Future of Retail Stock Management: Machine Learning-Driven Inventory Accuracy

Sidratul Ahmed sidratulmuntaha.ahmed@coles.com.au

Chamil Jayasundara chamil.jayasundara@coles.com.au

Sachintha Karunaratne sachintha.karunaratne@coles.com.au

Advanced Analytics and Artificial Intelligence

Coles Supermarkets

Abstract

Inventory inaccuracy is a significant problem for retailers, whether they operate brick-and-mortar stores or e-commerce platforms. The economic implications of inaccurate inventory records include lost sales opportunities, increased wastage, excess inventory costs, and the costs associated with counting and other corrective processes. High-tech methods to mitigate inventory inaccuracy, such as RFID technology, often demand considerable capital investment, and their effectiveness can be limited by the scope of their application—typically more suited to high-value or easily tagged items. Similarly, traditional cycle counting techniques and gap-scanning approaches that are governed by fixed rules may struggle to keep up with the fast-paced retail landscape where demand patterns and supply chain dynamics are constantly changing. Consequently, these approaches, being labour-intensive and prone to errors, underline the urgent need for a more efficient solution. This paper introduces an innovative approach that combines machine learning and operations research techniques to model and rectify inventory discrepancies in retail settings, which significantly improves the effectiveness of manual inventory correction activities. This has been introduced and implemented at Coles Supermarkets to make stock counts more efficient and improve inventory accuracy. Our solution has resulted in a 25% reduction in inventory errors (in terms of Root Mean Squared Error (RMSE)), 50% increase in inventory corrections compared to the traditional inventory counts, and a 57% reduction in annual labor cost spent on corrective processes. As a result, store team members are spending less time performing manual stock checks and inventory adjustments. This means they can devote more of their time into what truly matters: creating positive experiences for our customers. Furthermore, by keeping track of our inventory positions more accurately, we are improving product availability, enhancing the shopping experience, and fostering loyalty.

The ripple effects of this solution extend beyond our stores. More accurate inventory management means we are cutting down on unnecessary stock and wastage, significantly contributing to our sustainability goals. It

also means that we can place orders with confidence and consistency, giving our suppliers a stable foundation for their own planning and sustainability efforts. This strengthens the entire supply chain, making the journey from farm to table smoother and more efficient for everyone involved.

Keywords: *Inventory Accuracy, Machine Learning, Operations Research, Optimization, Supply Chain, Bayesian Inference, Sustainability*

1 Introduction

One of the fundamental challenges faced by many retailers is the discrepancies in their inventory records. In [1], authors have estimated that 65% of the inventory records examined at a U.S. retailer were inaccurate. Further empirical evidence confirms the presence of inventory inaccuracies at similar levels across various retail settings [2, 3, 4]. These inconsistencies between the system stock-on-hand (SOH) and the true SOH can arise from various sources, giving rise to a variety of problems for retailers. Although maintaining inventory precision is essential for any retailer striving to uphold both profitability and sustainability, it continues to pose as one of the most significant challenges for retailers despite the unprecedented advancements in various technologies.

Inaccuracies in retail inventory can stem from various sources including but not limited to:

- **Theft**: Theft stands out as a major contributor to retail shrinkage. There has been a noticeable rise in theft incidents recently correlating with increased cost of living [5]. These occurrences could be either external or internal, that is for example, employee fraud.
- **Supply chain-related inaccuracies**: Discrepancies in the supply chain may arise when retailers receive goods in quantities different from what was ordered from suppliers due to genuine data entry errors, misplaced goods during transportation, or deliberate fraud by suppliers.
- **Human errors**: Shrinkage can occur due to various human mistakes such as customer errors at self-checkouts, incorrect bar code scanning etc.
- **Incorrect store processes**: Improperly executed store processes may contribute to inventory inaccuracies. For instance, improper waste recording, misplacement of goods, inaccurate stock counts, and shortcomings in inventory management systems could result in significant inaccuracies in inventory records.

Irrespective of the origin, errors in inventories pose a pervasive challenge across the retail sector causing diverse issues such as decrease in sales performance and customer satisfaction and increase in wastage and lost sales. Previous research highlights the substantial impact of inventory errors, demonstrating that a typical European grocery retailer, with annual sales of $15.5 billion, could enhance its revenue by $0.6 to $1.2 billion through effective inventory management [6]. The analysis further revealed that approximately 60% of stock-keeping units (SKUs) exhibit inaccuracies, with an average deviation of 6.6 units for overestimation and 6 units for underestimation. Rectifying these discrepancies could potentially boost sales by 4%-8%.

Inaccurate inventories can adversely impact various areas of the retail business. These repercussions may include:

- **Stock outs**: When the levels of SOH recorded in the system exceed the actual quantity available in the store, it may lead to stock outs. This scenario has the potential to cause lost sales and customer dissatisfaction, especially when sought-after items are not available for purchase.
- **Overstocking**: If the actual inventory levels exceeds what is indicated in the system SOH, it could result in overstocking. This can directly contribute to wastage, especially for perishable items where correcting excess inventory through future orders might be challenging. Moreover, overstocking can escalate storage costs and potentially create storage issues for other critical items in certain circumstances.
- **Inefficient processes**: The ripple effect of retail inaccuracies extends beyond individual stores. Retail operations rely on accurate inventory data to optimize supply chain activities; incorrect data may strain suppliers and logistics unnecessarily. Furthermore, store processes such as replenishment heavily depend on recorded inventory positions, inaccuracies of which, can compromise the efficiency of these processes.
- **Incorrect insights**: Flawed inventories can distort the insights derived from data. For instance, demand may be underestimated if stockouts go unnoticed due to discrepancies between actual and system SOH values. Similarly, inventory inaccuracies could obscure underlying problems such as theft or supplier fraud, making them harder to identify and rectify.

The traditional strategy adopted by retailers to rectify their inventory situation is conducting physical inventory counts. They periodically perform an exhaustive count of all store inventory, often driven by accounting requirements. Additionally, they engage in pre-scheduled or cycle counting, as outlined by [1], where specific subsets of items are counted at regular intervals. However, these predetermined counting schedules lack adaptability to real-time data and often fail to identify inventory discrepancies beyond established patterns. Consequently, Coles employs a process known as “Gap scans” to uncover inventory issues on a day to day basis. During these scans, store staff traverse through store premises and storage areas to identify items that seem to be out of stock or have insufficient inventory levels. Historically, gap scans have been considered inefficient, with a low correlation between zero stock situations and inaccurate inventory records. Typically, only a fraction (20-30%) of products flagged during gap scans actually require inventory adjustments. Hence, our primary objective is to understand how to prioritize items for counting, by identifying items in each store that are most likely to have inaccurate inventory records

In this paper, we present an innovative data driven approach that has been successfully implemented at Coles to enhance inventory accuracy and make stock counts more efficient. Our approach comprises of two key components: (1) a machine learning-powered inventory estimation model, which we call the Total Inventory Model (TIM) and (2) a prioritization algorithm called Guided Counts, which selects ideal candidate products for inspection based on mathematical optimization.

TIM leverages a variety of machine learning algorithms and Bayesian techniques, along with a thoughtfully curated set of features, to comprehend the dynamics associated with store inventories. The model utilizes these insights to predict inventory positions for each item/store combination based on the most recent feature values. TIM is designed to be probabilistic, which means alongside predicting the mean inventory levels, it also provides a probability distribution. The probabilistic beliefs of SOH generated by TIM are then input into our count prioritization algorithm, along with other factors such as demand forecast to generate a list of items for each store to be physically counted. The prioritization algorithm works akin to a multi-period Knapsack problem [7] where the objective is to allocate items to be counted across multiple days in a way that minimises a cost metric i.e., lost sales due to stock out scenarios. Prioritization of counts were needed because there is limited labour hours allocated for the counting tasks. By implementing this approach, we were able to triple the efficiency of our inventory counts. This means that by investing the same or even fewer labor resources, we are reconciling significantly more inventories than before. Furthermore, the applications of TIM extend well beyond its current use in prioritising stock counts. The outputs of TIM has the potential to replace the use of system SOH in business processes such as in replenishment.

The rest of the paper is organized as follows. In Section 2, we discuss how TIM works in detail and the optimization approach used for prioritizing the counts with constraints on the amount of time or labour available.

In Section 3, we present our results and findings, our insights from applying this method in Coles supermarkets throughout Australia, highlighting the benefits that have been realized. In Section 4, the paper is concluded by providing a concise summary of our findings.

2 Methodology

The first part of the methodology section focuses on TIM, which is our machine learning model that estimates the true inventory positions. This includes an overview of the training data, model features, architecture, and learning algorithms employed. In the second part of this section, we delve into Guided Counts, which is the prioritization algorithm designed to identify the list of products for physical counting in order to optimize benefits from inventory counting procedures. We elaborate on establishing a cost metric known as "Unknown Lost Sales" to prioritize items at the risk of experiencing sales losses that are currently unidentified by the store. Additionally, we explore methods for incorporating uncertainty surrounding predictions through a Monte Carlo approach when an analytical solution is not available. Finally, we look at a heuristic approach to prioritize items to count before transitioning into an optimization strategy utilizing linear programming techniques.

2.1 Total Inventory Model

Our solution is centered around TIM, which is our inventory estimation model that accurately estimates the inventory levels for each item at every store, while also accounting for uncertainty. We leverage a variety of machine learning algorithms and Bayesian techniques, along with a thoughtfully curated set of features, to model the dynamics associated with store inventories. The overall idea of the TIM is to estimate the "should be" inventory positions for each item in each of the locations at the time of model scoring. Going beyond these fundamental requirements, we have developed the TIM model to provide inventory estimations as probability distributions rather than single point estimates, allowing for deeper insights into inventory statuses from the model outputs.

Features and the Data Pipeline

Our training data consists of a variety of inventory counts carried out by Coles team members at Coles stores throughout Australia over the years. These inventory counts can be categorized into three main types:

- **Scheduled counts**: These are periodic counts conducted at regular intervals following predefined static rules. They involve counting a small selection of items on a daily basis.

- **Stocktake counts**: Less frequent than scheduled counts, these counts encompass a broader range of items and typically occur monthly to every six months. They can cover all items in a single department to all items in the store, depending on the frequency that they are carried out.
- **Gapscan counts**: This type of counts are performed very frequently in an ad-hoc manner, especially when the team members observe low stock levels or discrepancies in inventories.

During these counts, if team members identify a discrepancy between the system SOH and the actual (counted) SOH, they will make adjustments to the system SOH. This process is known as inventory adjustments. By conducting these numerous inventory counts over many years, we have accumulated a comprehensive dataset that includes data from different types of items, stores, and days.

The SOH is an attribute that can experience notable fluctuations during the day as a result of sales and deliveries. In order to capture these dynamic changes effectively, we make use of a diverse set of features in

Store	Item	Day	Intra-day	Engineered
- State - Store size info: big, small, medium - Affluence information coming from store clustering model - Shopping center type - Parking type - allocated shelf capacity - Number of facings	- Category - TPC - Price - Product Volume - Product weight - Large/heavy item ind - Carton size - promotional status - Seasonal indicator - Sales velocity	- Day of week - Month - Holiday status	- Sales - System SOH - Deliveries - Inventory adjustments - Backroom data - Waste - intra-day demand forecast	- 3/7/30 day aggregations for – Sales – Deliveries – Inventory adjustment – End of the day SOH – Waste – Online picks - 1-7 day lag features for – Sales – Inventory adjustments – Start of the day and end of the day SOH – Forecast – Deliveries

Table 1: Features of TIM Model

TIM. These features span from stable attributes to those that evolve at different rates over time. For example, static attributes like product size and facing allocation remain constant in a specific location, while metrics such as average sales over the past 7 days change gradually across few days. On the other hand, daily sales exhibit significant variations even within a single day. To enhance clarity, we categorize and summarize some of these features utilized in our current model in Table 1.

As illustrated in Table 1, our dataset includes store level, item level, day level, intra-day level, and engineered features. Intra-day features are those attributes that have the potential to change within a single day. These particular features undergo batch processing multiple times throughout the day to ensure that they are near-realtime when it comes to model scoring. Additionally, we incorporate engineered features by applying rolling aggregates or lags to day-level features with varying window sizes.

Considering the fact that Coles operates over 800 stores throughout Australia, each of which carries a substantial range of items, the amount of data we have to process to generate these features is substantial. To address this challenge, we utilize multiple Extract, Transform, and Load (ETL) jobs to populate various feature stores, which are reusable data repositories containing features that are readily available to build and score machine learning models. Subsequently, these features are accessed by the model at runtime. This process is illustrated in the Figure 1.

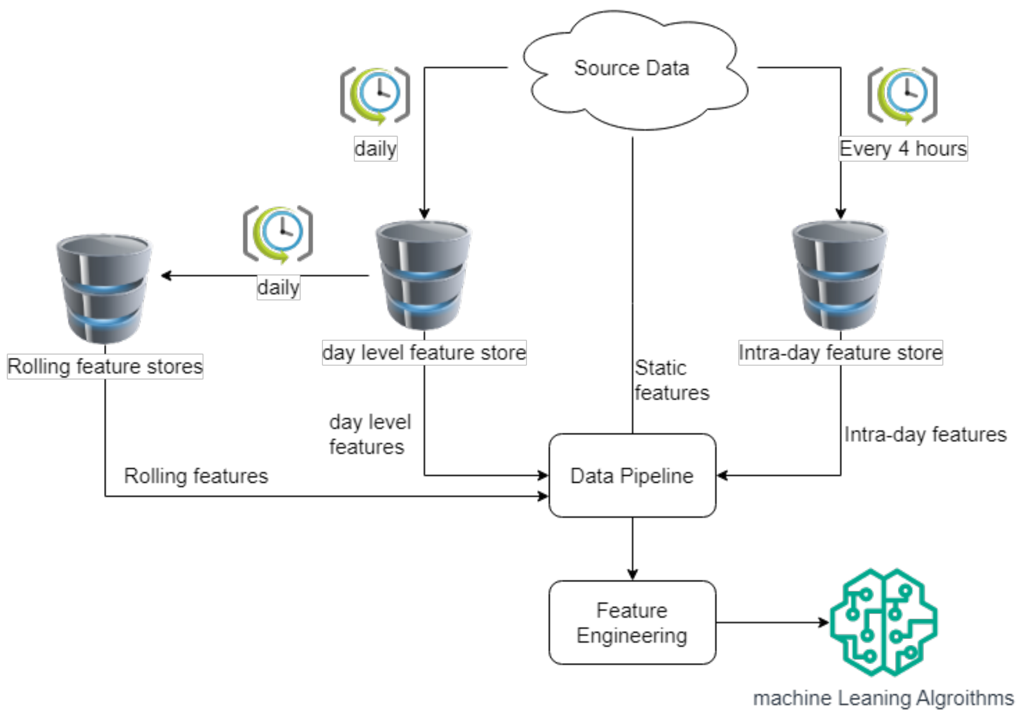


Figure 1: Data Pipeline of TIM

Machine Learning Architecture

As discussed above, TIM provides an estimate of current SOH position for each item and store combination as a probability distribution. The means of these probability distributions can be thought of as the point estimate of the current SOH positions, whereas the variance gives us a measure of uncertainty of these estimates. We achieve this functionality by leveraging a machine learning model architecture, which consists of tree based machine learning model and a Bayesian approach.

Learning Algorithms

The model is designed to output a probability distribution for each item-location, reflecting the current SOH positions. More specifically, the model generates a parametric distribution with unique parameters for every item-location pair. Drawing from our previous experience, the Gamma distribution has demonstrated its effectiveness in modeling items with positive SOH, offering a high degree of flexibility and favorable mathematical properties. Nevertheless, as some items may have zero SOH, the Gamma distribution on its own is unsuitable for capturing this discontinuity. To address this challenge, we adopt a hurdle model to represent the SOH. An illustrative output from TIM is shown in Figure 2.

Suppose that the SOH position for item i in location l is given by the random variable x_{il}. We assume that distribution of x_{il} is given by a hurdle Gamma model:

$$p(x_{il}|\theta_{il}, \alpha_{il}, \beta_{il}) = \begin{cases} \theta_{il} & \text{for } x = 0 \\ (1-\theta_{il})\Gamma(\alpha_{il}\beta_{il}), & \text{for } x > 0, \end{cases} \tag{1}$$

where θ_{il} is the probability of zero SOH, $\Gamma(\alpha_{il}, \beta_{il})$ is the probability density function (PDF) of Gamma distribution, and α_{il} and β_{il} are the shape and the rate parameters of the Gamma distribution respectively.

The mean of the Gamma distribution is $\nu_{ij} = \alpha_{il}/\beta_{il}$, and its variance $s_{il}^2 = \alpha_{il}/\beta_{il}^2$. Furthermore, ν_{ij} and s_{il}^2 are related to the mean μ_{il} and variance σ_{il}^2 of the mixture model by;

$$\nu_{il} = \frac{\mu_{il}}{1-\theta_{il}}, \tag{2}$$

$$s_{il}^2 = \frac{\mu_{il}^2 + \sigma_{il}^2}{1-\theta_{il}} - \mu_{il}^2. \tag{3}$$

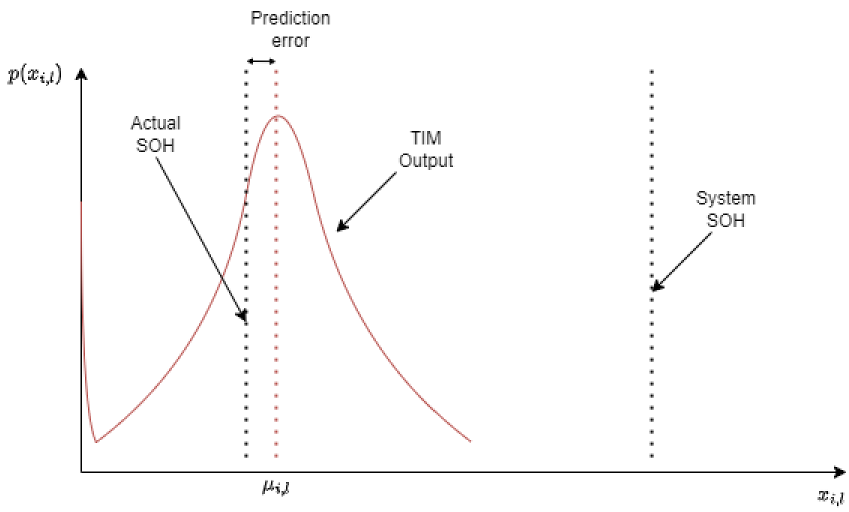


Figure 2: A sample SOH distribution from TIM

During the model training phase, we train models that can estimate μ_{il} and θ_{il} using historical data and subsequently derive σ_{il}^2 using a Bayesian approach. Figure 3 shows a graphical representation of this process.

As shown in Figure 3, first we train two machine learning models to estimate the μ_{il} and θ_{il}, $\forall l \in \mathbb{L}$, $i \in \mathbb{I}_l$, where $\mathbb{L}$ is the set of all locations and $\mathbb{I}_l$ is the set of all items in location l. Next, we use these in-conjunction with the data to estimate σ_{il}^2. During the scoring phase, the model uses the latest feature values to estimate these parameters to represent current SOH positions. In the following subsections, we explain how we estimate each of these parameters in more detail.

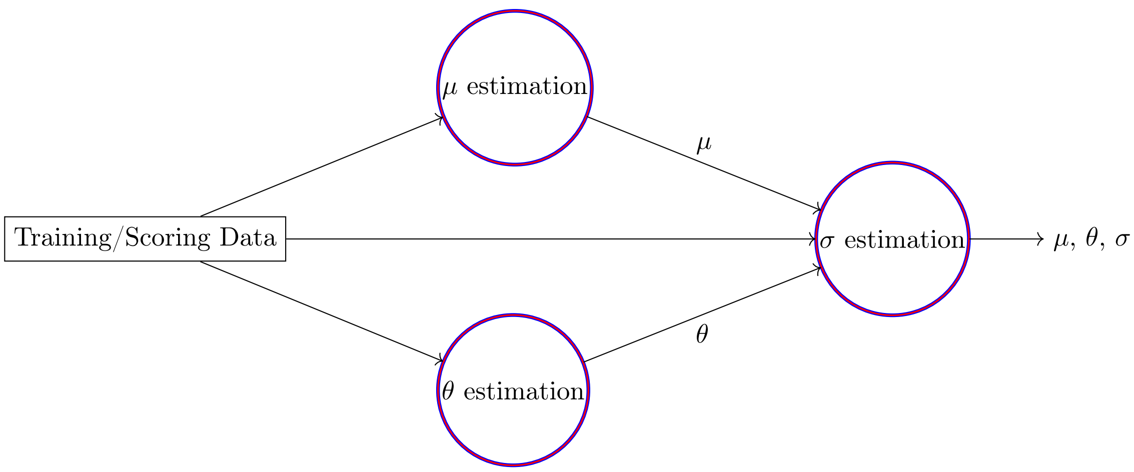


Figure 3: Machine Learning Model Architecture

Estimating the mean (μ_{il}) and the probability of zero SOH (θ_{il})

Our strategy involves initially estimating μ_{il} and θ_{il} using non-Bayesian machine learning models, and subsequently using those estimates to calculate σ_{il}^2 using a Bayesian approach. By using this approach, we benefit from the power of Bayesian modelling while using methods such as gradient boosting to estimate the mean (μ_{il}) and probability of zero SOH (θ_{il}). Typically, machine learning algorithms such as gradient boosted trees perform well to exploit non-linear relationships in the data and thus, the reason for the two stage approach where μ_{il} and θ_{il} are estimated in the first stage before passing on those values to a Bayesian model to estimate σ_{il}^2.

The dependent variable of the regression model that estimate the mean of the distribution, $\mu_{i,j}$, is the actual SOH as counted by store team members during the inventory counts. We use the features that were discussed in Section 2.1 for this model.

On the other hand, to estimate the probability of zero SOH, θ_{il}, we convert this regression model architecture into a binary classification framework, which identifies whether the observed SOH is zero or not. The resultant output of this binary classification model, which falls within the range of 0 to 1, serves as a reliable indicator for the probability of zero SOH. The target labels to train this classification model are established based on the presence or absence of zero SOH in our training data. Considering that a portion of our training data originates from the gap scan process outlined in Section 2.1, we observe a reasonable number of zero SOH counts within our training dataset. This results in a balanced distribution of training labels, essential for constructing this model. We use a set of features similar to that of the regression model and employ a gradient-boosted classification algorithm to estimate the probability of zero SOH.

Estimating the variance of SOH distribution (σ_{il}^2)

Once we have calculated the mean, μ_{il}, and the probability of zero SOH, θ_{il}, for the desired SOH distribution, our next step is to estimate its variance σ_{il}^2. We use a Bayesian approach to achieve this estimation. In particular, we use a variance regression where we model the logarithm of the variance of the Gamma density function (in the mixture distribution specified in equation 1) as a linear function of some predictor variables;

$$\log(s_{il}^2) = \beta_{0,il} + \beta_{1,il}\log(\nu_{il}) + \sum_{j=2}^{j=n} \beta_{j,il} x_{j,il}, \tag{4}$$

where ν_{il} and s_{il}^2 are the mean and variance of the Gamma part of the mixture distribution given in equation

1, and $x_{j,il}, j \in \{1, 2, \ldots, n\}$ is a set of n features selected from the feature space. In equation 4, the log function ensures that the variance remains positive. Also, note that the variance varies across different items and locations, as its depends on ν_{il}, $\beta_{1,il}$, and features $x_{j,il}, j \in \{1, 2, \ldots, n\}$. The parameters ν_{il} and s_{il}^2 can be calculated easily from the μ_{il} and θ_{il}, using equations 2 and 3. Due to these relationships, variance is also dependent on the θ_{il}, which is the output of the classification model. Using our business knowledge, we pick four features related to average inventory adjustments, average sales, and product size as the features $x_{j,il}$, $j = \{1, 2, 3, 4\}$ in equation 4.

To estimate the parameters $\beta_{0,il}, \beta_{1,il}, ..., \beta_{n,il}$, we employ a Bayesian approach. In particular, we assume log-normal pior distributions for coefficients $\beta_{0,il}$ and $\beta_{1,il}$, ensuring that they always remain positive, and determine the posterior distributions of parameters $\beta_{0,il}, \beta_{1,il}, ..., \beta_{n,il}$ by maximizing the log-likelihood of the mixture distribution function $p(x_{il}|\alpha_{il}, \beta_{il})$ given in Equation 1. To implement this effectively, we leverage Stan platform [8, 9] and apply variational Inference [10] whenever possible instead of Markov Chain Monte Carlo (MCMC) sampling, in order to achieve faster convergence .

2.2 Guided Counts

As the second part of the solution, we shift our focus towards rectifying inventory inaccuracies by establishing an effective counting policy that can uncover critical inventory inaccuracies in the presence of time and labor constraints. In each Coles store, there are approximately 15,000 unique products, with around 30% of these products showing incorrect inventory levels based on TIM prediction on any given day. However, it is impractical to recount all these items. Therefore, we need a method to prioritize which items need to be recounted based on their importance. This counting policy is known as Guided Counts.

Our objectives are to reduce unknown out-of-stocks, increase customer-facing availability, and enhance customer satisfaction. Essentially, we want to uncover products that the store believes are adequately stocked to fulfill the demand, but in reality are not. In order to address this, we need to consider the relative positions of the System SOH record, the probabilistic estimation of SOH as provided by TIM, and the forecasted demand until the next replenishment. For demand prediction, we use Coles proprietary demand forecasting engine known as Smarter Forecast. Similar to TIM, Smarter Forecast also generates a probabilistic estimation of demand, modelled as a Gamma random variable.

We define lost sales as the shortfall between the projected demand and the Stock on Hand (SOH) level. There are two measures of SOH: the system SOH and the TIM predicted SOH, leading to two distinct definitions of lost sales. Lost Sales according to System SOH is calculated as:

$$\text{Lost Sales according to System SOH} = \max(\text{Demand} - \text{System SOH}, 0). \tag{5}$$

Furthermore, Lost Sales according to TIM Predicted SOH is determined as:

$$\text{Lost Sales according to Pred SOH} = \max(\text{Demand} - \text{Pred SOH}, 0). \tag{6}$$

In addition to these definitions, we introduce a new cost measure called "Unknown Lost Sales". This measure quantifies the portion of lost sales that occurs without the store's knowledge. It helps distinguish between situations where (a) the store anticipates some lost sales and has placed a replenishment order and (b) cases where no action is taken due to lack of awareness. Mathematically, it is expressed as:

$$\text{Unknown Lost Sales} = \max(\max(\text{Demand} - \text{Pred SOH}, 0) - \max(\text{Demand} - \text{System SOH}, 0), 0). \tag{7}$$

We can categorize the products with incorrect inventory into one of the following 6 buckets based on the relative positions of the aforementioned attributes:

Scenario 1: Pred SOH<Demand<System SOH	**Scenario 2:** Pred SOH<System SOH<Demand	**Scenario 3:** System SOH<Pred SOH< Demand
System SOH = 50 Pred SOH = 30 Demand = 40 Lost sales (System SOH) = 0 Lost sales (Pred SOH) = 10 Unknown lost sales = 10	System SOH = 40 Pred SOH = 30 Demand = 50 Lost sales (System SOH) = 10 Lost sales (Pred SOH) = 20 Unknown lost sales = 10	System SOH = 30 Pred SOH = 40 Demand = 50 Lost sales (System SOH) = 20 Lost sales (Pred SOH) = 10 Unknown lost sales = 0
Scenario 4: Demand<System SOH<Pred SOH	**Scenario 5:** Demand<Pred SOH< System SOH	**Scenario 6:** System SOH<Demand<Pred SOH
System SOH = 40 Pred SOH = 50 Demand Forecast = 30 Lost sales (System SOH) = 0 Lost sales (Pred SOH) = 0 Unknown lost sales = 0	System SOH = 50 Pred SOH = 40 Demand Forecast = 30 Lost sales (System SOH) = 0 Lost sales (Pred SOH) = 0 Unknown lost sales = 0	System SOH = 30 Pred SOH = 50 Demand Forecast = 40 Lost sales (System SOH) = 10 Lost sales (Pred SOH) = 0 Unknown lost sales = 0

Table 2: Categorization based on relative positions of system SOH, predicted SOH and demand

In Scenario 1 depicted in Table 2, the store anticipates no lost sales as the system SOH is greater than the demand, yet TIM estimates there to be 10 units of lost sales as the predicted SOH is less than the demand.

Similarly, in scenario 2, the store is projecting 10 units of lost sales, while TIM based lost sales amounts to 20 units. In the other scenarios, even though there are discrepancies between the system SOH and the predicted SOH, we are not anticipating any immediate availability issues. Since we have limited time and remuneration budget per day to count and correct inventory positions, we rank the items in descending order of unknown lost sales and pick the top candidates.

Algorithm 1 Prioritise items to count

```
for each item in the store do
    Unknown Lost Sales = max(max(Demand - Pred SOH, 0) - max(Demand - System SOH, 0), 0)
end for
Rank items in descending order of unknown lost sales
Pick top n items
```

If a store team member physically counts the SOH for items recommended by the prioritization algorithm, we expect to find the true inventory level closer to the model's prediction. Once the inventory record is adjusted, actual SOH quantity aligns with the system SOH. The post adjustment SOH levels will subsequently trigger automated replenishment decisions ensuring that there is sufficient SOH to meet upcoming demand.

2.3 Accounting for prediction uncertainty

To calculate the expected values of unknown lost sales as described above;

1. we can utilize point estimates or the mean of the demand and SOH distributions for estimation purposes or,
2. alternatively, opt to use the full distributions which contain valuable information regarding spread or uncertainty. This approach is considered more robust as it grants us a well-rounded perspective on the uncertainties associated with our estimates.

Let Predicted SOH p and Demand d be two independent random variables with Gamma distributions:

$$p \sim \Gamma(\alpha_1, \beta_1), \tag{8}$$

$$d \sim \Gamma(\alpha_2, \beta_2). \tag{9}$$

Note that d and p are item-location level predictions, but we omit the subscripts in these equations for brevity.

The joint PDF is given by:

$$f_{p,d}(p,d) = f_p(p) \cdot f_d(d). \tag{10}$$

Lets substitute the Gamma PDFs in equation 10 with

$$f_{p,d}(p,d) = \frac{p^{\alpha_1 - 1} e^{-p/\beta_1}}{\beta_1^{\alpha_1}\Gamma(\alpha_1)} \cdot \frac{d^{\alpha_2 - 1} e^{-d/\beta_2}}{\beta_2^{\alpha_2}\Gamma(\alpha_2)}. \tag{11}$$

Now, to find $E[\max(d-p,0)]$, we integrate equation 11 over the regions where $d > p$:

$$E[\max(d-p,0)] = \int_0^\infty \int_0^d (d-p) f_{p,d}(p,d)\, dp\, dd. \tag{12}$$

Equation 12 can be further simplified to:

$$E[\max(d-p,0)] = \int_0^\infty \int_0^d (d-p) \left(\frac{p^{\alpha_1 - 1} e^{-p/\beta_1}}{\beta_1^{\alpha_1}\Gamma(\alpha_1)} \cdot \frac{d^{\alpha_2 - 1} e^{-d/\beta_2}}{\beta_2^{\alpha_2}\Gamma(\alpha_2)} \right) dp\, dd. \tag{13}$$

The integral in equation 13 provides the expected value of the difference when d exceeds p. However, it can be challenging to compute it analytically due to the fact that the distribution of the difference between two independent Gamma random variables does not conform to a straightforward well-known distribution. Hence, we opt for a Monte Carlo method where we generate numerous samples from both demand and current stock distributions. Subsequently, we determine $max(\text{Demand} - \text{Pred } SOH, 0)$ for each sample set and calculate the mean of those values in order to derive the expected outcome. Algorithm 2 presents the procedure in algorithmic form.

Algorithm 2 Prioritise Items to Count using Monte Carlo

for each item in the store **do**
    Generate p samples from the Demand distribution
    Generate p samples from the Pred SOH distribution
    for each set of the samples **do**
        Unknown Lost Sales = max(max(Demand - Pred SOH, 0) - max(Demand - System SOH, 0), 0)
    end for
    Take the mean of the unknown lost sales values for the item
end for
Rank items in descending order of mean unknown lost sales
Pick top n items

2.4 Extending to multi-period optimization problem

In the methodology discussed so far, our focus has been on prioritizing items based on the potential benefits of inspecting and correcting their inventories within the same day. This approach has certain limitations. Firstly, we are overlooking the possibility that the item doesn't necessarily have to be counted on the same day but may actually be more beneficial to be counted in upcoming days. For example, this heuristic solution may not be optimal when the lead times for replenishment are longer than a single day. Secondly, replenishment/inspection decisions in the current period can impact future inventory positions and thus, incur costs beyond the current period. There can be scenarios where counting an item tomorrow results in recovering the same amount of lost sales had it been counted today instead. For instance, consider an item with its next demand expected in 4 days and a replenishment lead time of 2 days. Correcting it's inventory tomorrow instead of today would still allow replenishment to take place in time for the next demand. In such cases, it might be more strategic to delay counting this particular item and instead, focus on another critical item that requires immediate attention, like an item that has an earlier demand or longer lead time. This demonstrates that ranking items solely based on short-term views is not always optimal. We can liken this situation to solving a multi-period Knapsack problem [7] where our capacity to count items daily is limited and our goal is to maximize recovered sales within a defined time frame.

In this section, we consider the problem of finding the optimal list of items to count on any given day as an optimization problem. For this purpose, we frame it as a Linear Programming (LP) problem. We define $x_{i,d}$ as a binary decision variable where

$$x_{i,d} = \begin{cases} 1 & \text{if item } i \text{ is counted on day } d, \\ 0 & \text{otherwise.} \end{cases} \tag{14}$$

The objective function of our optimization problem is to maximize the total benefit:

$$\sum_{i=1}^{n} \sum_{d=1}^{5} B_{i,d} \cdot x_{i,d}, \tag{15}$$

where $B_{i,d}$ represents the benefit of recounting item i on day d. We define the benefit $B_{i,d}$ of counting an item x_i on a particular day d_i, as the difference between the lost sales if the item was not counted at all and the lost sales if the item counted on d_i. The benefits associated with recounting each item on each day are represented in a matrix B of size $n \times 5$. For instance, for 3 items and 5 days, B takes the following form:

$$B = \begin{pmatrix} B_{1,1} & B_{1,2} & B_{1,3} & B_{1,4} & B_{1,5} \\ B_{2,1} & B_{2,2} & B_{2,3} & B_{2,4} & B_{2,5} \\ B_{3,1} & B_{3,2} & B_{3,3} & B_{3,4} & B_{3,5} \end{pmatrix}.$$

We use Coles automatic replenishment/ordering system to calculate the forecasted orders and post-replenishment SOH positions based on current SOH, demand, lead time, spacing, and display requirements. Assume we are at period $d-1$ and about to decide which products to inspect and correct inventory for on period d. I_d represents inventory position at the beginning of period d, D_d represents demand on period d, and R represents replenishment at the end of period d. The ordering system allows us to calculate the forecasted replenishment for each of the next 5 days, $R_{1,2,3,4,5}$, given current inventory level I_{d-1}, forecasted demand $D_{1,2,3,4,5}$, lead time, and other factors such as shelf spacing, display requirements, etc.

We calculate inventory available on period t as:

$$I_{i,d} = I_{i,d-1} + (R_{i,d-1} - S_{i,d-1}) \tag{16}$$

and the demand shortfall or lost sales for item i for the whole period as:

$$L_i = \sum_{d=1}^{5} D_{i,d} - I_{i,d}. \tag{17}$$

This is the lost sales we are expecting if the inventory position was not corrected. Similarly, we can recalculate the replenishment R', inventory position I' and ultimately lost sales L' if we corrected the inventory position by replacing the system SOH record, I_{d-1}, in the ordering system with the probabilistic prediction, I'_{d-1}, from TIM. Since we want to calculate the benefit of counting an item on each of the days, we repeat the process for days 1 through to 5, replacing the I with the TIM prediction for each day. The benefit of inspecting item i on day d is calculated as:

$$B_{i,d} = L'_{i,d} - L_i, \tag{18}$$

where $L'_{i,d}$ is the lost sales we expect to incur for item i over the period if inventory was corrected on day d and L_i is the lost sales we expect to incur if inventory was not corrected at all.

Additionally, the model includes the following constraints:

1. We limit each item to be inspected at most once, over the five day window:

$$\sum_{d=1}^{4} x_{i,d} \leq 1 \quad \forall i. \tag{19}$$

We define this constraint as a matrix with n rows (one for each item), with each row having a '1' in the columns corresponding to the five days for that item, and '0' elsewhere. This enforces that the sum of these variables is at most 1. The matrix representation of constraints for 3 items (each color represents an item) is as follows:

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_{1,1} \\ x_{1,2} \\ x_{1,3} \\ x_{1,4} \\ x_{2,1} \\ x_{2,2} \\ x_{2,3} \\ x_{2,4} \\ x_{3,1} \\ x_{3,2} \\ x_{3,3} \\ x_{3,4} \end{pmatrix} \leq \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.$$

2. Since there is limited time and resources available for inventory correction activities each day, we put a constraint on the number of items to be inspected: No more than L items to be allocated to any given day. This is mathematically represented as:

$$\sum_{i=1}^{n} x_{i,d} \leq L \quad \forall d. \tag{20}$$

We represent this constraint as a matrix with 5 rows (one for each day). Each of the rows has '1' s in the columns corresponding to all items for that specific day, ensuring the sum does not exceed L. The matrix representation of this constraint for a daily limit L and 3 items (each color represents an item) is:

$$
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
x_{1,1} \\ x_{1,2} \\ x_{1,3} \\ x_{1,4} \\ x_{2,1} \\ x_{2,2} \\ x_{2,3} \\ x_{2,4} \\ x_{3,1} \\ x_{3,2} \\ x_{3,3} \\ x_{3,4}
\end{pmatrix}
\leq
\begin{pmatrix}
L \\ L \\ L \\ L
\end{pmatrix}.
$$

We use the HiGHS package [11] in R to solve this problem. It is an interface to the High-Performance Software for Linear Optimization (HiGHS) solvers. It efficiently solves large-scale linear programming (LP) and mixed-integer programming (MIP) problems. The model is run everyday generating a store-specific prioritized list of items to count each morning. We present the efficiency gains and cost savings from implementing the prioritization algorithm in the next section.

3 Results

3.1 TIM Model Performance

In this section, we delve into the evaluation of the TIM model's performance through a variety of metrics. During the model building phase, we partition the dataset into training, evaluation, and test sets. The test set remains withheld until the end for validation purposes. All metrics discussed in this section are based on analyses conducted using this reserved test dataset, which comprises approximately 12 million inventory count instances from various Coles stores across Australia.

It is important to understand that the TIM model aims to provide precise estimates of inventory positions, especially when the system SOH may not be entirely reliable for various items. As a result, it is crucial for

TIM estimates to be more accurate than those derived from the system SOH, ultimately leading to fewer errors. Therefore, where applicable, we will also present corresponding metrics for system SOH in order to facilitate comparison.

	Metric	Model	System
Regression Model	RMSE	4.8	6.5
	MAE	1.5	2.1
	R^2	0.87	0.8
Classification Model	Accuracy	0.8	0.75
	Precision	0.94	0.95
	Recall	0.77	0.65
	AUC	0.91	0.78

Table 3: TIM Performance

Performance of the regression and classification models

In this section, we present the performance of both the regression and classification components within the TIM model. We use RMSE, Mean Absolute Error (MAE), and R-squared to evaluate the regression part of the TIM model, which are standard metrics used to evaluate regression models [12]. The RMSE is also the loss function used within the training and serves as a key indicator of prediction error. On the other hand, MAE is particularly useful in scenarios where outliers exist within the dataset. Both RMSE and MAE are measures of error that should be minimized to enhance model performance. Furthermore, R-squared (R^2) is an indicator of how effectively the model responds to variations in the predicted variable. It is worth noting that R^2 values range from 0 to 1, with higher values indicating better model performance.

When evaluating the classification model, we primarily focus on the Area Under the ROC Curve (AUC). The AUC is equivalent to the probability that a randomly chosen positive example has a higher score than a randomly drawn negative example [10]. AUC serves as a comprehensive metric that evaluates performance across all potential classification thresholds. The AUC value falls within the range of 0 to 1, with a perfect model achieving an AUC of 1.0 when its predictions are entirely accurate. Furthermore, we set a threshold of 0.6 and compute other relevant metrics such as accuracy, precision, and recall. Higher values for these metrics indicate superior performance levels.

Table 3 shows the summary of the performance analysis when model is evaluated on the test dataset. In Table 3, we show the relevant metrics comparing the model outputs with actual SOH and the system versus actual SOH positions. As shown in Table 3, it is apparent that the TIM model's estimated SOH positions exhibit higher accuracy levels by reducing errors within inventory records by over 25%. Furthermore, its capability to effectively identify out-of-stock scenarios (i.e., zero SOH) is prominently highlighted.

TIM Bayesian Model Performance

In contrast to the straightforward evaluation process of regression and classification models, assessing a Bayesian model requires special attention. Various methods exist in literature for evaluating the performance of probabilistic forecasts such as Continuous Ranked Probability Scores (CRPS) [13], Brier Score [14], Pinball Loss [15], and calibration plots. For our evaluation, we use CRPS, Pinball Loss, and calibration plots. CRPS offers several benefits for evaluating probabilistic forecasts. Importantly, it does not depend on distributional assumptions, making it dependable even when our chosen distribution is incorrect. Additionally, it can be viewed as an extension of MAE for point forecasts, simplifying the interpretation. On the other hand, the Pinball Loss is a loss function used in quantile regression. Ideally, we want the qth quantile to be less than SOH q% of the time. We measure this by averaging the pinball loss over a range of quantiles. Table 4 summarises the average CRPS and average pinball loss that we have observed when we score the model for the test set. Thus, it is evident that the Bayesian component of the TIM model captures the uncertainties around the SOH estimation successfully.

Average CRPS	**Average Pinball Loss**
1.64	0.35

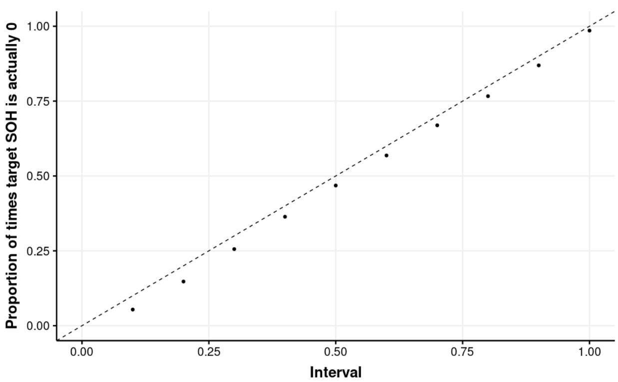
Table 4: TIM Bayesian Model Performance

In the context of probabilistic models, it is important for it to be well calibrated. If a model is well calibrated, then approximately 90% of actual targets should fall within the 90% prediction interval and should hold true for not just 90%, but for other intervals as well. Calibration plots can be used to graphically visualise how well the model achieves this. First, the range of predicted probabilities is divided into bins, often equally spaced. For each bin, the plot shows the fraction of events that actually happened. In a perfectly calibrated model, predictions match observed frequencies exactly, represented by points lying on a diagonal line $y = x$. Points below this line indicate overconfidence, while points above indicate under-confidence.

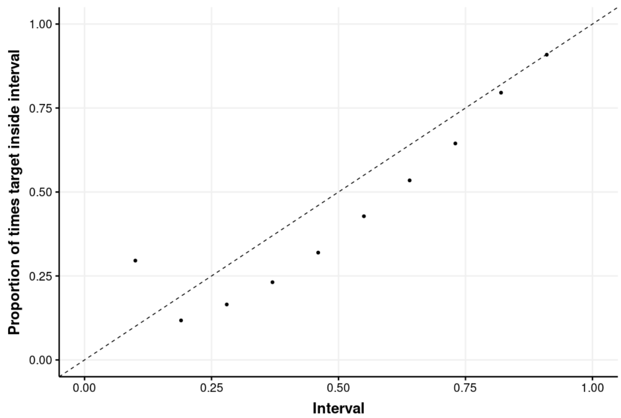
Given the presence of a discontinuity occurring at 0 (due to the mixture model), displaying the calibration plot for both zero and non-zero segments of SOH distribution in a single plot is not straightforward. To address this challenge, we use two separate calibration plots for these two scenarios as shown in Figure 4.

Figure 4a shows the calibration plot for the probability of zero SOH, which is the output of the TIM classification model. This visualization illustrates the proportion of data points originating from the test dataset that exhibit precisely zero SOH across various intervals of probability related to zero SOH. It is evident from

Figure 4a that our classification model maintains good calibration, aligning with our observations detailed in Section 3.1. The calibration plot shown in Figure 4b illustrates the distribution of non-zero SOH. The plot is constructed by focusing on probabilities that exceed the probability of zero SOH for each data point. We recognize that there is room for improvement in this calibration plot. Nevertheless, it offers valuable insights into the performance of the non-zero segment of the TIM's output SOH probability distribution. As illustrated in Figure 4b, despite displaying some overconfidence, the model demonstrates satisfactory performance.



(a) Calibration plot for Probability of Zero SOH



(b) Calibration plot for non-zero SOH distribution

Figure 4: Calibration plots

3.2 Performance of Guided Counts Optimization Algorithms

To compare the effectiveness of the optimal solution with the myopic heuristic-based approach, we conducted a simulation for a single store with a daily limit of 100 items. In the heuristic method, we simply selected the top 100 items based on their benefits for that specific day. Conversely, the optimal solution involves solving an optimization problem to maximize the overall benefit over a span of 5 days.

Figure 5 shows the benefit (lost sales recovered in units) from each solution. We observe that the heuristic solution yields a greater benefit on the first day as it uses a greedy approach. However, it falls short in the subsequent days compared to the optimal solution, as it fails to account for costs and benefits beyond the current day. Ultimately, the optimal solution outperforms the heuristic method by delivering a higher total benefit over the course of 5 days.

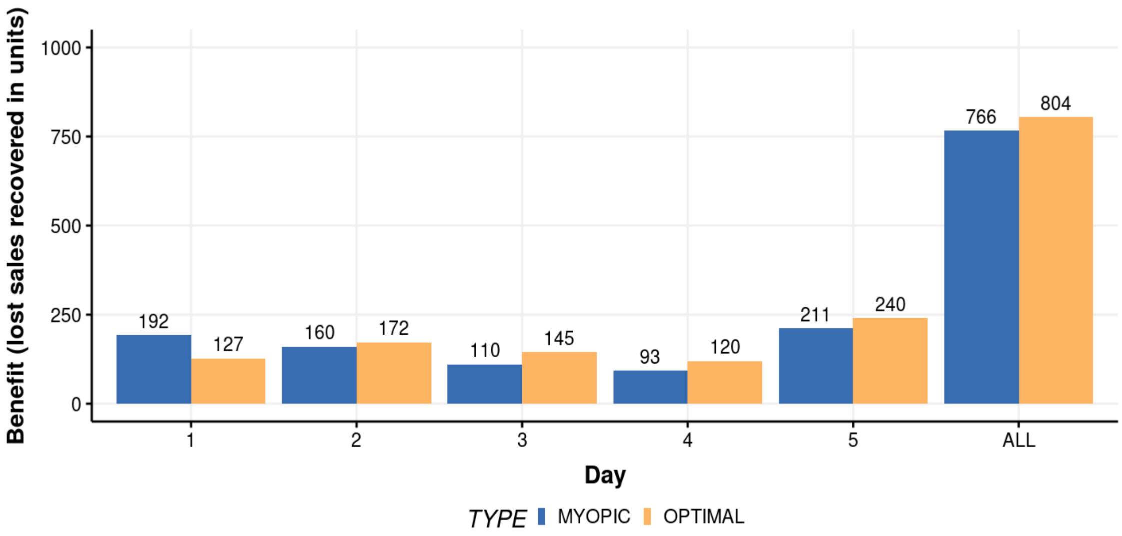


Figure 5: Comparison of Myopic (heuristic) vs. optimal solution

3.3 Benefits Realized from Implementing TIM and Guided Counts In Practice

The implementation of TIM and Guided Counts at Coles has had a profound impact on organizational efficiency and performance. Key outcomes include:

1. **Improved Inventory Accuracy:** The adoption of TIM has led to a significant reduction in inventory errors, with more than 20% improvement in the RMSE of inventory estimates across all product categories. This accuracy translates into better replenishment decisions, reducing both overstock and stock-out situations.

2. **Increased Efficiency of Stock Counts:** The guided counts approach has resulted in a 50% increase in the accuracy of items selected for recounts as compared to traditional methods such as cycle counting or gap scans. Gap scans can take between 2-3 hours based on store size and has a 25% accuracy on average. In comparison, guided counts are a targeted list of products that are highly likely to have incorrect SOH as informed by TIM predictions and the downstream optimisation algorithm. This list takes roughly 1 hour to complete and has 70% accuracy on average. This efficiency gain means that labour efforts are more effectively utilized, targeting items that are most likely to have inaccurate SOH positions.

3. **Efficient inventory corrections with reduced labor efforts:** By targeting the physical inventory counts to where they are most needed, TIM and Guided Counts allow us to efficiently correct more inventories with minimal labor effort. Figure 7 illustrates the average monthly inventory counts and average monthly adjustments per store over time, presented as a time-series of monthly granularity. It is

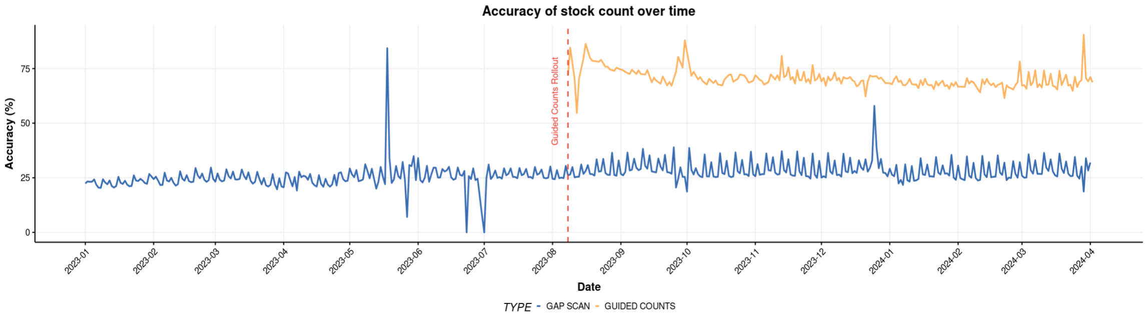


Figure 6: Comparison of gap scan and guided counts accuracy

important to note that both metrics displayed on the y-axis of Figure 7 have been normalized by dividing them by their respective maximum values. The x-axis denotes a timeline spanning from month 1 to month 29. The implementation of our solution commenced in month 11 and concluded in all Australian stores by month 22. As can be clearly seen from Figure 7, our solution is shown to correct more inventories per store while utilizing fewer labor resources.

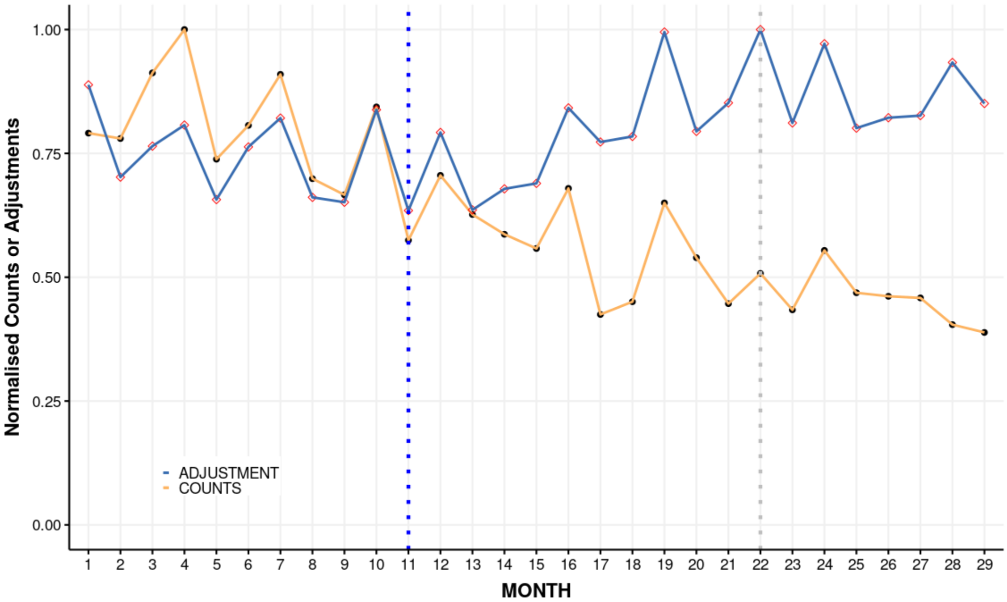


Figure 7: Normalised number of Inventory counts and adjustments across months

4. **Cost Savings:** The operational efficiencies and improvements in inventory management have culminated in substantial cost savings for Coles. The guided counts alone are estimated to have brought down labor costs for stock counting by as much as 57% annually, highlighting the significant financial impact of this project. Additionally, it has already resulted in significantly lower Out of Stock (OOS) events leading to a reduction in lost sales.

4 Conclusion

The presence of inaccuracies in retail inventory records has been a persistent challenge for retailers. This paper introduces a data science-driven approach to inventory management that has been successfully implemented at Coles Supermarkets, marking a substantial advancement in store operations related to inventory management. The solution includes TIM, a machine learning-driven model that accurately estimates current inventory levels and Guided Counts, a robust optimization algorithm to identify items that offer the greatest potential benefits through physical counting. In contrast to traditional inventory counting methods, this approach effectively captures the dynamic nature of retail environments and results in significantly more precise inventory adjustments with equal or reduced labor inputs. By providing accurate predictions of inventory positions and prioritizing items for counting, TIM and Guided Counts together contributed to a 20% reduction in inventory errors, a 50% increase in corrections made during counts, and a 57% reduction in labor costs for inventory corrective activities. The adoption of these innovative techniques highlights the importance of developing solutions that are both analytically rigorous and operationally impactful.

Furthermore, the proposed method has contributed significantly to Coles' sustainability efforts. More accurate inventory management reduces unnecessary stock and waste, directly supporting Coles' sustainability goals. By ensuring that stock levels closely match actual demand, TIM helps minimize the environmental impact associated with overproduction and disposal of unsold goods. This responsible inventory management strengthens the entire supply chain, making it more efficient and environmentally friendly, and its positive impacts extend beyond Coles itself to its suppliers and the broader Australian community.

References

[1] N. DeHoratius and A. Raman, "Inventory Record Inaccuracy: An Empirical analysis," *Management science*, vol. 54, pp. 627–641, 4 2008.

[2] Y. Kang and S. B. Gershwin, "Information inaccuracy in inventory systems: stock loss and stockout," *IIE transactions*, vol. 37, pp. 843–859, 9 2005.

[3] A. Beck and C. Peacock, *New loss Prevention: Redefining shrinkage Management.* 9 2009.

[4] B. C. Hardgrave, J. A. Aloysius, and S. Goyal, "RFID-Enabled Visibility and Retail Inventory Record Inaccuracy: Experiments in the field," *Production and operations management*, vol. 22, pp. 843–856, 7 2013.

[5] A. Cook, "The rising cost of retail theft? Trends in steal from retail to June 2023 (Bureau Brief No. 168). NSW Bureau of Crime Statistics and Research," 2023.

[6] Y. Rekik, A. Syntetos, and C. Glock, "Inventory inaccuracy in retailing: Does it matter," *URL www. ecr-shrink. group. com. ECR Shrinkage and OSA Group*, 2019.

[7] B. H. Faaland, "The multiperiod knapsack problem," *Operations Research*, vol. 29, no. 3, pp. 612–616, 1981.

[8] B. Carpenter, A. Gelman, M. D. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. Brubaker, J. Guo, P. Li, and A. Riddell, "Stan: A probabilistic programming language," *Journal of statistical software*, vol. 76, no. 1, 2017.

[9] Stan Development Team, "RStan: the R interface to Stan," 2018. R package version 2.17.3.

[10] K. P. Murphy, *Machine learning: a probabilistic perspective.* MIT press, 2012.

[11] F. Schwendinger and D. Schumacher, *highs: 'HiGHS' Optimization Solver*, 2023. R package version 0.1-10.

[12] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer, 2009.

[13] T. Gneiting, F. Balabdaoui, and A. E. Raftery, "Probabilistic forecasts, calibration and sharpness," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 69, no. 2, pp. 243–268, 2007.

[14] G. W. Brier, "Verification of forecasts expressed in terms of probability," *Monthly weather review*, vol. 78, no. 1, pp. 1–3, 1950.

[15] I. Steinwart and A. Christmann, "Estimating conditional quantiles with the help of the pinball loss," 2011.