OTA PROJECT REPORT

A2 G11

**Topic**

Inventory Management: Compare inventory stocking and reordering policies with the EOQ (Economic Order Quantity) model

**Problem Definition**

1. **Inventory Management**

A warehouse sells a product A. It has a contract to supply 60 units at the end of the 1st, 2nd and 3rd month. The cost of producing x units of A is x2 in the 1st month, 2x2 in the 2nd month and 3x2 in the 3rd month. The warehouse can produce more units of A in any month and carry them to subsequent month. The carrying cost of Rs. 25 per unit is charged for carrying units of A from one month to the next month. Determine the number of units of A that should be produced in each month to minimize the total cost.

2. **EOQ**

A warehouse sells 3 products with different demand for each product. Each Product has a different holding cost, different ordering price and different price for each unit. What should be the ordering policy for each of the products for the warehouse for minimum total cost.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Product | Demand (per month) | Ordering Cost | Price (per unit) | Holding cost |
| A | 200 | 25 | 50 | 25 |
| B | 350 | 21 | 45 | 19 |
| C | 400 | 25 | 68 | 36 |

**Formulation**

**First problem** is a Non-linear Programming Problem.

Objective Function: Z = x12 + 25\*(x1-60) + 2x22 + 25\*(x1+x2-120) + 3x32

Sub. To the constraints:

x1 >= 60

x1 + x2 >= 120

x1+ x2+ x3 = 180

**Second Problem** is Economic Order Quantity Calculation

Total Cost= d\*p+h\*q/2+d\*o/q

Where,

d- demand

p- price per unit

q- quantity in each order

h- holding cost

o- price per order

**Solution**

**For Problem 1:**

Z= x12 +2x22 + 3x32+50x1+ 25x2-4500

Sub to constraints:

x1>=60 🡪 -x1<=-60

x1+x2>=120 🡪 -x1-x2<=-120

x1+x2+x3=180

Solving using Kuhn Tucker Method:

λ1, λ2, λ3 are KKT Multipliers

λ1, λ2, λ3 >=0

(2x1+50,4x2+25,6x3) +λ1(-1,0,0) +λ2(-1,-1,0)+λ3(-1,-1,-1)

= (0,0,0)

🡪(2x1+50-λ1-λ2-λ3,4x2+25-λ2,6x3-λ3)= (0 0 0)

🡪

2x1+50-λ1-λ2-λ3=0

4x2+25-λ2-λ3=0

6x3-λ3=0

G1(x1,x2,x3)= -x1+60

G2(x1,x2,x3)= -x1-x2+120

G3(x1,x2,x3)= -x1-x2-x3+180

Now, According to Kuhn Tucker Method

λ1G1=0

λ2G2=0

λ3G3=0

As x1+x2+x3=180

We know that G3=0 so there’s no need to check for λ3.

So :

**Case 1**:

x1+x2=120, x1=60

**Case 2** :

λ1=0 ,x1+x2=120

**Case 3**:

λ2=0 ,x1=60

**Case 4**:

λ1=0 , λ2=0

Checking for case 1:

x1=60

x1+x2=120

Therefore, x1=60, x2=60

And from G3

x1+x2+x3=180

Therefore,

x1=x2=x3=60

Now,

6x3=λ3

λ3=360

4x2+25-λ2-λ3=0

265-λ3=λ2

265-360=λ2

λ2= -95

Now, as we know λ2 can’t be negative , so this case **fails**.

Checking for case 2:

λ1=0

x1+x2=120

Now,

2x1-λ1-λ2-λ3=0

2x1-λ2-λ3=0

x1= (λ2+λ3)/2

4x2+25-λ2-λ3=0

x2= (λ2+λ3-25)/4

Now,

(λ2+λ3)/2 + (λ2+λ3-25)/4 = 120

3(λ2+S3)=505

λ2+ λ3 = 505/3

x1+x2=120

x1+x2+x3=180

x3=60

λ3=6x3=360

λ2=505/3-360= -191.66

But λ2 should be positive

Hence case **fails.**

Checking for case 3:

λ2=0

x1=60

x1+x2+x3=180

x2+x3=120

Now,

4x2+25-λ2-λ3=0

x2=(λ3-25)/4

x3= (λ3)/6

(λ3-25)/4 + λ3/6 = 120

λ3=303

2x1+50-λ1-λ2-λ3=0

120+50-S1-0-303

λ1=-133

But λ1 should be positive

Hence case **fails**

Checking for case 4:

λ1=0

λ2=0

2x1+50-λ1-λ2-λ3=0

2x1+50=λ3

x1= (λ3-50)/2

4x2+25-λ2-λ3=0

4x2+25=λ3

x2=(λ3-25)/4

6x3-λ3=0

6x3=λ3

x3=λ3/6

Now,

x1+x2+x3=180

Therefore,

(λ3-50)/2+(λ3-25)/4+λ3/6=180

λ3= 2535/11

**Answer**

**x1=1985/22 (90.22)**

**x2=565/11 (51.36)**

**x3=845/22 (38.40)**

**Z(optimized) = 19140**

**For Problem 2:**

cost = d\*p+h\*q/2+d\*o/q;

on differentiating cost function w.r.t. q, we get,

dc/dq= h/2-d\*o/q2

On calculating minimum value for cost

dc/dq=0

Therefore,

h/2-d\*o/q2=0

q=sqrt(2\*d\*o/h)

**EOQ= sqrt(2\*d\*o/h)**

**1. For Product A:**

**EOQ= sqrt(2\*200\*25/25)= 20**

**Integer(20) = 20**

**Total Cost= 200\*50+25\*20/2+200\*25/20= 5750**

**2. For Product B:**

**EOQ= sqrt(2\*350\*21/19)= 27.81**

**Integer(27.81) = 27**

**Total Cost= 350\*45+19\*27/2+350\*21/27= 8203.79**

**3. For Product C:**

**EOQ= sqrt(2\*400\*25/36)= 23.57**

**Integer(23.57) = 23**

**Total Cost= 400\*68+19\*23/2+400\*25/23= 11408.08**

**MATLAB CODE**

**Problem 1:**

clc;

syms d p q h o

%our objective function

objective = @(x) x(1)^2+2\*x(2)^2+3\*x(3)^2+50\*x(1)+25\*x(2)-4500;

% initial guess

x0 = [60,60,60];

%linear inequality constraints

%{

x1>=60 ==> -x1<=-60

x1+x2>=120 ==> -x1-x2<=-120

%}

A=[-1,0,0;-1,-1,0];

B=[-60;-120];

%linear equality constraints

% x1+x2+x3=180

Aeq=[1,1,1];

Beq=180;

% optimize with fmincon

%[X] = fmincon(FUN,X0,A,B,Aeq,Beq)

x = fmincon(objective,x0,A,B,Aeq,Beq);

disp(x);

%converting solutions into integers

x(1)=ceil(x(1));

x(2)=floor(x(2));

x(3)=floor(x(3));

% show final objective

disp(['Final Objective: ' num2str(objective(x))])

fprintf(1, '\n');

% print solution

disp('Solution')

disp(['x1 = ' num2str(x(1))])

disp(['x2 = ' num2str(x(2))])

disp(['x3 = ' num2str(x(3))])

**Problem 2:**

clc;

%{

d- demand

p- price per unit

q- quantity in each order

h- holding cost

o- price per order

%}

syms d p q h o

%our cost function

cost = d\*p+h\*q/2+d\*o/q;

%differentiation of cost

diffcost = diff(cost,q);

%calculating EOQ function

q = solve(diffcost==0,q);

q = matlabFunction(q(1,1));

cost = matlabFunction(cost);

%Given problems

%{

1- demand

2- price per unit

3- holding cost

4- price per order

%}

x1 = [200,50,25,25];

x2 = [350,45,19,21];

x3 = [400,68,36,25];

%calculating EOQ

q1 = q(x1(1),x1(3),x1(4));

q2 = q(x2(1),x2(3),x2(4));

q3 = q(x3(1),x3(3),x3(4));

%converting solutions into integers

q1 = floor(q1);

q2 = floor(q2);

q3 = floor(q3);

%calculating final cost

c1 = cost(x1(1),x1(2),x1(3),x1(4),q1);

c2 = cost(x2(1),x2(2),x2(3),x2(4),q2);

c3 = cost(x3(1),x3(2),x3(3),x3(4),q3);

disp('Solution')

fprintf(1, '\n');

%For Product A

disp(['Total Cost for Product A: ' num2str(c1)])

disp(['EOQ for Product A = ' num2str(q1)])

fprintf(1, '\n');

%For Product B

disp(['Total Cost for Product B: ' num2str(c2)])

disp(['EOQ for Product B = ' num2str(q2)])

fprintf(1, '\n');

%For Product C

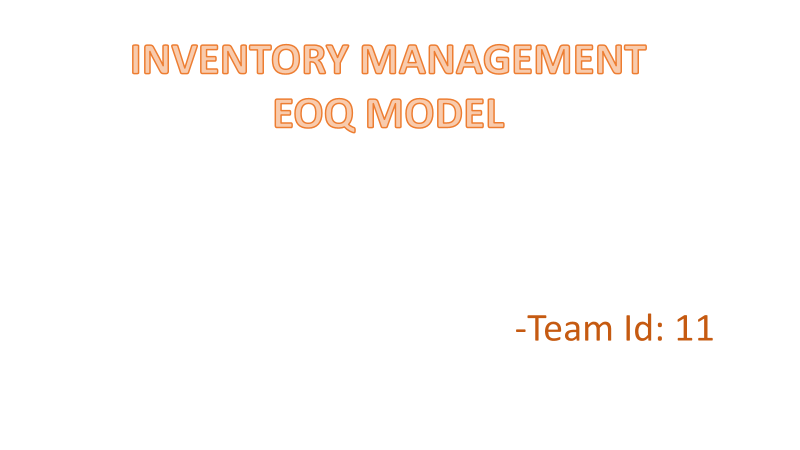
disp(['Total Cost for Product C: ' num2str(c3)])

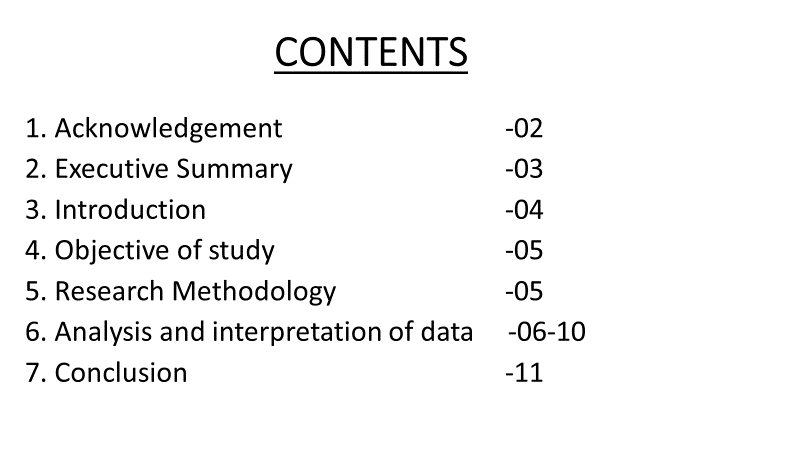
disp(['EOQ for Product C = ' num2str(q3)])

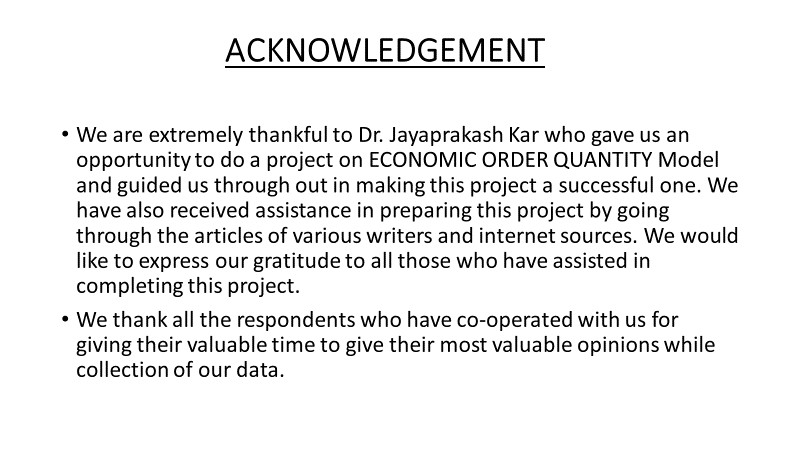
**Name of the Method**

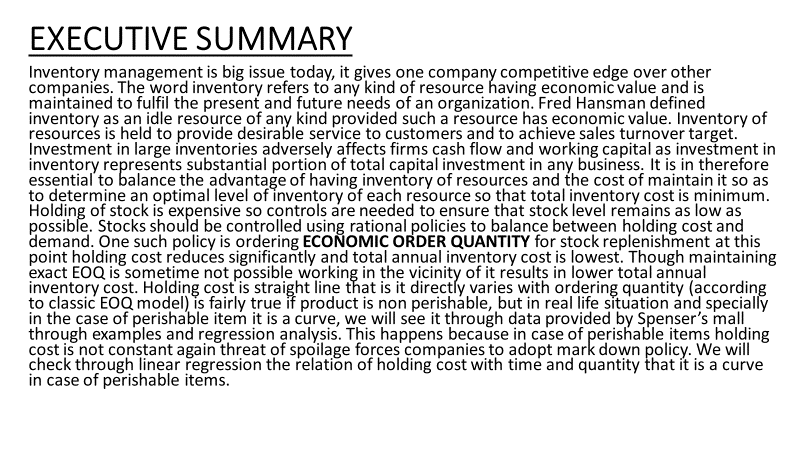
The method used in the formulation of the problems is Kuhn Tucker Method.

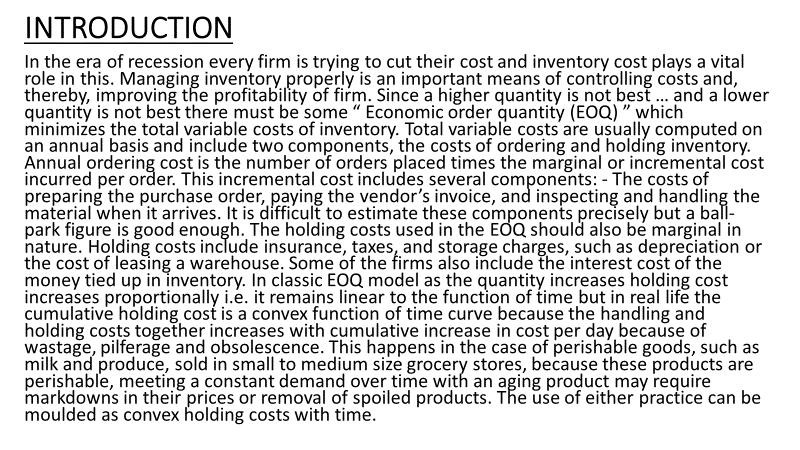
**Detailed Study of EOQ**

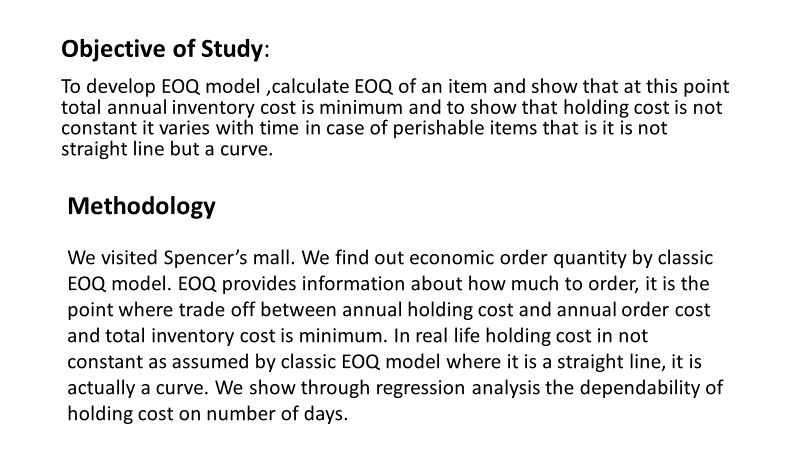


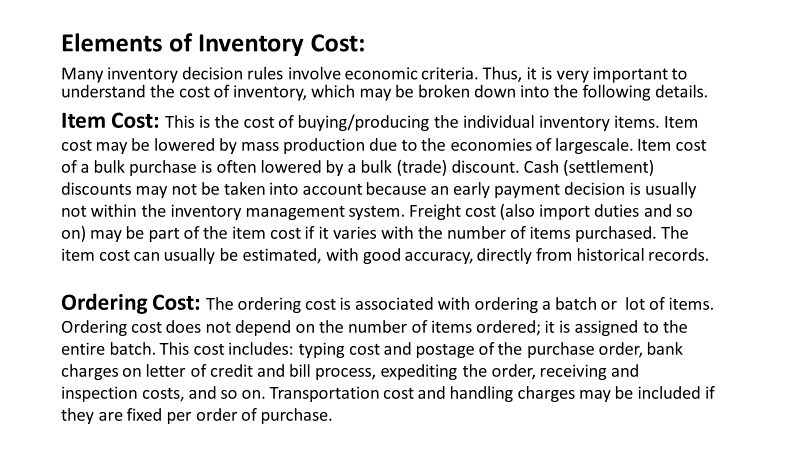


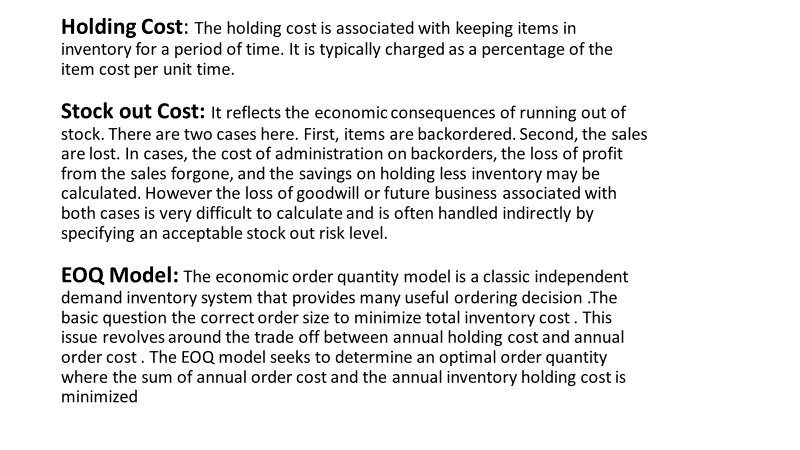


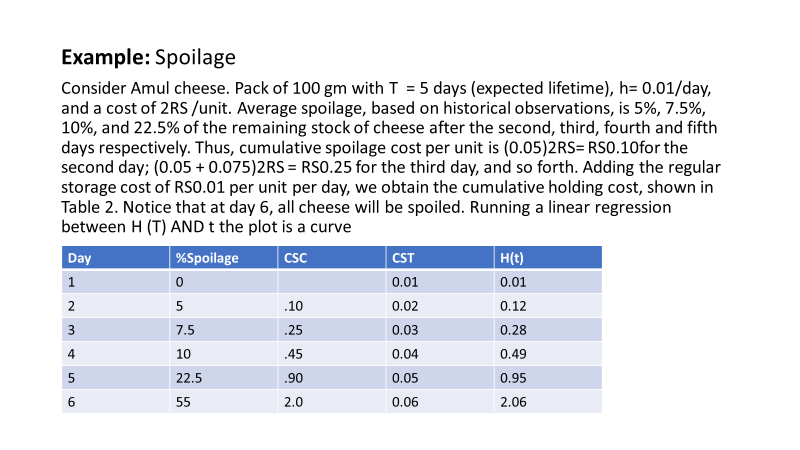


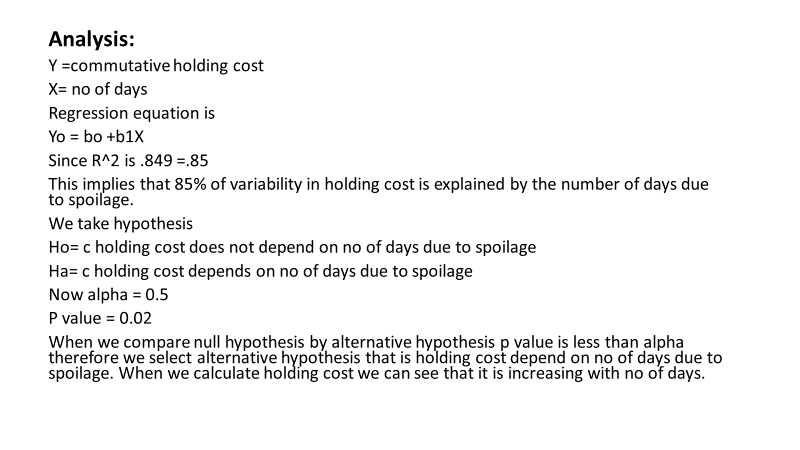


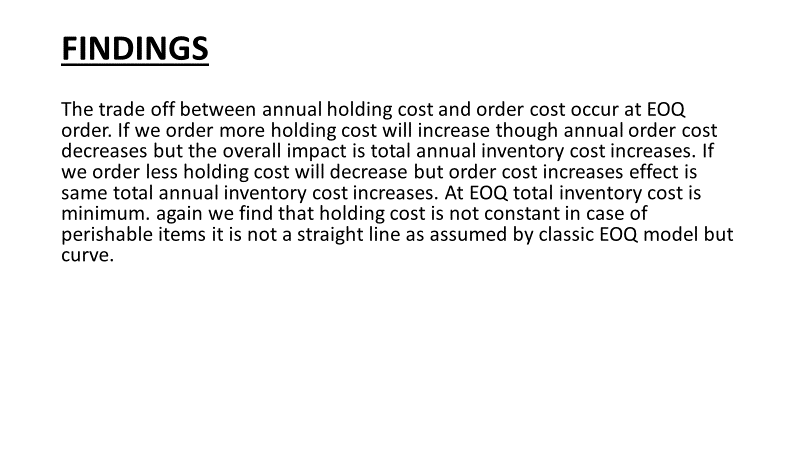


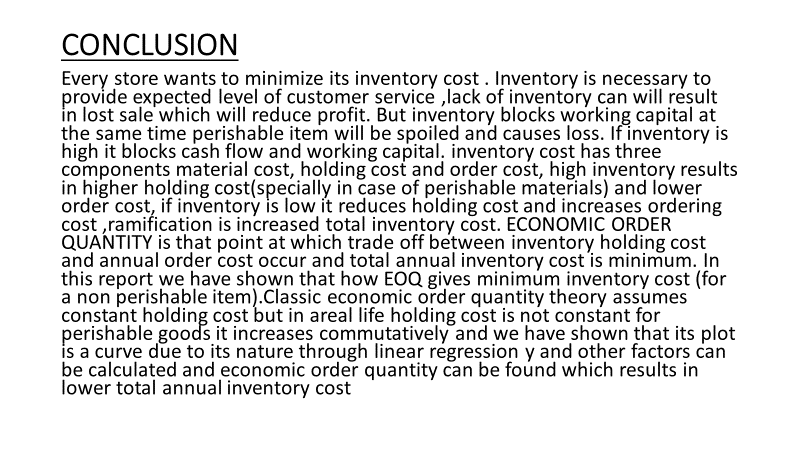












**Work Distribution:**

1.Problem Formulation:

Preksha Pandey (16ucs139)

Udish Nagpal (16ucs202)

Ujjawal Agarwal (16ucs203)

2.Design and Framework of solution:

Preksha Pandey (16ucs139)

Udish Nagpal (16ucs202)

Ujjawal Agarwal (16ucs203)

3.Implementation in MATLAB:

Preksha Pandey (16ucs139)

Udish Nagpal (16ucs202)

Ujjawal Agarwal (16ucs203)

4.Report Design:

Neelansh Sethi (16ucs116)

Suyash Agarwal (16ucs192)

Tirth M. Shah (16ucs199)

5.Case Study and Presentation:

Preksha Pandey (16ucs139)

Udish Nagpal (16ucs202)

Ujjawal Agarwal (16ucs203)