ASOS CUSTOMER LIFETIME VALUE PREDICTION

IDS 594 – DATA SCIENCE FOR ONLINE CUSTOMER ANALYTICS

FALL 2017 – TERM PROJECT

BHARGAVI SURENDRAN (UIN: 664072122)
LAAVANYA GANESH (UIN: 654324917)
MUGHUNDHAN CHANDRASEKAR (UIN:662895444)
SAVITHA TAMRAPARNI-VITTAL (UIN: 662405004)
SHAKUNTHALA NARAYANA SWAMY (UIN: 657499860)

1.1 Business Problem

The paper focuses on two main business problems:

- **Customer Lifetime Value Prediction**
  - CLTV prediction is an important problem in e-commerce where an accurate estimate of future value allows retailers to effectively allocate marketing spend, identify & nurture high value customers & mitigate exposure to losses. Free shipping is vital in online clothing retail because customers need to try on items without being charged. If a company can't recoup delivery costs for returned items, customers can easily have negative lifetime value. For this reason, the Customer Lifetime Value (CLTV) problem is particularly important in online clothing retail.
  - CLTV is defined as the sales, net of returns, of a customer over a one-year period.
- **Churn**
  - A customer is defined as churned if they have not placed an order in the past year.

The objective is to improve three key business metrics:

a. The average customer shopping frequency
b. The average order size
c. The customer churn rate

1.2 Nature of the Industry

The state of the art in this industrial domain of CLTV prediction uses large numbers of handcrafted features and ensemble regressors to forecast value, predict churn & evaluate customer loyalty. These handcrafted features & classifiers (especially Random Forest regressors), are seen to perform well in highly stochastic CLTV prediction problems. Recently, domains including language, vision and speech have shown dramatic advances by replacing handcrafted features with features that are learned automatically from data.

1.3 Company position in the Industry

As of July 6, 2017, there were 12.5 million active customers and the product catalogue contained more than 85,000 items. Products are shipped to 240 countries and territories and the annual revenue for 2016 was £1.4B, making ASOS one of Europe's largest pure play online retailers. An integral element of the business model at ASOS is free delivery and returns. Products are shipped to 240 countries and territories and the annual revenue for 2016 was £1.4B, making ASOS one of Europe's largest pure play online retailers. ASOS provides fashion advice, video content and magazines.

2.1 Data Sampling

Neural embeddings are a technique pioneered in Natural Language Processing (NLP) for learning distributed representations of Words. Embeddings are compact, dense representations that encapsulate similarity. Typically, the data is a sequence and the context is a fixed-length window that is passed along the sequence. The model learns that objects frequently occurring in the same context are similar and will have embeddings with high cosine similarity. The embedding model used for data sampling is SkipGram with Negative Sampling (SGNS). SGNS are used for item-level embeddings in item-based collaborative filtering called item2vec. A basket of items that were purchased together are used for analysis. SGNS was used to generate a set of product embeddings called prod2vec, by mining receipts from email data. For each customer, a sequence of products was built (with arbitrary ordering for those bought together) and then a context window was run over it. The goal was to predict products that are co-purchased by a single customer within a given number of purchases. A hierarchical extension called bagged-prod2vec that groups products appearing together in a single email was also proposed. A variant of SGNS was used to predict the next app a mobile phone user would open. The key idea is to consider sequences of app usage within mobile sessions. The data had associated time stamps, which were used to modify the original model. Instead of including every pairwise set of apps within the context, the selection probability was controlled by a Gaussian sampling scheme based on the inter-open duration. Applying SGNC requires defining three key design decisions:

- How to define a context
- How to generate pairs of customers (C_{in}, C_{out}) from within the context
- How to generate negative samples

2.2 Potential problems/biases with data

The ASOS CLTV model uses a rich set of features to predict the net spend of customers over the next 12 months by training a random forest regressor on historic data. One of the major challenges of predicting CLTV is the unusual distribution of the target variable. A large percentage of customers have a CLTV of zero. Of the customers with greater than zero CLTV, the values differ by several orders of magnitude. To manage this problem, CLTV percentiles are explicitly modeled using a random forest regressor. Having predicted percentiles, the outputs are then mapped back to real value ranges for use in downstream tasks.

2.3 Main challenges in getting/handling data

- Handcrafted features introduce a human bottleneck, can be difficult to maintain and often fail to utilize the full richness of the data.
- It is difficult to incorporate the vast majority of the customer data available to modern e-commerce companies into the RFM/BYTD framework to incorporate automatically learned or highly sparse features.
- The ASOS e-commerce application is estimated to have close to 85,000 products as of July 6, 2017. Since the number of products are very high, it becomes more difficult to predict the products that are co-purchased by a single customer within a specified number of purchases. This increases the number of customer interactions. Thus, it becomes more cumbersome to implement the sparse matrix.

3.1 Modeling Methodology

The methodology involves two distinct approaches:

- Unsupervised neural embeddings are learnt using customer product views. Once learnt the embeddings are added to the feature set of the Random forest
- Hybrid model combining logistic regression and a deep feedforward neural network

Deep feedforward neural networks accept all continuous-valued features and dense embeddings of categorical features, as inputs. Rectified Linear Units (ReLU) activations are used in the hidden units and sigmoid activation is used in the output unit. The output of the neural network's final hidden layer is used alongside all continuous-valued features and sparse categorical features, as input. This is akin to skip connections in neural networks, with the inputs connected directly to the output instead of the next hidden layer. Training on the neural network part of the models is done via mini-batch stochastic gradient descent with Adagrad optimizer, with change of weights back propagated to all layers of the network. Regularization is applied on the logistic regression part of the hybrid models via the use of FTRL-Proximal algorithm.

The performance and scalability of the two models with different architectures is evaluated and compared with other machine learning techniques. The methodology was experimented with neural networks withtwo, three and four hidden layers, each with different combinations of number of neurons. For each neuron architecture, models were trained using a subset of customer data (the training set). The following details were recorded:

- The maximum AUC achieved when evaluating on a separate subset of customer data (the test set)
- The time taken to complete a pre-specified number of training steps.

The training/testing were repeated multiple times for each architecture to obtain an estimate on the maximum AUC achieved and the training time. All training/testing was implemented using the TensorFlow library on a Tesla K80 GPU machine. Introducing bypass connections in the hybrid models improves the predictive performance compared to a deep feed-forward neural network with the same architecture. The advantages and disadvantages of the methodology are described in the below section.

The unsupervised neural embeddings of customers are learnt directly because products are short-lived. The model has two large weight matrices, W_{in} and W_{out} that learn distributed representations of customers. The output of the model is W_{in} and after training, each row of W_{in} is the vector representation of a customer in the embedding space. The inputs to the model are pairs of customers (C_{in},C_{out}) and the loss function is the probability of observing the output customer C_{out} given C_{in}:

$$E = -\log P(C_{\text{out}}|C_{\text{in}}) = \frac{\exp\left(\mathbf{v}'^{T}_{\text{out}}\mathbf{v}_{\text{in}}\right)}{\sum_{j=1}^{|C|}\exp\left(\mathbf{v}'^{T}_{j}\mathbf{v}_{\text{in}}\right)},$$

where v'j represents the j^{th} row of W_{out}, v'_{out} represents the row of W_{out} that corresponds to the customer C_{out} and v_{in} represents the row of W_{in} that corresponds to the customer C_{in} and |C| is the total number of customers. This method is explained in Fig. 1

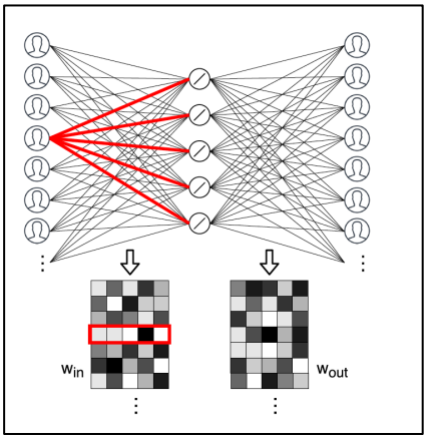


Fig 1: Neural network architecture and matrix representation of the network's input/output weights (embedding representation) in the Skipgram model on customer embeddings

The above methodology is borrowed from NLP where the context is usually a sliding window over word sequences within a document. The word at the center of the window is the input word and ($word_{in}$, $word_{out}$) pairs are formed with every other word in the context. The negative samples are drawn from a modified unigram distribution of the training corpus. Applying this concept, each product in

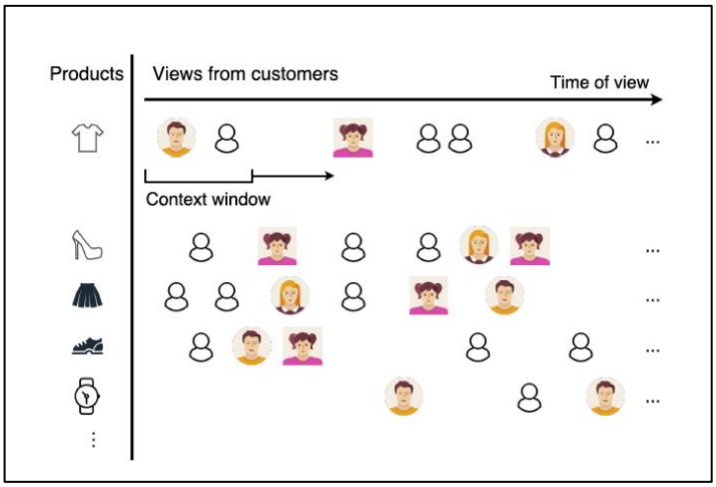


Fig. 2: Customer pair generation in the skip-gram model based on product views

the catalogue is associated with a sequence of customer views. A sliding context window is then passed over the sequence of customers. For every position of the context window, the central customer is used as C_{in} and all other customers in the window are used to form (C_{in}, C_{out}) pairs. In this way, a window of length three containing (C_1,C_2,C_3) would generate customer pairs (C_2,C_1) and (C_2,C_3). It was empirically found that a window of length 11 worked well. The above application of the NLP concept is described in the Fig. 2

The embedding algorithm begins by randomly initializing W_{in} and W_{out}. Then, for each customer pair (C_{in},C_{out}) with embedded representations (v_{in}, v_{out}), k negative customer samples C_{neg} are drawn. A.er the forward pass, k + 1 rows of W_{out} are updated via gradient descent, using backpropagation:

$$\mathbf{v}'^{\text{new}}_j = \begin{cases} \mathbf{v}'^{\text{old}}_j - \eta\,(\sigma(\mathbf{v}'^{T}_j \mathbf{v}_{\text{in}}) - t_j)\,\mathbf{v}_{\text{in}} & \forall j:\ C_j \in C_{\text{out}} \cup C_{\text{neg}} \\ \mathbf{v}'^{\text{old}}_j & \text{otherwise} \end{cases},$$

where η is the update rate, σ is the logistic sigmoid and tj = 1 if Cin = Cout and 0 otherwise.
Finally, only one row of Win, corresponding to vin is updated according to:

$$\mathbf{v}^{\text{new}}_{\text{in}} = \mathbf{v}^{\text{old}}_{\text{in}} - \eta \sum_{j:\, C_j \in C_{\text{out}} \cup C_{\text{neg}}} (\sigma(\mathbf{v}'^{T}_j \mathbf{v}_{\text{in}}) - t_j)\,\mathbf{v}'_j$$

3.2 Advantages and Disadvantages with the Modeling Methodology

Advantages:

- The novel customer embeddings are shown to improve CLTV prediction performance significantly as compared to the benchmark classifier.

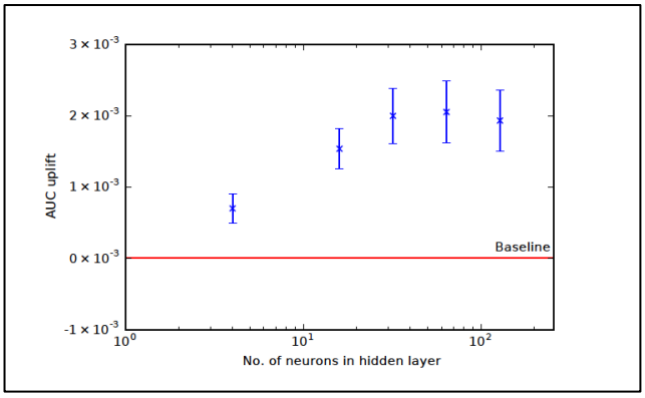
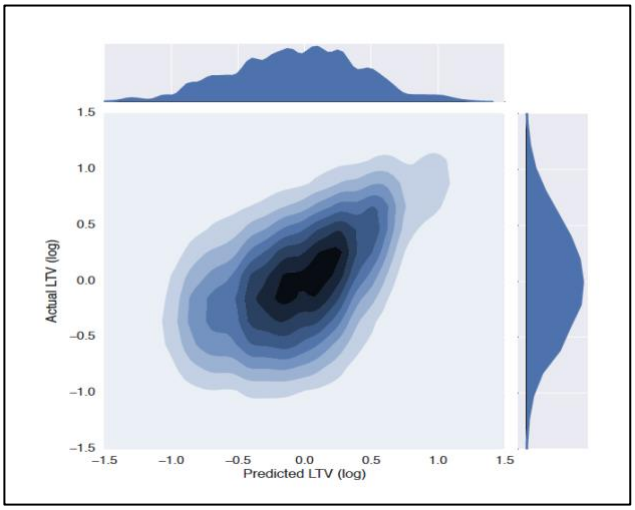


Fig 3: Uplift in the area under the ROC curve achieved on random test sets of 20,000 customers with product view-based embeddings against number of neurons in the hidden layer of the Skipgram model.



- Performing calibration is useful for the following reasons:
  1. The model becomes more robust to the existence of outliers
  2. The predictions obtained, when aggregated over a set of customers, match the true values accurately

- From the results obtained, range and density obtained the predicted CLTV matches the actual CLTV.

Fig 4: Predicted CLTV against actual CLTV. The distribution of the prediction and the actual CLTV are similar in log scale. The central plot shows the fit between the predictions and the actual values, which have a Spearman rank-order correlation coefficient of 0.46

- From the results obtained, the predicted probabilities match closely with the actual probabilities in the case of churn prediction

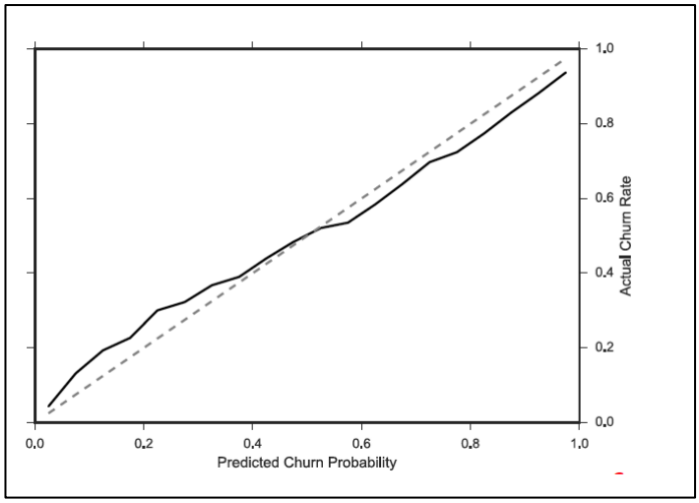


Fig 5: Churn prediction density and match between predicted probabilities and actual probabilities versus the optimal calibration.

- Despite being more difficult to interpret, learnt features avoid the resource intensive task of constructing features manually from raw data and have been shown to outperform the best handcrafted features in the domains of speech, vision and natural language.
- A comparison of the maximum AUC achieved by DNNs to the hybrid logistic and DNN models on a test set of 50,000 customers shows a statistically significant uplift at least 1.4 * 10^{-3} in every configuration.

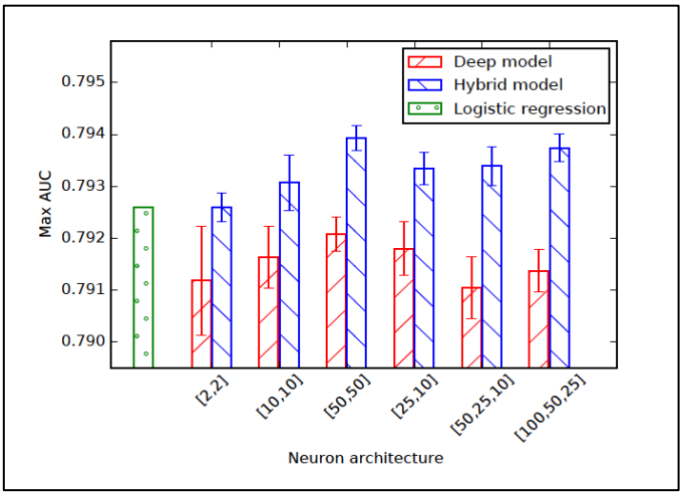


Fig 6: Maximum area under the ROC curve achieved on a test set of 50,000 customers in deep feed-forward neural networks and hybrid models with different numbers of hidden layer neurons.

Disadvantages:

- Incorporating embeddings into long-term prediction models is challenging because, the features are not easily identifiable.
- When components of embedded customer vectors are used as features for forecasting, each column represents a component of the vector representation of customers. The vector components randomly permute between train and testing time. Applying the learned parameters from training time directly to the embeddings in test time will not work as they are no longer attached to the correct component of the embedded vectors.

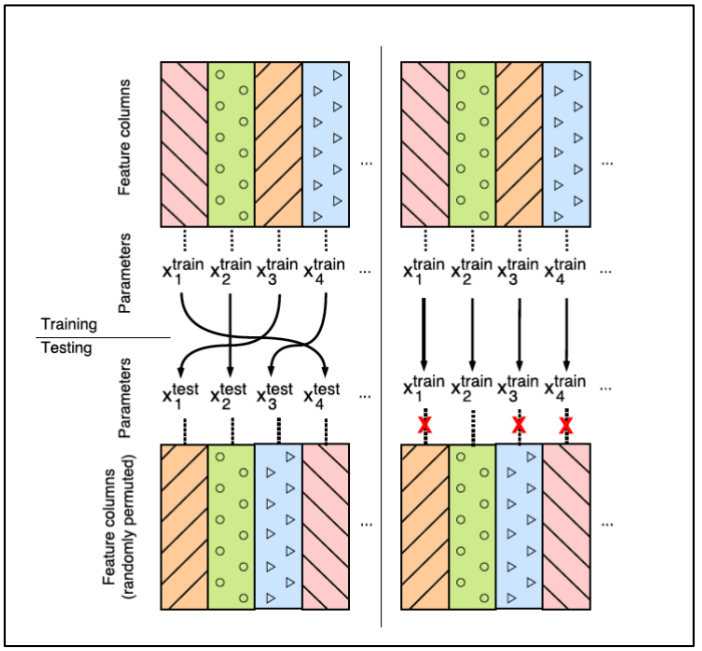
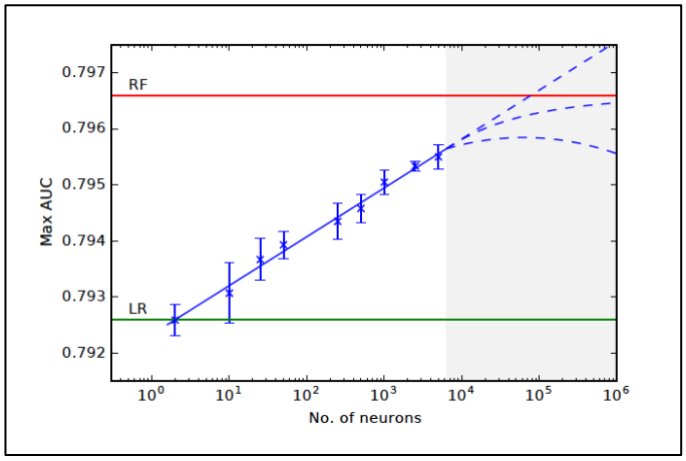


Fig 7: Illustration of the challenges of using the components of embedded customer vectors as features for forecasting

- Intuitively, high-value customers tend to browse products of higher value. By contrast, lower value customers will tend to appear together in product sequences during sales periods or for products that are priced below the market. This information is difficult to incorporate into the model using hand-crafted features as the number of sequences of product view grows combinatorically.



There is a linear relationship between the maximum AUC achieved on the same test set of customers and the number of neurons in each hidden layer in logarithmic scale.

- A hybrid model with a small number of neurons in each hidden layer already gives statistically significant improvement in maximum AUC achieved compared with logistic regression, but within the range of experiments it is difficult to exceed the performance of the RF model.

Fig 8: Maximum Area Under the ROC Curve (AUC) achieved on a test set of 50,000 customers in hybrid models against number of neurons in the hidden layers (in log scale). The bottom (green) and top (red) horizontal line represent the maximum AUC achieved by a vanilla logistic regression model (LR) and our random forest model (RF) on the same set of customers. The dashed lines in the shaded region represent different forecast scenarios for larger architectures.

- While the experiments suggest it is possible for the hybrid model, which incorporates a deep neural network, to outperform the calibrated RF model in churn classification, the monetary cost required to perform such training outweighs the benefit of gain in performance, making it impractical on cost grounds. The case is similar for CLTV prediction, in which a hybrid model on handcrafted features can achieve better performance than the deployed random forest model, though with a much higher cost that is not commercially viable.

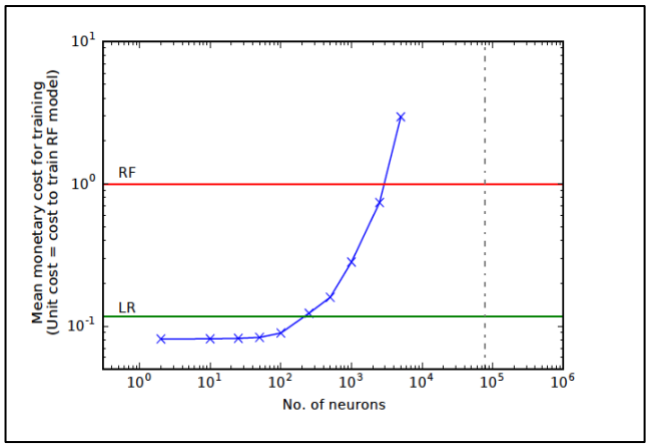


Fig 9: Mean monetary cost to train hybrid models on a training set of 100,000 customers against the number of neurons in the hidden layers (both in log scale). The bottom (green) and top (red) horizontal line represents the mean cost to train a vanilla logistic regression model (LR) and our RF model on the same set of customers. The vertical dash dotted (grey) line represents the estimated number of neurons in each layer required for a two-hidden layer hybrid model to out-perform our random forest model.

3.3 Features Used

Random forest feature importance was used to rank the 132 handcrafted features of which the top ones were as listed in Figure 8.

Feature Name	Importance
Number of orders	0.206
Standard deviation of the order dates	0.115
Number of session in the last quarter	0.114
Country	0.064
Number of items from new collection	0.055
Number of items kept	0.049
Net sales	0.039
Days between first and last session	0.039
Number of sessions	0.035
Customer tenure	0.033
Total number of items ordered	0.025
Days since last order	0.021
Days since last session	0.019
Standard deviation of the session dates	0.018
Orders in last quarter	0.016
Age	0.014
Average date of order	0.009
Total ordered value	0.008
Number of products viewed	0.007
Days since first order in last year	0.006
Average session date	0.006
Number of sessions in previous quarter	0.005

Fig 10: Top Features

3.4 Predictive Features

As seen from Fig 8, the number of orders, the number of sessions in the last quarter and the nationality of the customer were very important features for CLTV prediction. It was not expected that the number of items purchased from the new collection would be one of the most relevant features. This is because newness is a major consideration for high value fashion customers.

4. Learnings from Data

The ASOS model incorporates features from the full spectrum of customer information available at ASOS. The following are four broad classes of data:

a. Customer Demographics
b. Purchase History
c. Returns History
d. Web and App Session Logs

Fig. 9 shows the feature importance breakdown by the broad classes of data. By far the largest and richest of these classes are the session logs.

Data class	Overall Importance
Customer demographics	0.078
Purchases history	0.600
Returns history	0.017
Web/app session logs	0.345

Fig 11: Feature importance breakdown by the broad classes of data

5. Model Deployment

A high-level overview of the CLTV system is shown in Fig 12. The solid arrows represent the flow of data, and the dashed arrows represents interaction between stakeholders and systems/data. Customer data is collected and pre-processed by the data warehouse and stored on Microsoft Azure blob storage. The processed data is used to generate handcrafted features in Spark clusters, with web/app sessions additionally used to produce experimental customer embeddings in Tensorflow. The handcrafted features and customer embeddings are then fed through the machine learning pipeline on Spark, which trains calibrated random forests for churn classification and CLTV regression. The resulting prediction are piped to operational systems.

The live system uses a calibrated random forest model with features from the past twelve months that is re-trained every day. The twelve months are used because there are strong seasonality effects in retail and failing to use an exact number of years would cause fluctuations in features that are calculated by aggregating data over the training period.
To train the model, historic net sales over the last year is used as a proxy for CLTV labels and the random forest parameters are learned using features generated from a disjoint period prior to the label period. This is illustrated in Figure 13.

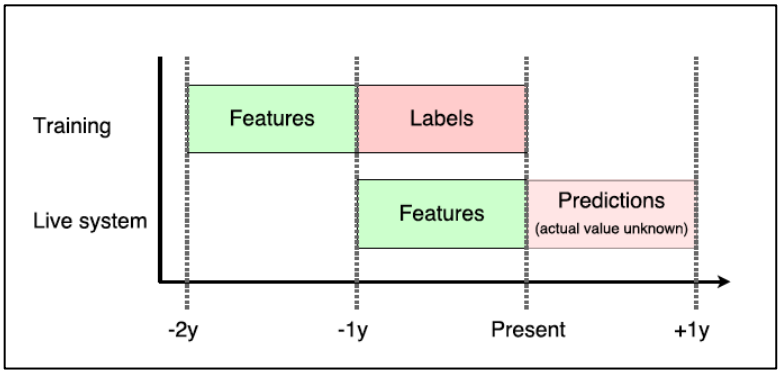


Fig 13. Training and prediction time-scales for CLTV

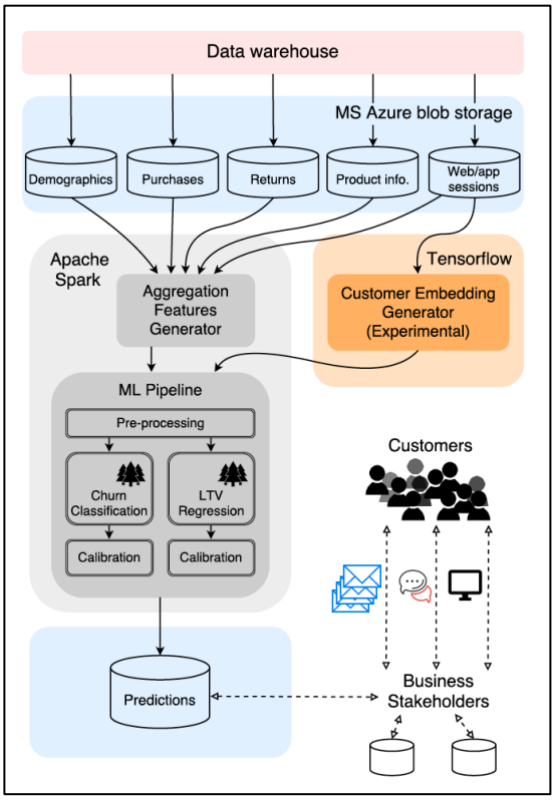


Fig 12: High-level overview of the CLTV system

The following is generated every day:

- A set of aggregate features and product view-based embeddings for each customer based on their demographics, purchases, returns and web/app sessions between two years and one year ago
- Corresponding target labels, including the churn status and net one-year spend, for each customer based on data from the last year. The feature and label periods are disjoint to prevent information leakage.

Area Under the receiver operating characteristic Curve (AUC) is used as a performance measure.

Calibration refers to the efforts to ensure that the statistics of the model predictions are consistent with the statistics of the data. Model predictions are derived from RF leaf distributions and we perform calibration for both churn and CLTV prediction. To generate consistent probabilities, calibration is performed by learning a mapping between the estimates and the realized probabilities. This is done by training a one-dimensional logistic regression classifier to predict churn based only on the probabilities returned by the random forest. The logistic regression output is interpreted as a
calibrated probability. Similarly, to estimate CLTV there is no guarantee that the regression estimates achieved by minimizing the Root Mean Squared Error (RMSE) loss function will match the realized CLTV distribution. To address this problem, analogously to churn probability calibration, the CLTV percentile is first forecasted and then the predicted percentiles are mapped into monetary values. In this case, the mapping is learnt using a decision tree.

6. General Lessons About Data Science

Today, businesses can collect data along every point of the customer journey. This information might include mobile app usage, digital clicks, interactions on social media and more, all contributing to a data fingerprint that is completely unique to its owner.
In the era of growing digital data, it is imperative to understand the benefits that businesses can reap from data science in terms of driving positive outcomes for their own business and their customers, while still maintaining and facilitating the highest level of data protection.
In our opinion, most of the media and e-commerce Chief Data Officers dream of an active collaboration between their Marketing and Data Science teams because the knowledge of Data Science can prove profitable to an organization for the following cases:

- Increase response rates, customer loyalty and, ultimately, ROI by contacting the right customers with highly relevant offers and messages.
- Reduce campaign costs by targeting those customers most likely to respond.
- Decrease attrition by accurately predicting customers most likely to leave and developing the right proactive campaigns to retain them.
- Deliver the right message by segmenting customers more effectively and better understanding target populations.

7. Recommendations and Future Scope

ASOS could incorporate the following 3 techniques to monitor and improve the lifetime value of its customers, for acquiring new customers, for customer portfolio management, for retention decisions and for increasing advertisement ROI.

- **Segmentation framework**
  Total customer base is segregated into segments based on their transaction behavior. For each segment a formula is derived, as a function of customers' vintage and recency, for the calculation of customer's lifetime. Customer's profitability is calculated based on measured historical values and predicted values.
  $CLTV = CLTV_{History} + P(t)*T*[Monthly\ Potential]$
- **Vintage based forecast framework**
  Customers are segmented on their transaction behavior and for each segment customer's lifetime value curve is obtained. Customer Lifetime Value is the area under the curve.
- **Survival analysis**
  Survival model is developed based on customer's past behavior and trends, to calculate the probability of a customer's survival for next "n" years. CLTV is calculated based on historic and predictive Customer Lifetime Value for each customer.
  $CLV = CLV_{History} + CLV_{Future}$
  $CLV_{Future} = Survival\ (t) * T * [Monthly\ Potential]$

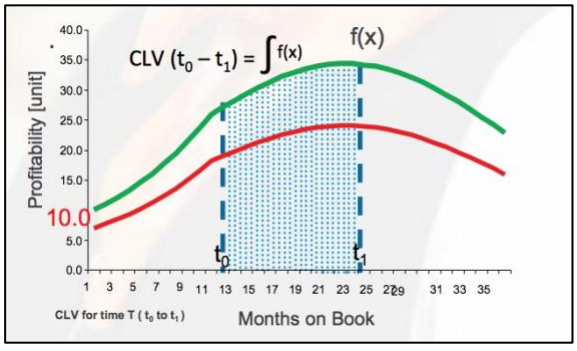


Fig 14: Vintage based forecast framework