DATA-51100: Statistical Programming
Programming Assignment 4 – Estimating Probabilities

Introduction

Probability is a number that indicates the likelihood of some outcome occurring, where each outcome comes from a set called the *sample space*, denoted by Ω. Probabilities are used in situations where there is uncertainty in data, either due to a lack of sufficient data or some inherent randomness associated with the data. Formally, probability of each outcome x is a value, $p(x)$, that satisfies the following properties:

1. $\forall x \in \Omega\ (p(x) \in [0,1])$ (each probability value has to be between zero and one)

and

2. $\sum_{x \in \Omega} p(x) = 1$ (sum of all probabilities needs to be one)

A set of outcomes defines an *event*. The probability of an event E is defined as

$$P(E) = \sum_{x \in E} p(x)$$

In many applications, it is necessary to estimate probabilities from data. If the data contains nominal (i.e. categorical) values, we can estimate the probability of a particular value occurring in the data by counting the number of instances in which the value occurs. In particular, assume the data consists of *N* instances, which is associated with a fixed number of feature values. Then the probability of a particular feature i having a particular value x can be computed as

$$P(feature_i = x) = \frac{\#(instances\ with\ feature_i = x)}{N}$$

We can also compute the *conditional probability* of a particular feature value, given some other features values as

$$P\big(feature_i = x | feature_j = f\big) = \frac{\#(instances\ with\ feature_i = x\ and\ feature_j = f)}{\#(feature_j = f)}$$

Note that the denominator is assumed to be non-zero. Such estimates can then be used for various data analysis applications, such as modeling or machine learning.

Requirements

You are to create a program in Python that performs the following:

1. Loads the 'cars.csv' file into a pandas DataFrame.
2. For each aspiration type a, computes the conditional probability of that aspiration, given each of the makes: $P(aspiration = a | model = m)$
3. Displays the conditional probabilities to the screen.
4. Computes the probability of each make and outputs to the screen.

Additional Requirements

1. The name of your source code file should be `ProbEst.py`. All your code should be within a single file.
2. You cannot import any package except for **pandas**. You need to use the pandas DataFrame object for storing data. You cannot use the groupby function!
3. Your code should follow good coding practices, including good use of whitespace and use of both inline and block comments.
4. You need to use meaningful identifier names that conform to standard naming conventions.
5. At the top of each file, you need to put in a block comment with the following information: your name, date, course name, semester, and assignment name.
6. The output of your program should **exactly** match the sample program output given at the end.

What to Turn In

You will turn in the **single ProbEst.py file** as well as **a screenshot of your output(s)** using BlackBoard.

Sample Program Output

```
DATA-51100, [semester] [year]
NAME: [put your name here]
PROGRAMMING ASSIGNMENT #4

Prob(aspiration=std|make=alfa-romero) = 100.00%
Prob(aspiration=turbo|make=alfa-romero) = 0.00%
Prob(aspiration=std|make=audi) = 71.43%
Prob(aspiration=turbo|make=audi) = 28.57%
Prob(aspiration=std|make=bmw) = 100.00%
Prob(aspiration=turbo|make=bmw) = 0.00%
Prob(aspiration=std|make=chevrolet) = 100.00%
Prob(aspiration=turbo|make=chevrolet) = 0.00%
Prob(aspiration=std|make=dodge) = 66.67%
Prob(aspiration=turbo|make=dodge) = 33.33%
Prob(aspiration=std|make=honda) = 100.00%
Prob(aspiration=turbo|make=honda) = 0.00%
Prob(aspiration=std|make=isuzu) = 100.00%
Prob(aspiration=turbo|make=isuzu) = 0.00%
Prob(aspiration=std|make=jaguar) = 100.00%
Prob(aspiration=turbo|make=jaguar) = 0.00%
Prob(aspiration=std|make=mazda) = 100.00%
Prob(aspiration=turbo|make=mazda) = 0.00%
Prob(aspiration=std|make=mercedes-benz) = 50.00%
Prob(aspiration=turbo|make=mercedes-benz) = 50.00%
Prob(aspiration=std|make=mercury) = 0.00%
Prob(aspiration=turbo|make=mercury) = 100.00%
Prob(aspiration=std|make=mitsubishi) = 53.85%
Prob(aspiration=turbo|make=mitsubishi) = 46.15%
Prob(aspiration=std|make=nissan) = 94.44%
Prob(aspiration=turbo|make=nissan) = 5.56%
Prob(aspiration=std|make=peugot) = 45.45%
Prob(aspiration=turbo|make=peugot) = 54.55%
Prob(aspiration=std|make=plymouth) = 71.43%
Prob(aspiration=turbo|make=plymouth) = 28.57%
Prob(aspiration=std|make=porsche) = 100.00%
Prob(aspiration=turbo|make=porsche) = 0.00%
Prob(aspiration=std|make=renault) = 100.00%
Prob(aspiration=turbo|make=renault) = 0.00%
Prob(aspiration=std|make=saab) = 66.67%
Prob(aspiration=turbo|make=saab) = 33.33%
Prob(aspiration=std|make=subaru) = 83.33%
Prob(aspiration=turbo|make=subaru) = 16.67%
Prob(aspiration=std|make=toyota) = 96.88%
Prob(aspiration=turbo|make=toyota) = 3.12%
Prob(aspiration=std|make=volkswagen) = 83.33%
Prob(aspiration=turbo|make=volkswagen) = 16.67%
Prob(aspiration=std|make=volvo) = 54.55%
Prob(aspiration=turbo|make=volvo) = 45.45%

Prob(make=alfa-romero) = 1.46%
Prob(make=audi) = 3.41%
Prob(make=bmw) = 3.90%
Prob(make=chevrolet) = 1.46%
Prob(make=dodge) = 4.39%
```

```
Prob(make=honda) = 6.34%
Prob(make=isuzu) = 1.95%
Prob(make=jaguar) = 1.46%
Prob(make=mazda) = 8.29%
Prob(make=mercedes-benz) = 3.90%
Prob(make=mercury) = 0.49%
Prob(make=mitsubishi) = 6.34%
Prob(make=nissan) = 8.78%
Prob(make=peugot) = 5.37%
Prob(make=plymouth) = 3.41%
Prob(make=porsche) = 2.44%
Prob(make=renault) = 0.98%
Prob(make=saab) = 2.93%
Prob(make=subaru) = 5.85%
Prob(make=toyota) = 15.61%
Prob(make=volkswagen) = 5.85%
Prob(make=volvo) = 5.37%
```