Problem Statement:
Ralph Lauren plans on releasing 100 ***new styles****, and they want to forecast the demand for each.*
How should they do it?

1. What ***kind of data*** *would you collect?*
2. How would you ***predict demand*** *of a new stlyle?*
3. How to gain credibility? : *i.* ***Evaluate performance*** *ii.****Explainanbility***

Can employ First Principles Thinking.

Solution:
It is intuitive to think that the sales of new styles can be predicted by clustering past styles with similar visual similarities and design attributes. However, this approach may not be accurate because the sales behavior of similarly clustered items can vary significantly. This is because sales are influenced by a variety of factors, including pricing, brand, and relative placement of a design. These factors can play a much more significant role in a customer's decision to purchase than the visual similarities or design attributes of a product alone.

The model we need to build, thus, should learn to identify similarly behaving time series across latent parameters, and also take into account discounting, promotions, visibility variations in comparing the time series.
A point in a time series is represented as
$\mathbf{y}_{it} = \mathbf{f}(\mathbf{A}_i, \mathbf{M}_{i,t}, \mathbf{M}_{i,t-1}, \ldots, \mathbf{M}_{i,t-p}, \mathbf{D}_{i,t}, \mathbf{D}_{i,t-1}, \ldots, \mathbf{D}_{i,t-p})$
where
$\mathbf{y}_{it}$ is **sales** for item 'i' at time 't',
$\mathbf{A}_i$ is **attribute of the item** 'i' like colour - blue, material - cotton etc.,
$\mathbf{M}_{it}$ indicate **merchandising factors** like discount, promotion for item 'i' at time 't',
$\mathbf{D}_{it}$ are **derived features** like trend, seasonality which are inferred from data and affect the sales, p is number of time lag.

Conventional time series methods face limitations when predicting forthcoming fashion items that are yet to be unveiled. The inherent unpredictability and swift fluctuations of fashion trends pose challenges for conventional time series models in accurately anticipating future demand for yet-to-be-launched fashion articles. Hence, we work with machine learning models ranging from tree based models like Random Forest and various flavours of Gradient Boosted Trees, to deep learning models.

1. Business objective **:**
Generate optimal revenue. This can also be stated otherwise like so: Find what will be the number of items sold for each of the new fashion article.

2. Data Collection :
We collect data on newly released fashion articles in previous n years. The data may include the span of each article's availability in the market (length of the time series for article), the number of items sold per fashion article, the color, print, material, sleeves, collaboration & partnerships, economic indicators (macroeconomic factors like disposable income, consumer confidence, and unemployment rates; economic conditions can influence consumers' willingness to spend on fashion) when the fashion article was released, the promotions, discount, page views (visibility), etc. of the fashion article.

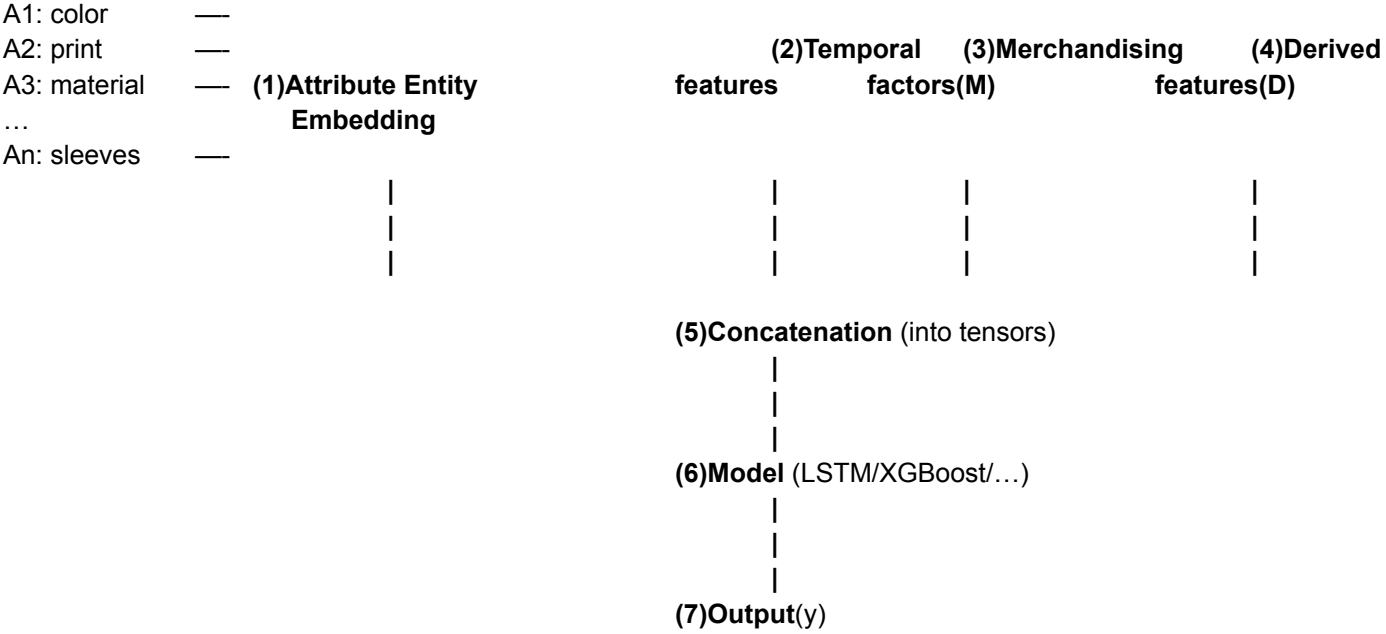
3. Feature Engineering :
We can engineer **temporal features** like
- The day of the week
- The month of the year
- The time of day
- The weather
- The holidays

The MLP model can learn to use the temporal features to predict the future sales of an item. For example, the model can learn that the sales of an item tend to increase in the first few days after it is released, and then level off after a few weeks. The model can also learn that the sales of an item are affected by holidays and other seasonal events.

To model the effect of sales drop just before and after a promotion, features like days to promotion and days from promotion can be used.

Derived Features :
– Shelf life of a style : With longer shelf life, the style's demand may decay with time.
– Trend and Seasonality : To model a trend in interest over time, the number of weeks between experiment start date and the current date is used. In order to model seasonality in purchase patterns, first 3 terms of the Fourier transform of week of year are used as features. For a new item, these can be derived during prediction.

4. Model Selection :
As mentioned in the intoduction, traditional time series models are not suitable choice for f . Hence, we work with machine learning models ranging from tree based models like Random Forest and various Gradient Boosted Trees, to deep learning models (tree based and deep learning models are chosen for their ability to model feature interactions even if transient in time, so that they capture non-linear relationship between target and regressors). Gist:



In most retail data, we see long tail behaviour, with fewer items contributing to a majority of the sales. Due to this, we see variation of sales over several orders of magnitude. To address this high variance problem, we train our models at different scales - log and linear, and try a different set of loss functions.

Model	**Criterion / Loss Function**
Random Forest (RF)	Mean Squared Error (MSE)
Gradient Boosted Regression Trees (GBRT)	MSE and Huber
Light Gradient Machine (LGBM)	MSE and Poisson
CatBoost (CB)	MSE and Poisson
XGBoost (XGB)	MSE and Poisson
Attribute Embedding + MLP	MSE and Poisson
Attribute Embedding + LSTM	MSE and Poisson

(If there is no variance problem i.e. the sales data is Gaussian, then we might as well use just MSE)
Tree based models and MLP are trained in non-linear ARIMA manner, where lagged values of time varying features are used to capture temporal dependencies.
We can use lagged values of temporal features up to last 4 time steps (p = 4) based on the intuition that temporal interactions over periods longer than 4 weeks are insignificant.

We model promotions, discount, and list page views (visibility) along with fashion attributes of the style as external regressors.

5. Training and Validation :
Let's say we collect data for 2 years. Then we might train the model on 1 year's data and validate it on next 6 months' data. The validation set is used to tune hyper-parameters of the models, using standard validation techniques.

6. Model Testing, Evaluation & Interpretation :

A test set of subsequent 6 months can be used for measuring and reporting performance.

We use weighted mean absolute percentage error (wMAPE), where the weight is the actual sales realised for an item.

$$wMAPE = \frac{\sum_{i=1}^{i=n} \sum_{t=1}^{t=t_i} |\hat{y}_{it} - y_{it}|}{\sum_{i=1}^{i=n} \sum_{t=1}^{t=t_i} y_{it}}$$

yit and ˆyit is actual and forecasted sales of an item ‘i’ at time ‘t’.
n is total number of items,
ti is the length of time series for item ‘i’.

We choose to weight our MAPE by the item’s actual sales in accordance with our tolerance for error in predicted values, so that the tolerance is lower with higher sales volumes.
Why wMAPE over MAPE? If actual sales for a set of items are 0, 5, and 10; and forecasted values are 1, 10 and 10; MAPE would be infinite, whereas wMAPE would be 0.4.
In under-forecasting scenarios, errors are upper bounded by a wMAPE of 1; in overforecasting scenarios, wMAPEs may be arbitrarily high.
Due to the specific nature of retail supply chains, especially in industries like fashion with long procurement lags and minimum order quantity requirements, the consequences of over-forecasting are more detrimental than those of under-forecasting. As a result, we prioritize addressing the challenges of over-forecasting and do not make efforts to balance the impact of both scenarios i.e. we do not symmetrise our under-forecasting and over-forecasting scenarios.

When working with fashion buyers to operationalize our plans and evaluate our forecasts on real buys, a relative priority of items is important to the procurement process since procurement happens in lots of minimum order quantity. An item with low forecasted sales may therefore not be ordered due to restrictions in buying budgets, time, and inventory holding capacity. Therefore, for an item that has higher actual sales realised relative to another, the forecasted sales should also be relatively higher so that ordering it ensures higher sell through rates as well as lesser inventory pile up. To capture this, we use the Pearson correlation and the Kendall tau.

$$\rho_{y_i, \hat{y}_i} = \frac{E[y_i \hat{y}_i] - E[y_i]E[\hat{y}_i]}{\sqrt{E[y_i^2] - E[y_i]^2}\sqrt{E[\hat{y}_i^2] - E[\hat{y}_i]^2}}$$

$$tau = \frac{(P - Q)}{\sqrt{((P + Q + T) * (P + Q + U))}}$$

P is the number of concordant pairs,
Q the number of discordant pairs,
T the number of ties only in yi,
U the number of ties only in ˆyi.

Pearson Correlation ensures that forecasted values and actual values move together in the same direction, and Kendall Tau takes into account relative ordering of the quantities between forecasted and actual values.

For model tuning, we use Mean Squared Error (MSE)

$$MSE = \frac{\sum_{i=1}^{i=n} \sum_{t=1}^{t=t_i} (\hat{y}_{it} - y_{it})^2}{\sum_{i=1}^{i=n} t_i}$$

Typically retail data shows long tailed distribution in linear scale, hence we use Poisson loss in linear scale for learning model parameters.

$$Poisson\ Loss = \sum_{i=1}^{i=n} \sum_{t=1}^{t=t_i} \hat{y}_{it} - y_{it} * log(\hat{y}_{it})$$

Huber Loss is used to minimize the effect of outlier on the training process.

$$Huber\ Loss = \frac{1}{2}\sum_{i=1}^{i=n}\sum_{t=1}^{t=t_i}\begin{cases}(\hat{y}_{it} - y_{it})^2 & \text{if } |\hat{y}_{it} - y_{it}| \leq \delta \\ \delta * |\hat{y}_{it} - y_{it}| - \frac{1}{2}\delta^2 & \text{otherwise}\end{cases}$$

For each model, we tabulate the wMAPE.
We can select the models with lowest wMAPE or use an ensemble of first n models (then again, checking if the loss is actually minimized or not).

7. Stakeholder Engagement :

The model that **predicts sales of new styles** can be helpful to a couple of stakeholders engaging in Seasonal Assortment Planning, Product Selection in Roadshows, etc.

They can plan their revenue ahead and know the quarterly, monthly, weekly sales.

i. Seasonal Assortment Planning : Retailers have to plan their assortment a year in advance due to manufacturing lead times. At the time planners do not have any information about the actual products, so they create all plans at attribute combination level and use an average based projection together with intuitive calls to allocate inventory budget. This model when used with appropriate simulations can generate forecasts for all possible attribute combinations of styles.

ii. Drop Planning : Currently, most retailers plan drops at a fixed interval irrespective of how demand for an item is going to be. This leads to either lost sales or lot of inventory at hand. This model provides good weekly *sales forecast*, evident from lower wMAPE at item-week level; this gives an opportunity to better plan drop by moving from manual to automated drop planning driven by data and machine learning

iii. Product Selection in Roadshows : Wherever a catalogue of items (with their descriptions and brands) is made available for a buyer to consume in events like Fashion Roadshows which buyers frequent to find out actually available assortment from different brands, our model can quickly compute *projected sales* for different items present in the roadshow. A buyer may use his/her intuition in addition to our model
output to get directional information on which products to spend budget on.