# Hospital SKU Forecasting — Technical Summary

## Overview

This document summarizes three types of forecasting details. Each set includes model choices, required raw columns, feature engineering, hyperparameters, target/output variables, and performance (MAPE/other metrics).

## Set 1 — All SKUs stock prediction (XGBoost best)

### 1) Why these models (and not others)?

- Tabular snapshot (no per-SKU time sequence) favors tree ensembles (RandomForest, XGBoost) for non-linear effects and mixed dtypes.
- Linear Regression serves as a baseline; its high MAPE indicates non-linearity.
- No time-series models here because the design predicts a single snapshot value using engineered features from two dates.

### 2) Required raw columns

- inventory_id
- hospital_id
- sku_id
- vendor_id
- current_stock
- reorder_level
- max_stock_level
- last_order_date
- expiry_date
- batch_number

### 3) Feature engineering performed

- Date parts from last_order_date & expiry_date: day, month, year, dayofweek
- days_to_expiry = expiry_date - last_order_date (in days)
- Ratios/flags: stock_utilization, reorder_ratio, stock_above_reorder
- Label encodings: sku_id, vendor_id, batch_number, hospital_id (→ *_encoded)
- Dropped identifier/date columns for modeling

## 4) Hyperparameters / tuning

| Field | Value |
|---|---|
| **RandomForest** | n_estimators=100, max_depth=10, min_samples_split=5, min_samples_leaf=2 |
| **XGBoost** | n_estimators=100, max_depth=6, learning_rate=0.1 |
| **LinearRegression** | defaults; no tuning |
| **Search** | None (fixed reasonable settings) |

## 5) Output variable & performance

| Field | Value |
|---|---|
| **Target (output)** | current_stock |
| **Best model (validation)** | xgboost (MAPE 6.12%) |
| **Test MAPE** | 5.73 |
| **Test MAE** | 2.77 |
| **Test RMSE** | 4.09 |

Top features (XGBoost importance):

| Feature | Importance |
|---|---|
| **stock_utilization** | 0.5626 |
| **max_stock_level** | 0.3335 |
| **reorder_level** | 0.0263 |
| **stock_above_reorder** | 0.0173 |
| **reorder_ratio** | 0.0123 |
| **expiry_year** | 0.0104 |
| **sku_id_encoded** | 0.0078 |
| **last_order_dayofweek** | 0.0061 |
| **batch_number_encoded** | 0.006 |
| **last_order_day** | 0.0053 |

## Set 2 — Clustered SKUs dataset, 16 models compared (LightGBM best)

### 1) Why these models?

- Same tabular framing as Set 1 but with a movement_cluster feature; gradient-boosting family (LightGBM/XGBoost/GBR/ExtraTrees) excels on mixed engineered features.
- Linear, KNN, and simple MLP underperformed due to bias/variance and sensitivity to scaling/sparsity.
- Scaling applied where appropriate (SVR, KNN, MLP, Ridge/Lasso/ElasticNet).

### 2) Required raw columns

- inventory_id
- hospital_id
- sku_id
- vendor_id
- current_stock
- reorder_level
- max_stock_level
- last_order_date
- expiry_date
- batch_number
- movement_cluster

### 3) Feature engineering performed

- Date parts from last_order_date & expiry_date: day, month, year, dayofweek
- days_to_expiry = expiry_date - last_order_date (in days)
- Ratios/flags: stock_utilization, reorder_ratio, stock_above_reorder
- Label encodings: sku_id, vendor_id, batch_number, hospital_id (→ *_encoded)
- Dropped identifier/date columns for modeling
- movement_cluster (label-encoded)
- Kept original IDs and dates to build an Actual_vs_Predicted sheet

### 4) Hyperparameters / tuning

- Fixed, sensible defaults per model; no grid search in the shared code
- Scaled features for SVR, KNN, MLP, and linear-regularized models

## Validation MAPE across models

| Model | Validation MAPE (%) |
|---|---|
| lightgbm | 5.81 |
| gradient_boosting | 5.84 |
| xgboost | 6.00 |
| extra_trees | 6.64 |
| bagging | 7.90 |
| random_forest | 7.97 |
| catboost | 8.64 |
| decision_tree | 11.60 |
| svr | 18.66 |
| ada_boost | 29.27 |
| ridge | 50.08 |
| linear_regression | 50.08 |
| lasso | 52.90 |
| neural_network | 54.48 |
| knn | 57.17 |
| elastic_net | 57.40 |

## 5) Output variable & performance

| Field | Value |
|---|---|
| Target (output) | current_stock |
| Best model (validation) | lightgbm (MAPE 5.81%) |
| Test MAPE | 5.7117 |
| Test MAE | 2.7931 |
| Test RMSE | 4.0518 |
| Test $R^2$ | 0.9748 |

Top features (LightGBM importance counts):

| Feature | Importance |
|---|---|
| stock_utilization | 525 |
| max_stock_level | 516 |
| reorder_level | 127 |
| reorder_ratio | 58 |
| sku_id_encoded | 53 |
| days_to_expiry | 46 |
| batch_number_encoded | 46 |
| expiry_day | 40 |
| last_order_day | 30 |
| vendor_id_encoded | 27 |
| expiry_month | 20 |
| last_order_dayofweek | 16 |
| last_order_month | 0 |
| expiry_year | 0 |
| last_order_year | 0 |

## Set 3 — Time-series consumption by SKU (Prophet / ES / XGB / LGBM)

### 1) Why these models?

- This set leverages daily transaction history per SKU, so time-series models (Prophet, Exponential Smoothing) are appropriate for trend/seasonality.
- Tree models (XGBoost/LightGBM) on lag/rolling/time features capture non-linearities and zero-inflation with good speed and scale.
- Category-specific settings (fast/medium/slow moving) balance bias/variance without heavy grid search.

### 2) Required raw columns

- sku_id
- transaction_date
- quantity_consumed
- total_cost
- patient_id

### 3) Feature engineering performed

- Daily aggregation: daily_consumption, total_cost, patient_count
- Movement scoring & category per SKU (fast/medium/slow) via consumption rate/frequency/mean percentiles
- Time parts & cyclic encodings: month, day_of_week, day_of_year, quarter, is_weekend, is_month_end, month_sin/cos, dow_sin/cos
- ML features: lags (1,2,3,7,14,28), rolling means/std (7/14/28), trend_1d, trend_7d, consumption_binary, movement_score
- Date-based train/val/test split; horizons: 1, 7, 30, 90 days; category-specific post-processing to reduce MAPE spikes

### 4) Hyperparameters / tuning

- Prophet: changepoint/seasonality priors, daily/weekly/yearly seasonality toggles, mode and range vary by category
- XGBoost/LightGBM: estimators, depth, learning_rate, subsample/colsample, regularization, min_child_* / num_leaves tuned per category with early stopping
- Exponential Smoothing: trend/seasonal/damped settings vary by category
- Improved MAPE computation (sMAPE, adjusted MAPE, weighted MAPE) – lowest reasonable value selected

### 5) Output variable & performance

- Target (output): daily_consumption per SKU
- Forecast horizons: 1, 7, 30, 90 days
- MAPE is computed per-SKU & per-horizon; the code exports CSVs (performance summary and best models)

## What the accompanying sheet contains (inputs, outputs, MAPE)

A companion Excel workbook aggregates the essentials. For each set  with engineered input variables used, the target (output) variable, validation MAPE per model and test metrics for the best model.

| Field | Value |
|---|---|
| **Set 1 — Inputs** | last_order_day, last_order_month, last_order_year, last_order_dayofweek, expiry_day, expiry_month, expiry_year, days_to_expiry, stock_utilization, reorder_ratio, stock_above_reorder, sku_id_encoded, vendor_id_encoded, batch_number_encoded, hospital_id_encoded |
| **Set 1 — Output** | current_stock |
| **Set 1 — Best MAPE (Val.)** | 6.12% |
| **Set 2 — Inputs** | last_order_day, last_order_month, last_order_year, last_order_dayofweek, expiry_day, expiry_month, expiry_year, days_to_expiry, stock_utilization, reorder_ratio, stock_above_reorder, sku_id_encoded, vendor_id_encoded, batch_number_encoded, hospital_id_encoded, movement_cluster_encoded |
| **Set 2 — Output** | current_stock |
| **Set 2 — Best MAPE (Val.)** | 5.81% |
| **Set 3 — Inputs** | month, day_of_week, day_of_year, quarter, is_weekend, is_month_end, month_sin, month_cos, dow_sin, dow_cos, lag_1, lag_2, lag_3, lag_7, lag_14, lag_28, mean_7d, mean_14d, mean_28d, std_7d, std_14d, std_28d, trend_1d, trend_7d, movement_score, sku_category, consumption_binary |
| **Set 3 — Output** | daily_consumption |
| **Set 3 — MAPE** | Per-SKU & horizon;  excel sheet |