

Retail Sales Forecasting at Walmart

Brian Seaman
WalmartLabs



The biggest challenge as a forecasting practitioner

The boss says:

I need a forecast of ...

A forecaster should respond:

Why?

Today's Focus

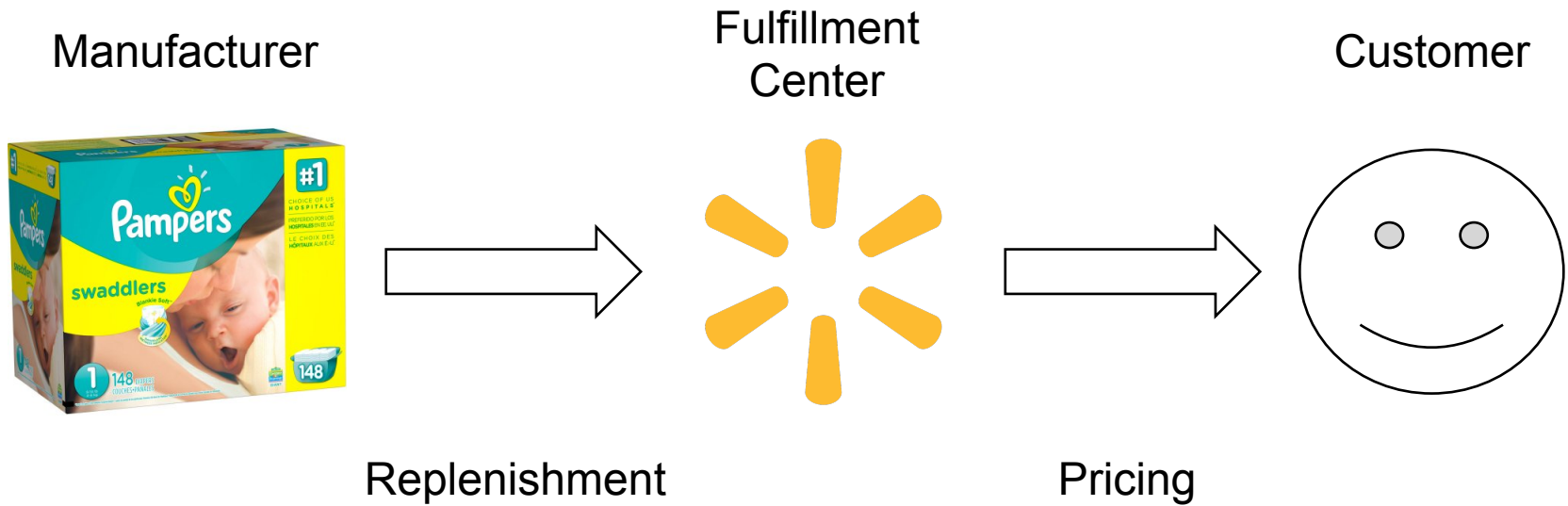
The boss says:

I need a better sales forecast

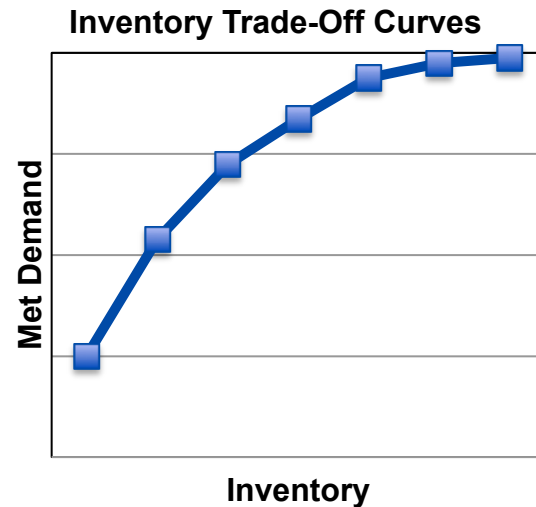
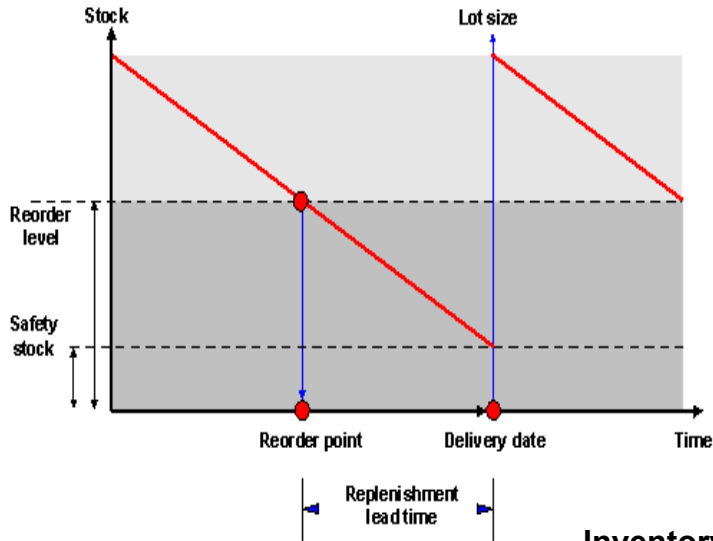
What the boss really means:

We have an issue staying
in-stock on certain items and
think that pricing may be
causing a problem

Domain Overview: Item Flow



Domain Overview: Replenishment



Objective

- Maintain lowest inventory level to meet the expected customer service level

Challenges

- Variable or long lead time items
- Uncertain forecasts
- Limited budgets
- Network flow constraints
- Holiday peaks
- Met demand and inventory trade off
- Leveraging store orders and inventory

Domain Overview: Pricing



Customers deserve the lowest price that we can offer at all times

Objective

- Set the item price to drive sales

Challenges

- What is a fair price?
- How much will we sell?
- How long will inventory last?
- What is impact to other items?

- Replenishment:
 - place order based on a forecast of an assumed future price
 - $O = F(f(P))$
- Pricing:
 - set the price based on a predicted forecasts of possible prices
 - $P = G(f(p))$
- Pricing happens after inventory is purchased

different problems

require

different solutions

Possible Forecast Objectives for an Item

- Forecast Demand
- Forecast a Distribution
- Forecast by Location
- Forecast by Customer
- Forecast Sales
- Forecast In-stock Rates
- Forecast Profit Margin
- Forecast Social Media Posts

Objective is to Improve the Customer Metrics

- Replenishment:
 - Forecast Demand Distribution
 - Assume in-stock in the future
 - Focus on upper percentiles of distribution
- Pricing:
 - Forecast Sales
 - Predict future in-stock rates
- Demand and Sales can differ based on in-stock rates
- Imputation of data can either improve or worsen forecasts depending on use
- A good mean forecast may generate a bad 95th percentile

- Error
 - Defines the accuracy of the forecasts
 - $E = \sum |f - a|^2$
- Bias
 - Defines if the forecasts on average are high or low
 - $B = \sum (f - a)$
- Volatility
 - Defines how much the forecast changes over time
 - $V = \sum |f_t - f_{t-1}|^2$
- Feel free to weight and normalize these metrics according to use case
- When deciding between models, you must make tradeoffs between metrics

- Replenishment:
 - High volatility can cause huge overstocks
 - Volatility trumps accuracy and bias
- Pricing:
 - Bias can be ignored depending on objectives
 - Accuracy trumps bias and volatility
- True impact must be determined through experimentation and simulation

- Replenishment:
 - Set by the ordering lead times
 - Generally days to months
- Pricing:
 - Set by pricing strategy
 - Varies from seconds to months
- Long time scales must include seasonality, with short time scales it is optional

- Replenishment:
 - Slow reaction times allow for slower forecasting
 - There is time to correct and change orders
- Pricing:
 - Fast reaction times demand faster forecasting
 - Must react quickly due to immediate customer impact

Tradeoffs

	Replenishment	Pricing
Objective	Demand Distribution	Mean Sales
Metric	Low Volatility	High Accuracy
Timescale	Long	Short
Speed	Slow	Fast

Item Size

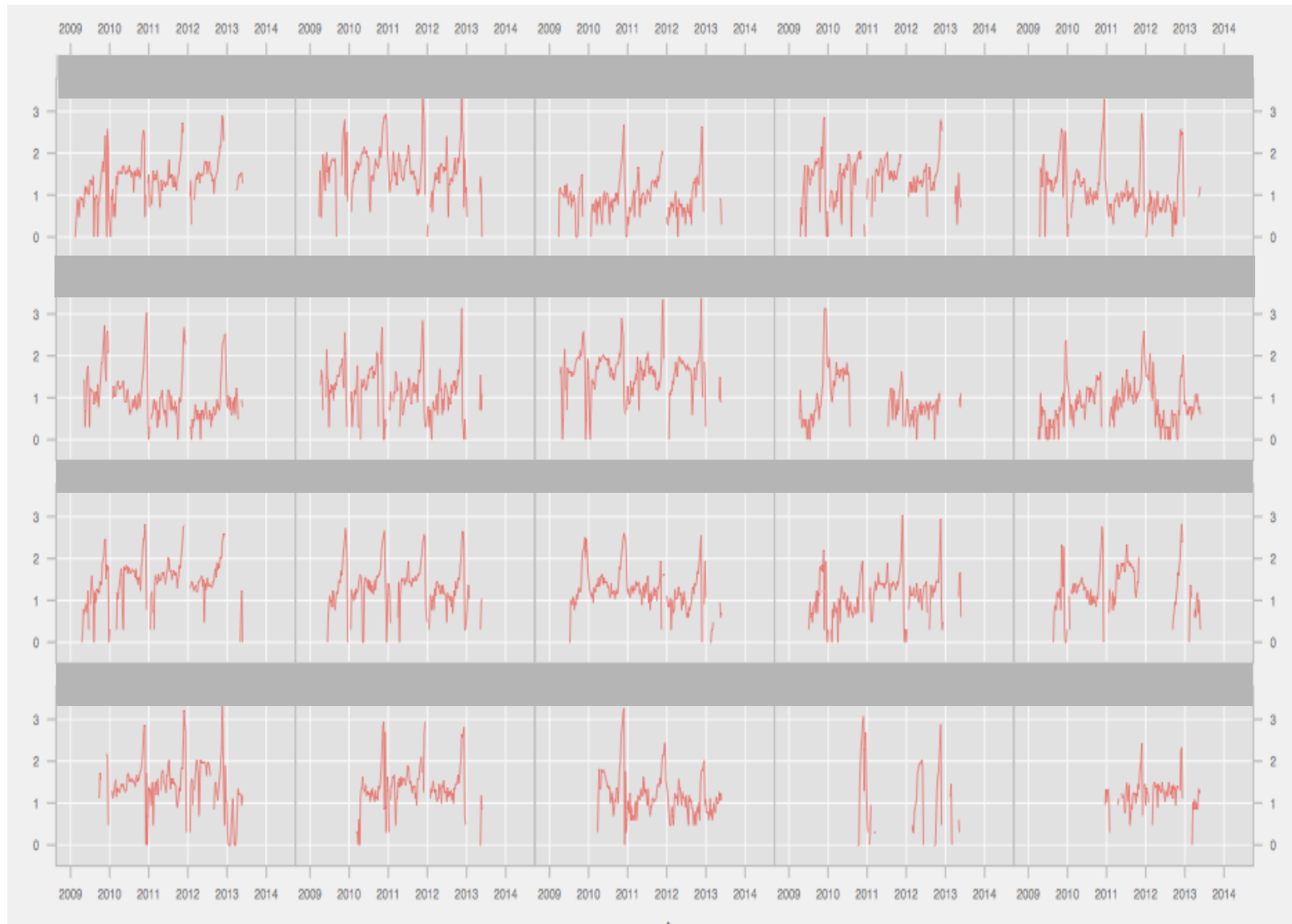
- Walmart Stores
 - ~12k Stores
 - ~200k items per store
 - ~2b unique store/items to forecast
 - ~40b item/item correlations
- Walmart.com
 - >50m items
 - 40k ZIP codes
 - ~2t unique ZIP/items to forecast
 - ~2000t item/item correlations

Walmart.com Item Information

- >50m items
- Each item has 100s to 1000s of attributes
- Few items sell consistently and have a long and complete time series
- Sales can be sparse and occasional stock-outs lead to missing data
- New items with no or relatively short sales history



Modeling: Sample Sales Data



New Tradeoffs

	Replenishment	Pricing
Objective	Demand Distribution	Mean Sales
Metric	Low Volatility	High Accuracy
Timescale	Long	Short
Speed	Fast	Very Fast

History

Complete

- Pick your favorite method
- Most are generally “good enough”

Partial

- Off the shelf methods fail quickly
- This is the hard part

None

- Can't use time series methods
- Machine Learning problem

Sales

Consistent

- “Regular” distribution
- Most are generally “good enough”

Intermittent

- Many zeros with occasional sales
- Be careful how the forecast is used

None

- This is pretty simple

Partial History

- Why do we have partial history?
 - Recently introduced item
 - Items go out of stock
 - Data feeds are corrupted
- What can we do?
 - We have lots of items, some items with excellent history
 - Share information across items
- New problems:
 - How should they share information?
 - Which items should share information?
 - What information should they share?

- Requirements
 - Fast
 - Include seasonality
 - Low volatility
 - Allow for missing data
 - Share information across items
- Our plan:
 - Cluster like items together
 - Calculate seasonal components
 - Forecast demand with multivariate DLMS
 - Estimate distribution function

Multivariate Dynamic Linear Model

$$\mathbf{Y}_t = \left[\begin{array}{c|c} 1 & \psi(t) \\ \cdot & \cdot \\ \cdot & \cdot \\ \hline & 1 \\ & \psi(t) \end{array} \right] \begin{bmatrix} \boldsymbol{\mu}_t \\ \boldsymbol{\alpha}_t \end{bmatrix} + \boldsymbol{\epsilon}_t,$$
$$\begin{bmatrix} \boldsymbol{\mu}_t \\ \boldsymbol{\alpha}_t \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mu}_{t-1} \\ \boldsymbol{\alpha}_{t-1} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\epsilon}_{1t} \\ \boldsymbol{\epsilon}_{2t} \end{bmatrix},$$
$$\boldsymbol{\epsilon}_t \sim N(0, \sigma^2 \mathbf{I}_n) \ \& \ \begin{pmatrix} \boldsymbol{\epsilon}_{1t} \\ \boldsymbol{\epsilon}_{2t} \end{pmatrix} \sim N \left(0, \sigma^2 \begin{bmatrix} g_1 \mathbf{W}_1 & \\ & g_2 \mathbf{W}_2 \end{bmatrix} \right)$$

- Local level seasonal DLM
- \mathbf{Y} is a vector of demand for a panel of n items
- g are variance scalars
- \mathbf{W} are correlation matrices
 - Diagonals of \mathbf{W} determine temporal smoothing
 - Off-diagonals of \mathbf{W} determine cross section smoothing
- Models like DLM can be computed iteratively = fast

Modeling: Clustering

- Computing a DLM across 1b entities is probably overkill
- Possible similarity metrics:
 - Euclidean distance
 - L1 or L2 norm has issues with scale
 - Pearson correlation
 - cosine angle is susceptible to outliers
 - Spearman correlation
 - ranked vectors is our Goldilocks metric
- Cluster via your favorite K-means variant
- Leverage semantic information to improve sparsity issues

Final Thoughts: Ensembling

- Generally, ensembling gives better forecasts with fewer outliers
- Fit models with different parameters and use CV to identify best combination

$$y_{it}^* = w_1 y_{it}^1 + w_2 y_{it}^2 + \dots + w_p y_{it}^p \text{ s.t. } w_i \geq 0 \text{ \& } \sum_i w_i \approx 1$$

- Forecasts may be correlated so you want parameters to be non negative
- Many ways to do this – LASSO, NNLS, Bagging, etc
- Boosting/Random Forests additionally helps to incorporate other predictors



Questions