



(<https://cognitiveclass.ai>)

## From Modeling to Evaluation

In [ ]:

```
#by Christopher Harrison
```

### Introduction

In this lab, we will continue learning about the data science methodology, and focus on the **Modeling** and **Evaluation** stages.

---

## Table of Contents

- 1. [Recap](#)
  - 2. [Data Modeling](#)
  - 3. [Model Evaluation](#)
- </div>
- 

## Recap

In Lab **From Understanding to Preparation**, we explored the data and prepared it for modeling.

The data was compiled by a researcher named Yong-Yeol Ahn, who scraped tens of thousands of food recipes (cuisines and ingredients) from three different websites, namely:

The screenshot shows the Allrecipes.com website. At the top, there's a navigation bar with links for search, shopping lists, menu planner, cooking school, and user account options. The main feature is the "Recipe of the Day," which is "Grilled Italian Pork Chops." Below the recipe title are five star ratings and a link to "See Reviews (23)". A small image of the dish is shown. To the right of the recipe is an advertisement for "Allrecipes Magazine" featuring a photo of lasagna and a "Subscribe" button. Below the main content, there's a "Most-Saved Recipes" section with six items, each with a thumbnail, a title, and a star rating. To the right, there's a "In Season" section with links to "Summer Fruit Desserts" and "Marinades Ramp It Up".

[www.allrecipes.com](http://www.allrecipes.com)

The screenshot shows the Epicurious.com website. At the top, there's a navigation bar with links for RECIPES & MENUS, EXPERT ADVICE, INGREDIENTS, HOLIDAYS & EVENTS, and COMMUNITY. Below the navigation, there are two large recipe features. On the left, under the heading "MENU", is a bowl of seafood soup with shrimp and mussels. The heading below it is "A Summery Seafood Dinner for Every Night of the Week". On the right, under the heading "GRILL", is a plate of grilled shrimp skewers with a small jar of dipping sauce. The heading below it is "This Weekend, Grill a Whole Fish". Both sections include author information at the bottom.

[www.epicurious.com](http://www.epicurious.com)

레스토랑	주소	전화번호	평점
곰바위	서울 강남구 삼성동	(02) 511-0068	★ ★★★★ 2.9
유원	서울 송파구 잠실동	(02) 416-7466	★ ★★★★ 4.3
공리	서울 강남구 대치동	(02) 562-0110	★ ★★★★ 4.3
요리하는남자	서울 송파구 잠실동	(02) 419-1511	★ ★★★★ 4.6

[www.menupan.com](http://www.menupan.com)

For more information on Yong-Yeol Ahn and his research, you can read his paper on [Flavor Network and the Principles of Food Pairing.](http://yongyeol.com/papers/ahn-flavornet-2011.pdf) (<http://yongyeol.com/papers/ahn-flavornet-2011.pdf>).

**Important note:** Please note that you are not expected to know how to program in Python. This lab is meant to illustrate the stages of modeling and evaluation of the data science methodology, so it is totally fine if you do not understand the individual lines of code. We have a full course on programming in Python, [Python for Data Science](http://cocl.us/PY0101EN_DS0103EN_LAB4_PYTHON_Coursera) ([http://cocl.us/PY0101EN\\_DS0103EN\\_LAB4\\_PYTHON\\_Coursera](http://cocl.us/PY0101EN_DS0103EN_LAB4_PYTHON_Coursera)), which is also offered on Coursera. So make sure to complete the Python course if you are interested in learning how to program in Python.

## Using this notebook:

To run any of the following cells of code, you can type **Shift + Enter** to execute the code in a cell.

Download the library and dependencies that we will need to run this lab.

In [1]:

We already placed the data on an IBM server for your convenience, so let's download it from server and read it into a dataframe called **recipes**.

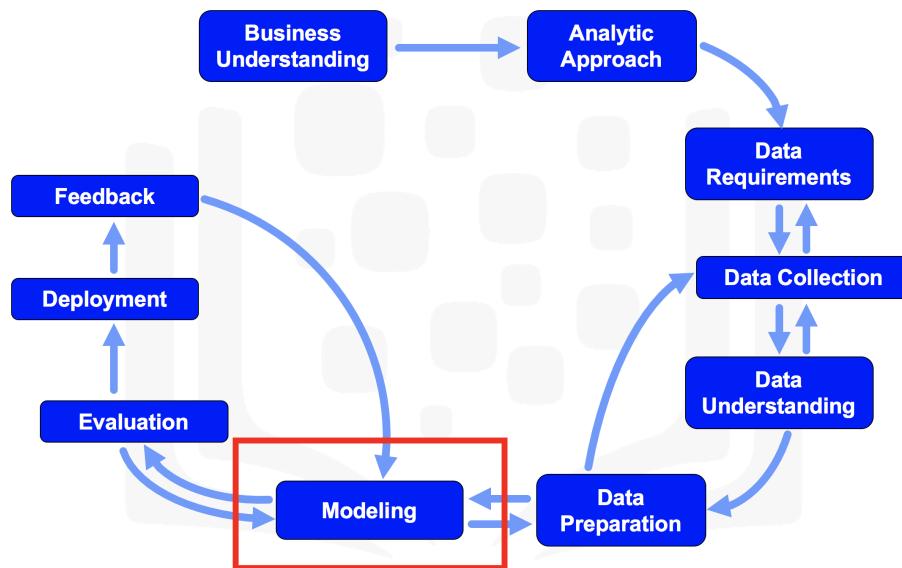
In [2]:

```
Data read into dataframe!
```

We will repeat the preprocessing steps that we implemented in Lab **From Understanding to Preparation** in order to prepare the data for modeling. For more details on preparing the data, please refer to Lab **From Understanding to Preparation**.

In [3]:

## Data Modeling



Download and install more libraries and dependies to build decision trees.

In [4]:

```
Collecting package metadata (current_repodata.json): done
Solving environment: done
```

```
## Package Plan ##
```

```
environment location: /home/jupyterlab/conda/envs/python
```

```
added / updated specs:
- python-graphviz
```

The following packages will be downloaded:

package		build
ca-certificates-2020.12.8		h06a4308_0
certifi-2020.12.5		py36h06a4308_0
openssl-1.1.1i		h27cf23_0
python-graphviz-0.15		pyhd3eb1b0_0
		Total: 2.8 MB

The following NEW packages will be INSTALLED:

```
python-graphviz      pkgs/main/noarch::python-graphviz-0.15-pyhd3eb1b0
_0
```

The following packages will be UPDATED:

```
ca-certificates      conda-forge::ca-certificates-2020.12.~ --> pkgs/main::ca-certificates-2020.12.8-h06a4308_0
```

The following packages will be SUPERSEDED by a higher-priority channel:

```
certifi              conda-forge::certifi-2020.12.5-py36h5~ --> pkgs/main::certifi-2020.12.5-py36h06a4308_0
openssl              conda-forge::openssl-1.1.1i-h7f98852_0 --> pkgs/main::openssl-1.1.1i-h27cf23_0
```

### Downloading and Extracting Packages

```
python-graphviz-0.15 | 20 KB      | #####
## | 100%
openssl-1.1.1i       | 2.5 MB     | #####
## | 100%
certifi-2020.12.5    | 140 KB     | #####
## | 100%
ca-certificates-2020 | 121 KB     | #####
## | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```



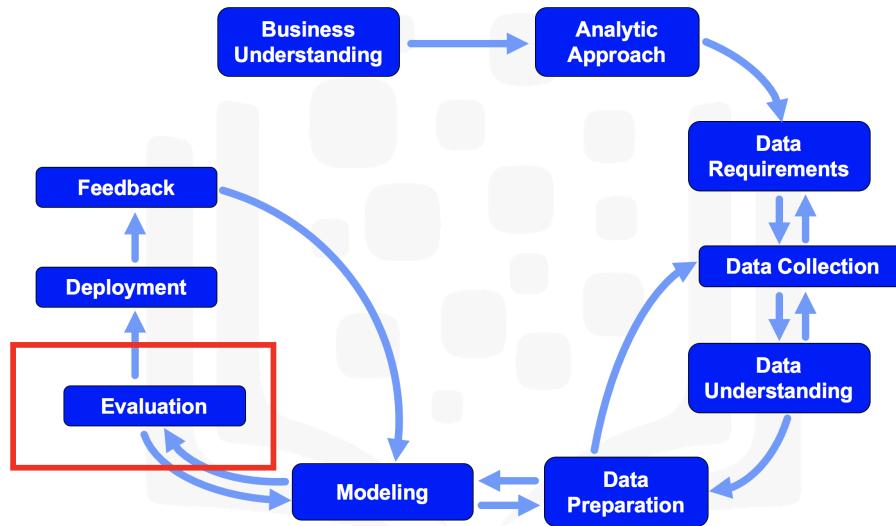
The decision tree learned:

- If a recipe contains *cumin* and *fish* and **no yoghurt**, then it is most likely a **Thai** recipe.
- If a recipe contains *cumin* but **no fish** and **no soy\_sauce**, then it is most likely an **Indian** recipe.

You can analyze the remaining branches of the tree to come up with similar rules for determining the cuisine of different recipes.

Feel free to select another subset of cuisines and build a decision tree of their recipes. You can select some European cuisines and build a decision tree to explore the ingredients that differentiate them.

## Model Evaluation



To evaluate our model of Asian and Indian cuisines, we will split our dataset into a training set and a test set. We will build the decision tree using the training set. Then, we will test the model on the test set and compare the cuisines that the model predicts to the actual cuisines.

Let's first create a new dataframe using only the data pertaining to the Asian and the Indian cuisines, and let's call the new dataframe **bamboo**.

In [8]:

Let's see how many recipes exist for each cuisine.

In [9]:

Out[9]:

```
korean    799  
indian    598  
chinese   442  
japanese  320  
thai      289  
Name: cuisine, dtype: int64
```

Let's remove 30 recipes from each cuisine to use as the test set, and let's name this test set **bamboo\_test**.

In [10]:

Create a dataframe containing 30 recipes from each cuisine, selected randomly.

In [11]:

Check that there are 30 recipes for each cuisine.

In [12]:

Out[12]:

```
thai      30  
chinese  30  
indian   30  
japanese 30  
korean   30  
Name: cuisine, dtype: int64
```

Next, let's create the training set by removing the test set from the **bamboo** dataset, and let's call the training set **bamboo\_train**.

In [13]:

Check that there are 30 *fewer* recipes now for each cuisine.

In [14]:

Out[14]:

```
korean    769  
indian    568  
chinese   412  
japanese  290  
thai      259  
Name: cuisine, dtype: int64
```

Let's build the decision tree using the training set, **bamboo\_train**, and name the generated tree **bamboo\_train\_tree** for prediction.

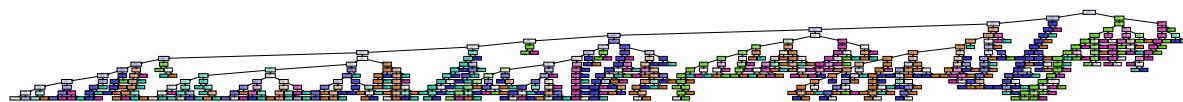
In [15]:

```
Decision tree model saved to bamboo_train_tree!
```

Let's plot the decision tree and explore it.

In [16]:

Out[16]:



Now that we defined our tree to be deeper, more decision nodes are generated.

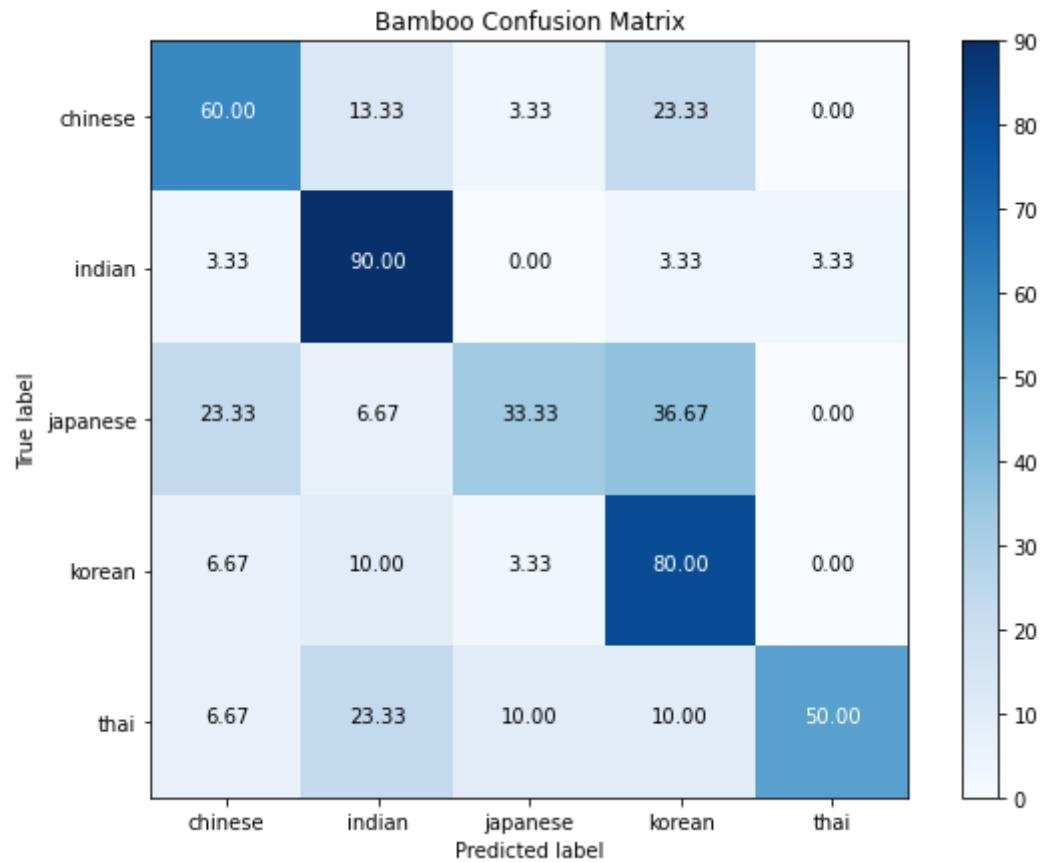
Now let's test our model on the test data.

In [17]:

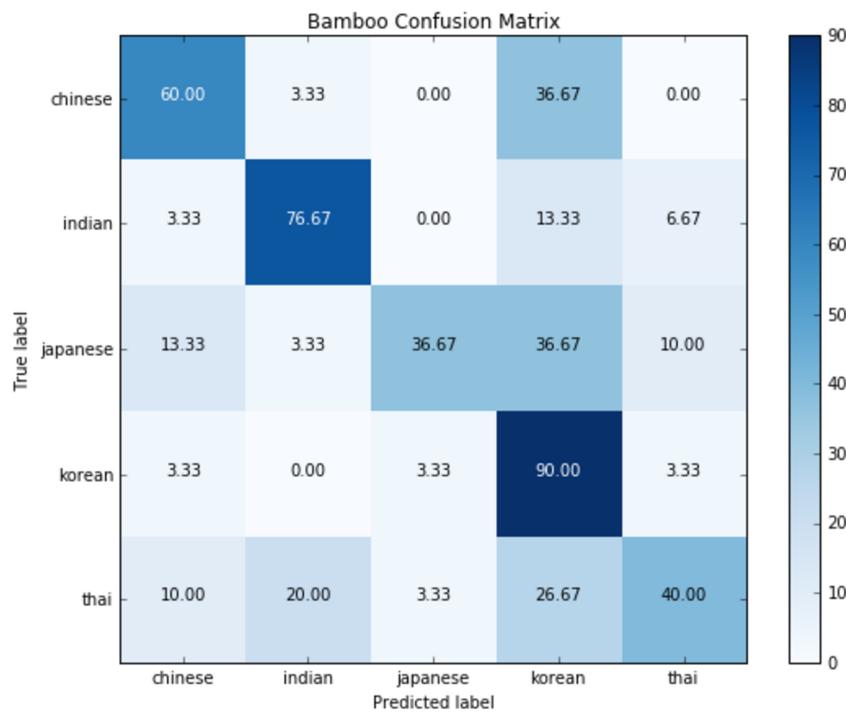
To quantify how well the decision tree is able to determine the cuisine of each recipe correctly, we will create a confusion matrix which presents a nice summary on how many recipes from each cuisine are correctly classified. It also sheds some light on what cuisines are being confused with what other cuisines.

So let's go ahead and create the confusion matrix for how well the decision tree is able to correctly classify the recipes in **bamboo\_test**.

In [18]:



After running the above code, you should get a confusion matrix similar to the following:



The rows represent the actual cuisines from the dataset and the columns represent the predicted ones. Each row should sum to 100%. According to this confusion matrix, we make the following observations:

- Using the first row in the confusion matrix, 60% of the **Chinese** recipes in **bamboo\_test** were correctly classified by our decision tree whereas 37% of the **Chinese** recipes were misclassified as **Korean** and 3% were misclassified as **Indian**.
- Using the Indian row, 77% of the **Indian** recipes in **bamboo\_test** were correctly classified by our decision tree and 3% of the **Indian** recipes were misclassified as **Chinese** and 13% were misclassified as **Korean** and 7% were misclassified as **Thai**.

**Please note** that because decision trees are created using random sampling of the datapoints in the training set, then you may not get the same results every time you create the decision tree even using the same training set. The performance should still be comparable though! So don't worry if you get slightly different numbers in your confusion matrix than the ones shown above.

Using the reference confusion matrix, how many **Japanese** recipes were correctly classified by our decision tree?

Your Answer:36.67%

Double-click **here** for the solution.

Also using the reference confusion matrix, how many **Korean** recipes were misclassified as **Japanese**?

Your Answer:3.33%

Double-click **here** for the solution.

What cuisine has the least number of recipes correctly classified by the decision tree using the reference confusion matrix?

Your Answer:36.67

Double-click **here** for the solution.

---

## Thank you for completing this lab!

This notebook was created by [Alex Aklson](https://www.linkedin.com/in/aklson/) (<https://www.linkedin.com/in/aklson/>). We hope you found this lab session interesting. Feel free to contact us if you have any questions!

This notebook is part of a course on **Coursera** called *Data Science Methodology*. If you accessed this notebook outside the course, you can take this course, online by clicking [here](https://cocl.us/DS0103EN_Coursera_LAB4) ([https://cocl.us/DS0103EN\\_Coursera\\_LAB4](https://cocl.us/DS0103EN_Coursera_LAB4)).

---

Copyright © 2019 [Cognitive Class](https://cognitiveclass.ai/?utm_source=bducopyrightlink&utm_medium=dswb&utm_campaign=bdu) ([https://cognitiveclass.ai/?utm\\_source=bducopyrightlink&utm\\_medium=dswb&utm\\_campaign=bdu](https://cognitiveclass.ai/?utm_source=bducopyrightlink&utm_medium=dswb&utm_campaign=bdu)). This notebook and its source code are released under the terms of the [MIT License](https://bigdatauniversity.com/mit-license/) (<https://bigdatauniversity.com/mit-license/>).