Students for Machine Learning in Business

Challenge 1

- Sell more cars! -

Congratulations on deciding to become a tech entrepreneur and starting your first consulting business! After doing some extensive market research in Alberta, you have found that the hottest up and coming industry right now is automotive sales! Thinking long and hard about how you can get your piece of the pie, you have come up with your big idea:

You will create a machine learning model
that will analyze people based on demographics,
and it will predict which car they want to buy!

Perfect! You can sell your model to car dealerships so that they can improve their marketing, know what cars to stock, and increase sales efficiency! You have determined that the most important variables when predicting the vehicle preference of a person are:

- **If their daily commute is under 5 miles**
- **If their household contains over 2 people**
- **If they have a post secondary education**

You will create your model in python to predict which vehicle a person will choose given vehicles with different features based on these demographic factors.

This challenge will run from **January 28, 2021, to February 8, 2021**. Your model must be submitted by the ending date to count. The submission details are outlined below in this document. If you're brand new to coding don't feel overwhelmed! Feel free to ask any questions in the SMLB Slack (and Discord coming soon) as well!

Please use the Challenge 1 Guide document for a walkthrough and tips!

Now it's time to get to work before competitors appear!

1st place: $50 Amazon Gift Card
2nd place: $30 Amazon Gift Card

Challenge Rules:

- ❏ Work in a group of up to 3 people or individually. Remember, collaborating shares the load, but also shares the rewards.
- ❏ Code must written in Python (modules/packages found on www.pypi.org are permitted)
- ❏ Must be submitted by end of the day on the submission deadline: **Feb 8, 2021**
- ❏ Code must be submitted to smlbualberta@gmail.com as covered in the Submission Details section
- ❏ Code must output in the correct format as covered in the Submission Details section
- ❏ Code will be marked based on number of correct predictions using the test data set (unreleased)

SMLB is meant to be a collaborative effort between individuals hoping to learn more about machine learning and its applications in the business world, as such, we intend to post examples of winning models on our website. As such, any code submitted for the SMLB Challenge 1 must be under an MIT License, which is described as 'A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications and larger works may be distributed under different terms and without source code.' by choosealicense.com.

Challenge 1 Guide

DATA

(Files linked)

The first step is to download and understand the data.
You can open the data in excel or a similar spreadsheet program.

- Education, Household, and Commute are the demographic factors
- The factors below are the vehicle features of each choice

Data Legend

Id:		The individuals numbered
choice:	(What your model is predicting!!!)	The vehicle they chose given 6 choices {choice1,choice2...choice6}

education:	Does the individual have a college education? 1 = Yes, 0 = No
household:	Is the individual's household size greater than 2? 1 = Yes, 0 = No
commute:	Is the individual's commute over 5 miles per day? 1= Yes, 0 = No
type(1-6):	The type of vehicle {van, regcar, truck, etc)
fuel(1-6):	The type of fuel (gasoline, electric, etc.)
price(1-6):	Price of the vehicle divided by the logarithm of the individual's income
range(1-6):	The number of miles the respective vehicle can travel before recharging/refueling
acc(1-6):	The acceleration of the vehicle in seconds from 0 - 30 mph
speed(1-6):	The highest attainable speed in mph of the vehicle
pollution(1-6):	The pollution emitted from the vehicle's tailpipe in comparison to a new gas vehicle (external benchmark not provided)
size(1-6):	Size of the vehicle. 0= Mini, 1= Subcompact, 2= compact, 3 = SUV or larger
space(1-6):	Fraction of vehicles luggage space in comparable new gas vehicle (external benchmark not provided)
cost(1-6):	Cost for a mile of travel in cents for the respective vehicle
station(1-6):	The fraction of stations in the country which can charge/refuel the respective vehicle

Submission Details

To submit your algorithm, send a zipped folder with all relevant codes and files to smlbualberta@gmail.com. In the main directory of the folder, please attach a README.txt file containing all of the instructions for somebody on a different machine to run your algorithm. Please be **very** clear on everything that is needed to run the code and in which order. Include a list of the python packages and their links on https://pypi.org/ with the version number that is required to run the submission. The organizers may contact you for clarification after submission. Please keep in mind that any edits to the source code will not be permitted after the deadline, so ensure that the code will run on a machine other than your own before submitting. Send yourself the zipped folder on a different machine and walk through the process of running it on the training set so that you are sure the organizers will be able to do so as well.

IMPORTANT - Code must output a table in this format: (feel free to use this template exactly)

```
my_predictions = None
name_of_file = 'TrainingData.csv'
name_of_test_file = 'We will modify this in your code. Just make sure
that when we rename this file, the code works and modifies the
"my_predictions" variable'
# This is where all your code runs
# After your code runs, we want the variable 'my_predictions' (or
whatever you've named it) to contain your predictions!
# You can use whatever method you like (eg. For loop to append
predictions, or converting some other DataFrame to match the desired
format!)


my_predictions = [['id1', 'choiceX'],
                  ['id2', 'choiceX'],
                  ...
                  ['idN', 'choiceX']]

'choiceX' is the prediction, where X can be a number from 1-6.
If you wish, you can put a number which corresponds to the choice
instead! See below:

my_predictions = [['id1', X],
                  ['id2', X],
                  ...
                  ['idN', X]]

Here, X is a number from 1 to 6. Eg.

['id1', 'choice3'] will be considered equivilant to ['id1',3]

Just remain consistent accross your results!
```

Instructions on how to set up this format are in the Challenge 1 Guide.

How will your code be marked?

The marking scheme is simple; we have some (test) data in the same format as the training data provided to you. This data does not include the *choice* column. We will run this test data through your code, and in return, will obtain the required output.

```
1  my_predictions = None
2  name_of_file = 'TrainingData.csv'
3  name_of_test_file = 'We will modify this in your code. Just make sure that when we rename
   this file, the code works and modifies the "my_predictions" variable'
4  # This is where all your code runs
5  # After your code runs, we want the variable 'my_predictions' (or whatever you've named it)
    to contain your predictions!
6  # You can use whatever method you like (eg. For loop to append predictions, or converting
   some other DataFrame to match the desired format!)
7
```

In your program, be sure to include some variables which we can change to test your model. In this example, we would modify line 3 and expect 'my_predictions' to hold your algorithm's predictions.

Your algorithm will be marked based on the percentage of correctly guessed choices for individuals.

Teams will be provided with a training dataset to train their algorithm so that when given the test set, a prediction will be made about their car preferences. Teams are welcome to utilize any python package found on https://pypi.org/ for use during their project, but since each submission will be run on a fresh virtual environment, they must provide a .txt file specifying all python packages which are used.

Additional Tips:

The process of developing algorithms often requires much tinkering and bug fixing. Regardless of your skill level, it is recommended that you try and get a working prototype before you attempt to get too ambitious. Get something that works before you make something that is perfect!

For Beginner Programmers:

- ❏ A detailed walk-through is provided for complete beginners who wish to receive instructions on how to progress!
- ❏ Install libraries such as Pandas and Scikit-Learn libraries which are crucial to machine learning.
- ❏ What are Decision Trees? https://youtu.be/DCZ3tsQIoGU (8:45)
- ❏ Before moving ahead with your split your data into 2 parts: Training and testing! You can learn this here https://youtu.be/fwY9Qv96DJY (Watch up to 3:43, the rest is irrelevant!)
- ❏ Check out Scikit-Learn's "Tree" module. See what methods it has to offer
    - ❏ Examine the various hyperparameters which you can fine-tune
- ❏ Final Tip: Do all the factors such as (Type, Fuel, Acc, etc.) necessarily make our predictions more accurate?
    - ❏ Some factors may in fact reduce the model's accuracy due to their lack of impact/correlation on the choice!

For Experienced Programmers:

- ❏ Look at the data closely! For instance, is 'choice1' referring to the same vehicle across all the individuals?
- ❏ How can you work-around this issue?

Good Luck! - SMLB Team