**VISVESVARAYA INSTITUTE OF TECHNOLOGY**

**Jnana Sangama, Belagavi-590010**



MINI PROJECT REPORT

ON

“**IMPLEMENTATION OF ELECTION MANAGEMENT SYSTEM**”

Submitted in partial fulfillment for the requirements for the fifth semester curriculum

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

For the Academic year 2018-2019

Submitted by:

**ASHISH KUMAR DUBEY 1MV16CS021**

**ARUN SINGH 1MV16CS020**

Project carried out at:

**Sir M. Visvesvaraya Institute of Technology**

Bengaluru-562157

Under the guidance of:

**Mrs. SAPNA**

Asst. Professor, Department of CSE

Sir M. Visvesvaraya Institute of Technology, Bengaluru

****

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**SIR. M VISVESVARAYA INSTITUTE OF TECHNOLOGY**

HUNASAMARANAHALLI, BENGALURU-562157

SIR M. VISVESVARAYA INSTITUTE OF TECHNOLOGY

BENGALURU -562157

  DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CERTIFICATE

It is certified that the project work entitled “**IMPLEMENTATION OF ELECTION MANAGEMENT SYSTEM**” is a bonafide work carried out **by ASHISH KUMAR DUBEY(1MV16CS021) , ARUN SINGH(1MV16CS020)** in partial fulfilment for the requirements of mini project for the V semester curriculum  Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belagavi during the year 2018-2019 . It is certified that all corrections and suggestions indicated for Internal Assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the course of Bachelor of Engineering.

Name & Signature of Guide       Name & Signature of HOD

**Mrs. SAPNA Prof.**

Asst. Prof & Internal Guide HOD, Dept of CSE

Dept. Of CSE, Sir MVIT Sir MVIT Bengaluru - 562157 Bengaluru -562157

External Examination:

Name of the Examiners                                                        Signature with date

1)

2)

**ACKNOWLEDGMENT**

It gives us immense pleasure to express our sincere gratitude to the management of **Sir M. Visvesvaraya Institute of Technology,** Bangalore for providing the opportunity and the resources to accomplish our project work in their premises.

On the path of learning, the presence of an experienced guide is indispensable and we would like to thank our guide **Mrs. SAPNA,** Assistant Professor, Dept. of CSE, for her invaluable help and guidance.

We would also like to convey our regards and sincere thanks to **Prof. \_\_\_\_\_\_\_\_\_\_\_\_** HOD, Dept. of CSE for his suggestions, constant support and encouragement, Heartfelt and sincere thanks to **Prof. MANJUNATH,** Principal, Sir. MVIT for providing us with the infrastructure and facilities needed to develop our project.

We would also like to thank the staff of Department of Computer Science and Engineering and lab-in-charges for their co-operation and suggestions. Finally, we would like to thank all our friends for their help and suggestions without which completing this project would not have been possible.

* ASHISH KUMAR DUBEY (1MV16CS021)
* ARUN SINGH (1MV16CS020)

DECLARATION

We hereby declare that the entire mini project work embodied in this dissertation has been carried out by us and no part has been submitted for any degree or diploma of any institution previously.

Place: Bengaluru

Date:

Signature of Students:

ASHISH KUMAR DUBEY (1MV16CS021)

ARUN SINGH(1MV16CS020)

ABSTRACT

TABLE OF CONTENTS

**Certificate I**

**Acknowledgement II**

**Declaration III**

**Abstract IV**

**Table Of Contents V**

**List Of Figures VIII**

Chapter 1 **INTRODUCTION** 1

1.1 Introduction to SCM 3

1.2 Need For SCM 3

1.3 Aim / Objectives Of This Project 5

Chapter 2 **FRONTEND AND BACKEND** 6

2.1 History Of SCM 6

2.2 Base Paper / Existing System 8

2.3 Proposed System 8

Chapter 3 **SPECIFICATIONS** 9

3.1 Hardware Requirements 9

3.2 Software Requirements 9

3.3 Functional Requirements 9

3.4 Non-Functional Requirements 10

Chapter 4 **METHODOLOGY** 11

4.1 Introduction 11

4.2 Tables and datatypes 11

4.3 Schema Diagram 11

4.4 E R Diagram 12

Chapter 5 **IMPLEMENTATION** 23

5.1 Modules

(Description and code of each module in order.) 23

Chapter 6 **RESULTS AND SNAPSHOTS 46**

**CONCLUSION** 62

**REFERENCES**

**CHAPTER 1**

**INTRODUCTION**

**1.1 INTRODUCTION TO EMS**

Supply chain management (SCM) is the oversight of materials, information, and finances as they move in a process from supplier to manufacturer to wholesaler to retailer to consumer. Supply chain management involves coordinating and integrating these flows both within and among companies. It is said that the ultimate goal of any effective supply chain management system is to reduce inventory. As a solution for successful supply chain management, sophisticated software systems with Web interfaces are competing with Web-based application service providers (ASP) who promise to provide part or all of the SCM service for companies who rent their service.

A supply chain is the collection of steps that a company takes to transform raw components into the final product. Typically, supply chain management is comprised of five stages: plan, develop, make, deliver, and return.

1. **Plan:**

A plan or strategy must be developed to address how a given good or service will meet the needs of the customers. A significant portion of the strategy should focus on planning a profitable supply chain. A big piece of SCM planning is developing a set of metrics to monitor the supply chain so that it is efficient, costs less and delivers high quality and value to customers.

1. **Develop:**

It involves building a strong relationship with [suppliers](http://www.wisegeek.com/what-are-suppliers.htm) of the raw materials needed in making the product the company delivers. This phase involves not only identifying reliable suppliers but also planning methods for shipping, delivery, and payment. SCM managers can put together processes for managing their goods and services inventory, including receiving and verifying shipments, transferring them to the manufacturing facilities and authorizing supplier payments.

1. **Make:**

At this stage the product is manufactured, tested, packaged, and scheduled for delivery. This is the manufacturing step. This is the most metric-intensive portion of the supply chain—one where companies are able to measure quality levels, production output and worker productivity.

1. **Deliver:**

This stage, at the logistics phase, customer orders are received and delivery of the goods is planned. This is the part that many SCM insiders refer to as logistics, where companies coordinate the receipt of orders from customers, develop a network of warehouses, pick carriers to get products to customers and set up an invoicing system to receive payments.

1. **Return:**

As the name suggests, during this stage, customers may return defective products. The company will also address customer questions in this stage. This can be a problematic part of the supply chain for many companies. Supply chain planners have to create a responsive and flexible network for receiving defective and excess products back from their customers and supporting customers who have problems with delivered products.

This project in the Supply Chain Management will provide us an understanding of key areas of logistics and supply chain management where relevant analysis of data needed. This will focus on several key supply chain functions and analyse data that may be available for the supply chain. This will be done through Excel-based approaches for analysing. `If a company expects to achieve benefits from their supply chain management process, they will require some level of investment in technology. The backbone for many large companies has been the vastly expensive Enterprise Resource Planning (ERP) suites, such as SAP and Oracle.

Since the wide adoption of Internet technologies, all businesses can take advantage of Web-based software and Internet communications. Instant communication between vendors and customers allows for timely updates of information, which is key in management of the with today’s emphasize on cutting costs and streamlining expenses, many companies are supply chain.

**1.2 NEED FOR SUPPLY CHAIN MANAGEMENT**

Looking to improve their bottom lines with more effective supply chains. Unfortunately, many people involved with companies don't have a clear understanding of what a supply chain is or how it fits into the company’s overall strategy. Supply Chain Management (SCM) software can have tremendous financial benefits for companies Supply chains include a company’s entire manufacturing and distribution process. They involve every step of the production from planning to manufacturing to handling defective goods. The overall goal of these chains is to keep the process running smoothly at all times and to keep all of the components (vendors, warehouses, etc.) connected.

It is well known that supply chain management is an integral part of most businesses and is essential to company success and customer satisfaction.

**Boost Customer Service**

• Customers expect the correct product assortment and quantity to be delivered.

• Customers expect products to be available at the right location.

• Right Delivery Time – Customers expect products to be delivered on time.

• Right After Sale Support – Customers expect products to be serviced quickly.

**Reduce Operating Costs**

• Decreases Purchasing Cost – Retailers depend on supply chains to quickly deliver expensive products to avoid holding costly inventories in stores any longer than necessary.

• Decreases Production Cost – Manufacturers depend on supply chains to reliably deliver materials to assembly plants to avoid material shortages that would shut down production. Decreases Total Supply Chain Cost – Manufacturers and retailers depend on supply chain managers to design networks that meet customer service goals at the least total cost. Efficient supply chains enable a firm to be more competitive in the market place

**Improve Financial Position**

• Increases Profit Leverage – Firms value supply chain managers because they help control and reduce supply chain costs. This can result in dramatic increases in firm profits.

• Decreases Fixed Assets – Firms value supply chain managers because they decrease the use of large fixed assets such as plants, warehouses and transportation vehicles in the supply chain. Increases Cash Flow – Firms value supply chain managers because they speed up product flows to customers.

• Lesser known, is how supply chain management also plays a critical role in society. SCM knowledge and capabilities can be used to support medical missions, conduct disaster relief operations, and handle other types of emergencies.

Whether dealing with day-to-day product flows or dealing with an unexpected natural disaster, supply chain experts roll up their sleeves and get busy. They diagnose problems, creatively work around disruptions, and figure out how to move essential products to people in need as efficiently as possible.

**Ensure Human Survival**

• SCM Helps Sustains Human Life – Humans depend on supply chains to deliver basic necessities such as food and water. Any breakdown of these delivery pipelines quickly threatens human life

• SCM Improves Human Healthcare – Humans depend on supply chains to deliver medicines and healthcare. During a medical emergency, supply chain performance can be the difference between life and death.

• SCM Protects Humans from Climate Extremes – Humans depend on an energy supply chain to deliver electrical energy to homes and businesses for light, heat, refrigeration and air conditioning. Logistical failure (a power blackout) can quickly result in a threat to human life.

**Improve Quality of Life**

• Foundation for Economic Growth – Societies with a highly developed supply chain infrastructure (modern interstate highway system, vast railroad network, numerous modern ports and airports) are able to exchange many goods between businesses and consumers quickly and at low cost. As a result, the economy grows. In fact, the one thing that most poor nations have in common is no or a very poorly developed supply chain infrastructure.

• Improves Standard of Living – Societies with a highly developed supply chain infrastructure (modern interstate highway system, vast railroad network, numerous modern ports and airports) are able to exchange many goods between businesses and consumers quickly and at low cost. As a result, consumers can afford to buy more products with their income thereby raising the standard of living in the society

• Job Creation – Supply chain professionals design and operate all of the supply chains in a society and manage transportation, warehousing, inventory management, packaging and logistics information

• Opportunity to Decrease Energy Use – Supply chain activities involve both human and product transportation. As a by-product of these activities, scarce energy is depleted.

**1.3 AIM THIS PROJECT**

Supply chain management (SCM) is the oversight of materials, information, and finances as they move in a process from supplier to manufacturer to wholesaler to retailer to consumer. Supply chain management involves coordinating and integrating these flows both within and among companies

**CHAPTER 2**

**LITERATURE REVIEW**

**2.1 HISTORY OF SUPPLY CHAIN MANAGEMENT**

Over the last 100 plus years of the history of supply chain management has evolved from an initial focus on improving relatively simple, but very labour-intensive processes to the present-day engineering and managing of extraordinarily complex global networks

In the 1940s and 1950s, the focus of logistics research was on how to use mechanisation to improve the very labour intensive processes of material handling and how to take better advantage of space using racking and better warehouse design and layout. In the mid 1950 s, this concept was extended transportation management with the development of intermodal containers together with ships, trains, and trucks to handle these containers.

By the 1960s, a clear trend had developed in shifting more time-dependent freight transportation to truck rather than rail. This led to the need for joint consideration of warehousing, material handling, and freight transportation, which emerged under the label of "Physical Distribution." The National Council of Physical Distribution Management was formed in 1963 to focus industry attention on this area and quickly became the predominant organization in the field.

Prior to the 1960s, virtually all transactions and record keeping were done manually. The computerization of this data opened the door to a huge opportunity for innovations in logistics planning, from randomized storage in warehouses to optimization of inventory and truck routing.

The 1980s marked the beginning of a sea-change in logistics in the history of supply chain management. The emergence of personal computers in the early 1980s provided tremendously better computer access to planners and a new graphical environment for planning. Perhaps the most important trend for logistics in the 1980s was that it had begun to get tremendous recognition in industry as being very expensive, very important, and very complex. Company executives became aware of logistics as an area where they had the opportunity to significantly improve the bottom line if they were willing to invest in trained professionals and new technology.

The logistics boom was fuelled further in the 1990s by the emergence of Enterprise resource planning **(ERP)** systems. These systems were motivated in part by the successes achieved by Material Requirements Planning systems developed in the 1970s and 1980s, in part by the desire to integrate the multiple databases that existed in almost all companies and seldom talked to each other, and in part by concerns that existing systems might have catastrophic failures as a result of not being able to handle the year 2000 date. In spite of some significant problems in getting the ERP systems installed and working, by 2000 most large companies had installed ERP systems.

The widespread recognition of the term "supply chain" has come primarily as a result of the globalization of manufacturing since the mid 1990 s, particularly the growth of manufacturing in China. U.S. imports from China grew from about $45 billion per year in 1995 to more than $280 billion per year in 2006.

**2.2 BASE PAPER / EXISTING SYSTEM**

**2.2.1 Base Paper**

Present scenario competition between organizations is among supply chain, Global economic climate is that most of companies experiencing decline in gross margin due to lack of visibility in respective supply chain. Therefore, effective supply chain management plays key role in improving organizational performance.

Scope of this project is to study Supply Chan application at L&T Construction & Mining Business to improve Customer support at after sales market and organizational performance. After sales & service division deals with 25000 to 30000 SKUs to support 70 to 75 verity of Equipment’s like Wheel Loaders, Excavators, Dozers and Dump trucks. This is a real challenge to the organization to balance demand and supplies with optimum cost. This project thus analyses the data to have clear visibility and stream lining its operations through Supply Chain Analytics to respond them proactively, increasing the both efficiency and profitability. Further it stipulates the linkage between organizational information, alerts and key performance indicators and its management.

**2.2.2 Existing system:**

L&T Construction & Mining Business uses ERP - SAP ECC 6.0 (Modules – Sales and Distribution, Materials Management & Finance) and uses standard reports and few of customized reports. Data analysis is done outside SAP using MS Excel on case to case basis. It is a challenge to get right data at right time for taking right decision making.

**2.3 PROPOSED SYSTEM:**

It is recommended to overcome present challenges in data, proposed to use Analytical techniques to get right data at right time for right decision. There are many analytical software solution are available, it is proposed to use R & R Studio and Excel. L&T data primarily on SAP and pulling data directly to R is difficult as it requires customization, hence it is proposed to take SAP dump in Excel and load data in R / R Studio to fulfil the requirement of this project.

**CHAPTER 3**

**SPECIFICATIONS**

**3.1 HARDWARE REQUIREMENTS**

* PROCESSOR : Intel Pentium or Higher Version
* RAM : Minimum 1GB
* HARD DISK : 60GB and above

**3.2 SOFTWARE REQUIREMENTS**

* + SOFTWARE : Python 3.3 or greater
  + SUPPORTED BROWSERS : Google Chrome / Mozilla Firefox / Internet Explorer
* Anaconda Software or Jupyter notebook
* Windows Vista\* or Newer Version, or MACos, or Linux (32/64 bit)

**3.3 FUNCTIONAL REQUIREMENTS**

The Functional Requirements Specification documents the Operations and activities that a system must be able to perform. Functional Requirements include:

* + Descriptions of data to be entered into System
  + Descriptions of operations performed by each system
  + Descriptions of work-flows performed by the system
  + Descriptions of system reports or other outputs
  + Who can enter the data into the system
  + How the system meets applicable regulatory requirements

The Functional Requirements Specifications is designed to be read by a general audience. Readers should understand the system, but no particular technical knowledge should be required to understand the document.

These are the functional requirements specification documents for the project analysis. A software requirement specification helps to attenuate the time and energy needed by the developers to attain their desired goals and additionally minimizes the value of development.

**Following Factors are used to measure software development quality:**

Each attribute may be accustomed measure of the product performance. These attributes may be used for Quality assurance similarly as quality control. Quality assurance activities are directed towards prevention of introduction of defects and internal control activities are aimed toward detecting defects in product and services.

1. **Reliability**

Measure if product is reliable enough to sustain in any condition. Give systematically correct results. Product dependability is measured in terms of operation of project underneath different operating atmosphere and different conditions.

**2. Maintainability**

Different versions of the product ought to be easy to maintain. For development it ought to be easy to feature code to existing system, ought to be easy to upgrade for brand new options and new technologies time to time. Maintenance ought to be value effective and simple. System be easy to take care of and correcting defects or making a change within the software system.

**3. Usability**

This can be measured in terms of ease of use. Application should be user friendly. Easy to use for input preparation, operation and also for interpreting of output.

**4. Portability**

This can be measured in terms of Costing issues related to porting, Technical issues related to porting, Behavioural issues related to porting.

**3.4 NON-FUNCTIONAL REQUIREMENTS**

Satisfactory will probably not be assessed on the system where the program is developed, tested or first installed.

**CHAPTER 4**

**SYSTEM DESIGN**

**4.1 INTRODUCTION**

System design is the first design stage for devising the basic approach to solving the problem. During system design, developers decide the overall structures and styles. The system architecture determines the organization of the system into subsystems. In addition, the architecture provides the context for the detailed decisions that are made in later stages .during design, developers make decisions about how the problem will be solved, first at the high level and then with more detail.

**4.2 ARCHITECTURAL DIAGRAM**

**4.3 DATA FLOW DIAGRAM**

A Data Flow Diagram (DFD) is a graphical representation of the “flow” of data through an information system. Data Flow modules are used to show how data flows through a sequence of processing steps. The data is transformed at each step before moving on to the next stage. These processing steps or transformations are program functions when Data Flow Diagrams are used to document a software design.

Data flow modules are an intuitive way of showing how data is processed by a system. At the analysis level, they should be used to module the way in which data is processed in the existing system. The notation used in these modules represents functional processing, data stores and data movements between functions.

With a dataflow diagram, users are able to visualize how the system will operate, what the system will accomplish and how the system will be implemented. Old system dataflow diagrams can be drawn up and compared with the new system dataflow.

These are several common modelling rules to be followed while creating DFD’s are as follows:

* + All processes must have at least one data flow in and one data flow out.
  + All processes should modify the incoming data, producing a new form of outgoing data.
  + Each data store must be involved with at least one data flow.
  + Each external entity must be involved with at least one data flow.
  + A data flow must be attached to at least one process.

**4.4 USE CASE DIAGRAMS**

The use case module is a catalogue of system functionality described using UML use cases. Each use case represents a single, repeatable interactions that a user or actor experiences when using the system. A use case typically includes one or more “scenarios” which describes the interactions that go on between the actor and the system, and documents the results and exceptions that occur from the user’s perspective. Use case may include other use cases as part of a larger pattern of interaction and may also extended by other use cases to handle exceptional conditions.

A use case is a coherent piece of functionality that a system can provide by interacting with actors. For example, a customer actor can buy a beverage from a vending machine. The customer inserts money in to the machine, makes a selection, and ultimately receives a beverage. Similarly, a repair technician can perform scheduled maintenance on a vending machine.

Proposed Model will make use of the following:

**4.5 DATA ANALYTICS USING PANDAS DATA FRAME**

Pandas is an open source, BSD-licensed library providing high performance, easy to use data structures and data analysis tool for the Python Programming language. Pandas is a NUMFocus sponsored project. This will help ensure the success of development of pandas as a world-class open-source project, and makes it possible to donate to the project.

Pandas Library Features

* A fast and efficient **Data Frame** object for data manipulation with integrated indexing.
* Tools for reading and writing data between in-memory data structures and different formats: CSV and text files, Microsoft Excel, SQL databases, and the fast HDF5 format.
* Intelligent **data alignment** and integrated handling of **missing data**: gain automatic label-based alignment in computations and easily manipulate messy data into an orderly form.
* Flexible **reshaping** and pivoting of data sets.
* Intelligent label-based **slicing**, **fancy indexing**, and **sub setting** of large data sets.
* **Time series-functionality**: date range generation and frequency conversion, moving window statistics, moving window linear regressions, date shifting and lagging. Even create domain-specific time offsets and join time series without losing data.
* Highly **optimized for performance**, with critical code paths written in [Cython](http://www.cython.org/) or C.
* Python with pandas is in use in a wide variety of **academic and commercial** domains, including Finance, Neuroscience, Economics, Statistics, Advertising, Web Analytics, and more.

**CHAPTER 5**

**IMPLEMENTATION**

**5.1 STAGE 1: LOADING AND PRE-PROCESSING OF THE DATA.**

The first stage in implementation involves loading the company’s data from the excel sheet format to pandas dataframes, and then carrying out the early pre-processing on the data to detect and manipulate on the null values , missing values, duplicates and merge tables with strong relations for further implementation .

**5.1.1 Load All The Modules We Need**

1. For plotting

import matplotlib.pyplot as plt

import matplotlib

import seaborn as sns

import datetime

2. For ML

import sklearn

3. For data manipulation

import numpy as np

import pandas as pd

4. This makes all the plots to be shown within jupyter

%matplotlib inline

#Setting the default plot size

matplotlib.rcParams['figure.figsize'] = (20.0, 10.0)

**5.1.2 Loading And Parsing Of Data From Excel To Pandas Dataframes**

xlsx = pd.ExcelFile(‘data\_File’)

customer\_order = xlsx.parse('customer order')

material\_master = xlsx.parse('Material Master')

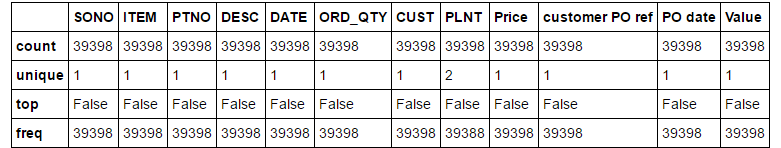
sales\_past\_demand = xlsx.parse('sales past demand')

**5.1.3 Pre-Processing And Cleaning Of The Data**

Considering the customer order dataframe.

**5.1.3.1 Finding If There Are Null Values.**

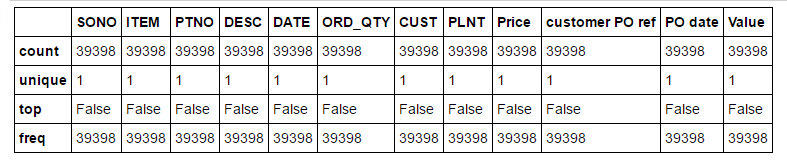
pd.isnull(customer\_order).describe()



We observe that there is missing data in PLNT . We fill it with the most common PLNT values.

customer\_order.PLNT.fillna(930, inplace=True)

pd.isnull(customer\_order).describe()



Null values have been successfully removed from the dataframe. We carry out similar operations on all dataframes.

**5.1.3.2 Processing Data With Incorrect Formats Or Ambiguous Values**

We see that customer\_order['PO date'] gives has some strings which are set to 00:00:00. We remove these by setting PO date to the DATE .

Fill in the missing 'PO date' with the corresponding values from 'DATE'. To do this, first get the month and day from each corresponding 'DATE' value. and using this and the year (2016) create a datetime.date and assign it to 'PO DATE'

tmp = customer\_order[customer\_order['PO date'] == datetime.time(0)]

day, month = [], []

for i in tmp['DATE'].str.split('-').values:

day.append(int(i[0]))

month.append(int(i[1]))

customer\_order.loc[customer\_order['PO date'] == datetime.time(0), 'PO date'] = [datetime.date(2016, m, d) for m, d in zip(month, day)]

**5.1.3.3 Merging tables with strong relations**

Since there are multiple tables and there is a strong relation amongst these tables, we could merge the tables for easier access and manipulation. We use unique keys or if no such keys are present, we form them by concatenating two or more attributes.

Customer order and Customer master on customer’s unique code :

customer\_order.head(2)

customer\_master.head(2)

customer = pd.merge(customer\_master, customer\_order, left\_on=['customer code'], right\_on=['CUST'])

customer.drop(['CUST'], axis=1, inplace=True)

**5.1.3.4 Analysis Of Duplicates**

We need to drop the duplicate values in the dataframes to eliminate redundancy. Considering the ‘Bill’ dataframe :

bill[bill.duplicated(['Sales ord', 'Delivery no', 'Description', 'Consignment details', 'Value'])]

bill.drop(bill[bill.duplicated(['Sales ord', 'Delivery no', 'Description', 'Consignment details', 'Value'])].index, inplace=True)

**5.2 STAGE 2: DATA ANALYSIS**

**5.2.1 Time To Market Or Customer**

The order to delivery analysis comprises finding the following:

1. Order to delivery note generation: The time between when order is placed by the customer to when the delivery note is generated by the company.
2. Delivery note to Invoice generation: The time interval between when the delivery note is generated to when the invoice for the corresponding order is sent to the customer.
3. Invoice to consignment: The time interval between when the invoice note is generated to when the consignment of product takes place.
4. Consignment to reaching customers (Recpt date): The time interval when the consigned product is delivered to the respective customer.

**5.2.1.1 Order To Delivery Note Generation**

We consider Customer and Delivery data table and map them using (SONO + Item). We then

Code:

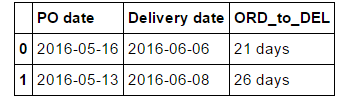
tmp = customer[['SONO', 'PO date', 'ITEM']]

bill = bill.merge(tmp, how='left', left\_on=['Sales ord', 'delivery Item'], right\_on=['SONO', 'ITEM'])

order\_to\_delivery = bill[['PO date', 'Delivery date']]

order\_to\_delivery.loc[:, 'ORD\_to\_DEL'] = order\_to\_delivery['Delivery date'] - order\_to\_delivery['PO date']

order\_to\_delivery.head(2)

****

Implementation of function to segregate according to increasing order in the difference between dates

def display\_days\_difference(tmp, title):

df = pd.DataFrame(columns=['Days', 'cum'])

for i in xrange(7):

df.loc[i] = [str(i) + ' days', tmp[tmp['Days'] <= i]['Count'].sum()]

df.loc[7] = ['< 1 week', tmp[tmp['Days'] <= 7]['Count'].sum()]

df.loc[8] = ['< 2 weeks', tmp[tmp['Days'] <= 14]['Count'].sum()]

df.loc[9] = ['< 3 weeks', tmp[tmp['Days'] <= 21]['Count'].sum()]

df.loc[10] = ['< 1 month', tmp[tmp['Days'] <= 30]['Count'].sum()]

df.loc[11] = ['> 1 month', tmp['Count'].sum()]

df['Count'] = df['cum']

for i in xrange(11, 0, -1):

df.loc[i, 'Count'] = df.loc[i, 'Count'] - df.loc[i - 1, 'Count']

Finding percentage and cumulative percentage of delivery note generation under each of the category

df['percentage'] = 100\*df.Count/df.Count.sum()

df['cum percent'] = df.percentage.cumsum()

Display dataframe as table

from IPython.display import display

print "TABLE"

display(df)

Displaying line graphs and corresponding bar graphs on the same axes plane.

for (i, j) in [('Count', 'percentage'), ('cum', 'cum percent')]:

fig, ax1 = plt.subplots()

ax2 = ax1.twinx()

ax1.bar([x for x in xrange(len(df))], df[i], width=.5, label=i)

ax2.plot([x for x in xrange(len(df))], df[j], color='red', marker='o')

for k in xrange(len(df)):

ax1.text(k, df[i][k] + 300, str(float(df[j][k]))[:4], horizontalalignment='center')

plt.xticks([x for x in xrange(len(df))], df['Days'])

h1, l1 = ax1.get\_legend\_handles\_labels()

h2, l2 = ax2.get\_legend\_handles\_labels()

ax1.legend(h1+h2, l1+l2, loc=0)

plt.title(title)

plt.show()

## Table For Order To Delivery Note Generation

## 

## 5.2.1.2 Delivery To Invoice Generation.

Finding difference between delivery date and bill date :

## delivery\_to\_bill = bill[['Bill date', 'Delivery date']]

## delivery\_to\_bill.loc[:, 'DEL\_TO\_BILL'] = delivery\_to\_bill['Bill date'] - delivery\_to\_bill['Delivery date']

## tmp = delivery\_to\_bill['DEL\_TO\_BILL'].value\_counts().reset\_index()

## tmp['Days'] = tmp['Days'].apply(lambda x: x.days)

## tmp.sort\_values(by=['Days'], inplace=True)

## Calling the function display\_days\_difference to display the graph :

## display\_days\_difference(tmp, 'Service Efficiency: Allocation To Invoice')

Table For Delivery to Invoice Generation

## 

## 5.2.1.3 Invoice To Consignment Generation

## Determining time interval between invoice to consignmemt and calling function to display respective table and corresponding graphs.

bill\_to\_GC = bill[['Bill date', 'GC date']]

bill\_to\_GC.loc[:, 'BILL\_TO\_GC'] = bill\_to\_GC['GC date'] - bill\_to\_GC['Bill date']

bill\_to\_GC.loc[:, 'BILL\_TO\_GC'] = bill\_to\_GC['BILL\_TO\_GC'].dt.components.days

tmp = bill\_to\_GC['BILL\_TO\_GC'].value\_counts().reset\_index()

tmp.columns = ['Days', 'Count']

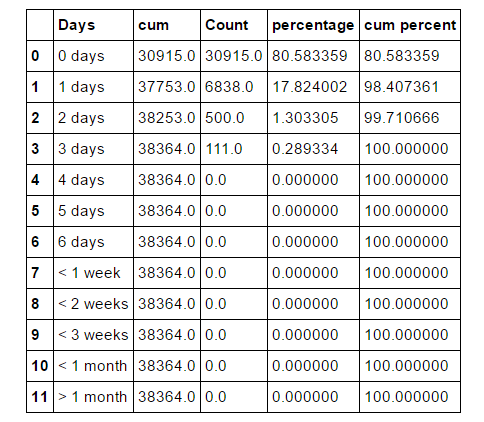
tmp['Days'] = tmp['Days'].apply(lambda x: x.days)

tmp.sort\_values(by=['Days'], inplace=True)

tmp = tmp.reset\_index().drop('index', axis=1)

display\_days\_difference(tmp, 'Service Efficiency: Invoice to Consignment')

Table For Invoice To Consignment Generation



## 5.2.1.4 Consignment To Receipt

## Determining time interval between consignmemt to receand calling function to display respective table and corresponding graphs.

GC\_to\_recpt = bill[['GC date', 'Recpt date']]

GC\_to\_recpt.loc[:, 'GC\_TO\_RECPT'] = GC\_to\_recpt['Recpt date'] - GC\_to\_recpt['GC date']

tmp = GC\_to\_recpt['GC\_TO\_RECPT'].value\_counts().reset\_index()

tmp.columns = ['Days', 'Count']

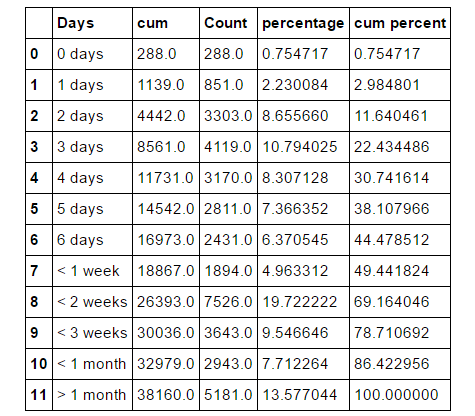
tmp['Days'] = tmp['Days'].apply(lambda x: x.days)

tmp.sort\_values(by=['Days'], inplace=True)

tmp = tmp.reset\_index().drop('index', axis=1)

display\_days\_difference(tmp, 'Service Efficiency: Consignment to Reciept')

Table for Consignment to Receipt



**5.2.2 Customer wise, Region wise and Model wise sales analysis**

**5.2.2.1 Customer wise sales**

The next step involves determining the customers providing maximum profit to the company. Here we display the top 50 value adding customers.

Finding the total price corresponding to each customer

customer['Total\_Price'] = customer['ORD\_QTY'] \* customer['Price']

customer.head(1)

tmp = customer[['customer code', 'Total\_Price']]

tmp = tmp.groupby(['customer code']).sum().reset\_index()

tmp.sort\_values(by='Total\_Price', ascending=False, inplace=True)

tmp = tmp.reset\_index(drop=True)

Considering top 50 profit giving customers

tmp2 = tmp.loc[50:]

tmp = tmp.loc[:49]

tmp.loc[50] = ['Others', tmp2['Total\_Price'].sum()]

Code for plotting the bar graph

sns.barplot(x='customer code', y='Total\_Price', data=tmp)

plt.xticks(rotation=60)

plt.title("Top customers")

plt.plot()

# **5.2.2.2 Region wise Sales**

Similar to customer wise sales, we now analyse on the profit incurred by the company in various regions within India

df = customer.groupby('Region')['Total\_Price'].sum().reset\_index()

Sorting by income

df = df.sort\_values(by='Total\_Price', ascending=False)

df = df.reset\_index(drop=True)

Plotting the region wise profit graph

sns.barplot(x = 'Region', y= 'Total\_Price', data=df)

plt.xticks(rotation=60)

plt.plot()

**5.2.2.3 Model Wise Sales**

We now explicate on the profit incurred by various model releases by the company .

model\_demand = material\_master

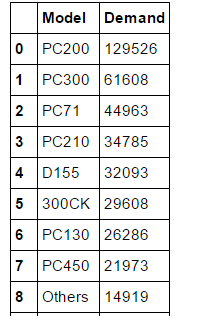
model\_demand = model\_demand.groupby('Model').sum().reset\_index()

model\_demand.drop(['safety stock'], axis=1, inplace=True)

model = model\_demand.sort\_values(by='Demand', ascending=False)

model = model.reset\_index(drop=True)

model.head(15)

****

**5.2.3 SEASON WISE DEMAND**

Here we try to find out how the demand for various products in the company vary with the seasons.

First, we extract the month from ‘PO date’ of customer\_order and find the profit for each month. Then, the months are grouped into seasons as mentioned below.

We consider 5 seasons:

1. Winter - November to Feb (11,12,1,2)

2. Spring - March to April (3,4)

3. Summer - May to June (5,6)

4. Monsoon - July to August (7,8)

5. Autumn - September to October (9,10)

Then, Total\_Profit for each month is found.

# Grouping the months into seasons and finding the profit in each season.

def test (row):

if (row['month'] ==1)|(row['month']==2) :

return 'winter'

if (row['month'] ==3)|(row['month'] ==4) :

return 'spring'

if (row['month'] ==5)|(row['month'] ==6) :

return 'summer'

if (row['month']==7)|(row['month']==8) :

return 'monsoon'

if (row['month'] ==9)|(row['month'] ==10) :

return 'autumn'

if (row['month'] ==11)|(row['month'] ==12) :

return 'winter'

df['season'] = df.apply (lambda row: test(row),axis=1)

df = df.drop(['month'], axis=1)

df = df.groupby('season').sum().reset\_index()

df['percentage'] = 100\*df.Total\_Price/df.Total\_Price.sum()

|  | **season** | **Total\_Price** | **percentage** |
| --- | --- | --- | --- |
| **0** | autumn | 115282947 | 16.943341 |
| **1** | monsoon | 190583361 | 28.010378 |
| **2** | spring | 136098629 | 20.002659 |
| **3** | summer | 229683886 | 33.757052 |
| **4** | winter | 8753860 | 1.286570 |

**5.2.4 Finding Materials OUT\_OF\_STOCK**

In this part of data analysis, we find materials which are out of stock, in stock, and materials for which only few stocks are left.

First, we will merge material\_master and stock\_master to obtain safety stock and quantity for each materials in one table. Unwanted columns are dropped.

Then, the materials are grouped in to IN\_STOCK, OUT\_OF\_STOCK and NEW\_STOCKS\_NEEDED as show below.

# checking for various conditions

def test (row):

if row['safety stock'] < row['Quantity'] :

return 'IN\_STOCK'

if row['safety stock'] > row['Quantity'] :

return 'OUT\_OF\_STOCK'

if row['safety stock'] == row['Quantity'] :

return 'NEW\_STOCK\_NEEDED'

return 'Other'

stock\_df['status'] = stock\_df.apply (lambda row: test (row),axis=1)

|  | **Material code** | **Model** | **safety stock** | **Quantity** | **status** |
| --- | --- | --- | --- | --- | --- |
| **0** | 01010-61435I. | PC450 | 8 | 86 | IN\_STOCK |
| **1** | 01010-61455I. | D65 | 1 | 4 | IN\_STOCK |
| **2** | 01010-61635I. | GD511 | 1 | 34 | IN\_STOCK |
| **3** | 01010-61645I. | D475 | 8 | 4 | OUT\_OF\_STOCK |
| **4** | 01010-61650I. | HD465 | 5 | 8 | IN\_STOCK |

After grouping the materials into out\_of\_stock, in\_stock and new\_stocks\_neeeded, we will focus our anlalysis on the materials which are out\_of\_stock.

#Retriving materials which are out of stock from stock\_df table

ofs=stock\_df.loc[stock\_df['status'] == 'OUT\_OF\_STOCK']

#Finding how much stock is needed

ofs['Stock Needed'] = ofs['safety stock'] + ofs['Quantity']

#sorting by the value of stock needed(difference)

ofs=ofs.sort\_values(by='Stock Needed',ascending=False)

ofs=ofs.reset\_index(drop=True)

ofs.head(5)

|  | **Material code** | **Model** | **safety stock** | **Quantity** | **status** | **Stock Needed** |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | 07063-51210I | PC200 | 229 | 4 | OUT\_OF\_STOCK | 233 |
| **1** | 01010-61865I. | D65 | 113 | 85 | OUT\_OF\_STOCK | 198 |
| **2** | 6136-51-5121I | D65 | 172 | 12 | OUT\_OF\_STOCK | 184 |
| **3** | 20Y-60-21311I | PC200 | 183 | 1 | OUT\_OF\_STOCK | 184 |
| **4** | 07063-01100I | D275 | 151 | 5 | OUT\_OF\_STOCK | 156 |

**5.2.5 CHECKING POSSIBLE REASONS FOR DELAY FROM**

**CONSIGNMENT TO DELIVERY**

In section 5.2.1, we found the time taken between consignment to delivery(receipt). Here we are trying to find out the possible reasons for delay from consignment to delivery.

First, we extract consignment to delivery time from bill data frame. Then the table is grouped according to consignment details.

GC\_to\_recpt = bill[['GC date', 'Recpt date', 'Consignment details']] GC\_to\_recpt.loc[:, 'GC\_TO\_RECPT'] = (GC\_to\_recpt['Recpt date'] – GC\_to\_recpt['GC date']) .dt.days GC\_to\_recpt = GC\_to\_recpt.sort\_values(by='GC\_TO\_RECPT', ascending=False) GC\_to\_recpt = GC\_to\_recpt.groupby('Consignment details') .sum(). reset\_index() .sort\_values(by='GC\_TO\_RECPT', ascending=False).reset\_index(drop=True)

|  | **Consignment details** | **GC\_TO\_RECPT** |
| --- | --- | --- |
| **0** | 442176696 GATI 21-07-2016 | 3922 |
| **1** | 22364639 NECC 26-07-2016 | 3834 |
| **2** | 442176603 GATI 23-07-2016 | 2313 |
| **3** | 21041 UNITY 11-07-2016 | 1895 |
| **4** | LOCAL COLLECTION | 1682 |

**5.2.6 QUERY DRILL DOWN**

The next part of data analysis is query drill down. Here, we will check if the order fluctuates as per the date, as we know there would be season for most orders. Like at the end of the fiscal year or start of new fiscal year.

Here we will first find the profit and demand for each month of the year and then the same for each day of the month. For this we are using customer data frame.

#For month of the year

df = customer[['PO date', 'Total\_Price', 'ORD\_QTY']] df.loc[:,'month'] = df['PO date'].apply(lambda x: x.month) df = df.drop(['PO date'], axis=1) df = df.groupby(['month']).sum().reset\_index()

#For day of the month

df = customer[['PO date', 'Total\_Price', 'ORD\_QTY']] df.loc[:,'day'] = df['PO date'].apply(lambda x: x.day) df = df.drop(['PO date'], axis=1) df = df.groupby(['day']).sum().reset\_index()

Month of the year

|  | **month** | **Total\_Price** | **ORD\_QTY** |
| --- | --- | --- | --- |
| **0** | 1 | 534144 | 191 |
| **1** | 2 | 3298831 | 1099 |
| **2** | 3 | 27143439 | 1439 |
| **3** | 4 | 108955190 | 38685 |
| **4** | 5 | 120713266 | 38275 |
| **5** | 6 | 108970620 | 38315 |
| **6** | 7 | 104511662 | 34957 |
| **7** | 8 | 86071699 | 29733 |
| **8** | 9 | 115282947 | 35748 |
| **9** | 11 | 2934394 | 215 |
| **10** | 12 | 1986491 | 378 |

|  | **day** | **Total\_Price** | **ORD\_QTY** |
| --- | --- | --- | --- |
| **0** | 1 | 12893564 | 3604 |
| **1** | 2 | 30516498 | 8774 |
| **2** | 3 | 24459472 | 5732 |
| **3** | 4 | 26489072 | 7810 |
| **4** | 5 | 26719262 | 10069 |

Day of the month

**5.2.7 DEMAND FORECASTING**

The next step of data analysis (predictive analysis) is demand forecasting. Here we are predicting the future demand of materials from the previous demands.

From sales past demand table we have demand for 36 months. We are implementing “Regression Analysis” to predict the future demands.

First, we will check for periodicity. This is done by analysing the subplots for materials. Then if periodicity is present we will proceed with prediction by applying various regression methods.

Instead of data per month, we divide the data as to have data per 3 months. This allows us to predict the demand for the next three months which would be aggregated better than data per month. Since, we had demand for 36 months, know it is divided into 12 intervals of three months.(Q\_0 to Q\_11)

#Grouping the demand into interval of three months

df = pd.DataFrame() df['Material code'] = sales\_past\_demand['Material code'] for i in range(1, 37, 3):

df['Q\_' + str((i-1)/3)] = sales\_past\_demand[['DEM' + str(x) for x in range(i, i + 3)]]. sum(axis=1)

Demand for interval of 3 months

|  | **Material code** | **Q\_0** | **Q\_1** | **Q\_2** | **Q\_3** | **Q\_4** | **Q\_5** | **Q\_6** | **Q\_7** | **Q\_8** | **Q\_9** | **Q\_10** | **Q\_11** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 01124-82250I. | 0 | 6 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 0 | 3 |
| **1** | 01310-21216I. | 100 | 102 | 112 | 83 | 87 | 39 | 173 | 74 | 59 | 60 | 40 | 75 |
| **2** | 01435-00814I. | 0 | 2 | 0 | 0 | 0 | 0 | 8 | 0 | 1 | 0 | 0 | 3 |
| **3** | 01435-00865I. | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **4** | 01435-01025I. | 2 | 1 | 7 | 20 | 1 | 1 | 14 | 9 | 4 | 6 | 12 | 13 |
| **5** | 015424KB. | 0 | 1 | 6 | 5 | 7 | 3 | 1 | 4 | 2 | 0 | 0 | 1 |

We split the data into quarters (3 months). Since we have data of 36 months, this gives us 12 quarters. So, the idea is to train the model on 11 quarters so that it is able to predict the 12th quarter.

Now we have our desired inputs and desired outputs. But it wouldn't make sense to train the machine learning algorithm and test it on the same data, so we'll now split our data into training and tests sets (70% - 30%).

#Dividing the data into train and test set

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(df.drop('Q\_11', axis=1), df.Q\_11,

test\_size=.3, random\_state=42)

Finding the most accurate Regression method for demand prediction:

1. ElasticNet regression

from sklearn.model\_selection import GridSearchCV

from sklearn.linear\_model import ElasticNet

param = {'alpha': [1.0, 2, 5, 10, 50, 100, 1000], 'normalize': [True, False]}

reg = GridSearchCV(ElasticNet(), param)

reg.fit(X\_train, y\_train)

reg.score(X\_test, y\_test)

Accuracy = 0.8616707598397878

2. Lasso

from sklearn.model\_selection import GridSearchCV

from sklearn.linear\_model import Lasso

param = {'alpha': [1.0, 2, 5, 10, 50, 100, 1000], 'normalize': [True, False]}

reg = GridSearchCV(Lasso(), param)

reg.fit(X\_train, y\_train)

reg.score(X\_test, y\_test)

Accuracy = 0.84968164809060198

3. Ridge

from sklearn.model\_selection import GridSearchCV

from sklearn.linear\_model import Ridge

param = {'alpha': [1.0, 10, 100], 'normalize': [True, False], 'solver' : ['auto', 'svd', 'cholesky', 'lsqr', 'sparse\_cg', 'sag']}

reg = GridSearchCV(Ridge(), param)

reg.fit(X\_train, y\_train)

reg.score(X\_test, y\_test)

Accuracy = 0.87221892970289294

4. AdaboostRegressor

from sklearn.model\_selection import GridSearchCV

from sklearn.ensemble import AdaBoostRegressor

param = {'n\_estimators': [50, 100, 500], 'loss': ['linear', 'square', 'exponential']}

reg = GridSearchCV(AdaBoostRegressor(), param)

reg.fit(X\_train, y\_train)

reg.score(X\_test, y\_test)

Accuracy = 0.77895308772704985

Picking up the best model:

Since we got the highest accuracy with the Ridge model, we'll use it to do our predictions. Now using the Ridge regression lets train and then test on our data set.

#Training and testing the data using ridge regression

from sklearn.linear\_model import Ridge

reg = Ridge(alpha=100, normalize=False, solver='sag')

reg.fit(X\_train, y\_train)

We will perform two iterations of ridge regression to predict the demand for future two quarters.

1. Iteration one:

In first iteration, we will split the data into test(X) and train(y) sets. Where, the ridge model is trained using train set and test the predicted values using test set.

We will assign y as the required output (Q\_11) and the others as X the input (Q\_1, Q\_2 ... Q\_10).

y = df['Q\_11']

X = df.drop(['Q\_11'], axis = 1)

Now we have our desired inputs and desired outputs. But it wouldn't make sense to train the machine learning algorithm and test it on the same data, so we'll now split our data into training and tests sets (70% - 30%).

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

Now using the Ridge regression lets train and then test on our data set

from sklearn.linear\_model import Ridge

reg = Ridge(alpha=100, normalize=False, solver='sag')

reg.fit(X\_train, y\_train)

Now our model is trained on our data, we will try to predict the test set.

pred = reg.predict(X\_test)

tempdf = pd.DataFrame()

tempdf.loc[:, 'real'] = y\_test

tempdf.loc[:, 'pred1'] = pred

tempdf.reset\_index(inplace=True, drop=True)

tempdf.head(10)

#Predicted values got from the implementation of Ridge regression

|  | **real** | **pred1** |
| --- | --- | --- |
| **0** | 0 | 0.649993 |
| **1** | 0 | 0.300784 |
| **2** | 0 | 0.191031 |
| **3** | 0 | 0.534626 |
| **4** | 0 | 3.720385 |
| **5** | 44 | 10.751811 |

Now, we will put these predicted values as demand for first future quarter i.e. Q\_12.

And the sales\_past\_demand table would look like;

|  | **Material code** | **Q\_0** | **Q\_1** | **Q\_2** | **Q\_3** | **Q\_4** | **Q\_5** | **Q\_6** | **Q\_7** | **Q\_8** | **Q\_9** | **Q\_10** | **Q\_11** | **Q\_12** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 01124-82250I. | 0 | 6 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 0 | 4 | 2 |
| **1** | 01310-21216I. | 100 | 102 | 112 | 83 | 87 | 39 | 173 | 74 | 59 | 60 | 40 | 77 | 75 |
| **2** | 01435-00814I. | 0 | 2 | 0 | 0 | 0 | 0 | 8 | 0 | 1 | 0 | 0 | 0 | 3 |
| **3** | 01435-00865I. | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **4** | 01435-01025I. | 2 | 1 | 7 | 20 | 1 | 1 | 14 | 9 | 4 | 6 | 12 | 5 | 13 |
| **5** | 015424KB. | 0 | 1 | 6 | 5 | 7 | 3 | 1 | 4 | 2 | 0 | 0 | 0 | 1 |
| **6** | 01580-10806I. | 2 | 12 | 3 | 2 | 5 | 4 | 4 | 8 | 0 | 0 | 6 | 0 | 5 |
| **7** | 01599-01011I. | 4 | 0 | 0 | 1 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **8** | 01602-20825I. | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **9** | 01643-31445. | 9 | 25 | 0 | 0 | 26 | 16 | 46 | 34 | 24 | 32 | 22 | 34 | 36 |
| **10** | 01643-32460. | 0 | 24 | 0 | 0 | 16 | 0 | 36 | 0 | 0 | 0 | 0 | 0 | 9 |

2. Iteration two:

Second iteration is similar to first iteration. But, here we will train the ridge model according to train data Q\_1 to Q\_11 and test the predicted value with test data Q\_12. The predicted values will be demand for second future quarter i.e. Q\_13.

And the sales\_past\_demand table after implementing second iteration would look like,

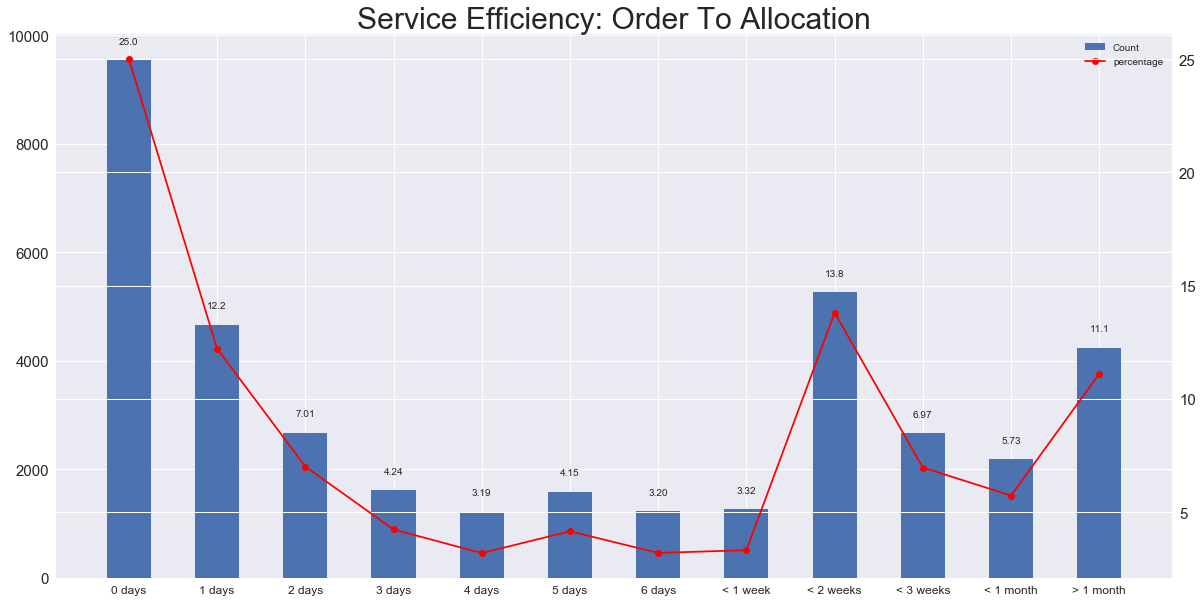
|  | **Material code** | **Q\_0** | **Q\_1** | **Q\_2** | **Q\_3** | **Q\_4** | **Q\_5** | **Q\_6** | **Q\_7** | **Q\_8** | **Q\_9** | **Q\_10** | **Q\_11** | **Q\_12** | **Q\_13** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 01124-82250I. | 0 | 6 | 0 | 0 | 0 | 0 | 8 | 0 | 0 | 0 | 0 | 4 | 2 | 1 |
| **1** | 01310-21216I. | 100 | 102 | 112 | 83 | 87 | 39 | 173 | 74 | 59 | 60 | 40 | 77 | 75 | 58 |
| **2** | 01435-00814I. | 0 | 2 | 0 | 0 | 0 | 0 | 8 | 0 | 1 | 0 | 0 | 0 | 3 | 1 |
| **3** | 01435-00865I. | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **4** | 01435-01025I. | 2 | 1 | 7 | 20 | 1 | 1 | 14 | 9 | 4 | 6 | 12 | 5 | 13 | 11 |
| **5** | 015424KB. | 0 | 1 | 6 | 5 | 7 | 3 | 1 | 4 | 2 | 0 | 0 | 0 | 1 | 1 |
| **6** | 01580-10806I. | 2 | 12 | 3 | 2 | 5 | 4 | 4 | 8 | 0 | 0 | 6 | 0 | 5 | 5 |
| **7** | 01599-01011I. | 4 | 0 | 0 | 1 | 2 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **8** | 01602-20825I. | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **9** | 01643-31445. | 9 | 25 | 0 | 0 | 26 | 16 | 46 | 34 | 24 | 32 | 22 | 34 | 36 | 29 |
| **10** | 01643-32460. | 0 | 24 | 0 | 0 | 16 | 0 | 36 | 0 | 0 | 0 | 0 | 0 | 9 | 3 |

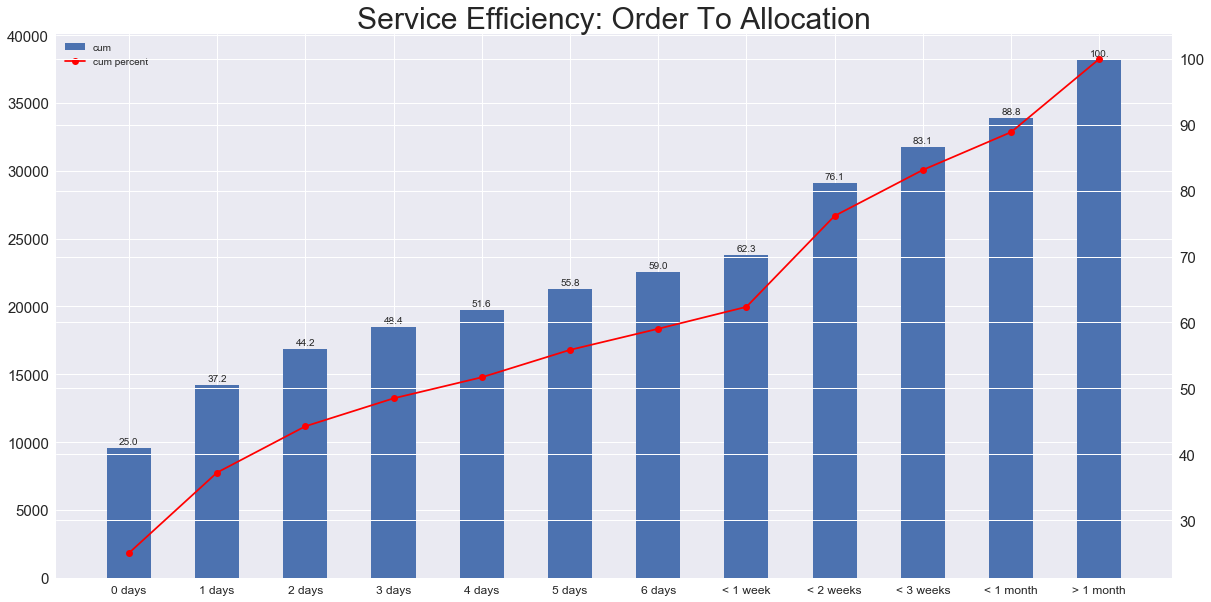
**CHAPTER 6**

**RESULTS AND SNAPSHOT**

**6.1 ORDER TO DELIVERY REPORT**

**6.1.1 Order To Allocation (Delivery Note Generation)**

****

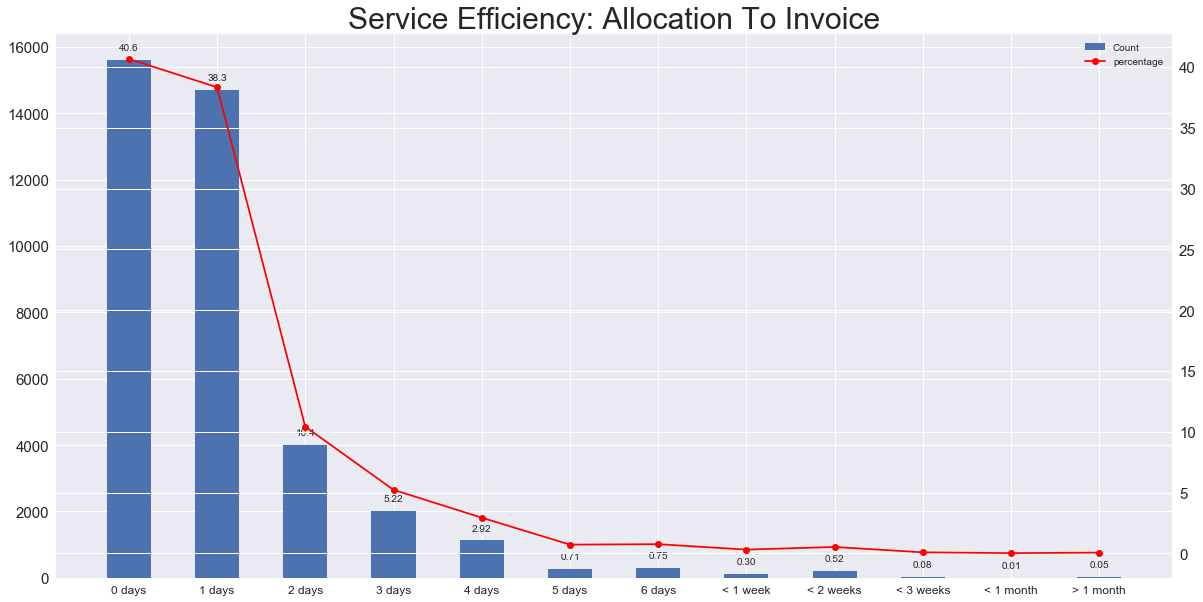


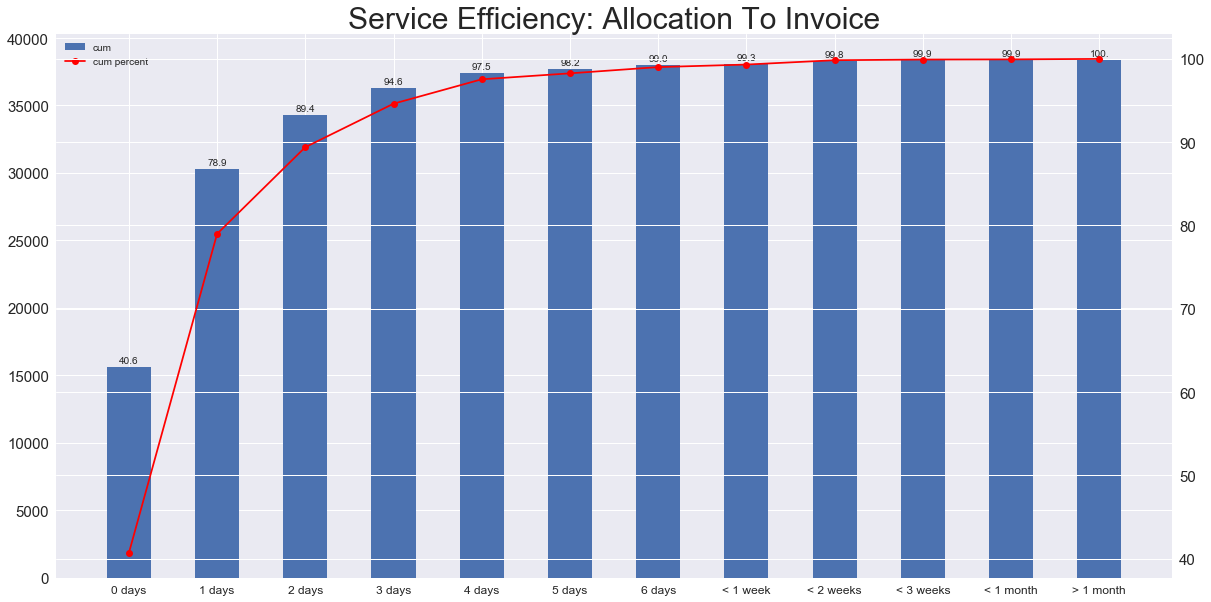
From the graphs above, the following observations are construed . We see that only around 50% of the alloction of goods ordered takes place within the first 3 days, and it takes around 1 month for 90% of the allocations to be completed.

This requires the company to focus on the main reasons behind this delay . One major reason could be the shortage of stocks in the company’s warehouses as and when ordered.

We thereby try to find out the stocks which are consistently in demand and those unavailable at the time of order . We also predict on the probable future demands for each of these materials .

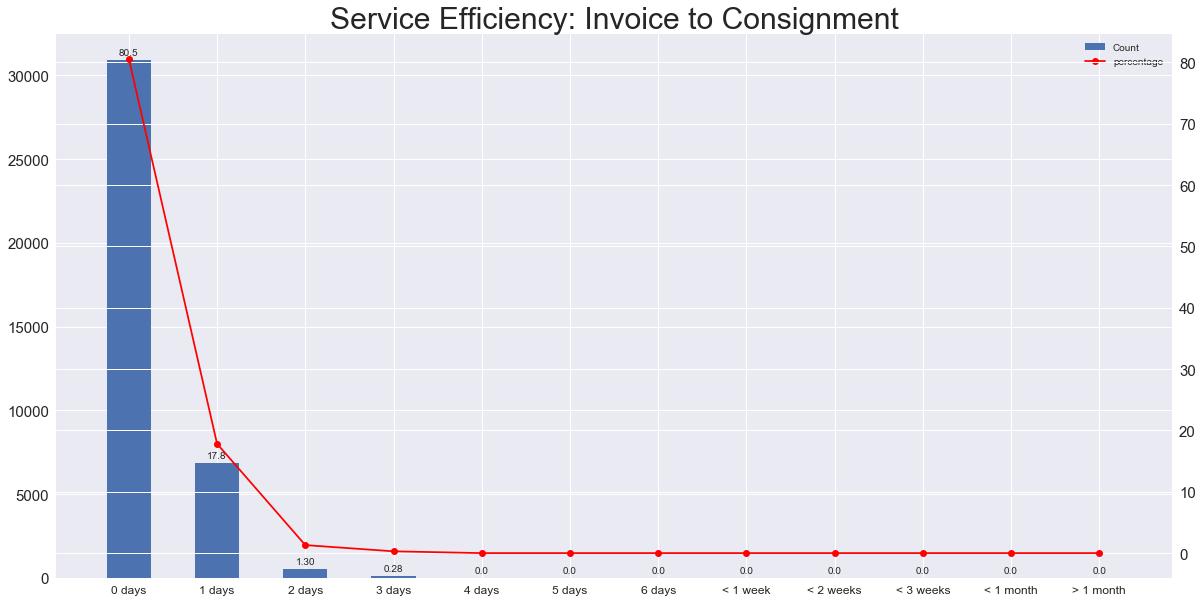
**6.1.2 Allocation To Invoice Generation**

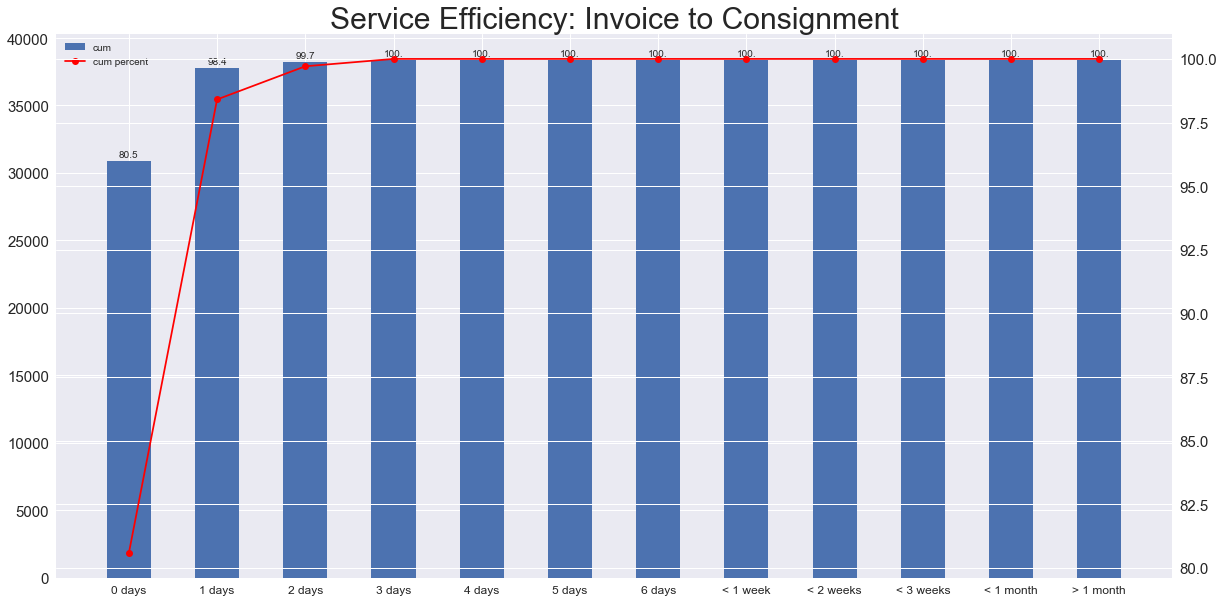




From the graphs above we infer that around 95% of the invoices for the orders made by the customers are generated within first 3 days . Hence no immediate measures are necessary on this aspect .

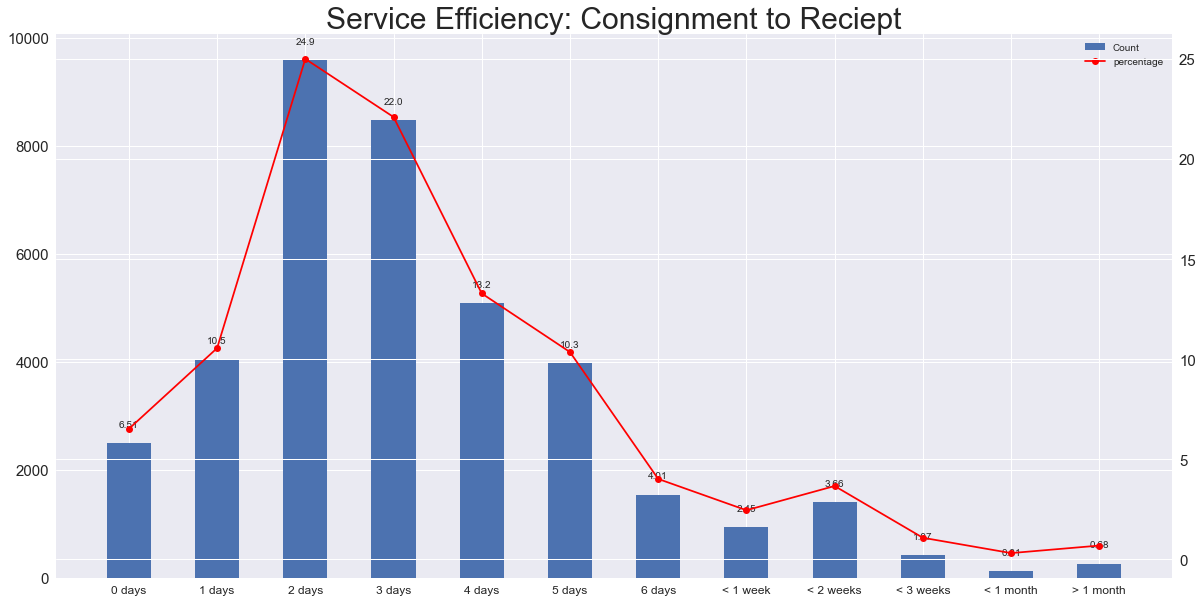
**6.1.3 Invoice To Consignment**

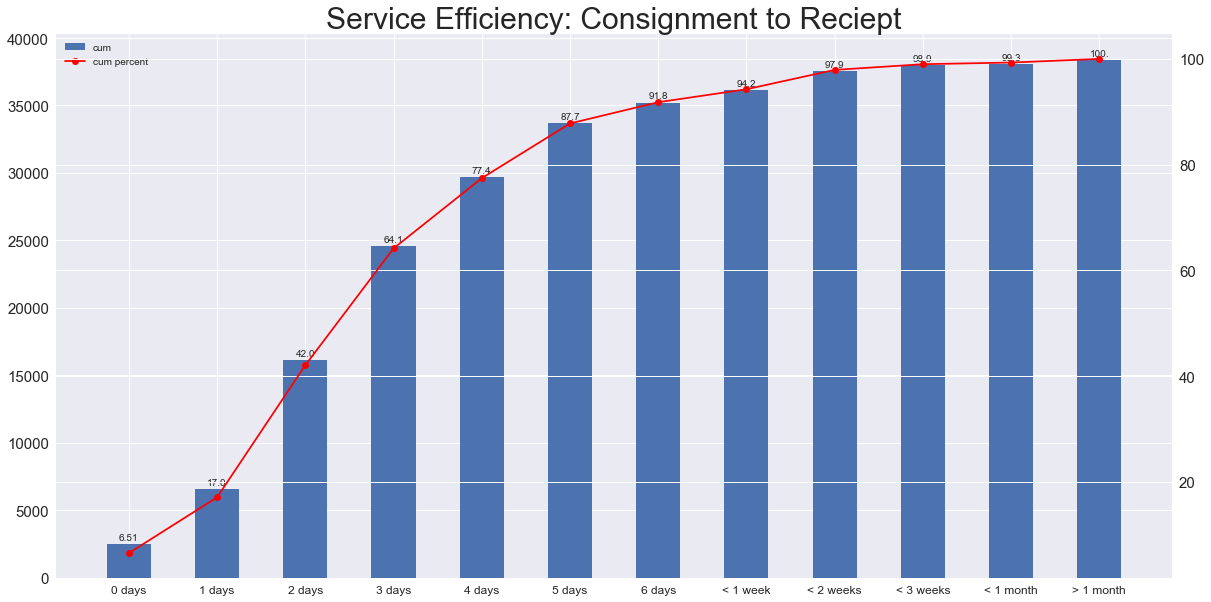




From the graphs above we infer that around 95% of the job is completed within 1 day. Hence no immediate measures are necessary on this aspect either.

**6.1.4 Consignment To Reaching Customers (Receipt Date)**

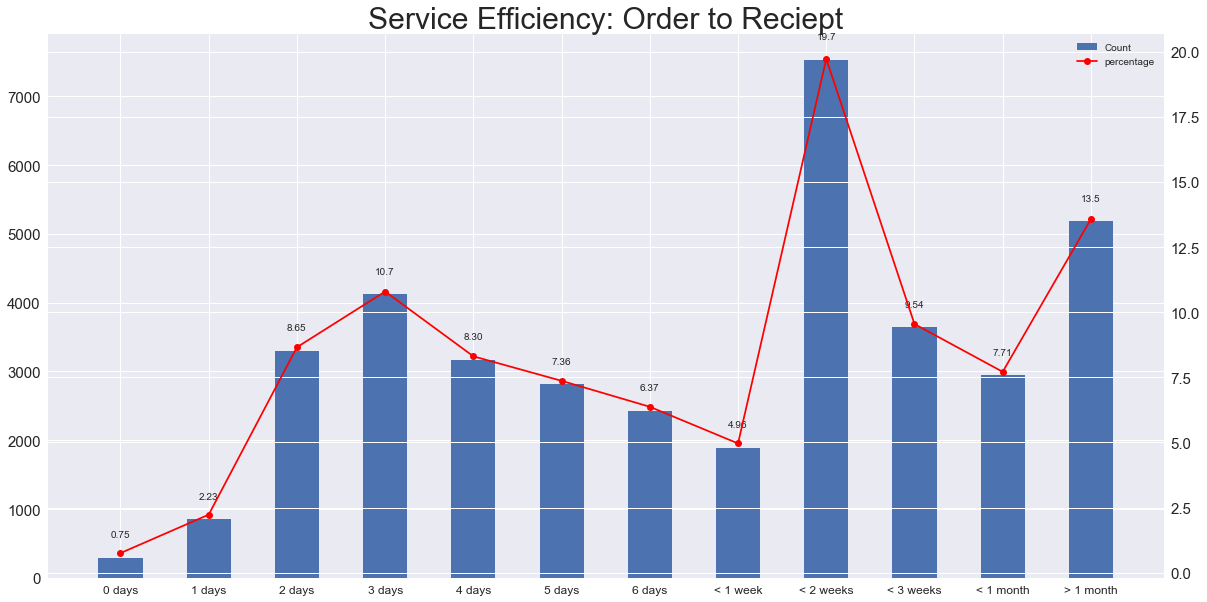


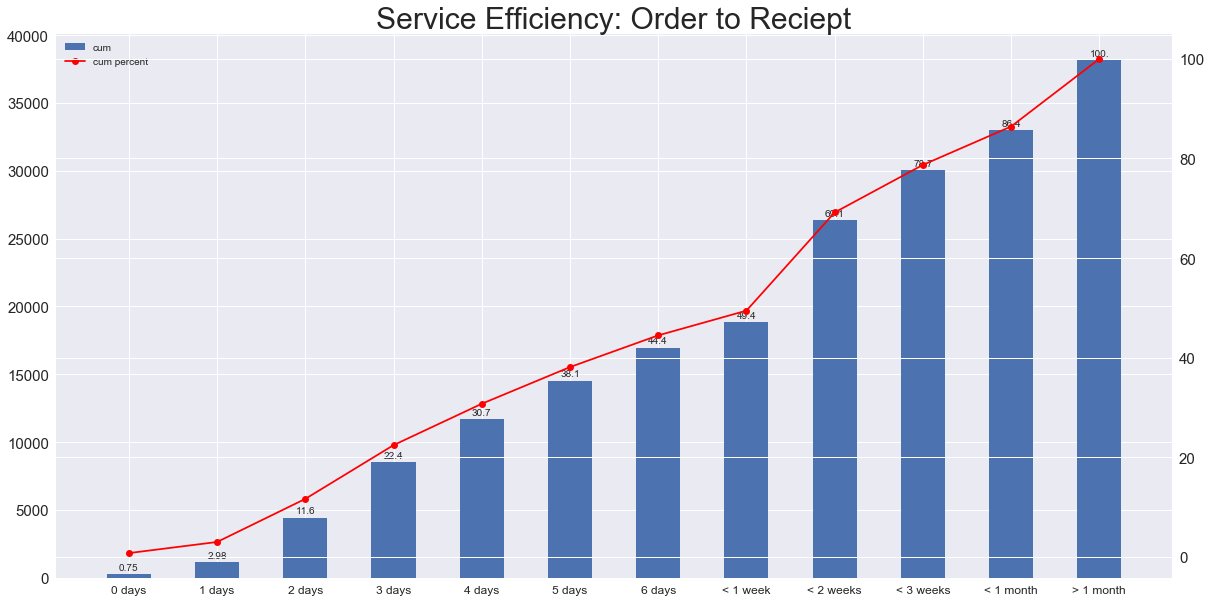


From the graphs plotted , we construe it takes more than 1 month for all of the deliveries to reach their respective customers .

We decipher that one of the major reasons could be the mode of delivery being used . We thereby determine the modes of consignment taking major amount of time , the company needs to find alternatives inorder to reduce the delivery time and increase their market value .

**6.1.5 Order To Receipt**



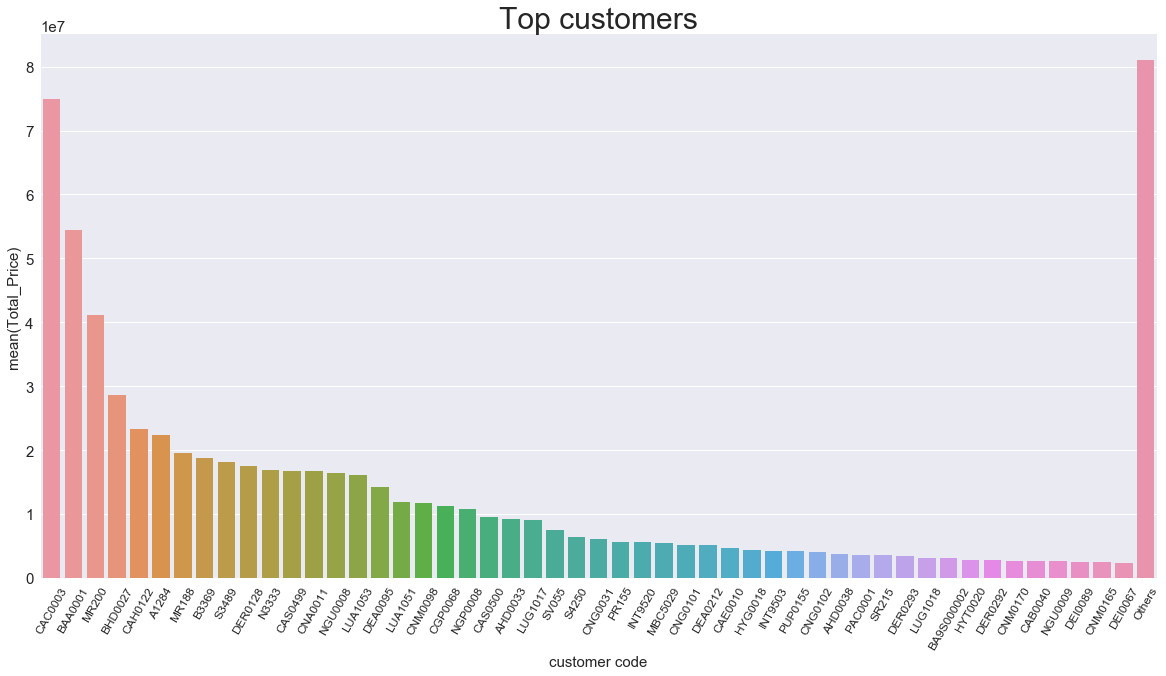


The above graph shows the total no. of days taken from the time order was placed by the customer to the day the order was delivered to him . It is observer that it takes around a month for all the orders to be delivered .

**6.2 CUSOMER WISE, REGION WISE, MATERIAL WISE AND**

**MODEL WISE PROFITS, AND SEASON WISE SALES.**

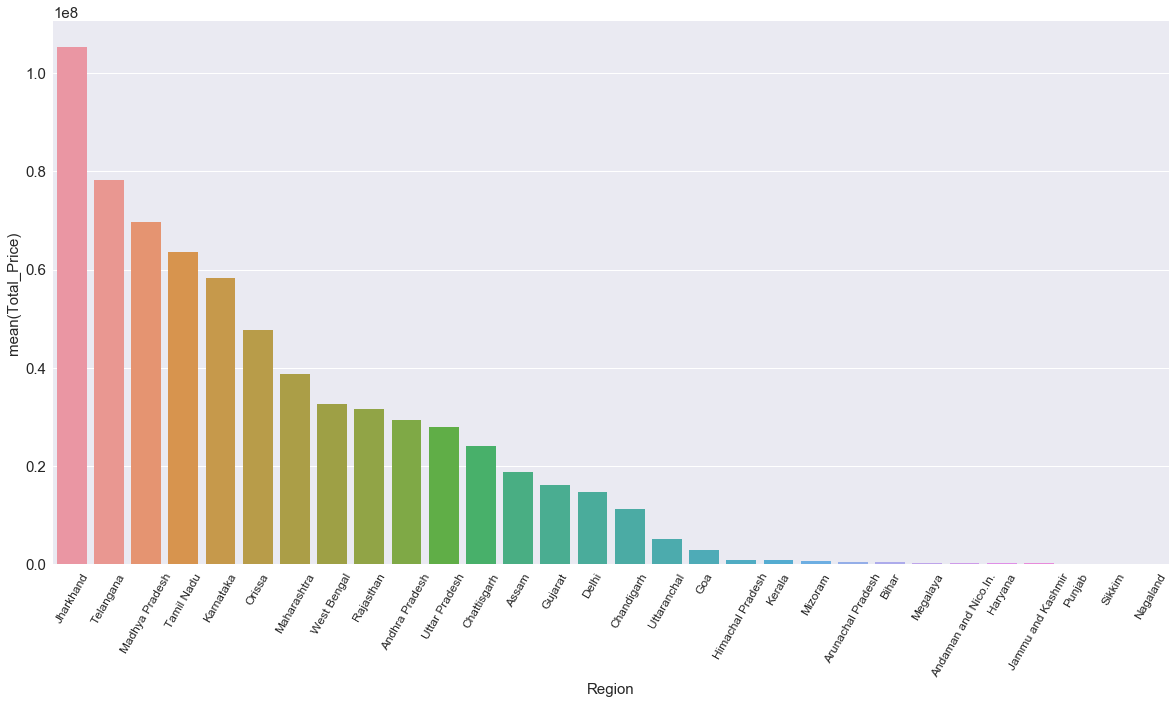
**6.2.1 Customer Wise Profit**



The above graph shows the top 50 value adding customers , facilitating the company’s profit. It is inferred that more than 90% of the customer side profit is from these top 50 customers , hence they are a major asset to the company.

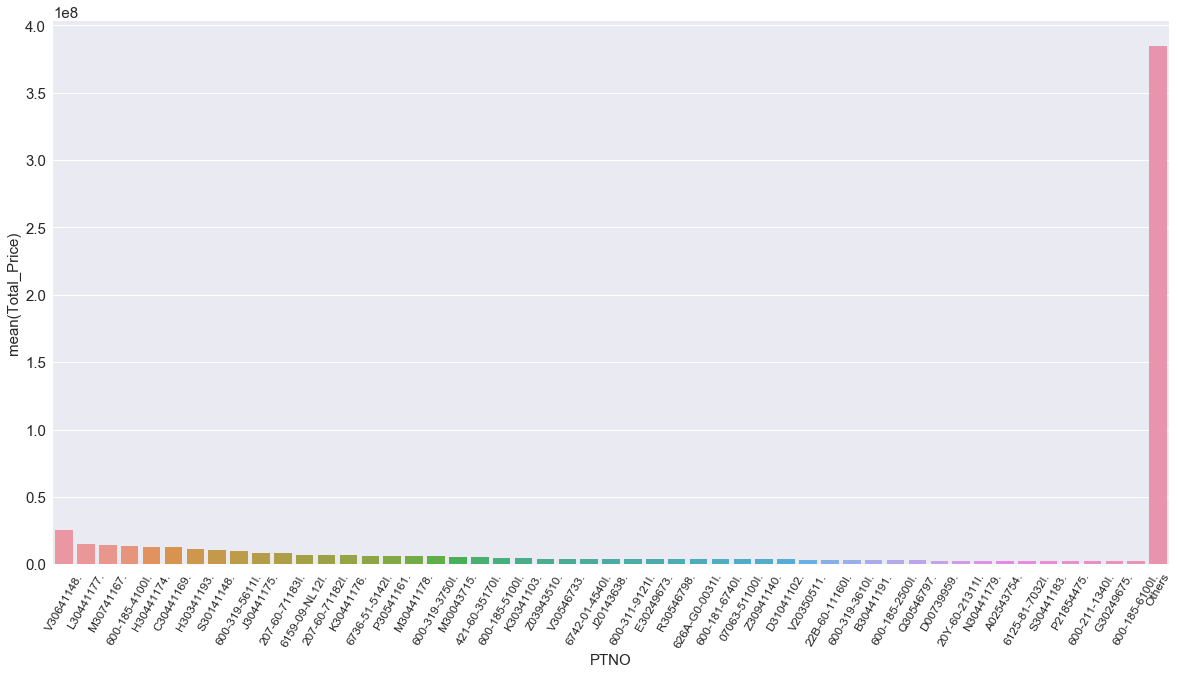
Hence it would be highly valuable if company focuses more on these customers to maintian their trust , so they continue to invest . At the same time, attracting the customers who are not currently investing majorly on the company’s products would be valuable for growth of company .

**6.2.2 Region Wise Profit**



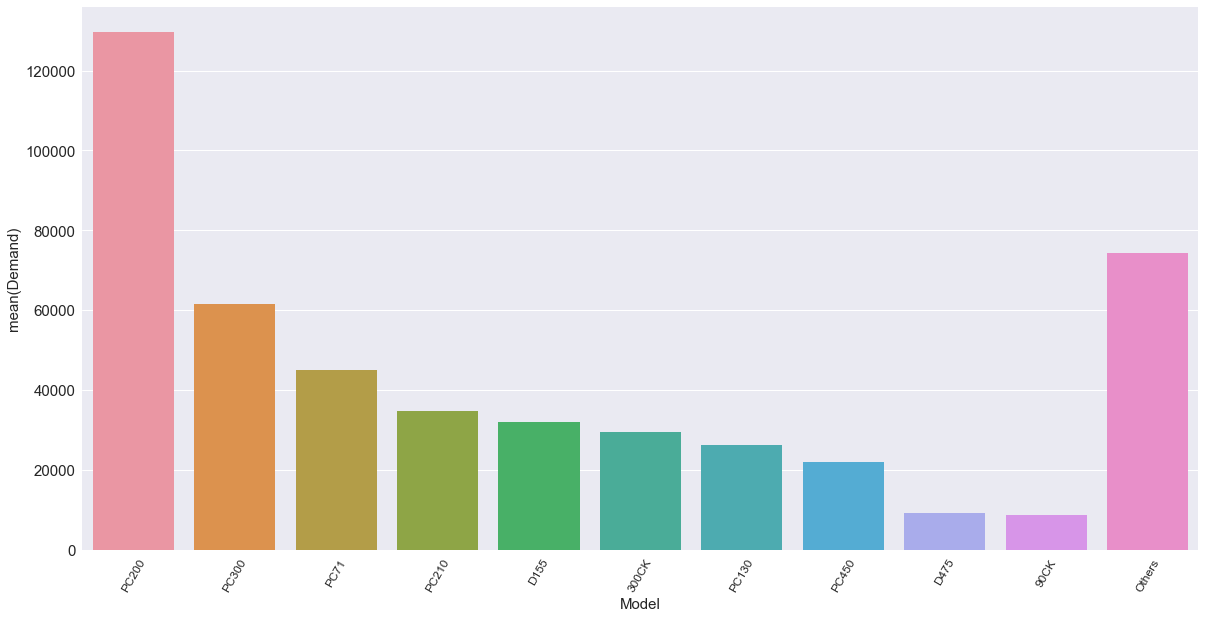
The graph shows the major regions investing on the company’s products.Similar measures are to taken, as suggested in case of customer sales.

**6.2.3 Material Wise Profit**



Above graph shows the major value adding materials produced by the company. The highest profit adding products should be noted and measures similar to customer sales are advised.

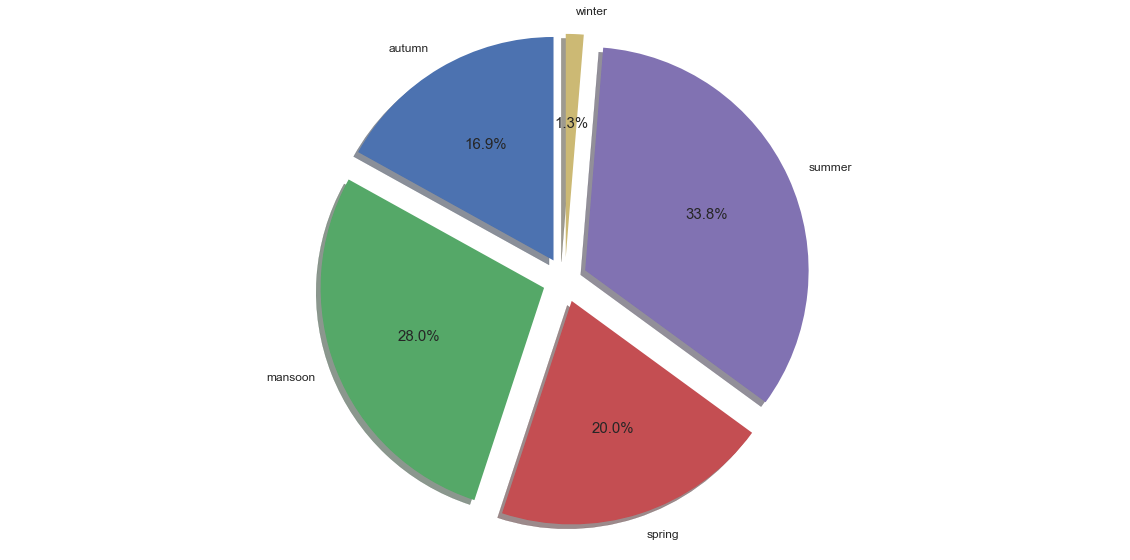
**6.2.4 Model Wise Profit**



The above graph shows the top 10 value adding models released by the company , facilitating to it’s profit. It is inferred that more than 90% of the profit is from these top 10 models , hence they are major value adders to the company, performing significantly in the market .

Hence it would be highly valuable if the company works on releasing more such models, with similar features and performance .

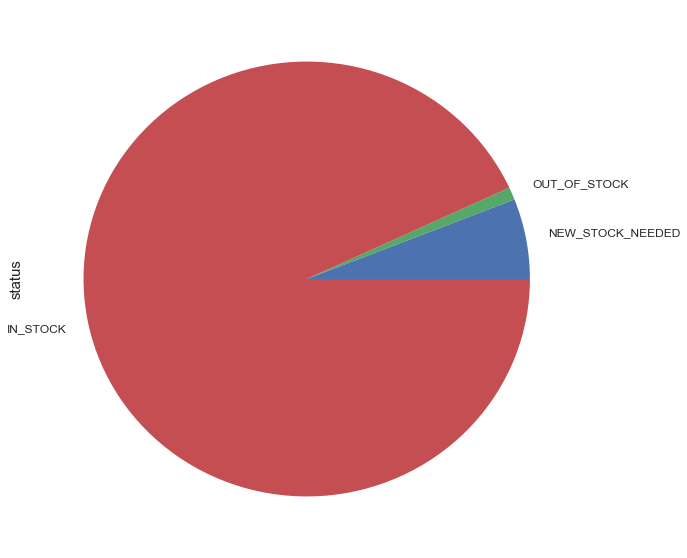
**6.2.5 Season Wise Sales**



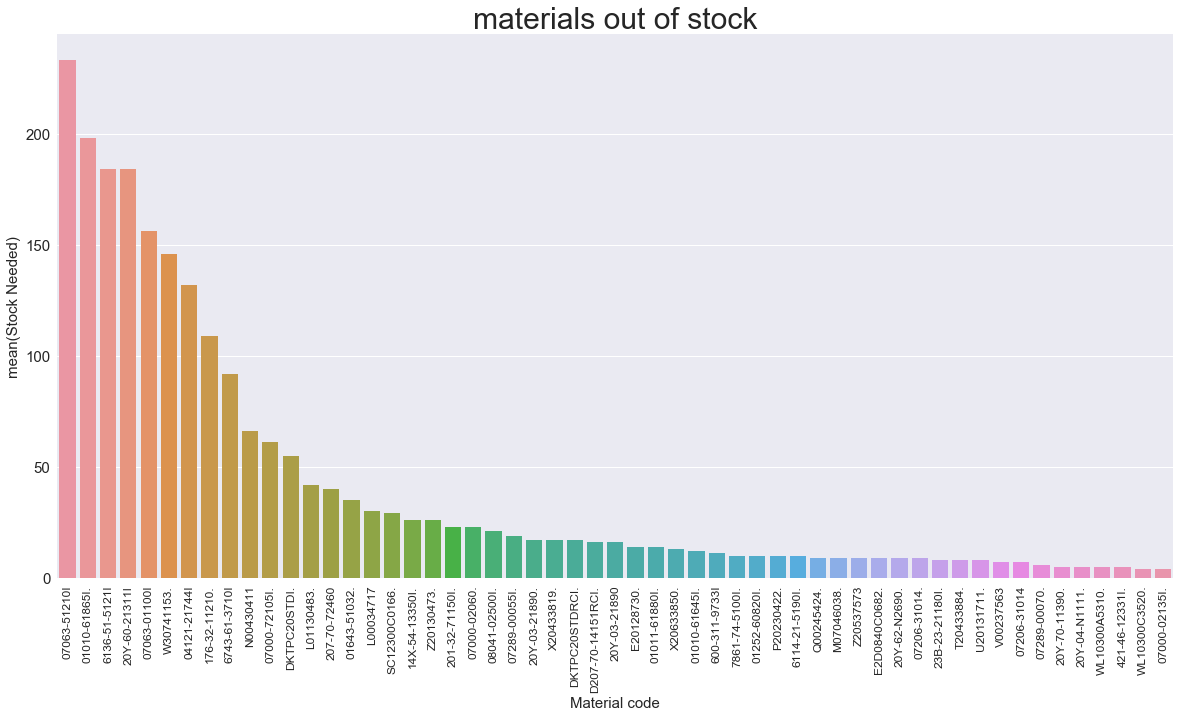
The above pie plot gives the season wise demand forecast.i,e how demand of the product in the company varies with seasons.

From the above graph we can see that sales in summer is maximum i,e 33.8% and sales in winter is minimum i,e 1.3%.From this we can infer that, since the salse will be maimum in summer, company should boost up its production to satisfy its customers.And since the salse is minmum in winter company should measures to increase the sales.

**6.3 FINDING MATERIALS WHICH ARE OUT OF STOCK**



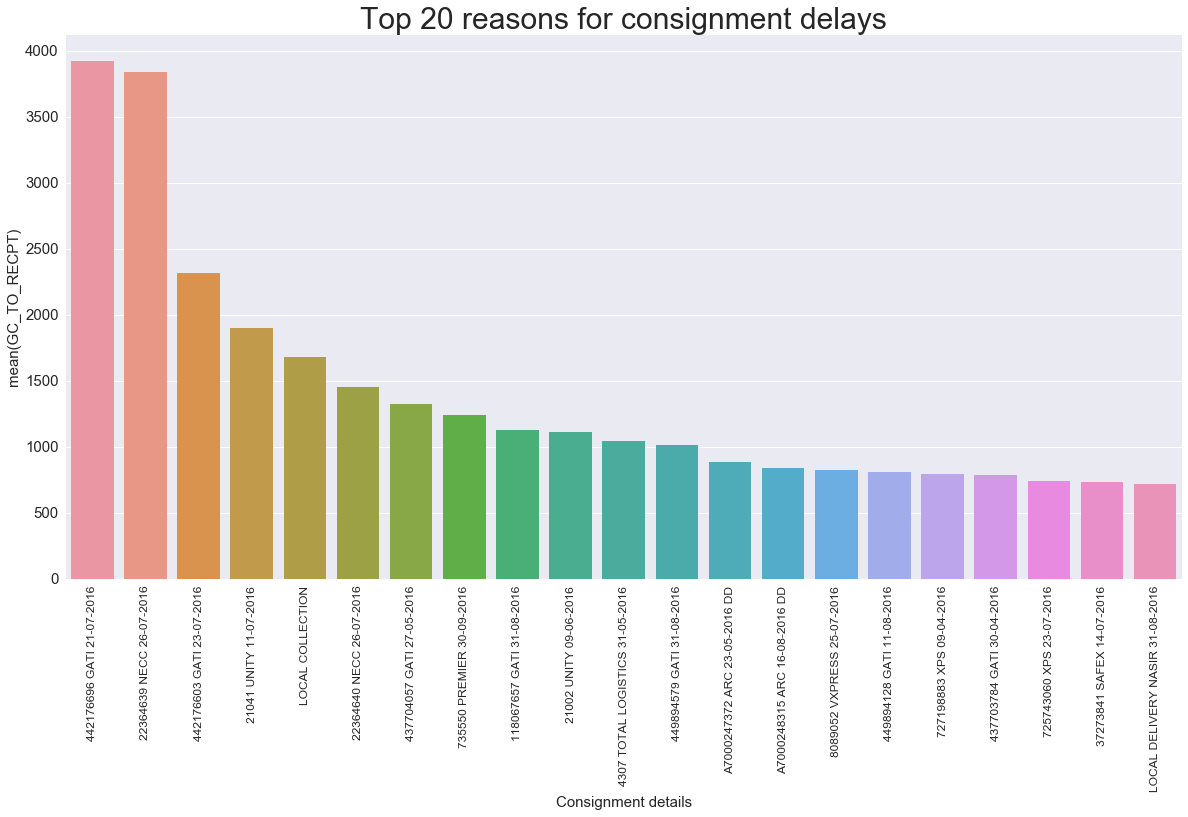
The above pie plot shows the amount of materials which are out\_of\_stock, in\_stock and few\_stock\_needed. From this we can infer that few materials are in out\_of\_stock and new stocks needed. Action must be taken from the company to overcome this.



The above graph shows the materials which are out\_of\_stock . Here we can see top 4 materials which are in out of stock with higher values .We can say that, these materials are needed immediately and also the stock of rest of the materials should be added.

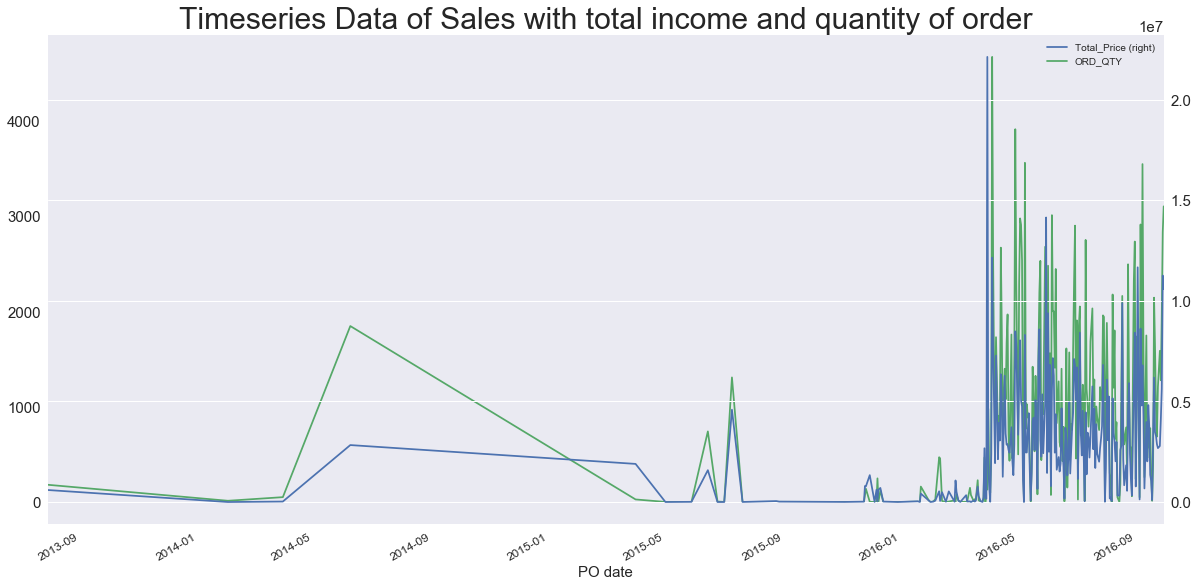
**6.4** **CHECKING POSSIBLE REASONS FOR DELAY FROM**

**CONSIGNMENT TO DELIVERY**



The above graphs shows he possible reasons for delay in consignment to delivery .The x asis gives the consignment details and y axis gives the mean time take by them. This can be reduced by making sure the materials are in stock and by improving the mode of courier services .

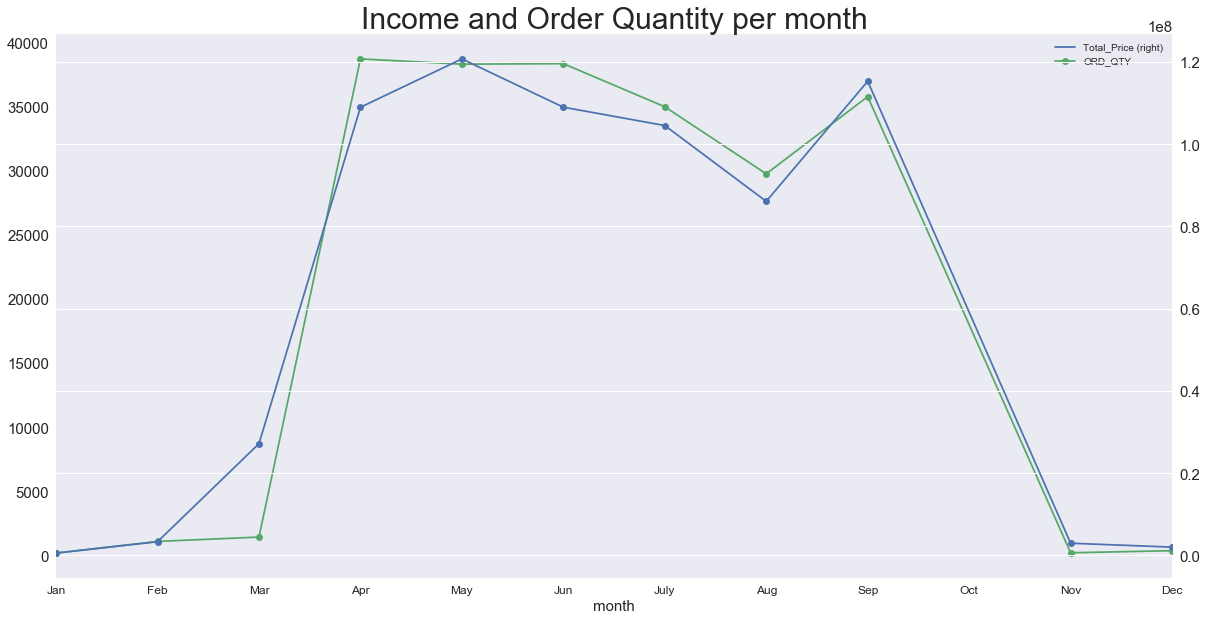
**6.5 QUERY DRILL DOWN**



The above graph gives the total income and quantiy of order with time. Here, almost 3 years of data. And from the above graph we can say that that is huge improvement in the salse from previous year to next year.

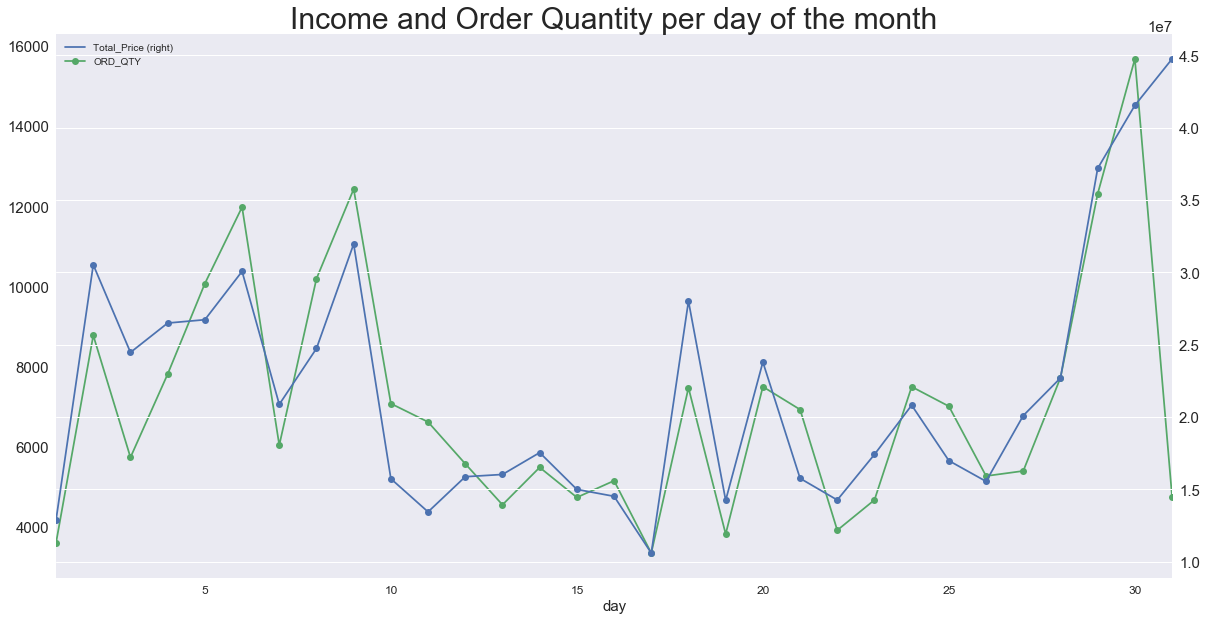
We will further narrow down this analysis to each month of the year and day of the month.

**6.5.1 Income And Order Quantity For Months Of The Year**



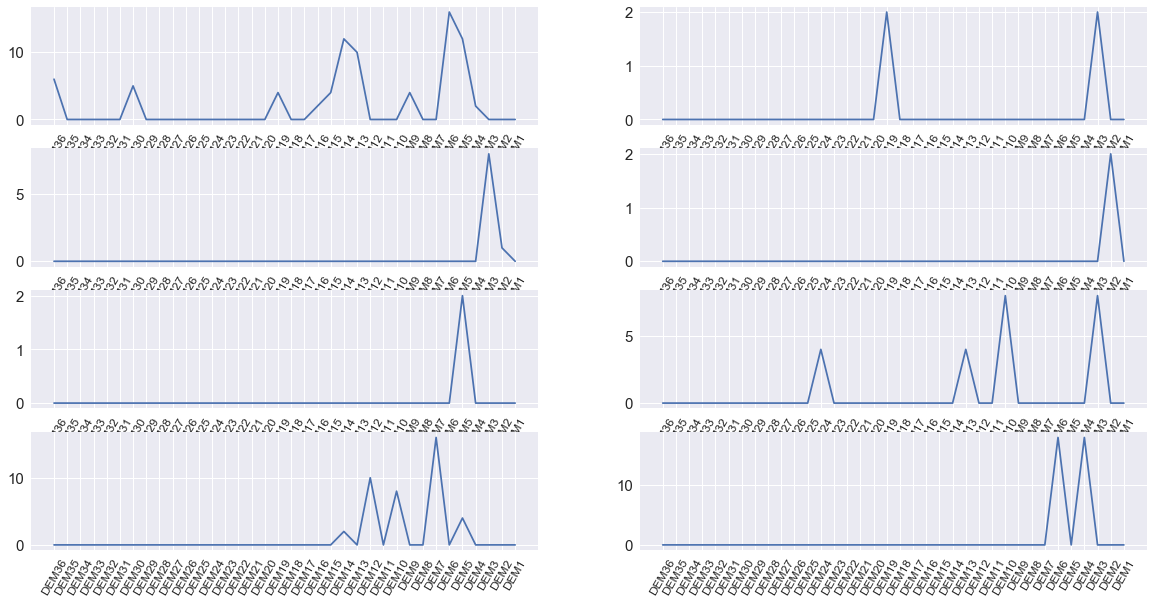
The above graphs gives the sales for each month of the year. Here we can see that, the orders is higher in the middle of the year and lower at the beginning and end.This shows the optimal time to pump up the production of parts.

**6.5.2 Income And Order Quantity For Days Of The Month**

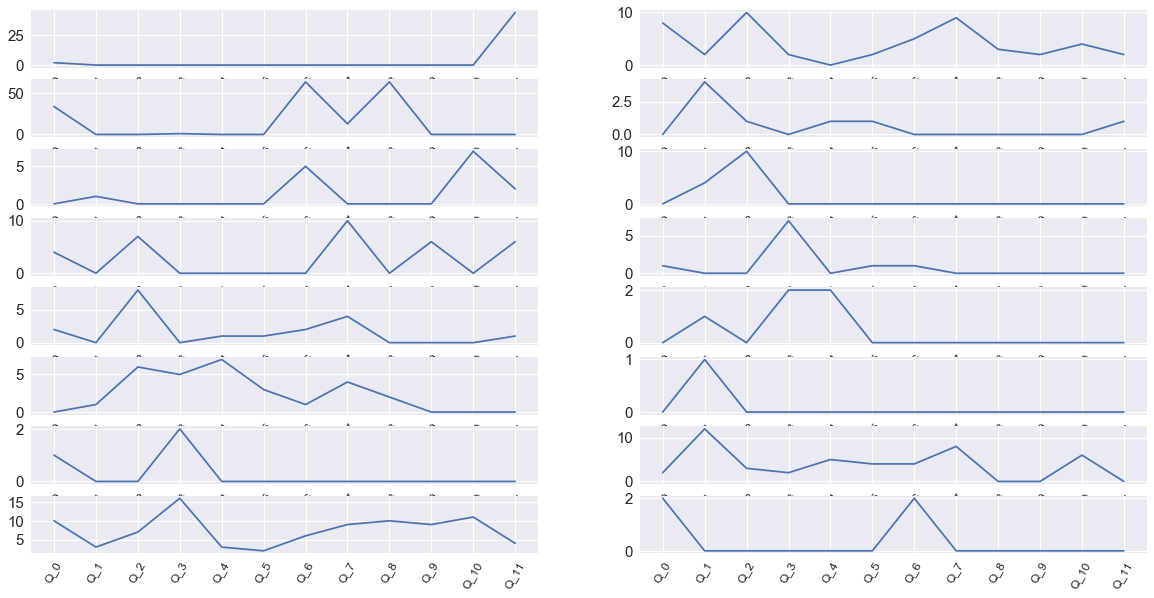


The above graph gives the sales for each day of he month. Here we can see that the demand is maximum at he end of the month.

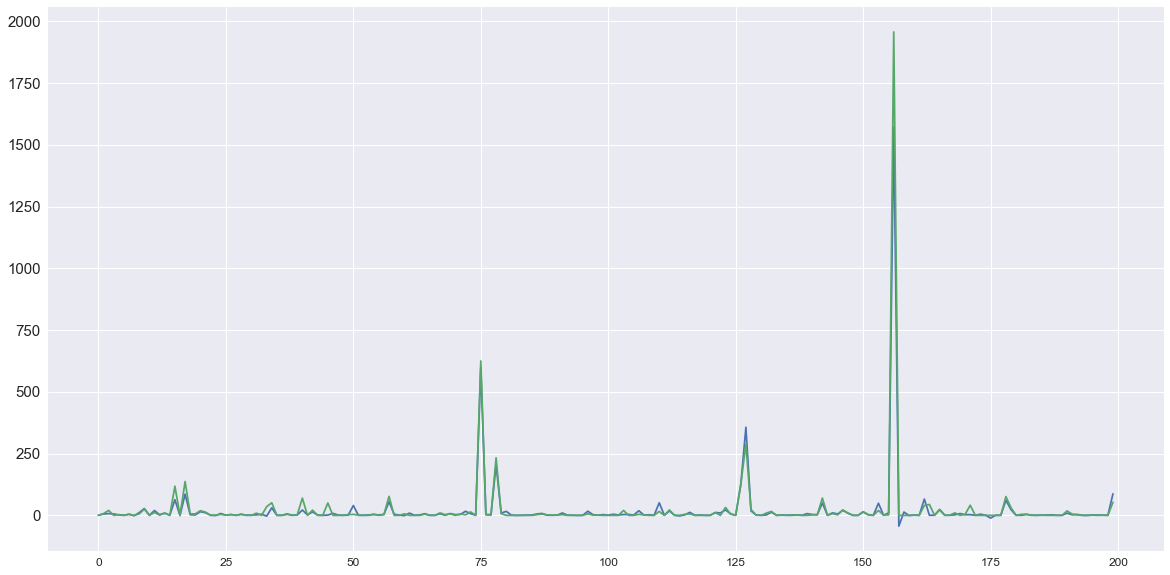
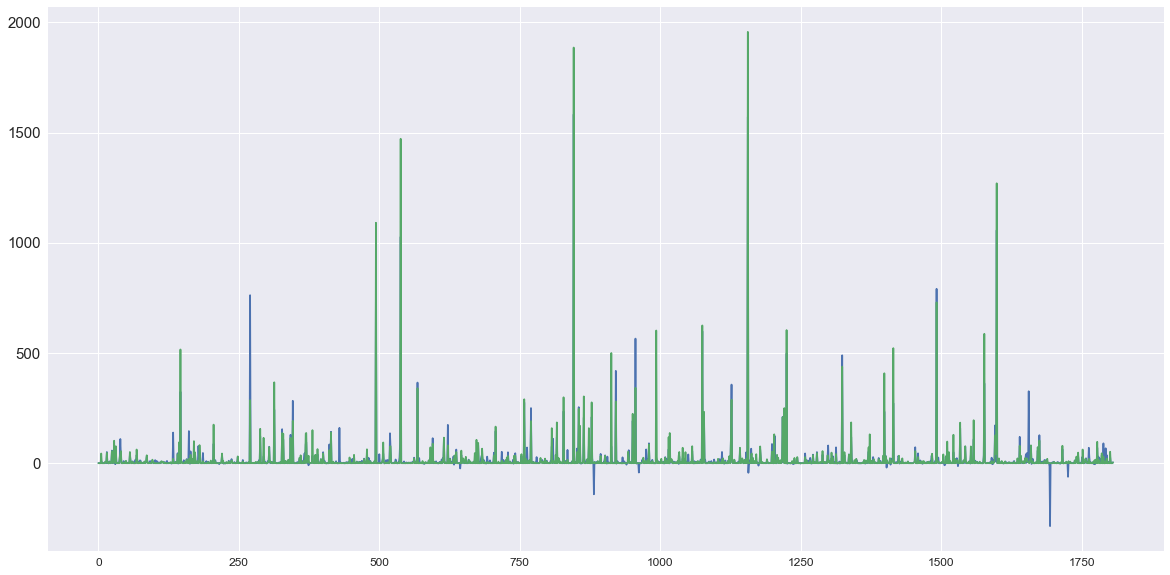
**6.6 DEMAND FORECASTING**



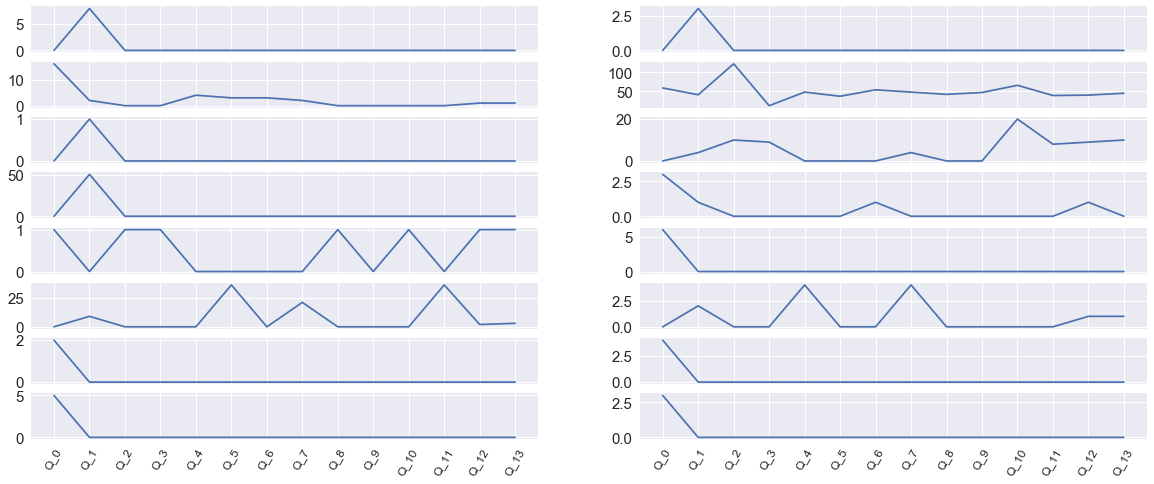
The above graph gives the demand of few materials for 36 months . Here we can see that the is no periodicity, hence we have to fiind other factors which influence these purchases or we could try to represent data in some other form.So, instead of datat per month, we divide the data as to have data per three months. This allows us to predict the demand for next three months which would be aggregated better than data per month



The above graph shows demand for interval of three months. Here we can see some sort of patterns. This could be predicted well .



The above three graphs give the comparison between predicted values and actual values.



This graph gives the demand for materials after predicting for two more intervals. We have use ElasticNet regression to predict these values.

**CONCLUSION**

Efficient material & supply chain management is crucial for the success of any small scale manufacturing & fabrication project and can be the deciding factor between a successful project and a project full of delays and claims. Better material management methods and decision models are needed to improve the electrical industry current practices, thus increasing efficiency and minimizing costs.

An effective supply management system is essential for managing efficient material management to avoid material shortages, misplacements, loss, and theft which might result in increases in crew idle times, loss of productivity and delay of activities. Small scale Electrical industry should implement an efficient material management system due to the fact that in most of the cases they are asked to squeeze their bids in order to keep the costs of project under budget. In such a case, failures to effectively manage materials could result in decreases in profit or even a loss. The primary goal is to have the material needed, in the amounts needed, with the quality required, and the time that they are needed. Most electrical companies have a material management system that serves their 205 needs, although it could be improved. Standardization of the material management system could be a step forward in improving the system and eliminating some of the bottlenecks.

The research presented in this document aimed at designing an integrated system of decision- support tools for material procurement for the small-scale industry particularly an electrical industry. An integrated approach for material procurement provides better decisions on what to order, how much to order and where to deliver. Future research will be needed to develop a more complete framework integrating other decisions needed in areas such as supplier selection and preliminary material scheduling during the prefabrication phase. A fully integrated approach will better improve communication and minimize gaps in information flow among all the parties and departments involved.

**BIBILOGRAPHY**

**BOOKS**

1. [Essentials of Supply Chain Management](http://www.amazon.com/Essentials-Supply-Chain-Management-Edition/dp/0470942185/ref=as_li_bk_tl/?tag=scm0fa-20&linkId=ef365705ab27e7eb6ddd452f7256cb1b&linkCode=ktl)**https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=ef365705ab27e7eb6ddd452f7256cb1b&_cb=1459001852153** by Michael H. Hugos
2. [Logistics and Supply Chain Management](http://www.amazon.com/Logistics-Supply-Chain-Management-5th/dp/1292083794/ref=as_li_bk_tl/?tag=scm0fa-20&linkId=f08c27d26b17dc6f1dc7da9e83e9cbca&linkCode=ktl)https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=f08c27d26b17dc6f1dc7da9e83e9cbca&_cb=1493874319315https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=operations0a-20&bm-id=default&l=ktl&linkId=8dc3c60cae331f71acee40f16ffb2a35&_cb=1493874213635https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=5c8f7703811ed0f84bf2f8bb886185be&_cb=1459001867128 by Martin Christopher
3. [Strategic Supply Chain Management: The Five Core Disciplines for Top Performance](http://www.amazon.com/Strategic-Supply-Chain-Management-Disciplines/dp/007181308X/ref=as_li_bk_tl/?tag=scm0fa-20&linkId=5fcd66c520d3e7e3c7eb3d61eb705ee2&linkCode=ktl)**https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=5fcd66c520d3e7e3c7eb3d61eb705ee2&_cb=1459002016041** by Shoshanah Cohen and Joseph Roussel
4. [Supply Chain Management: Strategy, Planning, and Operation](http://www.amazon.com/Supply-Chain-Management-Strategy-Operation/dp/0133800202/ref=as_li_bk_tl/?tag=scm0fa-20&linkId=783f86b2846f4bd4ebe40f7dc0a09564&linkCode=ktl)https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=783f86b2846f4bd4ebe40f7dc0a09564&_cb=1493874340983**https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=fa5c219b9caaa68be7b8e7c83a70d02f&_cb=1459002000529** by Sunil Chopra and Peter Meindl
5. [Designing and Managing the Supply Chain: Concepts, Strategies and Case Studies](http://www.amazon.com/Designing-Managing-Supply-Chain-Student/dp/0073341525/ref=as_li_bk_tl/?tag=scm0fa-20&linkId=f5adfcd33a56a317d212a010f76d5152&linkCode=ktl)**https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=f5adfcd33a56a317d212a010f76d5152&_cb=1459001831535** by David Simchi-Levi et al
6. [Integral Logistics Management: Operations and Supply Chain Management Within and Across Companies](http://www.amazon.com/Integral-Logistics-Management-Operations-Companies/dp/1439878234/ref=as_li_bk_tl/?tag=scm0fa-20&linkId=aa3100d6ea9752a3b80efe86b1acbae5&linkCode=ktl)https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=aa3100d6ea9752a3b80efe86b1acbae5&_cb=1460252485490https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=e1957b9d3b09cdb9dacf46c86fe4f077&_cb=1460252436757https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=20de63e647f54ec2eb66ac272e7e4b0e&_cb=1460252315484 by Paul Schönsleben
7. [Logistics Management and Strategy: Competing through the Supply Chain](http://www.amazon.com/Logistics-Management-Strategy-5th-Competing/dp/1292004150/ref=as_li_bk_tl/?tag=scm0fa-20&linkId=06a5951f7a860ef63a0f48e37398f1bc&linkCode=ktl)https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=06a5951f7a860ef63a0f48e37398f1bc&_cb=1459001897694 by Alan Harrison and Remko Van Hoek
8. [Manufacturing Planning and Control for Supply Chain Management](http://www.amazon.com/Manufacturing-Planning-Control-Supply-Management/dp/0071750312/ref=as_li_bk_tl/?tag=scm0fa-20&linkId=d829a6d406b7574207e2377a3a956a57&linkCode=ktl)https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=d829a6d406b7574207e2377a3a956a57&_cb=1459001919825 by F. Robert Jacobs et al
9. [Supply Chain Logistics Management](http://www.amazon.com/Supply-Logistics-Management-Donald-Bowersox/dp/0078024056/ref=as_li_bk_tl/?tag=scm0fa-20&linkId=72268ecff7792ea3a885927a4f9e2577&linkCode=ktl)https://ir-na.amazon-adsystem.com/e/ir?source=bk&t=scm0fa-20&bm-id=default&l=ktl&linkId=72268ecff7792ea3a885927a4f9e2577&_cb=1459001982744 by Donald Bowersox et al

[10] The Handbook of Logistics and Distribution Management: Understanding the Supply Chain by Alan Rushton et a

**LINKS**

1. <https://www.analyticsvidhya.com/blog/2016/01/complete-tutorial-learn-data-science-python-scratch-2/>
2. <https://www.lynda.com/Numpy-tutorials/Introduction-Data-Analysis-Python/419162-2.html>
3. <http://bigdata-madesimple.com/step-by-step-approach-to-perform-data-analysis-using-python/>
4. <https://www.coursera.org/learn/python-data-analysis>
5. <https://pandas.pydata.org/pandas-docs/stable/10min.html>
6. <https://www.dataquest.io/blog/pandas-python-tutorial/>
7. <http://jupyter.readthedocs.io/en/latest/install.html>
8. <https://pythonprogramming.net/python-pandas-data-analysis/>

LIST OF FIGURES

Sl. No. Title of the figure Page No.

Fig 4.1 Architectural Diagram 19

Fig 4.2 Data Flow Diagram 20

Fig 4.3 Use Case Diagram 21

Fig 4.4 Project Work Flow Diagram 22