**STEPS FOR SUPERVISED LEARNING**

1. **Data Understanding**
   1. Find out if the dependent variable / response var / predicted var / y is continuous (regression algos) / binary categorical( binary classification algos) or multi-class categorical (multi-class classification algo)
   2. Visual Inspection
   3. Datatypes (info, describe, dtypes, head, tail)
   4. Making sure that my data is structured data: Locate the rows and cols – we need to make sure that observations are present in the rows and variables in the columns.
   5. Freq. counts (using table function / using group-by commands) – for factor variables
   6. Aggregate data according to factor variables to understand the bi-variate relationships
   7. Summary stats (using summary fn) – primarily for numerical variables – distribution is normal, left skewed / right skewed / outliers
   8. Subsets
   9. Data Visualization
      * Visual dashboards
      * Reports
2. **Data Pre-processing** 
   1. Data Cleaning – Converting unstructured / semi-structured data to structured format
   2. Missing Values Imputations / Deletion
   3. Data entry errors like –ve values in a column – can be treated as null values and handled in the same way.
   4. Outliers Imputation / deletion
   5. Feature Engineering:
      * Add / Delete columns (Create as many variables as possible which intuitively explain the variance in the response variable: Speed / rate / Total taxi time = taxi in + taxi out, title (extracting from the name variable), family size;
      * Delete the useless variable (which are intuitive as not useful / redundant for the prediction) – to handle the overfitting
   6. Handling duplicate rows / columns
   7. Data Transformations – taking a log / exponent incase the column distribution is right /left skewed.
   8. Taking square-root / cube-roots also sometimes help when the right skewness is not extreme
   9. Taking squares / cubes when left skewness is not extreme
   10. Delete all the character columns, the only columns that should be there with you should be either factor columns or numerical columns
   11. Factor columns also, have to be converted numerical (either numpy array or pandas df) mandatorily.
3. **Predictive Modelling**
   1. Divide the entire data into training and test / validation data in the ratio 70:30 or 80:20
   2. Train the model on training data
   3. Test our model performance on the test data / validation data
   4. Check the over-fitting using cross-validations
4. **Deploy the Model on the unseen data for future predictions of response variable.**