1)Data Scientist at AI- Consultancy:

Client: Fortune 100 retail company

SCM Team - Supply Chain Management DS team:

30 members: SWEs, DS, PMs, QE, Test Eng,...etc…

DS- 3 members:

Building an AI based supply chain tool:

-provides visibility of KPIs to supplier and replenishment managers

-DS team:

-increase revenue of the client by x%

-client **suspects** a lot of sales has been lost due to out of stock scenarios

-client wants you to solve this problem, and also look into other factors causing these decline in revenue

Statement of Work doc–

1)Budget

2)Timeline

3)Goals

4)Members

5)Impact expected

**#Problem Solving:-**

**#Framework to solve a problem:-**

**1)Supply Chain**

1) Understand the client’s domain and their industry (Overall - retail level)

2) Understand the team’s work and domain- supply chain

**Amazon:-**

**Supply chain:-**

Manufacturer Philips → Supplier A (philips, Nivea, Kelloggs,etc…) → DC (distribution center)/ warehouse → Stores-DMart/Online -Amazon

Purchase Order or PO\_NBR

Supplier→ DC→ Store

1) Understand the client’s domain and their industry (Overall - retail level)

2) Understand the team’s work and domain- supply chain

3) Define the Problem

**Current State→**

1)F-100 retail client see decline in revenue

2) OOS scenarios causing revenue drop by y%

3)The client wants to implement an AI tool to have a visibility for suppliers

Into this decline of revenue

4)SCM DS team is expected to validate the hypothesis and consider factors causing this drop and increase revenue by x%

**Gap:**-- Reduce OOs scenarios by identifying factors

**Future State→**

1) Increase rev in **next 3 months** by x% using data science

2) Identify factors causing drop and take suitable actions

3)Ensure this isn’t repeated

4)Look into interconnected problems

**Factors for OOS:-**

1)Demand Forecasting Issue—

2)Lead time Issue—raw materials, not etting approval,sys defect,

3) Raw materials scarcity

*4) Transport vehicle is broken*

5) Manufacture couldn’t manufacture so many pdts

*6)Custom Dept issues*

*7) Payment issues*

8) Quality Issues – resent

*9)Global Issues - Covid, recession*

*10)Manual labour – efficiency– less*

**Hypothesis:-**

1. There is loss of sales due to OOS scenarios
2. There is loss of sales due to bad marketing – another team
3. Higher lead time causes more OOS scenarios
4. Total qty is higher leads to OOs

5)Risk1 is negatively impact OOS

…..

**Validate using data**

–Search for data sources and available data

**Data sources:** lead time, inventory data, transport data, product data, quality data,demand ..etc…

**—--end of chapter 1—---**

Next meeting with the client:--

–Gives you 80% of data sources

-rn client has a sample of data, work with this for now

-now we need to validate hyp using sample of data and generalize for

population

**—**

**Beginning of ch-2**

–clarification

Tell clients short terms goals –week1, week2,....weekn –tell them using ML

-business requirements - AWS—subc

Hyp Testing

-The data we have is sample data for limited POs

-Full data (entire POs) you’ll receive after a month

**-**Important factors affecting OOS

-Lead Time —- client says why?

-Lead time impacting OOS — OOS is increasing

**Fix lead time issues– how???**

Revenue → OOS —> Lead Time —> Factors?

Revenue → OOS —> factors!

**Chapter 3: Proof-Of-Concept < 3 months – Prototype/Demo**

**AI Smart SCM tool:**

-New feature which predicts future lead times - ML REGRESSION

-New feature which predicts if a PO is going to be OOS in future –PO OOS/NOOS – CLASSIFICATION

-Notify the store managers if HRS Score > above safe threshold

-Store managers take suitable actions to prevent loss of sales and decrease storage cost

-Your revenue will increases

**-Your final impact is going to be revenue increased by x%**

**Business managers will decide if that x% is good enough?**

**If it is good enough you go to ch-4 where you scale it up and you’ll get more budget for that from clients**

**-but if x% is not that good, then project may end up over there due to no additional ROI**

PO→ f1,f2,f3,f4…fn, LT Preds, OOS Prob

1. PO→ 1 days, 0.9→ Low risk
2. PO→ 10 days, 0.1→ Low risk
3. PO→ 10 days, 0.9→ High risk
4. PO→ 1 day, 0.1 →Highly Low risk

0.1 + 0.9 —>AM = 0.5

0.1 \* 0.9 —--> HRS = 0.09–LOW RISK

—>(**P+R**)/2

—>(**2\*P\*R**)/(P+R)--->CHOOSE THIS

**PO→ HRS**

**Business these are the POS having very high risk based on their HRS scores**

**Through the AI**

LT Preds, OOS Prob → Hybrid Risk Score

→ if HRS score high, then high risk else low risk

LT\*OOS —> 0.3 \* 0.5

Actual lt time, preds lt time

How far is preds lt time from mean actual lt time? –

MEAN LT TIME - CENTER

PRED LT TIME 1 – 5 units far from mean actual lt time –

PRED LT TIME 2 – 1 units far from mean actual lt time –

ML tools - 3 days

I’ll order 3 days prior

1. **Receives order very early** – issue→ not enough inventory at store to hold the items→ pay extra to store them —> additional costs
2. **Receives order very late** – issue→ loss of sales due to OOS scenarios → customers experience hampers –. Loss of $ or revenue

PO4 - 0.9 of OOS —immediately order

Demand forecasting if we have an expected demand for this PO—Order it immediately

But if no demand then can wait for some time

**Ch-3: ML**

#Framework for ML model building

–Linear, Non-Linear

–Feature Eng, Data Preprocessing

DS - Feature in this AI tool - predict lead time, OOS scenarios, analyze factors

Crosstab:

Sample

Ch4 : Population – 2M data — python ?

Business Problem:-

**Ch-3: Insights, EDA, ML framework, Python Coding**

**Chapter 4: Scale it up for entire population data**

**—> greater impact**

**For chapter-3:**

**ML Model Framework -**

1)Clarification - Platform , Resources, Impact,

2) Understand the business requirements:-

**-At what time period should the model run?** - daily, hourly, weekly, monthly?

-4 hours?

—> For future, how much data should we consider to give predictions? - past 1 week data, past 1 year data

–data is rapidly changing, so the model you trained 1 month ago, needs to be rebuilt

* **Decide when to re-train the models?**
* **How much data should we use?**

Finally where would the predictions be incorporated? – AI based tool → app, webapp, website→?

3)Data Sources→ Write **modularized** and **scalable automated** code with proper quality checks

India → code1

Singapore -> you don’t have to make a lot changes in the code - modularized code

Scalability→ 100K rows→ 1M rows→ 1000M rows

Parallel and distributed computing —? Cloud servers?

100K rows – all good

1M rows- things failed — now you need to migrate to another platform which can handle this

If in future, new data source, truck data is added, how can we add that to our model in automated way

You dont have to manually run the JN, platform →

Workflow which automatically triggers / runs the code at scheduled time

–Airflow —>

In-House tools → workflow where they can automatically run the pipeline

PCA - dimensionality reduce features – evs → loose understand of which features are important

Model explainabilty

L1, L2 regularization - features, avoid overfitting/underfitting

PCA → ev1,ev2,ev3,ev4

90% of variance from2 evs

ev1—> f1,f4,f5?

4)Technical Solution:-

* EDA and Insights generation
* Data Pre-processing
* Feature Engineering - **Polynomial Features, Lag variables, rolling features– 7 days of data— take mean of 7 days of data**
* Feature Selection → PCA, regularization? ML model built– check important features→VIF build another model using only important features instead of all the features : Tier-1 and Tier-2 models
* Training the Model: Linear, Non-Linear, NNs, SVMs
* Model Evaluation: Metrics – Regression/Classification - P/R?
* Productionalization → KNN isnt good for productionalization,

NNs might not be best for model explainability

Data is so non-linear that even your XGboost can

NN - shapley values - MLP – non l

**Occams Razor Rule**

If you have 2 ML models – both giving above benchmark performance and

Difference between their performance isnt too high…

Then always go with the simpler model

**LOR–90% Accuracy**

XGBoost - 91% Accuracy

NN - 91.5% Accuracy

In reality if you have time, budget and resources, try every single model

After you done your best at feature engineering, hyperparameter tuning for

the potential best model

ML engineers -

XGBoost – \

RF

DT

Po–sales past 1 day→ sales on past 2nd day—> sales on past 30th day→

1st Jan – Sales very high

2nd Jan - Sales very low

PO-2nd Jan Sales—1st Jan sales – sales is low if the previous day was a festival

Po- data → avg of sales data for previous 7 days

11th Jan – 4th -11th jan

12th Jan - 5th -12th Jan

13th Jan -

100 Pos– 5 items

500M items - 500M preds

100 pos- 100 preds

Category:

PO Beauty care:- 20 Pos

5 categories each having 20 Pos→ 5 preds

Beauty Care - Risky

Salary→

w1x1—> w1\*salary

Degree =2 → w1\*salary\*\*2

Linear Regression → Non Linear Regression

No of rows and no of cols:

1000 rows and 100 features — a lot – feature selection

1000 rows and 10 features – no fs

1M rows and 10 features — create features- feature engineering

1M rows and 1000 features