# **WEIGHT & DIMENTIONS UPDATE PROCESS IN EVP**

**Objective**: Predicting, based on historical shipping information, the weight, height, length, and width to improve shipping cost estimation, positively impacting business profitability. Once the predictions are obtained, this data will be uploaded to the EVP system weekly for timely updates.

**Context:** There is an issue with the weight and dimensions entered in the EVP system, which may differ from reality or be incorrect due to operational errors on the part of the supplier. This information is used to estimate shipping costs, so incorrect dimensions directly impact e-commerce operating costs.

## **Data Sources:**

The process extracts information from the following sources:

1. **FEDEXHISTORICO:** Contains billing information for packages sent by FEDEX.
2. **UPSNEWFORMAT:** Contains billing information for packages shipped by UPS.
3. **Endiciatransacprovision:** Contains billing information for packages shipped by Endicia.
4. **Box:** Contains historical information about Box transactions.
5. **DWShippingInfo:** Contains shipment details, including SKUs, unit of measure (UOM) codes, and sales dates.
6. **stg-MaxWarehouse.EvpCsv.EVPAlllSKUsDimNWeight:** Contains standard weight and dimension data for the SKUs.

## **Description of CTEs:**

Several Common Table Expressions (CTEs) are created to process and transform the data.

### **BoxFechas**

Extracts dimensions from the DIMENSIONS field in the BOX table, filtering records where OrderDate is a valid date

### **Factura3meses**

Extracts the weights and dimensions of packages sent in the last 3 months, grouping the data by tracking number:

* **FEDEX:** ActualWeightAmount is used as the maximum weight.
* **UPS:** BilledWeight is used as the maximum weight. The dimensions are extracted from the DetailKeyedDim column, separating them into height, length and width.
* **Endicia:** Weight\_(lb) is used as the maximum weight.

### **DWVentasUnicas**

Obtain unique sales in the last 4 months, identified by SalesOrderNumber and Sku.

### **CantidadSkusxVenta**

Filter sales orders with only one SKU.

* DWVentasUnicas is grouped by SalesOrderNumber.
* Those orders where the number of SKUs is 1 are filtered.

### **PesosfacturaVentas**

* It is joined with CantidadSkusxVenta to consider sales of only one product of the same SKU.

### **RankedData**

Obtains shipment data (tracking, SKU, units of measure, weight, dimensions) and assigns a ranking based on the invoice date (InvDate).

### **Max\_Rank**

Takes data from RankedData and groups by transaction and SKU, obtaining the maximum weight and dimensions.

### **EJD\_DIMS**

EJD supplier dimension and weight columns are added.

## **Business Rules:**

* Only transactions from the last 3 months are considered
* The dimensions of the Box are considered in case they are empty on the invoice databases
* Records with weight = 0 are excluded. Dimensions are validated to avoid erroneous values like ​​Dims != '0X0X0'
* Sales with a single SKU and a single unit are filtered
* The EJD dimensions will be used as a threshold to calculate a range as follows:
  + If the SKU's EJD weight is less than 1 pound:
    - Weight threshold: 5 ounces
    - Threshold for dimensions: 3 inches
  + If the SKU's EJD weight is greater than 1 pound:
    - Weight threshold: 3 pounds
    - Threshold for dimensions: 3 inches

## **Data Cleaning:**

In this phase, several transformations are performed to ensure data quality, such as:

* **UomCode transformations:** converting to uppercase, removing spaces, replacing variations such as EACH to EA, and standardizing unit formats such as PACK to PK.
* **BaseUnit field transformations:** empty fields were filled according to the UomCode, to ensure information consistency.
* **Data type conversion:** dimension values ​​are transformed to float type, dates to date.
* **Sort dimensions:** sort the Height, length, width dimensions from largest to smallest.
* **Outlier detection:** The interquartile range method is used to eliminate outliers. These data are replaced by the median of the SKU-UomCode.

## **Relevant Features for Prediction**

### Variable Selection:

* The Sku column is used as a unique identifier for each product.
* The Weight, Height, Length, and Width dimensions are considered as the variables to be predicted.
* The UomCode variable is included to allow differentiation by unit of measurement.

## **Prediction Model**

Two approaches were implemented to predict weight and dimension values ​​based on the number of records of each Sku-UomCode pair:

* **Machine Learning Approach for SKUs-UomCode with Many Records:**

A machine learning model (Random Forest) was trained for SKUs with more than five historical records. This model uses all available variables to predict product dimensions (weight, height, length, and width).

* **Median Approach for SKUs-UomCode with Few Records:**

For SKUs with fewer than 5 records, the median of historical data was used as a reference value for predictions, since lack of data does not allow for effective training of more complex models.

* **Division of the Machine Learning Model into Two Subgroups:**

The machine learning model was split into two additional approaches:

1. A model for SKUs with at least one record weighing less than 1 pound. This submodel focused on capturing the characteristics of lighter SKUs.
2. Another model for SKUs that consistently weigh more than 1 pound. This improved the model's sensitivity, especially for heavier products, allowing for more accurate predictions of weight and dimension values.

## **Prediction Pipeline**

## Process Automation

* **Data Extraction:** Data is extracted using SQL queries in Python, connecting to SQL Server. Extraction is automated to ensure the most recent data is used in predictions.
* **Data Preprocessing and Cleaning:** A data cleansing process is performed, identifying and removing erroneous or inconsistent records. In addition, columns relevant to predictions are transformed and standardized, such as converting dates to appropriate formats and normalizing dimensions and weight data.
* **Model Training:** The model is trained on preprocessed data. A cross-validation approach is applied to ensure model robustness.
* **Prediction Generation:** The model predicts the product dimensions and weight for the upcoming week. These predictions are automatically generated and exported to a CSV file for further analysis and visualization.
* **Output:** Four CSV files are generated with the results of the predictions and metrics, which will be used in different stages of the process:
  + **df\_evp:** This file is used to update the weights and dimensions in the EVP database. The data will be uploaded via FTP.
  + **df\_historico:** Contains historical data from previous predictions, used to keep track of past measurements.
  + **df\_metrica:** Contains metrics for SKUs that are outside the range of the EJD supplier's dimensions.
  + **df\_predicciones:** Contains the current predictions generated by the model, which will be consumed by the Shiny application for visualization and interactive analysis.

## **System Update and Maintenance**

## Weekly Automation

* The process runs automatically every week, specifically on **Tuesdays at 10am**, using the Windows Task Scheduler.
* The only manual part of the process is uploading the dimensions to EVP via FTP.

## **Challenges and Limitations**

## Technical challenges

* SKUs with fewer records represent a challenge due to the lack of sufficient historical data to train a reliable model.
* Handling missing or erroneous data, especially in dimensions and weights, is crucial to avoid bias in predictions.
* Data inconsistency results in a lengthy cleaning and transformation process.

## **Expected Results and Benefits**

* **Shipping Cost Optimization:** The main advantage of this model is optimizing shipping costs by more accurately predicting product weight and dimensions.
* **Improved Decision-Making:** Predictions allow the company to make informed decisions about packaging and transportation strategies, improving operational efficiency and detecting potential operational or data errors from the EJD supplier.

## **Pipeline Flowchart**

[Weights & Dims Process.vsdx](https://bitsmaxwarehouse-my.sharepoint.com/:u:/g/personal/ecalderon_maxwarehouse_com/EXGPtB0zLn1Fr-ChCDoNaR0B2VidYGQnHe1WC1mug5wFxA?e=ZglF6q)