Leveraging E-commerce Sites to Absorb Retail Stores’ Excess Inventories

by

HYUK JIN CHO
Bachelor of Engineering

and

KETAN LAHOTI
Master of Technology(M.Tech)

SUBMITTED TO THE PROGRAM IN SUPPLY CHAIN MANAGEMENT
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE IN SUPPLY CHAIN MANAGEMENT
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

JUNE 2019

Signature of Author: ____
Department of Supply Chain Management
May 10, 2019

Signature of Author: ____
Department of Supply Chain Management
May 10, 2019

Certified by: ____
Bruce C Arntzen
Executive Director, Supply Chain Management Residential Program
Capstone Advisor

Accepted by: ____
Dr. Yossi Sheffi
Director, Center for Transportation and Logistics
Elisha Gray II Professor of Engineering Systems
Professor, Civil and Environmental Engineering

Leveraging E-commerce Sites to Absorb Retail Stores' Excess Inventories

by
HYUK JIN CHO
and
KETAN LAHOTI
Submitted to the Program in Supply Chain Management
on May 10, 2019 in Partial Fulfillment of the
Requirements for the Degree of Master of Applied Science in Supply Chain Management

ABSTRACT

With an aim to avoid stock-outs, retail pharmacy business constantly deals with the problem of inventory build-up across the supply chain. There are myriad reasons for the rise in inventory such as promotions and forecasting errors. While there is a need to improve the forecasting accuracies with better data coming from the point of sale (POS), the increasing complexities associated with such factors as climate change affecting flu seasons and occurrences of disasters compound the forecasting errors. Hence, there is a need in the system to identify excess stocks being built up across various locations and reduce the inventory. The client is considering to address the problem of excess inventory by sending it from its stores to Mail Centers. The Mail Centers will then sell off this inventory quicker as it fulfills far more prescriptions per day in comparison to the stores.

The focus of this study is to develop a model to automate this process of quantifying excess inventory across thousands of retails location for the client. We then suggest a process to reduce the stocks by issuing operational guidelines for each store to pick top stock keeping units (SKUs) and pack and ship them from stores to respective Mail Centers. The frequency for doing out this activity is also suggested after carrying out the cost benefit analysis involved.

Capstone Advisor: Bruce C Arntzen
Title: Executive Director, Supply Chain Management Residential Program

ACKNOWLEDGMENTS

First, we want to thank Dr. Bruce C Arntzen for providing helpful feedback and guiding us in the right direction throughout the project, even when we ran into problems.

We also want to thank CVS Health for sponsoring our project, especially Thomas McHugh, Gary J. Misiaszek, and Lauren Mancini. They were all very supportive, and not only shared a vast amount of data required for analysis, but also discussed alternative solutions which were instrumental in the success of this project.

Finally, I need to thank Pamela Siska and Toby Gooley for helping us with the writing. They helped restructure the paper and allowed us to be more precise and concise.

We are very grateful for all of the support and valuable feedback on this Masters capstone project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1: Introduction

1.1 Background

CVS Health is the leading retail pharmacy chain in the Unites States of America. With the commitment to deliver innovative health solutions that create a simpler, more accessible experience for patients, customers and caregivers, CVS Health is always trying to ensure that the patient does not go empty handed from its pharmacies. In order to serve its customers to its fullest, CVS Health has built up its inventory to prevent stock outs. However, the resulting rise in inventory levels has become one of the major areas of concerns for the business. It currently operates a complex supply chain consisting of ~3000 SKUs from 13 Distribution Centers, over 9600 Stores, and multiple Mail Centers, Specialty Pharmacies, Infusion Centers and Clinics across the country, as well as Deliveries from Wholesalers to the stores.

In the retail business, it is essential to keep a track of the inventory turns and the return of assets (ROA). As a publicly traded company, these numbers are closely scrutinized and become major parameters for performance comparison in the industry. These pressures led the company to right size their Rx inventory levels by more than a billion dollars back in 2015. The process was facilitated by collecting data of inventory across the locations, interpreting the data to identify opportunities, and manually executing the operations steps necessary to reposition inventory for better utilization. The mission was accomplished; however, it took a lot of human capital as well significant shipping costs to be successful.

There are myriad reasons for growth in Rx inventory at CVS Health such as promotions, use of multiple pharmacies by patients and forecasting errors. While there is a need to improve the forecasting accuracies with better data coming from the point of sale (POS), the increasing complexities associated with such factors as climate change affecting flu seasons and occurrences of disasters compound the forecasting errors. Hence, there is a need in the system to identify excess stocks being built up across various locations and reduce the inventory. The company is considering to address the problem of excess inventory by sending it from stores to Mail Centers. The Mail Centers will then sell off this inventory quicker as it fulfills far more prescriptions per day in comparison to the stores.

1.2 Objective

The objective of the capstone is to develop a model to identify the excess inventory across locations for CVS Health by quantifying its monetary value and suggesting a process to reduce the inventory to the desired levels. This will help institutionalize the practice of tracking excess inventory in the system utilizing the data available from the inventory management system currently in use. After identifying the excess stocks, CVS Health could issue guidelines to the stores to pick up SKUs with excess stocks and pack and ship them to Mail Centers, where they would be sold off more quickly. Hence, the model developed will help CVS Health in keeping track of the inventory turns and ROA across locations, thereby helping to improve business performance.

Chapter 2: Literature Review

Inventory systems account for a large proportion of business' cost, especially in the retail sector, and hence need to be managed efficiently. The traditional inventory design is hierarchical and flows from manufacturer to wholesaler and then onwards to retailer. More flexible inventory systems allow for lateral transshipments, as shown in figure 1, within an echelon, i.e., redistribution within the retailers (same echelon) to pool their inventory to achieve similar service levels albeit with lower inventory costs.

In the LT (lateral transshipment) literature, two major types of transshipments categorized on the basis of timing of the transshipments (Paterson et al., 2011; Seidscher and Minner, 2013), are as follows:

1. *Proactive Transshipment* - Proactive LT is a preventive approach. Before an inventory shortage occurs, the LT decision is implemented at a predetermined point in time to avoid future stockout. Hence, it reduces the risk of future stockouts by rebalancing/redistributing inventory across different locations in the same echelon (Bertrand and Bookbinder, 1998; Jönsson and Silver, 1987; Tagaras and Vlachos, 2002).

2. *Reactive Transshipment* - The alternative approach is reactive LT (Chiu and Huang, 2003), which responds to an occurrence of inventory shortage after demand has been realized. As a result, location with surplus stock on hand transfers inventory to another location with stock deficiency (Krishnan and Rao, 1965; Robinson, 1990).

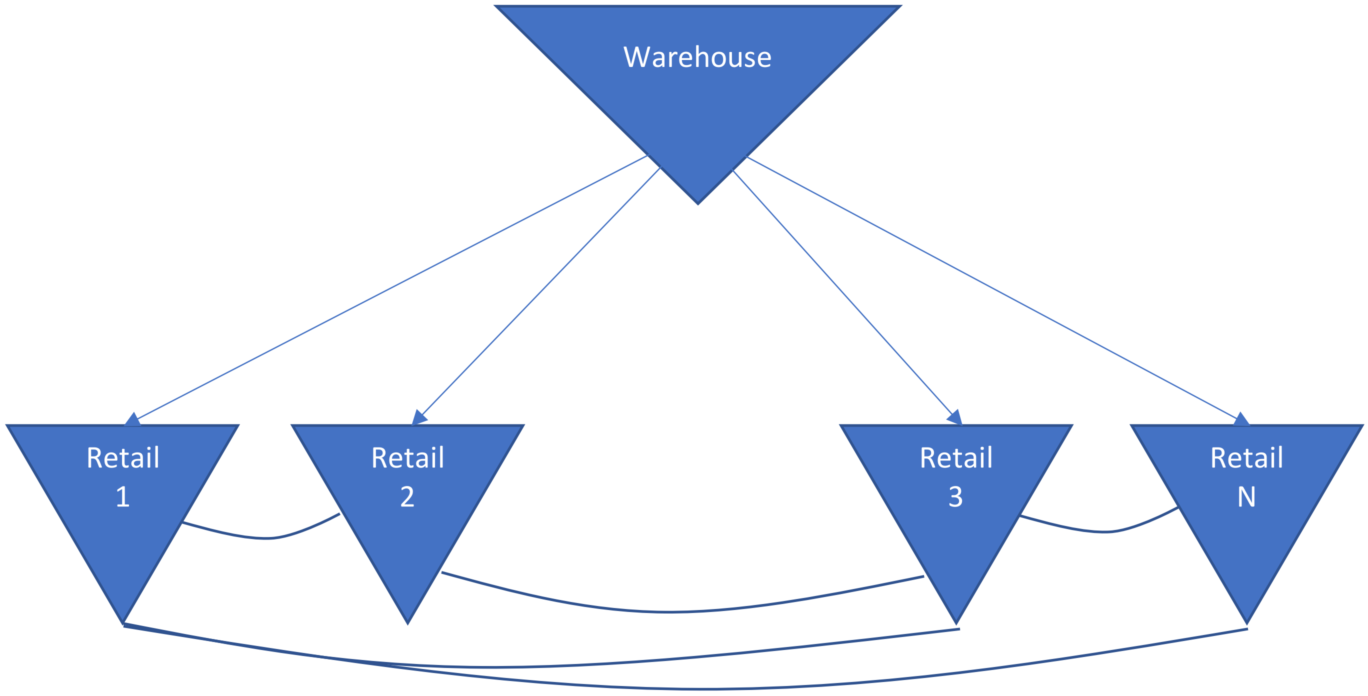


Figure 1 showing Lateral transshipment model

Our study concentrates on developing a proactive lateral transshipment model to reduce inventory levels across 9,600 retail outlets.

It becomes challenging to control, operate and optimize an inventory system with the allowable flexibility of lateral transshipments. Besides deciding when and how much to order from the supplier, decisions also need to be made on when, how much and from where the transshipments need to take place. As a result of the added complexity, the literature is found to be mainly restricted to systems with two echelons (as depicted in the figure 1) or in some cases restricted further by not considering the central warehouse and/or allowing only a limited number of stocking points in the second echelon (Paterson, Kiesmüller, Teunter and Glazebrook, 2011)

Due to complex nature of the problem, there are a few heuristic transshipment policies available which work reasonably well. Transshipment Based on Availability (TBA) and Transshipments for Inventory Equalization (TIE) are two such proactive transshipment rules introduced by Banerjee, Burton, and Banerjee (2003). Under these policies, if the inventory level at a particular location drops below the average demand rate, a transshipment is initiated. Under TBA, the system redistributes stocks between locations to prevent stockouts in next period and hence acts reactively. While TBA policy is slightly more effective in preventing stockout incidents (Banerjee et al. 2003), Burton and Banerjee (2005) show that TIE policy generally achieves lower overall system cost. This reflects how the objectives of the system impacts the transshipment policy to be adopted. These policies can be applied to multi-period or multi-location problem easily.

The comparison between TIE and TBA is further extended by Lee et al. (2007) by including another proactive heuristic – SLA (Service Level Adjustment) policy to the problem. The probability of stockout is calculated with an upper bound and a lower bound at the beginning of each sub-period. The retail outlet with stock over and above that required to achieve its upper bound of SLA replenishes the stock for retailer with highest requirement (stock required to reach the lower bound of the SLA). This policy is shown to be superior, by a simulation study, in terms of cost metric than both TBA and TIE combined, when costs for transportation are significantly low.

Although transshipments have been shown to reduce the total occurrence of lost sales by redistribution of stocks across the echelon, it also induces certain practical risks. These risks include reduced forecast accuracy, by frequent rebalancing of stocks, and increased chances of human error in a complex inventory system.

The thesis is organized as follows – In section 3, we describe the multi-location transshipment problem with assumptions and notations used and discuss the methodology used to arrive at answering two questions – 1) How much inventory is excess in the system and 2) How to carry out the activity of transfers from multi-locations keeping in mind the cost and frequency of such acts. In section 4, we discuss the results of both the questions together with limitations of model developed and future refinement of model with proactive local transshipment policy in tandem with the developed approach.

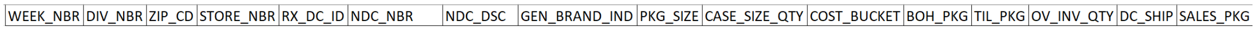
Chapter 3: Data and Methodology

The focus of this study is to find the excessive inventory (quantity as well as monetary value) at a particular instant in time across the various stocking locations. By comparing current weeks of inventory stock to the required target levels of inventory in weeks at each location, we can determine the excess inventory levels. The next question then is when and how to redistribute these stocks. The call to initiate a redistribution across locations would primarily take into account costs in the form of labor, packaging and shipping. These costs have to be factored in to arrive at the frequency levels of exercising this activity of sending stocks from outlets to Mail Centers (either Philadelphia or Chicago), where they are most likely to be sold off quickly.

3.1 Approach

3.1.1 Data Collection

We first requested the data for inventory across 3,000 SKUs and 9,000 stores of CVS Health for all 52 weeks in the past year. The data comprised following columns:



WEEK_NBR	DIV_NBR	ZIP_CD	STORE_NBR	RX_DC_ID	NDC_NBR	NDC_DSC	GEN_BRAND_IND	PKG_SIZE	CASE_SIZE_QTY	COST_BUCKET	BOH_PKG	TIL_PKG	OV_INV_QTY	DC_SHIP	SALES_PKG

- WEEK_NBR – Fiscal week of the year
- DIV_NBR – Unique number for the business division
- ZIP_CD – ZIP code of the store
- STORE_NBR – Unique store ID
- RX_DC_ID – ID of DC serving that pharmacy

- NDC_NBR – National Drug Code for the SKU
- NDC_DSC – Drug Name
- GEN_BRAND_IND –Generic or Brand indicator
- PKG_SIZE – Package size
- CASE_SIZE_QTY – Case size
- COST_BUCKET – Ranking based on package cost
- BOH_PKG – On-hand inventory at the end of fiscal week
- TIL_PKG – Target inventory level, used as order up to point when ordering to DC
- OV_INV_QTY – Net shipments from wholesaler
- DC_SHIP – Net shipments from DC in packages
- SALES_PKG – Net sales in packages for store in the fiscal week

In order to protect actual costs which are proprietary business information, the study uses Average Wholesale Price (AWP) a recognized industry proxy for the value of each drug. Drugs are also categorized as controlled or non-controlled by the DEA (Drug Enforcement Agency). Controlled drugs will not be included in the redistribution activity because of risks involved and tighter regulation around transfers.

3.1.2 Data Cleaning

We removed the unnecessary columns which were mostly the description of the drugs and chose columns which were required for further data processing. The final data set comprised 5 columns

WEEK_NBR	STORE_NBR	NDC_NBR	BOH_PKG	SALES_PKG

The negative value in the SALES_PKG & BOH_PKG were all changed to zero. These values were a case of very few returns taking place in the system which was less than 1% in the overall data. There were also instances in the data where the on-hand inventory was zero at the end of the week while sales did take place during that week. Such occurrences meant that there could have been instances of lost sales in the particular store after the on-hand inventory went down to zero. Hence, we increased the sales value by a factor of 10% to capture this in the average flow (discussed in the next section).

3.1.3 Data Processing

We first calculated the average outbound for each SKU at a store by taking the mean of the sales value across the 52 weeks.

$$Average\ Outbound = Sum\ of\ sales\ for\ a\ SKU\ in\ a\ store\ across\ 52\ weeks/52$$

We next calculated the Current weeks of supply for each SKU at a store for week 52 by dividing Average outbound obtained in the previous equation by BOH_PKG (current on-hand)

$$Current\ Weeks\ of\ Supply(WOS) = \frac{BOH_{PKG} for Week\ 52}{(Average\ Outbound)}$$

The above step was done for each of the 3000 SKUs across 9,600 stores to arrive at Weeks of Supply for each SKU at each store.

The target inventory levels mentioned by CVS Health (client) was 3 weeks but we compared the values of Weeks of Supply against 3 ,5 and 10 weeks of target inventory to see the range of excess inventory in the system with different target levels.

$$Excess\ Inventory\ for\ a\ SKU\ in\ a\ store = Current\ WOS - Target\ inventory\ levels$$

For security reasons, only packed whole units could be shipped. Therefore, in cases where the excess inventory was less than 1.0 unit, we didn't consider it to be excess. In cases where the excess inventory was in whole numbers with decimals, we rounded it down to the nearest whole number. But if the average outbound equaled zero (sales zero throughout the year), we considered two cases – (1) If excess was found to be 1.2 units for example, we shipped 1 unit and left the partially opened drug (0.2) at the pharmacy (2) If excess was found to be 2 units for example, we shipped the remaining to the Mail Center while keeping the 1 unit of inventory at store as per instructions from CVS Health.

After determining the excess inventory by subtracting the target inventory from WOS, the excess inventories were multiplied with AWP (Average wholesale price) to arrive at the monetary value of excess inventory across the outlets in the system.

$$Excess\ inventory\ value\ across\ the\ system = \sum(Excess\ inventory\ units\ for\ a\ SKU\ in\ store * \text{AWP})$$

(Note: The actual cost to the retailer will be different than the AWP as used in the analysis)*

3.2 Platform used for data analytics

Because of the enormous size of the data (more than 1 billion records) being processed, we used Python to carry out the data analytics and arrive at the results. The resulting code would be able to process data fed in a certain format and identify excess inventory in the system. Also, a sensitivity analysis in the form of change in target levels of inventory in the system has been carried out on quarterly basis to see the accumulation of the stock the variation in the excess inventory over time.

Chapter 4: Results & Analysis

This study formulates a model which can be applied to inventory-level data across stores in a stated format to determine excess stocks of each SKU in the outlet. These stocks can then be sent across to the Mail Centers (either Philadelphia or Chicago), where they can be sold off faster. Thereby, freeing up space in the stores to improve overall inventory turns and return on assets for the organization. This model helps to institutionalize the practice of finding excess stocks in the system and directing them to the Mail Centers (either Philadelphia or Chicago) this enables the client to reduce the inventory accumulated across the various outlets in the network.

In 2015, the client had undergone a massive inventory-reduction exercise across its various businesses due to pressures from Wall Street to bring the inventory turns to industry benchmark levels. The activity was carried out manually and expended a lot of resources in finding out the excess stocks in the system. Since then, there seems to again be a build-up of inventory in the system over and above the amount required to maintain the highest service levels to its consumers. As a retail pharmacy, the client cannot afford to have stockouts and send the consumer/patient home empty-handed. The analysis of excess stocks was conducted for each of the standard periods (3 weeks, 5 weeks or 10 weeks worth of inventory), separating out controlled (restricted in terms of transfers due to regulations) and non-controlled drugs.

Table 1 showing excess dollar amount by drug category

Target Weeks of Supply (WOS)	Excess dollar amount Controlled Drugs	Excess dollar amount Non-controlled Drugs	Excess dollar amount of All Drugs
3 Weeks	$69.49M	$1636.55M	$1706.04M
5 Weeks	$57.82M	$1014.6M	$1072.42M
10 Weeks	$48.0M	$472.52M	$520.53M

The results show that the excess inventory levels in the system have again climbed to more than a $1 billion worth of stocks with reference point at 5 weeks of supply. On average, every store across the country was found to have 230 SKUs (almost 50% of all SKUs carried by stores) with excess inventory, with a standard deviation of 41 SKUs as can be seen from figure 2 and figure 3.

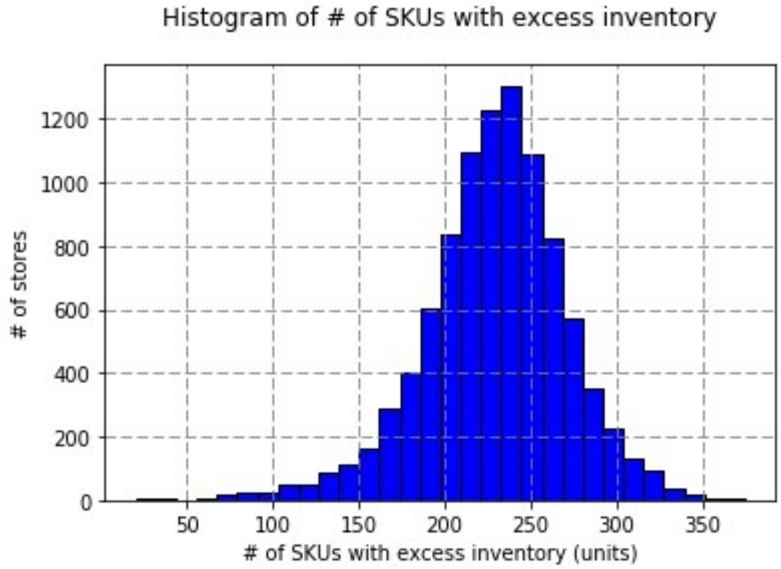


Figure 2 showing histogram of # of Store with # of SKUs with excess inventory

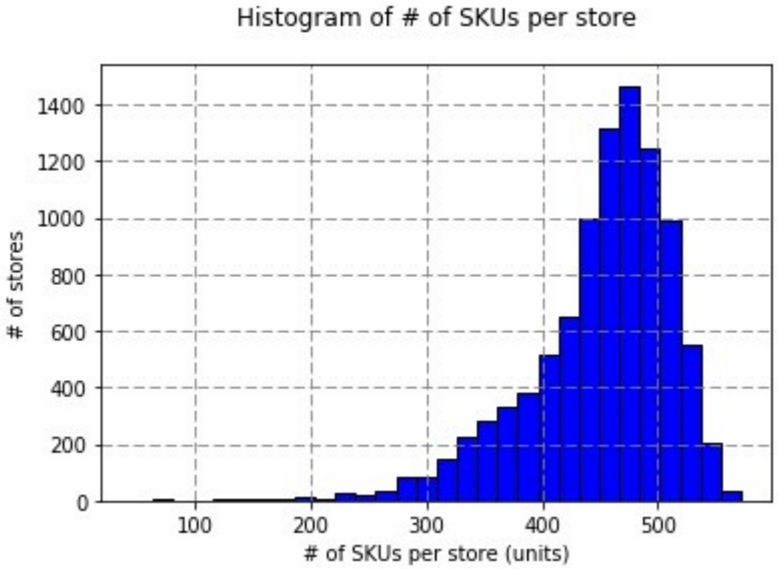


Figure 3 showing histogram of # of SKUs carried by # of Stores

The analysis comes up with specific SKU-level details for each store, with the amount of excess stocks lying at stores at the end of week 52 of the previous year. The excess can be sent to Mail Centers, where they can be sold off quickly. As discussed in the previous chapters, we have tried to do analysis with the available records of weekly data of all SKUs across all stores from the past year.

4.1 Limitation of the Model

The model in its current formulation takes into account last years’ worth of weekly data for ~ 3000 SKUs across 9,600 stores. To arrive at the Current weeks of Supply at week 52, on-hand inventory at week-52 was divided by average outbound (sales). However, there is an assumption that the sales pattern remains same weekly throughout the year which may not be the case especially with seasonality in case of several drugs. The weekly data of previous years for all SKU across various stores could have helped to ascertain the seasonality as well as occurrence of any outlier events. In that case, we would have accordingly modelled the trend to arrive at a more robust sales trend at both the store and SKU level by assigning weights to obtain weeks of supply for different periods.

Chapter 5: Discussion and Conclusion

The model developed identifies the excess inventory levels by store and by SKU type across retail locations in the country. This helps in realizing both tangible as well as intangible benefits for the organization. The tangible benefits are in form of inventory reduction (improvement of Return on Assets (ROA) and inventory turns ratio to achieve similar service levels. The intangible benefits are in the form of improved customer experience and employee morale. The model, with developed algorithm will help automate the process of determining excess inventory levels in the system and institutionalizing the practice to rebalance stocks.

5.1 Insights & Management Recommendations

The next step is to create guidelines for the business to achieve these inventory reductions. Guidelines need to be issued to each store for shipping SKUs with excess stock to the Mail Center (either Philadelphia or Chicago). A cost-benefit analysis of doing such transfers should be done to determine the frequency of carrying out this activity

1. *Operational Guidelines* - We created a Pareto Chart showing the impact of the reduction that can be achieved by shipping different numbers of SKUs from individual stores. For example, in figure 4 and figure 5 below if we ship the top 10 excess SKUs from each store, the client would be able to reduce inventory by $300 million, representing approximately 30% of the total potential reduction (reference taken as 5 weeks of supply for each SKU required at stores).

Table 2 showing Savings (in $million) v/s top SKUs(by dollar value) shipped per Store

Top SKUs shipped/Store	**Total Savings ($M)**	**% Savings**
1	84.33	8.11%
5	208.22	20.03%
10	300.46	28.91%
15	369.6	35.56%
20	426.16	41.00%
25	474.5	45.65%
30	516.83	49.72%
35	554.53	53.35%
40	588.49	56.62%
45	619.35	59.59%
50	647.61	62.31%
55	673.63	64.81%
60	697.71	67.13%
65	720.08	69.28%
70	740.93	71.28%
75	760.42	73.16%
80	778.67	74.92%
85	795.78	76.56%
90	811.84	78.11%
95	826.91	79.56%
100	841.08	80.92%

The table above shows if client shipped top 10 SKUs (arranged by descending order of dollar value) per Store, the inventory reduction can take place to the tune of $300 million. If this activity is repeated and next 10 SKUs are shipped, the savings will be $125.70 million ($426.16 million - $300.46 million). Further, client will need to make a strategic call on number of SKUs to be shipped (whether 10 or more) basis the cost to carry out this activity and the inventory reduction realization potential.

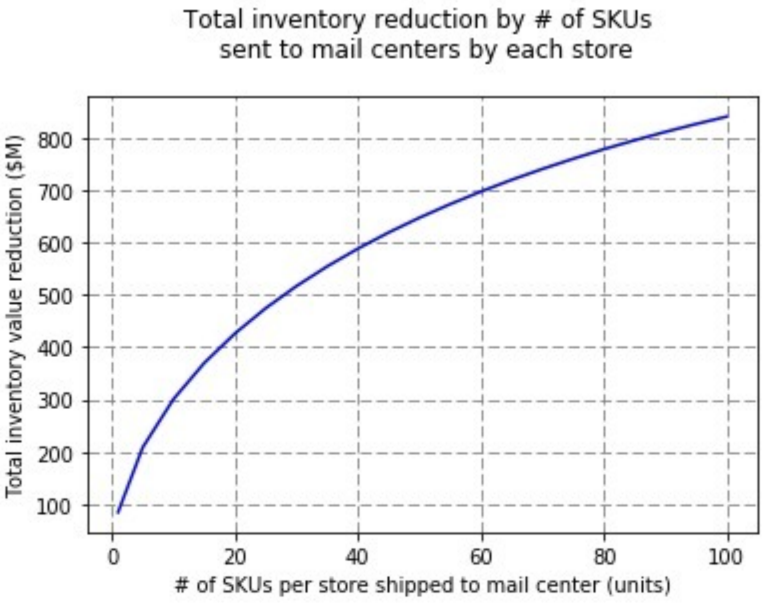


Figure 4 showing Pareto chart total Savings in ($M) vs. # of SKUs per store

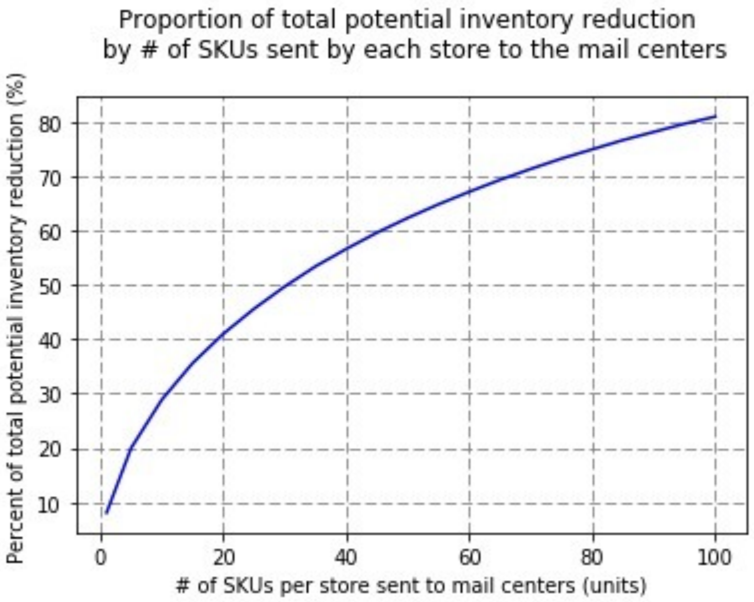


Figure 5 Pareto chart of % savings vs. # of SKUs shipped

2. *<u>Costs involved in Shipping from Stores to Mail Center</u>* - To evaluate the costs involved in the shipping of SKUs from stores to Mail Center (either Philadelphia or Chicago), we broke down the cost into 2 broad components -

   a. *<u>Labor Cost of pharmacists</u>* –To ship the top 10 SKUs from each store, the pharmacist would require one-hour time to collect all SKUs from the shelves and place them in a box. The hourly cost for a pharmacist is considered as $50.
   b. *<u>Cost of Packaging and Shipping via carriers like UPS, USPS, etc.</u>*- The store's proximity to the Mail Center (Chicago or Philadelphia) is determined on the basis of distance (using zip codes) between store and Mail Center locations. The excess stock is packed (cartooning and taping cost) and shipped to the respective Mail Center via carriers like UPS, USPS, etc. which would roughly cost another $50 per package

Thus, the total cost involved in shipping of identified excess goods for approximately 9,600 stores at $100 per store across the country would be approximately $ 1 Million.

3. *<u>Frequency of carrying out this activity</u>* – After doing due diligence on the cost for initiating such country wide transfer of stocks from stores to Mail Centre (either Philadelphia or Chicago), we recommend that to be done quarterly. This should occur at the middle of the quarter to have the reduction achieved by the end of quarter when the financial results are published. The frequency can be reviewed as CVS will reach a point where the quarterly inventory reduction will be so small as to no longer justify the quarterly $1 million in cost. At that time they may switch to just every 6 months or every year.

Hence, the model that we developed can be added to the inventory management system architecture and serve as a way to track excess inventory accumulating in the system across the business. The client can then make a strategic call to redistribute the stocks across locations basis the potential cost-benefit analysis. There does not seem to be much additional need for investments required as far as the extra capability of ‘rebalancing inventory’ across locations is concerned. The only investment that may be required would be to convert the logic developed in the appropriate programming language (currently in Python) and add a module to the information system currently in use.

5.2 Future Research

The current model in its present form identifies amount of excess inventory in stores and then directs stores to ship the SKUs with excess inventory to the nearest Mail Center (either Chicago or Philadelphia). The model can be extended to enable lateral transshipments whereby store to store transfers for excess stock may be feasible within a region to balance inventory levels across a geography while maintaining high service levels throughout. After the rebalancing of stocks across the stores in a region, any excess stocks can be then sent across to the Mail Centers, where it can be sold off quickly. This will help reduce the shipments from DC to the stores saving transportation costs as rebalancing inventory across the stores in the region would help the stores with stocks shortfall quickly and with lesser transportation costs involved.

References

Paterson, Colin & Kiesmüller, G.P. & Teunter, Ruud & Glazebrook, Kevin. (2011). Inventory models with lateral transshipments: A review*. European Journal of Operational Research. 210. 125-136*. 10.1016/j.ejor.2010.05.048.

L P Bertrand and J H Bookbinder (1998). Stock redistribution in two-echelon logistics systems. *Journal of the Operational Research Society 1998, 966—975*. 10.1057/palgrave.jors.2600613. https://doi.org/10.1057/palgrave.jors.2600613

Silver, Edward. (2008). Inventory Management: An Overview, *Canadian Publications, Practical Applications and Suggestions for Future Research. Infor.46*. 10.3138/infor.46.1.15.

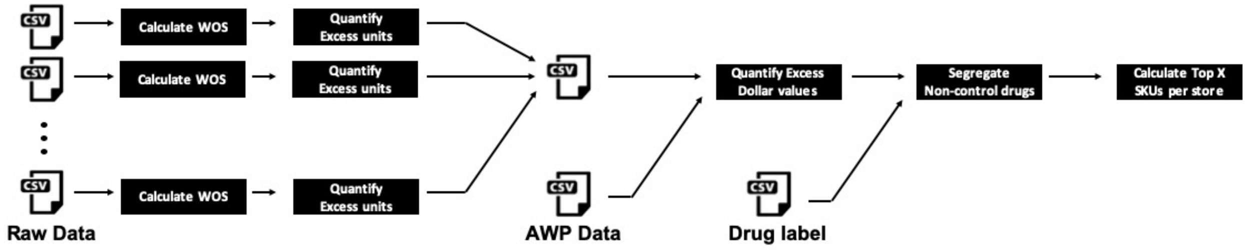
George Tagaras & Dimitrios Vlachos (2002). A Periodic Review Inventory System with Emergency Replenishments. *Management Science, 47(3)*. https://doi.org/10.1287/mnsc.47.3.415.9770

K. S. Krishnan and V. R. K. Rao, “Inventory control in n warehouses,” *Journal of Industrial Engineering, Vol. 16, pp. 212–215, 1965.*

Banerjee, Avijit & Burton, Jonathan & Banerjee, Snehamay. (2003). A simulation study of lateral shipments in single supplier, multiple buyers supply chain networks. *International Journal of Production Economics. 81-82. 103-114*. 10.1016/S0925-5273(02)00366-3.

Appendix

Appendix A. Inventory Reduction Model



Appendix B. Model Code

```
# In[78]:
import glob
import numpy as np
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib import rcParams
from scipy import stats


# # INDIVIDUAL RAW CSV FILE
# In[55]:
# Read raw data - Do it for all csv file name by changing the file name
df = pd.read_csv("M08.csv", sep = "|")

# In[56]:
df.drop(['DIV_NBR', 'RX_DC_ID', 'ZIP_CD', 'NDC_DSC', 'GEN_BRAND_IND', 'PKG_SIZE', 'CASE_SIZE_QTY',
'COST_BUCKET', 'TIL_PKG', 'OV_INV_QTY', 'DC_SHIP'], axis = 1, inplace = True)

# In[57]:
# Apply assumption:
# when Inventory on hand(BOH_PKG) is 0, lost sales are about 10% of the sales that week
df["TEMP"] = np.where(df["BOH_PKG"] == 0, df["SALES_PKG"] * 1.1, df["SALES_PKG"])
df["SALES"] = np.where(df["TEMP"] < 0, 0, df["TEMP"])
df.drop(["TEMP", "SALES_PKG"], axis = 1, inplace = True)

# In[58]:
# when Inventory on hand(BOH_PKG) is less than zero, convert it to zero
# Change column name: BOH_PKG -> IOH
df["IOH"] = np.where(df["BOH_PKG"] < 0, 0, df["BOH_PKG"])
df.drop(["BOH_PKG"], axis =1, inplace = True)
```

```
# In[59]:
# Caculate average weekly sales for each SKU at each Store
wf = df.groupby([df["STORE_NBR"], df["NDC_NBR"]])["SALES"].mean().reset_index().round(3)

# In[60]:
# Filter only week 52
now = df[df["WEEK_NBR"] == 201852].reset_index()
now.drop('SALES', axis = 1, inplace = True)

# In[61]:
# merge average weekly sales value to filtered(week 52) data
merge = pd.merge(now, wf, how = "left", left_on = ["STORE_NBR", "NDC_NBR"], right_on = ["STORE_NBR",
"NDC_NBR"])

# In[66]:
# Calculate Weeks of Supply for each SKU at each store
# when average weekly sales are zero, convert to 0.00001 to calculate
merge["W_SALES"] = (np.where(merge["SALES"] == 0, 0.00001, merge["SALES"]))
merge["N_WOS"] = (merge["IOH"] / merge["W_SALES"]).round(2)
merge.drop("SALES", axis = 1, inplace = True)

# In[74]:
merge.drop(["index"], axis = 1, inplace = True)

# In[76]:
# save dataframe to csv file
merge.to_csv("WOS08.csv")


# # COMBINE MODIFIED RAW DATA
# In[79]:
# Reading all csv files created above and concatinate
df = pd.concat([pd.read_csv(f) for f in glob.glob('WOS*.csv')], ignore_index = True)
df.drop(["Unnamed: 0"], axis = 1, inplace = True)

# In[80]:
# Creating columns - target weeks of supply (3, 5, and 10)
df["TWOS03"] = 3
df["TWOS05"] = 5
df["TWOS10"] = 10

# In[81]:
# Creating columns - excess inventory in weeks
df["TEMP03"] = df["N_WOS"] - df["TWOS03"]
df["TEMP05"] = df["N_WOS"] - df["TWOS05"]
df["TEMP10"] = df["N_WOS"] - df["TWOS10"]

df["EXC_WOS03"] = np.where(df["TEMP03"] > 0, df["TEMP03"], 0)
df["EXC_WOS05"] = np.where(df["TEMP05"] > 0, df["TEMP05"], 0)
df["EXC_WOS10"] = np.where(df["TEMP10"] > 0, df["TEMP10"], 0)

# In[82]:
df.drop(["TEMP03", "TEMP05", "TEMP10"], axis = 1, inplace = True)
```

```
# In[83]:
# Convert excess inventory in weeks to excess units
df["EXC_UNIT03"] = (df["EXC_WOS03"] * df["W_SALES"]).round(3)
df["EXC_UNIT05"] = (df["EXC_WOS05"] * df["W_SALES"]).round(3)
df["EXC_UNIT10"] = (df["EXC_WOS10"] * df["W_SALES"]).round(3)

# In[84]:
# Excess units to integer - remove decimal points
df["EXC_UNIT03"] = df["EXC_UNIT03"].astype(int)
df["EXC_UNIT05"] = df["EXC_UNIT05"].astype(int)
df["EXC_UNIT10"] = df["EXC_UNIT10"].astype(int)

# In[85]:
# Creating columns - removed decimal points
df["LEFT03"] = df["IOH"] - df["EXC_UNIT03"]
df["LEFT05"] = df["IOH"] - df["EXC_UNIT05"]
df["LEFT10"] = df["IOH"] - df["EXC_UNIT10"]


# In[86]:
# Condition - leave minimum quantity at the store
# When average weekly sale is zero and excess unit is whole number, remove one unit from excess unit to leave at
least one unit.
df["EXC_UNIT03"] = np.where((df["W_SALES"] < 0.0001) & (df["LEFT03"] == 0), df["EXC_UNIT03"] - 1,
df["EXC_UNIT03"])
df["EXC_UNIT05"] = np.where((df["W_SALES"] < 0.0001) & (df["LEFT05"] == 0), df["EXC_UNIT05"] - 1,
df["EXC_UNIT05"])
df["EXC_UNIT10"] = np.where((df["W_SALES"] < 0.0001) & (df["LEFT10"] == 0), df["EXC_UNIT10"] - 1,
df["EXC_UNIT10"])

# In[87]:
# Remove unnecessary columns
df.drop(['TWOS03', 'TWOS05', 'TWOS10', 'EXC_WOS03', 'EXC_WOS05', 'EXC_WOS10'], axis = 1, inplace =
True)


# # PRICE DATA FILE
# In[90]:
price = pd.read_csv("price.csv")

# In[91]:
price.drop(["Unnamed: 0", "SKU_NBR", "AWP/PKG"], axis = 1, inplace = True)


# # MERGE PRICE WITH DATA
# In[92]:
df_price = pd.merge(df, price, how = "left", left_on = "NDC_NBR", right_on ="UPC_NDC_NBR")

# In[93]:
df_price.drop(["UPC_NDC_NBR"], axis = 1, inplace = True)

# In[94]:
# Calculate excess dollar values for WOS of 3, 5, and 10
df_price["DOLLAR03"] = df_price["EXC_UNIT03"] * df_price["AWP"]
df_price["DOLLAR05"] = df_price["EXC_UNIT05"] * df_price["AWP"]
df_price["DOLLAR10"] = df_price["EXC_UNIT10"] * df_price["AWP"]
```

```
# In[95]:
# Creating new csv file
df_price.to_csv("complete.csv")

# In[96]:
df = df_price

# In[97]:
# Creating each dataframe for Q1 to Q4
q1 = df[df["WEEK_NBR"] == 201813].reset_index()
q2 = df[df["WEEK_NBR"] == 201826].reset_index()
q3 = df[df["WEEK_NBR"] == 201839].reset_index()
q4 = df[df["WEEK_NBR"] == 201852].reset_index()

# In[98]:
# Creating new csv file
q1.to_csv("201813.csv")
q2.to_csv("201826.csv")
q3.to_csv("201839.csv")
q4.to_csv("201852.csv")


# # FILTERING NON_CONTROLLED DRUGS
# In[99]:
control = pd.read_csv("control.csv")
control.drop(["Unnamed: 0", "SKU_NBR", "NDC_DSC", "PKG_SIZE", "SCHD_DRUG_CD"], axis = 1, inplace =
True)

# In[100]:
# Filtering Non-Controlled drugs
# Merge Q4(week 52) dataframe with controlled & non-controlled label
merge = pd.merge(q4, control, how = "left", left_on = "NDC_NBR", right_on = "NDC_NBR")
merged = merge[merge["CONTROL_IND"] == "NON-CONTROL"].reset_index()
merged_con = merge[merge["CONTROL_IND"] == "CONTROL"].reset_index()


# # RESULT
# In[190]:
# Assigning number of top X SKUs per store
X = 20

# In[191]:
# FILTERING TOP X number of SKUs from each store
df03 = merged.groupby(["STORE_NBR"]).apply(lambda x: x.sort_values("DOLLAR03", ascending =
False).head(X)).reset_index(drop = True)
df05 = merged.groupby(["STORE_NBR"]).apply(lambda x: x.sort_values("DOLLAR05", ascending =
False).head(X)).reset_index(drop = True)
df10 = merged.groupby(["STORE_NBR"]).apply(lambda x: x.sort_values("DOLLAR10", ascending =
False).head(X)).reset_index(drop = True)


# In[192]:
# Printing out result
print("TOP 20 dollar value products")
print("Target WOS =  3 weeks: $", (df03["DOLLAR03"].sum()/1000000).round(2), "M")
print("Target WOS =  5 weeks: $", (df05["DOLLAR05"].sum()/1000000).round(2), "M")
print("Target WOS = 10 weeks: $", (df10["DOLLAR10"].sum()/1000000).round(2), "M")
print("")
print("Total excessive dollar value (non-controlled)")
print("Target WOS =  3 weeks: $", (merged["DOLLAR03"].sum()/1000000).round(2), "M")
print("Target WOS =  5 weeks: $", (merged["DOLLAR05"].sum()/1000000).round(2), "M")
print("Target WOS = 10 weeks: $", (merged["DOLLAR10"].sum()/1000000).round(2), "M")
print("")
print("TWOS  3 -> Top 20 products are
",(((df03["DOLLAR03"].sum()/1000000)/(q4["DOLLAR03"].sum()/1000000))*100).round(2),"% of total dollar
value")
print("TWOS  5 -> Top 20 products are
",(((df05["DOLLAR05"].sum()/1000000)/(q4["DOLLAR05"].sum()/1000000))*100).round(2),"% of total dollar
value")
print("TWOS 10 -> Top 20 products are
",(((df10["DOLLAR10"].sum()/1000000)/(q4["DOLLAR10"].sum()/1000000))*100).round(2),"% of total dollar
value")


# Below is using only Target WOS of 5 weeks
# In[160]:
SkuCount = []
Dollar = []
Percentage = []

m = merged.groupby(["STORE_NBR"])

for i in range(0, 101, 5):
    d = m.apply(lambda x: x.sort_values("DOLLAR05", ascending = False).head(i)).reset_index(drop = True)
    SkuCount.append(i)
    Dollar.append((d["DOLLAR05"].sum()/1000000).round(2))
    Percentage.append((((d["DOLLAR05"].sum())/(q4["DOLLAR05"].sum()))*100).round(2))

create = {"# of SKUs": SkuCount, "Dollar": Dollar, "Percentage": Percentage}
df = pd.DataFrame(create)


## OPERATIONAL GUIDELINE
# In[217]:
# Entire list of top 20 SKUs at each store
operation = df05.groupby(["STORE_NBR"])[["STORE_NBR", "NDC_NBR", "EXC_UNIT05",
"DOLLAR05"]].head(20)
operation.head(50)

# In[218]:
operation.to_csv("operation.csv")

# In[210]:
# Specific store data
# Enter Store #
S = 3

# Return top 20 SKUs and quantities
df05[df05["STORE_NBR"] == 3][["NDC_NBR", "EXC_UNIT05", "DOLLAR05"]]
```

```
# # GRAPHS
# In[161]:
rcParams["axes.titlepad"] = 20

# In[162]:
x = SkuCount
y = DollarSave

plt.plot(x, y, "b")
plt.xlabel("# of SKUs per store shipped to mail center (units)")
plt.ylabel("Total inventory value reduction ($M)")
plt.title("Total inventory reduction by # of SKUs \n sent to mail centers by each store")

# In[163]:
x = SkuCount
y = Percentage

plt.plot(x, y, "b")
plt.xlabel("# of SKUs per store sent to mail centers (units)")
plt.ylabel("Percent of total potential inventory reduction (%)")
plt.title("Proportion of total potential inventory reduction \n by # of SKUs sent by each store to the mail centers")

# In[164]:
temp = merged.groupby(["STORE_NBR"])["NDC_NBR"].count().reset_index()
m_temp = merged[merged["EXC_UNIT05"]>0]
temp1 = m_temp.groupby(["STORE_NBR"])["NDC_NBR"].count().reset_index()

# In[165]:
plt.hist(temp["NDC_NBR"], color = 'blue', edgecolor = 'black', bins = 30)
plt.xlabel("# of SKUs per store (units)")
plt.ylabel("# of stores")
plt.title("Histogram of # of SKUs per store")

# In[179]:
print("Stores have mean of", round(temp["NDC_NBR"].mean(), 1),"SKUs with stdev of",
round(temp["NDC_NBR"].std(), 1))


# In[180]:
plt.hist(temp1["NDC_NBR"], color = 'blue', edgecolor = 'black', bins = 30)
plt.xlabel("# of SKUs with excess inventory (units)")
plt.ylabel("# of stores")
plt.title("Histogram of # of SKUs with excess inventory")

# In[181]:
print("Stores have mean of", round(temp1["NDC_NBR"].mean(),1),"excess SKUs with stdev of",
round(temp1["NDC_NBR"].std(),1))

# In[182]:
print("On average,", round((temp1["NDC_NBR"].mean() / temp["NDC_NBR"].mean())*100, 1),"% of SKUs are
excess at each store")
```