**Analysis Oscar Nominations using Two Class Decision Forest**

Yu-lin Shih [yulinshih@calstatela.edu](mailto:yulinshih@calstatela.edu)

Jiyang Liy jliu116@calstatela.edu

Tinayi Deng tdeng3@calstatela.edu

Biran li

**Overview**

In this tutorial, you will implement a classification model using **Two Class Decision Forest** that uses features of a movie to predict whether or not the movie will be nominated as **Best Picture** in Oscar award. You should follow the steps below to build, train and test the model

**What You’ll Need**

To complete this lab, you will need the following:

• An Azure ML account

• A web browser and Internet connection

• A Movie\_Oscar database

**Preparing and Exploring the Data**

In this tutorial you will work with a movie\_oscar dataset that contains data about movies. After train and evaluate the Two-class decision tree model we can predict whether the movie will be nominated as Oscar “Best Picture” or not.

**Upload the Data Set**

1. Create a new blank experiment, and give it a title .

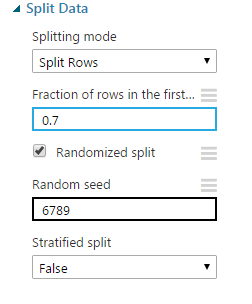
2. At the bottom left, click **NEW**. Then in the **NEW** dialog box, click **DATASET.**

3. Click **FROM LOCAL FILE**. Then in the **Upload a new dataset** dialog box, browse to select the **Movie\_Oscar.csv** file and then click the **OK** icon.

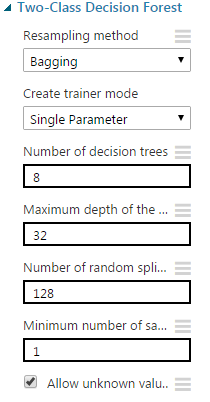
4. Expand **Saved Datasets** and **My Datasets** to verify that the **Movie\_Oscar** dataset is listed.

Building a Classification Model

1. Drag the **Movie\_Oscar** and onto the canvas.
2. Search for the **Select Columns in Dataset** module and drag it onto your canvas. Connect the **Results Dataset** output of the **Movie\_Oscar** module to the input port of the **Select Columns in Dataset module**.
3. With the **Select Columns in Dataset** module selected, in the properties pane, launch the column selector except title\_year and winner.
4. Search for the **Clean Missing Data** module and drag it onto your canvas. Connect the **Results Dataset** output of the **Select Columns in Dataset** module to the input port of the **Clean Missing Data**.
5. With the **Clean Missing Data** module selected, in the properties pane, launch the all th columns and set Cleaning mode as **remove entire row**.
6. Search for the **Split Data (Split)** module. Drag this module onto your experiment canvas. Connect the **Results dataset** output port of the **Clean Missing Data** module to the **Dataset** input port of the **Split Data (Split)** module. Set the **Properties** of the **Split Data (Split)** module as follows:



1. Search for the **Two Class Decision Forest** module. Make sure you have selected the regression model version of this algorithm. Drag this module onto the canvas. Set the Properties if this module as follows:



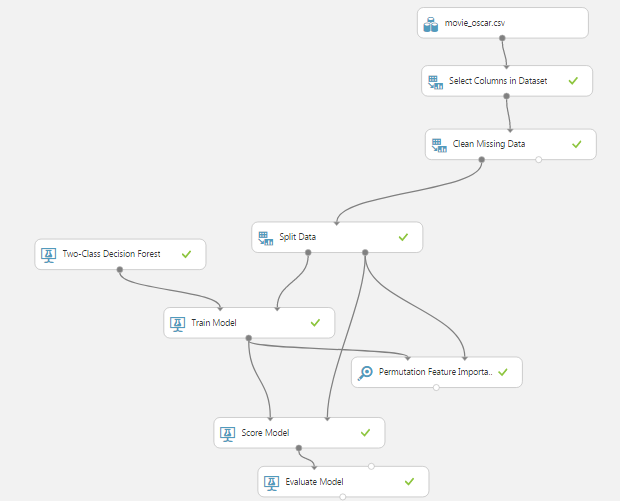
6. Search for the **Train Model** module. Drag this module onto the canvas.

7. Connect the **Untrained Model** output port of the **Two Class Decision Forest** module to the **Untrained Model** input port of the **Train Model** module. Connect the **Results dataset1** output port of the **Split Data (Split)** module to the **Dataset input** port of the **Train model** module. On the **Properties** pane, launch the column selector and select the **Award** column.

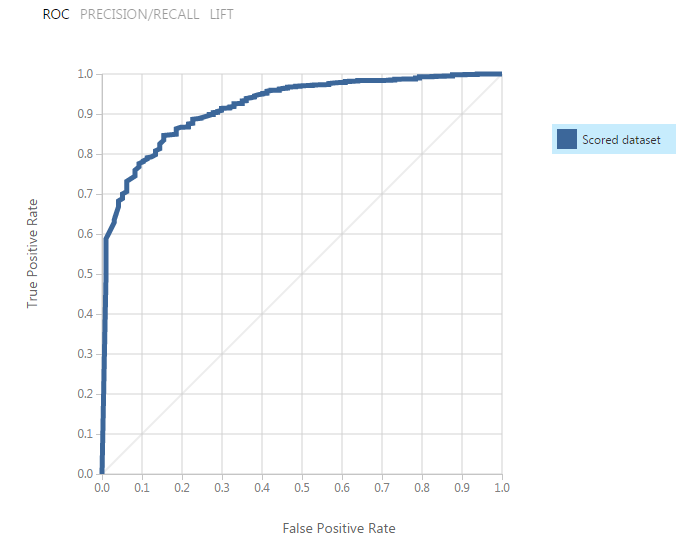
8. Search for the **Score Model** module and drag it onto the canvas.

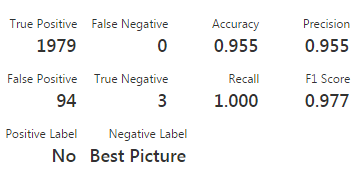
9. Connect the **Trained Model** output port of the of the **Train Model** module to the **Trained Model** input port of the **Score Model** module. Connect the **Results dataset2** output port of the **Split Data (Split)** module to the **Dataset** port of the **Score Model** module.

10. Search for the **Permutation Feature Importance** module and drag it onto the canvas. Connect the **Trained Model** output port of the **Train Model** module to the **Trained model** input port of the **Permutation Feature Importance** module. Connect the **Results dataset2** output port of the **Split Data (Split)** module to the **Dataset** port of the **Test data** input port of the **Permutation Feature Importance** module.

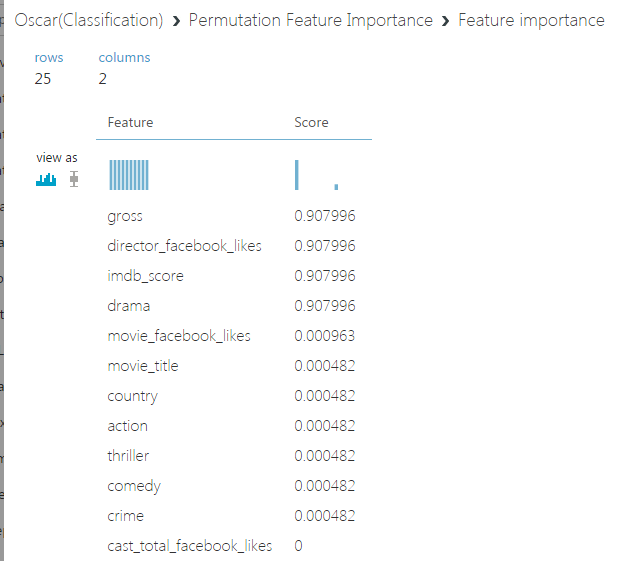
11. Search for the **Evaluate Model** module and drag it onto the canvas. Connect the **Scored Dataset** output port of the **Score Model** module to the left hand **Scored dataset** input port of the **Evaluate Model** module. Your mapping will look like below: 

12. Save and run the experiment. When the experiment is finished, visualize the **Evaluation Result** port of the **Evaluate Model** module and review the ROC curve and performance statistics for the model as shown below.





13. Visualize the output of the **Permutation Feature Importance** module. The upper portion of the list produced should resemble the following:



**This is the end of the tutorial**