

Pseudo Code

```
import data.csv # import testing data
```

```
# preprocess data, data cleaning feature engineering
```

```
drop Nan # drop any null values from the data set
```

```
drop column # drop any unnecessary columns (key, pick-up datetime and passenger count)
```

```
.labelEncoder(), one-hot # methods to convert String type to 0s 1s (ints) for machine readability
```

```
# Find Euclidean Distance
```

```
def euc_distance(...) # This function calculates distance between pickup and drop-off location  
    return ...
```

```
df['distance'] = euc_distance(df['pickup_latitude'])
```

```
# calculate normal standardization of the euc-distance to determine the mean distance (pickup_longitude, latitude, dropoff_longitude & latitude)
```

```
# Don't remember any lines of code for normal standardization
```

```
# Machine must calculate the standardization for whichever column has variety of ints for the machine to understand the values. Translating the data into "machine language" this allows us to program whatever it is we want to do with the set of data.
```

```
# Testing Data
```

```
relu = (8, -, -) # Input
```

```
relu = (10, -, -)
```

```
output = (1, -, -)
```

```
# With this line of code, the program identifies how many input, hidden, and output nodes there are. Most common function to use is "relu" and another separate function (tanh, etc.). Output layer has a different function.
```

Unsure as to where this part of the code goes perhaps after normal standardization?

calculate taxes paid for the amount traveled using rideshare services

The program should be able to predict the taxes paid when traveling a certain distance

Training Data: (forgot majority of codes but I know the steps needed for a successful Deep Learning process)

history = { ... }

usually use the epoch method to run the tests and the program should give us the "losses" and "accuracy" results.

Method #2 (different functions)

The 2nd method will implement a different function to get different results.

using ^{different} activation optimizer and loss function. This will affect the loss % as well as accuracy %.