Hide

Comments (–) Share Hide Toolbars

```
combi$Item_MRP_clusters = as.factor(Item_MRP_clusters$cluster)
```

Encoding Categorical Variables

In this stage, we will convert our categorical variables into numerical ones. We will use 2 techniques — Label Encoding and One Hot Encoding.

1. **Label encoding** simply means converting each category in a variable to a number. It is more suitable for ordinal variables — categorical variables with some order.
2. In **One hot encoding**, each category of a categorical variable is converted into a new bunary column (1/0).

We will use both the encoding techniques.

Label encoding for the categorical variables

We will label encode Outlet_Size and Outlet_Location_Type as these are ordinal variables.

Hide

```
combi[,Outlet_Size_num := ifelse(Outlet_Size == "Small", 0,
                                 ifelse(Outlet_Size == "Medium", 1, 2))]
combi[,Outlet_Location_Type_num := ifelse(Outlet_Location_Type == "Tier 3", 0,
                                          ifelse(Outlet_Location_Type == "Tier 2", 1, 2))]
# removing categorical variables after label encoding
combi[, c("Outlet_Size", "Outlet_Location_Type") := NULL]
```

One hot encoding for the categorical variable

Hide

```
ohe = dummyVars("~.", data = combi[,-c("Item_Identifier", "Outlet_Establishment_Year", "Item_Typ
e")], fullRank = T)
ohe_df = data.table(predict(ohe, combi[,-c("Item_Identifier", "Outlet_Establishment_Year", "Item
_Type")]))
combi = cbind(combi[,"Item_Identifier"], ohe_df)
```

PreProcessing Data

Before feeding our data into any model, it is a good practice to preprocess the data. We will do preprocessing on both independent variables and target variable

Checking Skewness

Skewness in variables is undesirable for predictive modeling. Some machine learning methods assume normally distributed data and a skewed variable can be transformed by taking its log, square root, or cube root so as to