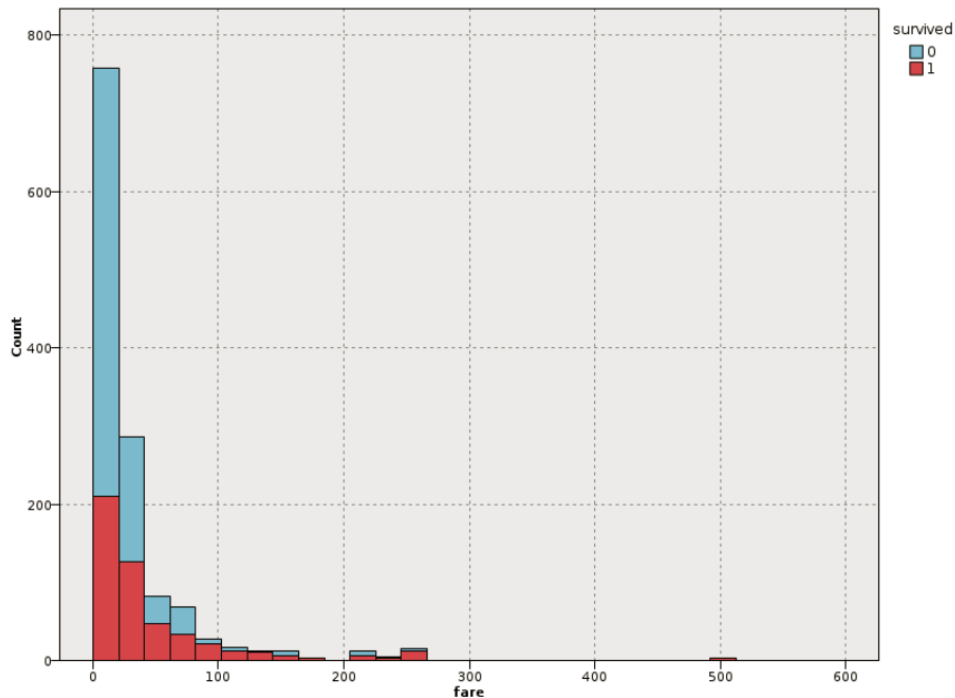


16. Double click on the Histogram of fare



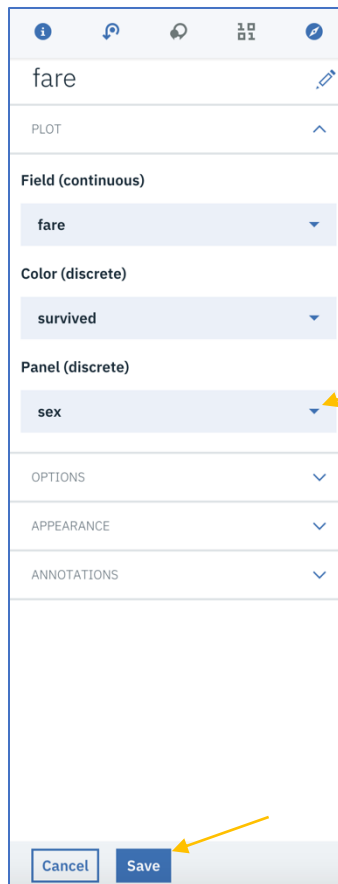
Histogram of fare

under the Outputs tab at the right of the screen.

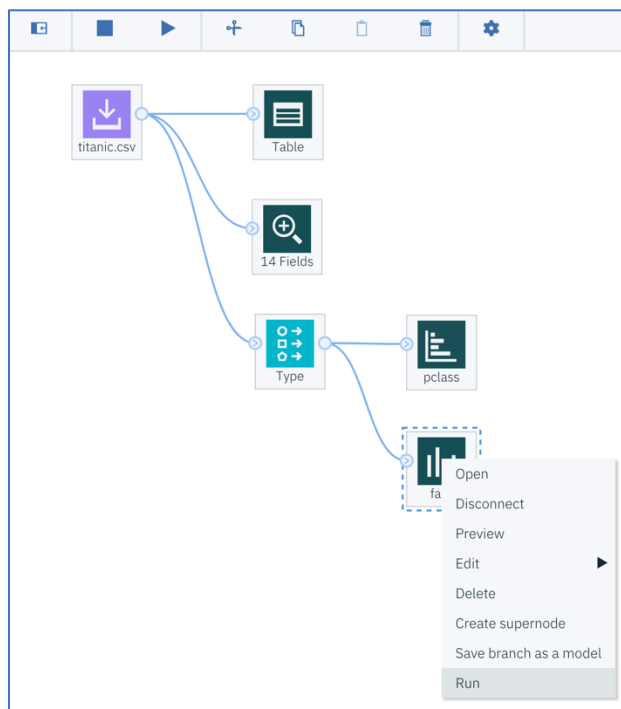


17. We can see that the higher fares have a higher percentage of survival. We can also see that the histogram is skewed. Skewness will impact the effectiveness of some machine learning techniques. One way to deal with skewness is to do a logarithmic transformation of the data. We will do this transformation in the preparing the data for modeling section below.

18. You can view the above graph separately for male and female passengers. DoubleClick the Histogram icon. In the **Panel (discrete)** select sex, and the click **Save**.



19. Right click on the Histogram and select **Run**.

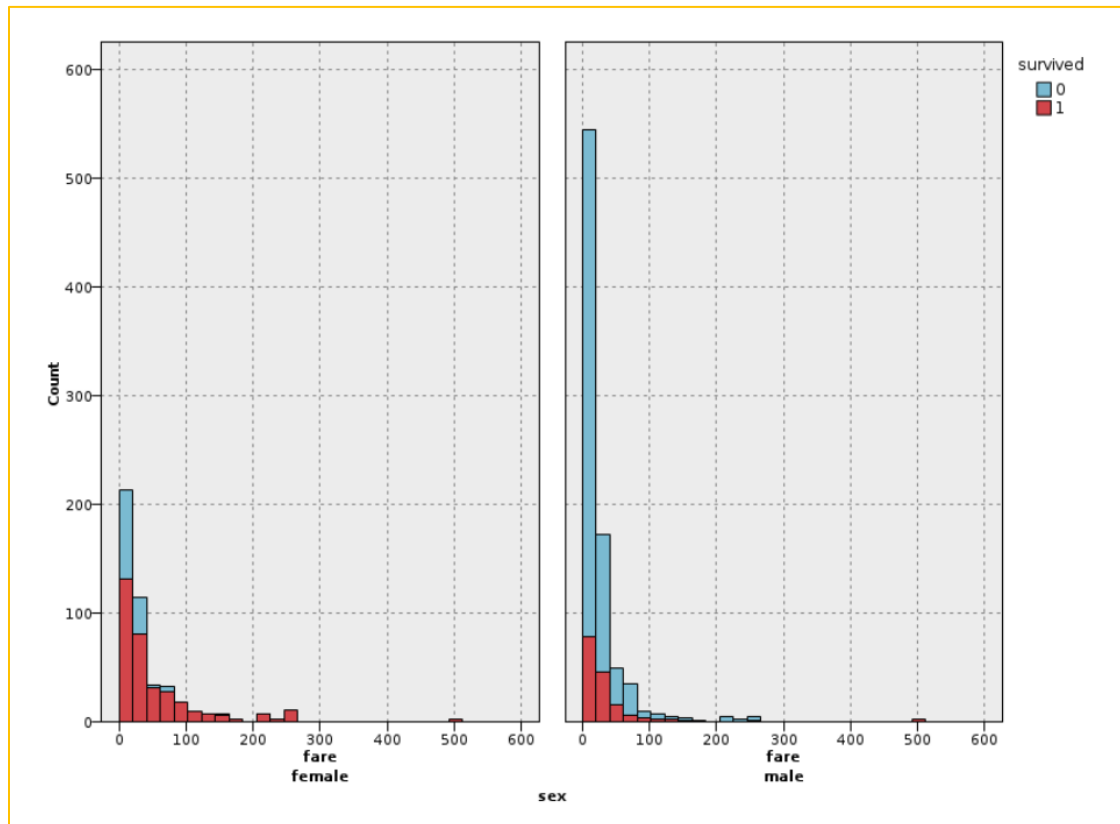




Histogram of fare #1

20. Double click on the Histogram of fare under the Outputs tab at the right of the screen.

under the Outputs tab




Step 2.4 Prepare the Data for Modeling

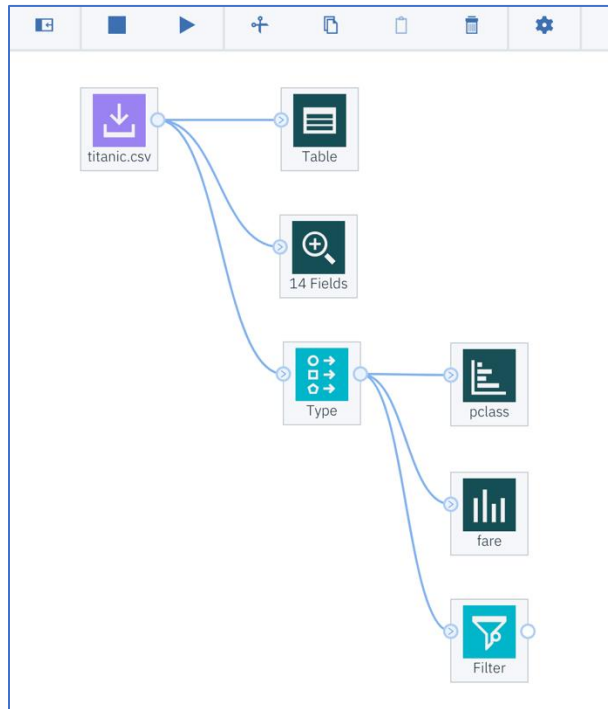
Based on our exploration of the data, there are several transformations that are needed to prepare the data for modeling. This section will introduce, the **Filter** node, the **Select** node, and the **Derive** node that will do the necessary transformations. The **Filter** and **Derive** nodes act on a field level, whereas the **Select** node acts on a record level.

Filter node – The **Filter** node performs two functions. It specifies fields that can be dropped. It also allows fields to be renamed. We will drop the fields cabin,boat,body, and home_dest.

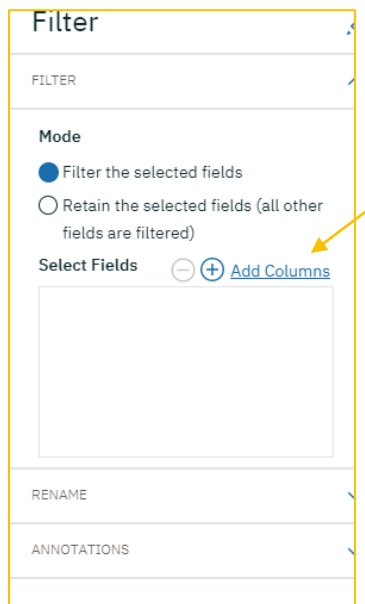
Derive node – The **Derive** node modifies data values or creates new fields from one or more existing fields. We will use the derive node to do a logarithmic transformation of the fare field. We will also use this node to bin the age and fare fields.

Select node – The **Select** node is used to select or discard a subset of records from the data stream based on a specific condition. We will remove the rows where there is missing information in the fare, age, or embarked fields.

1. Add a **Filter** node to drop fields with many missing values. Add the **Filter** node by clicking on the **Field Operations** menu item in the Node palette and dragging the **Filter** node onto the canvas underneath the fare **Histogram** node. If the Node Palette is not visible, click on the Node Palette icon  first. Connect the **Type** node to the **Filter** node. The canvas should appear as below.



2. Double click on the **Filter** node. Click on the **Filter** dropdown. In the Filter panel, click on **Add Columns**.



- Click on the checkboxes adjacent to the **cabin**, **boat**, **body**, and **home_dest** fields, and then click on **OK**.

Select Fields for Filter

Search in column Field name Filter: [Reset](#)

<input type="checkbox"/>	ticket	string
<input type="checkbox"/>	fare	double
<input checked="" type="checkbox"/>	cabin	string
<input type="checkbox"/>	embarked	string
<input checked="" type="checkbox"/>	boat	string
<input checked="" type="checkbox"/>	body	integer
<input checked="" type="checkbox"/>	home_dest	string

[Cancel](#) **OK**

- Click **Save** on the Filter panel.

Filter

FILTER

Mode

☒ Filter the selected fields

☐ Retain the selected fields (all other fields are filtered)

Select Fields

− + Add Columns

cabin

boat


body

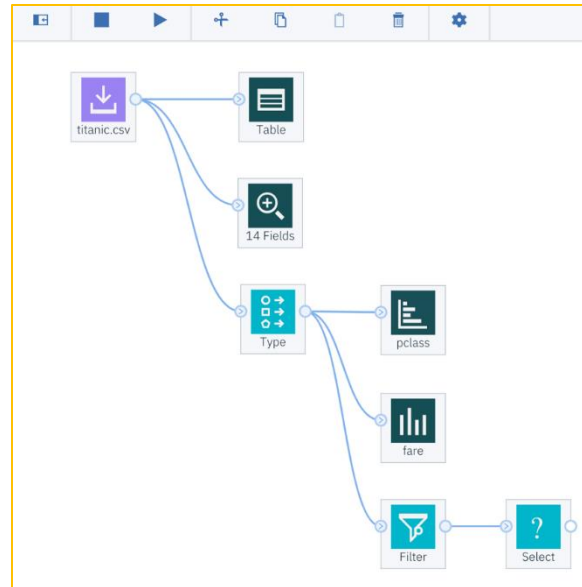
home.dest

RENAME

ANNOTATIONS

Cancel Save

5. Add a **Select** node by clicking on the **Record Operations** menu item in the Node palette, and then dragging the **Select** node to the canvas to the right of the **Filter** node. Connect the **Filter** node to the **Select** node. If the Node Palette is not visible, click on the Node Palette icon  first. The canvas should appear as below.



6. Double click on the **Select** node. Click on the **Settings** dropdown. In the **Select** panel, click on the **Discard** radio button, copy and paste (or type) the code shown below in the **Condition text box**, and then click **Save**.

@NULL (age) or embarked==" " or @NULL(fare)

Select


SETTINGS ^

Mode

☐ Include

☒ Discard


Condition

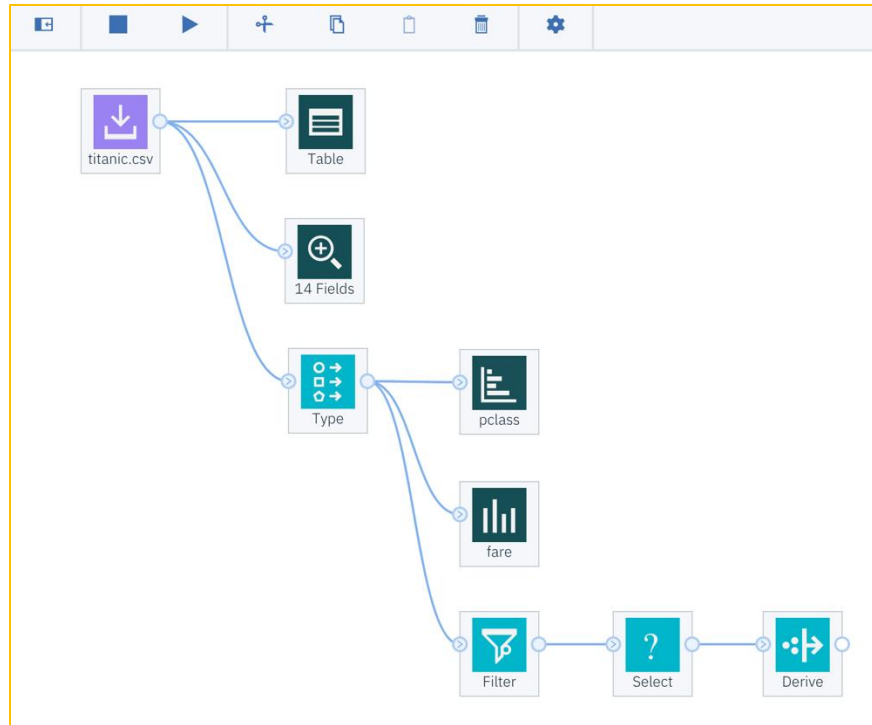


@NULL (age) or embarked==" " or @NULL

ANNOTATIONS v

Cancel Save

7. Add a **Derive** node to the canvas by clicking on the **Field Operations** menu item in the Node palette, and then dragging the **Derive node** onto the canvas to the right of the **Select** node. If the Node Palette is not visible, click on the Node Palette icon  first. Connect the **Select** node to the **Derive** node. The canvas should appear as below.



8. Double click on the **Derive** node. Click on the **Settings** Dropdown. Click on the **Single** radio button, enter log_fare for the **Derive** field, select **Continuous** for the measurement, copy and paste (or type) the following code in the **Expression** text box, and click Save.

```
if (fare /=0) then log(fare)
```

```
else 0
```

```
endif
```

Derive

SETTINGS

Mode

☒ Single field
 ☐ Multiple fields

Derived Field Name

log_fare

Derive As

Formula

Measurement

Continuous

Expression

```

if (fare /=0) then log(fare)
else 0
endif


```

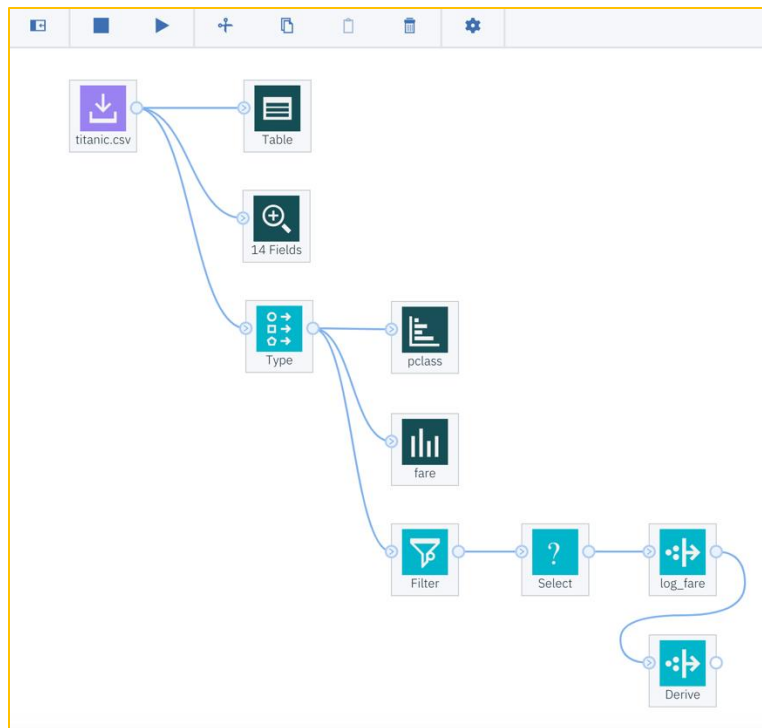
ANNOTATIONS

Cancel

Save

9. Binning of continuous fields is a technique sometimes used in preparing data for modeling. We will bin the age field, and the log_fare field. Add a **Derive** node by clicking on the **Field Operations** menu item in the Node palette and dragging the **Derive** node on the canvas underneath the log_fare **Derive** node.

If the Node Palette is not visible, click on the Node Palette icon  first. Connect the log_fare **Derive** node to the newly added **Derive** node. The canvas should appear as below.



10. Double click on the **Derive** node. Click on the **Settings** dropdown. Click on the **Single** radio button, enter age_bucket for the **Derive** field, select **Ordinal** for the **Measurement**, copy and paste the following code in the **Expression** text box, and then click **Save**.

```
if age >=0 and age < 6 then 0
else if age >=6 and age < 12 then 1
else if age>=12 and age< 18 then 2
else if age>=18 and age <40 then 3
else if age>=40 and age <65 then 4
else if age>=65 and age<80 then 5
else 6
endif
endif
endif
endif
endif
endif
```


The image shows the configuration window for a 'Derive' node. It has a title bar 'Derive' with an edit icon. Below is a 'SETTINGS' section with an expand/collapse arrow. The settings are organized into sections: 'Mode' with radio buttons for 'Single field' (selected) and 'Multiple fields'; 'Derived Field Name' with a text input containing 'age_bucket'; 'Derive As' with a dropdown menu set to 'Formula'; 'Measurement' with a dropdown menu set to 'Ordinal'; and 'Expression' with a text area containing an if-then statement. At the bottom are 'Cancel' and 'Save' buttons. Yellow arrows point to the 'Single field' radio button, the 'age_bucket' text input, the 'Ordinal' dropdown, the expression text area, and the 'Save' button.

Derive

SETTINGS

Mode

☒ Single field

☐ Multiple fields

Derived Field Name

age_bucket

Derive As

Formula

Measurement


Ordinal

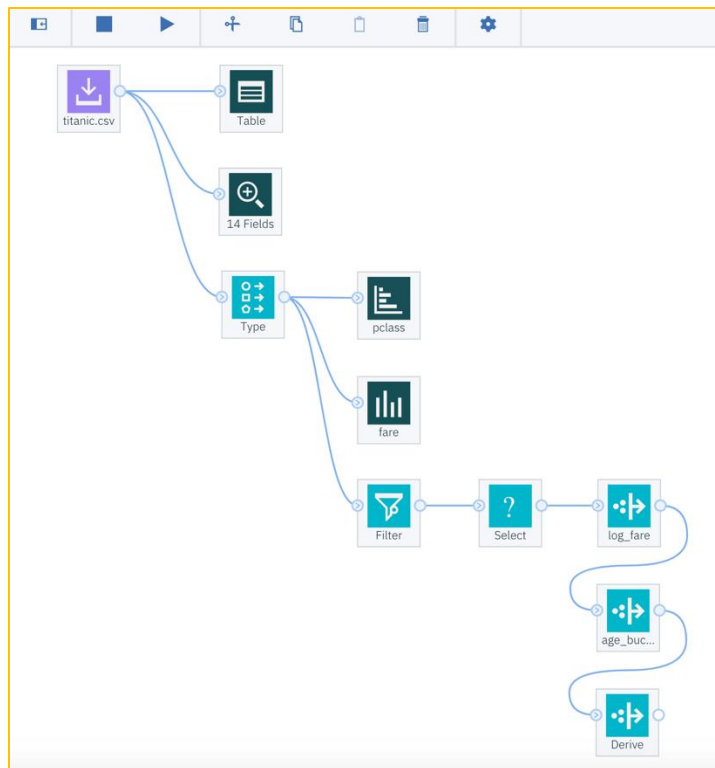
Expression

```
if age >=0 and age < 6 then 0
else if age >=6 and age < 12 the
else if age>=12 and age< 18 then
else if age>=18 and age <40 then
```

ANNOTATIONS

Cancel Save

11. Add a **Derive** node by clicking on the Field Operations menu item in the Node palette and dragging the **Derive** node onto the canvas underneath the age_bucket **Derive** node. You can click on the **zoom to fit** icon  in the top right to fit the flow to the canvas. Connect the age_bucket **Derive** node to the newly created **Derive** Node. The canvas should appear as below.



12. Double click the **Derive** node. In the **Derive** panel, click on the **Single** radio button, enter fare_bucket in the **Derive** field, click on Ordinal for the **Measurement**, copy and paste (or type) the following code in the **Expression** text box, and click on **Save**.

```

if log_fare < 0 then 0
else if log_fare > 8 then 9
else to_integer(log_fare)+1
endif
endif

```

Derive

SETTINGS

Mode

☒ Single field

☐ Multiple fields

Derived Field Name

fare_bucket

Derive As

Formula

Measurement

Ordinal

Expression

```
if log_fare < 0 then 0
else if log_fare > 8 then 9
else to_integer(log_fare)+1
endif
```

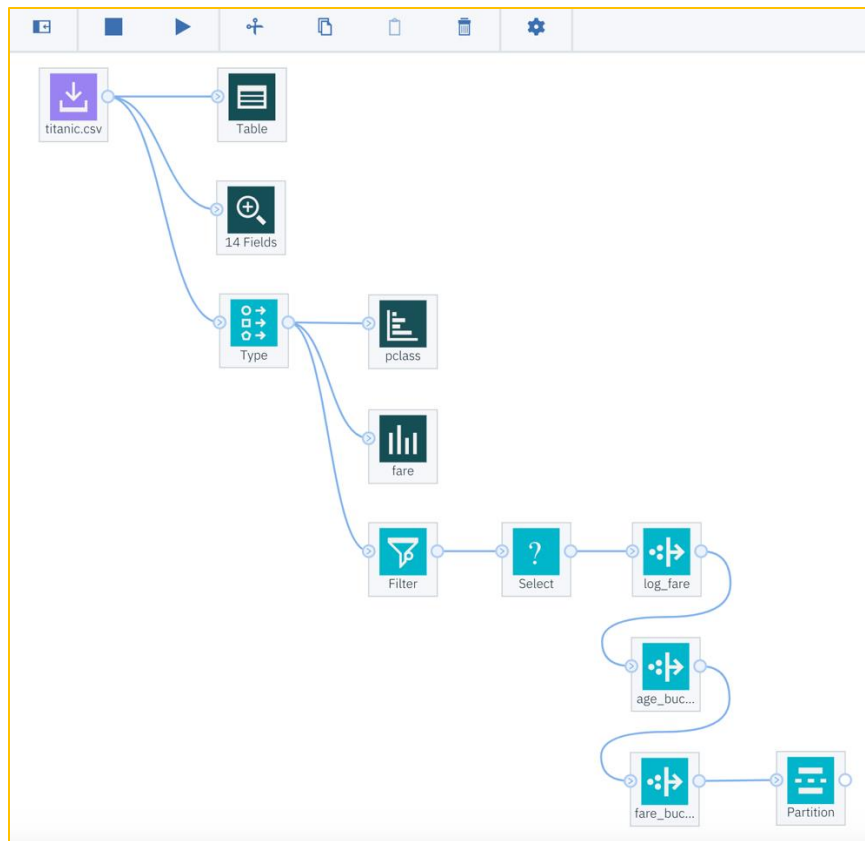
ANNOTATIONS

Cancel Save

Step 2.5 Modeling and Evaluation

Now that the data is prepared, we can start the modeling effort. First, we will add a **Partition** node to divide the data set into Training and Testing sets. In addition, a **Type** node is needed prior to modeling to type the new data fields that were created. Then we will add a **Logistic** node and use the Training set to train the model. Finally, we will add an **Analysis** node to evaluate the results.

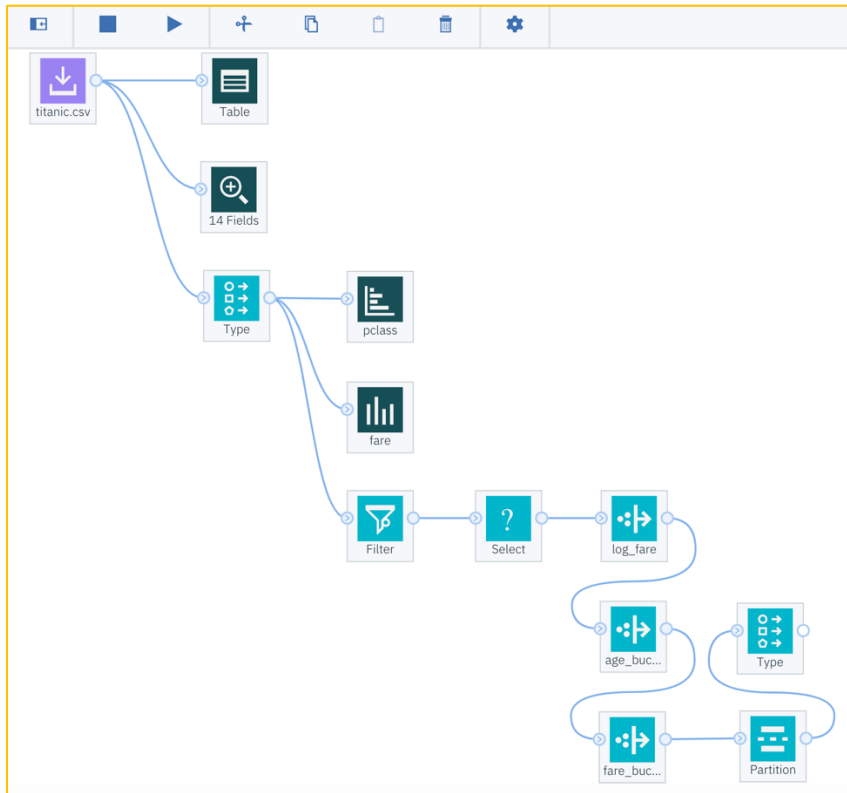
1. Add a **Partition** node by clicking on the Field Operations menu item in the Node palette and dragging the **Partition** node onto the canvas to the right of the fare_bucket **Derive** node. Connect the fare_bucket **Derive** node to the **Partition** node. The canvas should appear as below.



2. Double click on the Partition node. Set the **Training Partition** to 70 and the **Test Partition** to 30. Leave the other defaults and click on **Save**.

The screenshot shows the 'Partition' settings dialog box. It has a 'SETTINGS' section with the following options: 'Derived Field Name' (set to 'Partition'), 'Training Partition' (set to 70), 'Testing Partition' (set to 30), 'Create validation partition' (unchecked), 'Repeatable partition assignment' (checked), 'Seed' (set to 1234567), and 'Use unique field to assign partitions' (unchecked). There is an 'ANNOTATIONS' section at the bottom. At the very bottom are 'Cancel' and 'Save' buttons. Three yellow arrows point to the 'Training Partition' field, the 'Testing Partition' field, and the 'Save' button.

3. Add a **Type** node by clicking on the **Field Operations** in the Node palette and dragging the **Type** node onto the canvas above the **Partition** node. Connect the **Partition** node to the **Type** node. The canvas should appear as below.



4. Double click on the **Type** node. Click on **Read Values**.

Type

SETTINGS

Default Mode

☒ Read metadata ☐ Pass (do not scan)

Type Operations

Read Values Clear All Values

Field	Measure	Role	Value modeValues	Check
-------	---------	------	------------------	-------

Configure Missing Values

FORMAT

ANNOTATIONS

Cancel Save

5. For the log_fare, select **Continuous** for the **Measurement**. For the fare_bucket field, select **Ordinal** for the **Measurement**, and for the age_bucket, select **Ordinal** for the **Measurement**, (note these values should already be set correctly) and click **Save**.

Type

SETTINGS

Default Mode

☒ Read metadata ☐ Pass (do not scan)

> Type Operations

Read Values

Clear All Values

Field ^	Measure ^	Role ^	Value modeValues ^	Check
log_fare	Continuous	Input	Specify 0.0, 6.238967387...	None
fare_buc...	Ordinal	Input	Specify 1, 2, 3, 4, 5, 6, 7	None
age_buc...	Ordinal	Input	Specify 0, 1, 2, 3, 4, 5, 6	None

+ Configure Missing Values

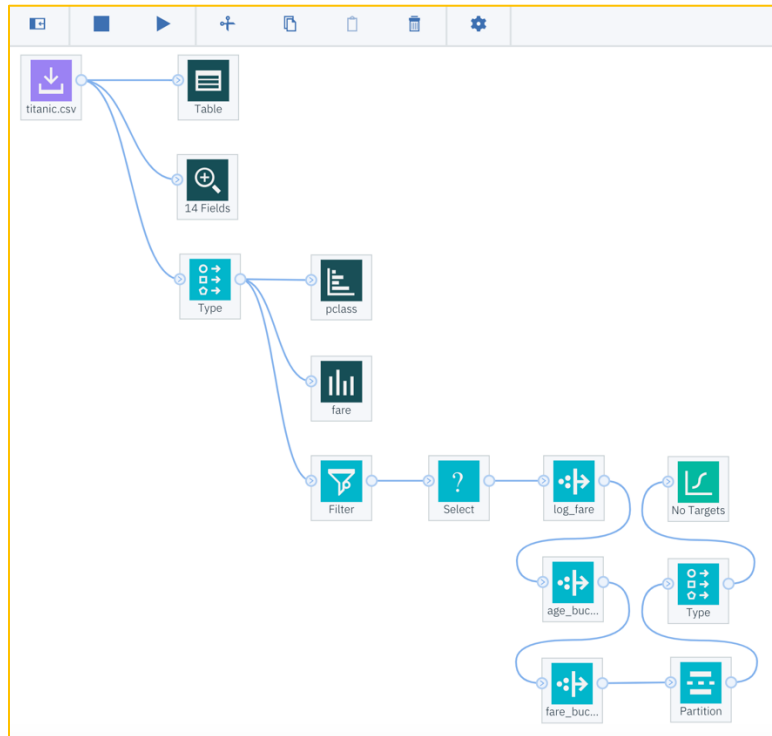
Missing Values

log_fare	false	[]
fare_bucket	false	[]

Cancel

Save

6. Add a **Logistic** node by clicking on the **Modeling** menu item in the Node palette and dragging the **Logistic** node onto the canvas above the **Type** node. Connect the **Type** node to the **Logistic** node. The canvas should appear as below.






7. Double click on the **Logistic** node. Click on the checkbox next to **Use custom field roles**, select **survived** for the **Target**, select **Partition** for the **Partition**, and click on **Add Columns** to add the input fields.


The image shows a configuration window titled "No Targets" with a pencil icon in the top right corner. The window is divided into several sections. At the top is a "FIELDS" section with an upward arrow. Below it is a checkbox labeled "Use custom field roles" which is checked; a yellow arrow points to this checkbox. Underneath is a "Target" section with a dropdown menu showing "survived"; a yellow arrow points to this dropdown. Below that is an "Inputs" section with a blue button labeled "Add Columns" (preceded by a minus and plus icon); a yellow arrow points to this button. Underneath the "Inputs" section is a large empty rectangular box. Below that is a "Partition" section with a dropdown menu showing "Partition"; a yellow arrow points to this dropdown. At the bottom of the window is an "ANNOTATIONS" section with a downward arrow. At the very bottom are two buttons: "Cancel" and "Save".













8. Click on the checkboxes next to **pclass**, **sex**, **sibsp**, **parch**, **embarked**, **age_bucket**, **fare_bucket** fields (you have to scroll down), and then click **OK**

Select Fields for No Targets

Search in column *Field name*


Filter:   

[Reset](#) 

<input type="checkbox"/>	Field name ^	Data type ^
<input checked="" type="checkbox"/>	pclass	 integer
<input type="checkbox"/>	name	 string
<input checked="" type="checkbox"/>	sex	 string
<input type="checkbox"/>	age	 double
<input checked="" type="checkbox"/>	sibsp	 integer
<input checked="" type="checkbox"/>	parch	 integer
<input type="checkbox"/>	ticket	 string
<input type="checkbox"/>	fare	 double
<input checked="" type="checkbox"/>	embarked	 string
<input type="checkbox"/>	log_fare	 double
<input checked="" type="checkbox"/>	age_bucket	 integer
<input checked="" type="checkbox"/>	fare_bucket	 integer

Cancel

OK



9. Click **Save**.

No Targets

FIELDS

☒ Use custom field roles

Target

survived

Inputs

[Add Columns](#)

pclass

sex

sibsp

parch

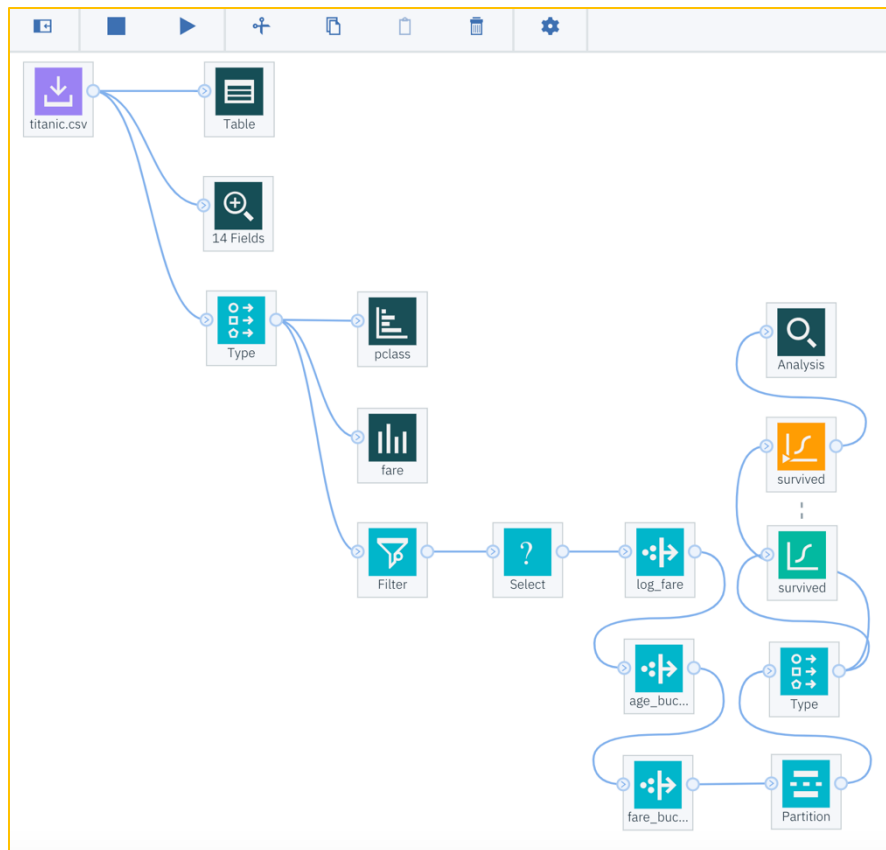
Partition

Partition

ANNOTATIONS

Cancel Save

10. Right click on the **Logistic** node and then click **Run**. A **Logistic** “nugget will be created” connected by a dotted line to the **Logistic** node. Note, it may be hidden under another node. Drag the nugget and place it above the **Logistic** node. The canvas should appear as below.



12. Double click on the Analysis node. Click on the **Settings** dropdown. Click on the **Evaluation metric** checkbox, and click on **Save**.

Analysis

SETTINGS

☐ Coincidence matrices (for symbolic targets)

☐ Performance evaluation

☒ Evaluation metric (AUC & Gini, binary classifiers only)

☐ Confidence figures (if available)

Threshold for pct. correct

90

Improve accuracy multiplier

2

Find predicted/predictor fields using

☒ Model output field metadata

☐ Field name format (for example, '\$<x>-<target field>')

Cancel

Save

13. Right click on the Analysis node and select Run. After completion, double click on



Analysis of [survived]

link in the Outputs tab on the right side of the screen. The results should be similar to those shown below.

Results for output field survived

Individual Models

Comparing \$L-survived with survived

'Partition'	1_Training		2_Testing	
Correct	579	78.99%	246	79.35%
Wrong	154	21.01%	64	20.65%
Total	733		310	

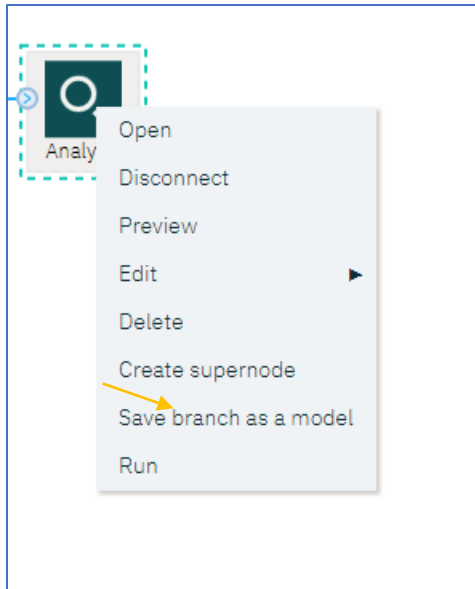
Evaluation Metrics

'Partition'	1_Training		2_Testing	
Model	AUC	Gini	AUC	Gini
\$L-survived	0.858	0.716	0.855	0.709

Step 2.6 Saving a Model

Now that we have created and evaluated a model, we will save the model as an asset. This saved model can be deployed at a future date, removing the need to recreate the same model from scratch.

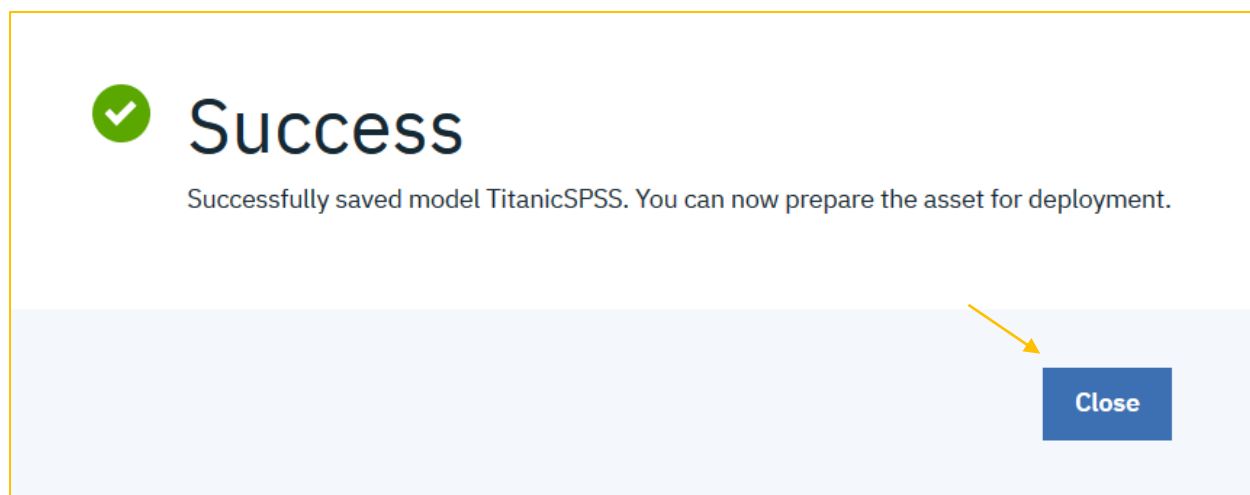
1. Right click on the Analysis node and then click on **Save branch as a model**.



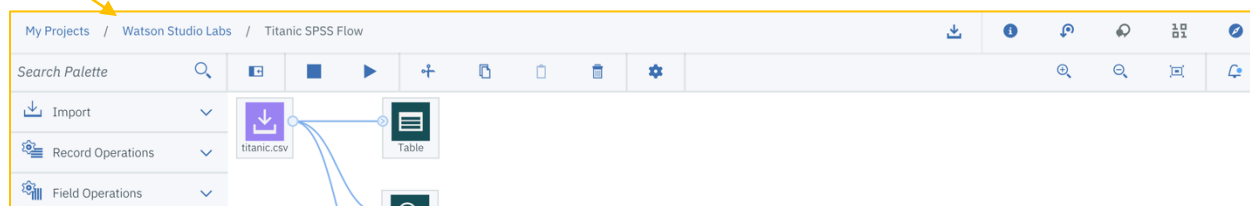
2. Type in “**TitanicSPSS**” as the Model Name and click **Save**.

A screenshot of the 'Save Model' dialog box in a software interface. The dialog has a title bar with 'My Projects / Watson Studio Labs / Titanic SPSS / Save'. The main content area is titled 'Save Model'. It contains several sections: 'Saving Mode' with radio buttons for 'Scoring branch' (selected) and 'Individual algorithm as PMML'; 'Branch Terminal Node*' with a dropdown menu showing 'Analysis'; 'Model name*' with a text input field containing 'TitanicSPSS'; 'Model description' with a large text area; and 'Machine Learning Service' with a dropdown menu showing 'WatsonMachineLearning'. At the bottom, there is a note: 'The model will be saved to your project. You can access your model and create deployments from the Models section under Assets.' and two buttons: 'Cancel' and 'Save' (highlighted with a yellow arrow).

3. Click **Close**.



4. Navigate to your project “assets” page. In this example, click on **Watson Studio Labs**.



5. Note that the model you built is now saved as an asset and the work you have completed can be easily reused in the future.

A screenshot of the "Models" section in Watson Studio Labs. It shows a table of "Watson Machine Learning models". An orange arrow points to the "TitanicSPSS" model entry in the table.

NAME	STATUS	TYPE	RUNTIME	LAST MODIFIED	ACTIONS
TitanicSPSS	trained	spss-modeler-18.1	spss-modeler-18.1	20 Jul 2019	
Titanic_AutoAI - P3 XGBClassifierEstimator	trained	wml-hybrid_0.1	hybrid_0.1	20 Jul 2019	
Single Convolution Layer on MNIST	trained	tensorflow-1.5	python-3.5	20 Jul 2019	
XGB_Heart_Disease_Detection	trained	scikit-learn-0.19	python-3.5	20 Jul 2019	