Airline Seat Allocation and Overbooking
ISEN 609 Team Project

Jung-Ho Park, EunSuk Ko, Yeong-In Kim
12/6/2011
1. Project Statement

1.1 Goal

The goal of this project is to find an optimal seats allocation which maximizes profits for Time-Flies Airline. The three questions are given as followings.

1. Assuming that the price is fixed, how to find an optimal seats allocation when overbooking is not allowed?

2. In the same circumstance, how to find an optimal seats allocation when overbooking is allowed?

3. What if the price on the flight depends on competition?

1.2 Given Information

1. The flight is single-leg and there are two fare classes for the flight.

2. Tickets for discount seats are inexpensive than tickets for regular seats.

3. Regular tickets are refundable whereas discounted tickets are not.

4. Discount seats purchased by leisure travelers are often sold in advance than regular seats mostly purchased by business travelers.

5. The sales data for previous 13 weeks below can be used as demand data for those two tickets, and it is given as follows:

<table>
<thead>
<tr>
<th>Week</th>
<th>Seats Sold(R)</th>
<th>No Show(R)</th>
<th>Seats Sold(D)</th>
<th>No show(D)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>120</td>
<td>9</td>
<td>43</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>161</td>
<td>3</td>
<td>46</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>98</td>
<td>8</td>
<td>64</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>74</td>
<td>1</td>
<td>94</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>112</td>
<td>6</td>
<td>89</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>67</td>
<td>2</td>
<td>91</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>23</td>
<td>0</td>
<td>24</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>99</td>
<td>9</td>
<td>69</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>175</td>
<td>3</td>
<td>97</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>12</td>
<td>1</td>
<td>32</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>111</td>
<td>8</td>
<td>42</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>148</td>
<td>8</td>
<td>51</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>82</td>
<td>2</td>
<td>76</td>
<td>3</td>
</tr>
<tr>
<td>Total</td>
<td>1,282</td>
<td>60</td>
<td>818</td>
<td>16</td>
</tr>
</tbody>
</table>

[Table 1] Sales data on Sunday flights for the last 13 weeks
1.3 The key approach for this problem

1. At first, we define several assumptions for each question. And then, parameters and random variables are introduced according to them.

2. For the first two questions, we apply ‘Newsboy’s problem’ to set up mathematical models where many expected values are calculated by conditioning.

3. The optimal solutions are numerically calculated by R program.

4. As an extension to our model, we introduce simple dynamic pricing model. The basic concept and approach are derived from relative research papers.
2. Static Pricing Model without Overbooking

2.1 Assumption

1. The seats should be fully allocated.
2. There is no cancellation ahead of boarding.
3. Each class has independent demand.
4. Each passenger on the same class has the same probability of buying a ticket.
5. Demand for each class is according to random distribution.
6. The ticket fares of all classes and are fixed.
7. Each passenger in a same class has a same probability of no-show.
8. An event of refund occurs only when a passenger on regular class doesn’t show-up (no-show).

2.2 Notation

2.2.1 Indices

\[ i = \{r, d\}: r \text{ is a indice for regular class and } d \text{ is a set for discounted class} \]

2.2.2 Parameters

\[ f_i: \text{ticket fare for a seat on class } i \]
\[ r_i: \text{refund rate for a no-\text{\text{--} show passenger on class } i} \]
\[ c_r: \text{fixed cost for a seat} \]
\[ M: \text{the maximum number of seats} \]
\[ A_i: \text{the number of allocated seats for class } i \]
\[ p_{N_i}: \text{probability that a passenger of class } i \text{ does not show-up} \]

2.2.3 Random Variables

\[ X_i: \text{demand of tickets on class } i \]
\[ S_i: \text{the number of tickets sold on class } i \]
\[ N_i: \text{the number of no-\text{\text{--} show passengers on class } i} \]
2.3 Mathematical Model

2.3.1 Evaluation of parameters and random variables

1. Ticket fare

We define ticket fares for each class as follows

\[ 0 < f_d < f_r \]
\[ f_r = 5,000 \]
\[ f_d = 3,000 \]

2. Refund rate for no-show passengers

Although a passenger does not show up for boarding, we guarantee that a regular passenger can get a refund a portion of their money by the assumption. The refund rate should be less than 100% of original fare rate.

\[ r_r = 0.8, \quad r_d = 0 \]

3. Fixed cost for a seat

We assume that all fixed costs are included in the deterministic parameter \( c_f \) to simplify the model.

\[ c_f = 1,000 \]

4. Maximum number of seats

Since we do not consider a problem with overbooking at this point, maximum number of tickets should be a fixed constant which is 164.

\[ M = 164 \]

5. The number of allocated seats

The sum of regular and discounted seats is equal to the maximum number of seats.

\[ A_r + A_d = M = 164 \]
6 Probability of no-show

As we assume that each passenger on the same class has the same probability of no-show, the probability of no-show on a certain class can be calculated as follows:

\[ p_{N_i} = \frac{\text{total number of no-show}}{\text{total number of passengers}} \]

\[ p_{N_r} = \frac{16}{818} \approx 0.0196 = 1.96\%, \quad p_{N_r} = \frac{60}{1282} \approx 0.0468 = 4.68\% \]

Note that these probabilities are independent.

7. Demand of tickets

Since we assume that each passenger on the same class has the same probability of buying a ticket, we can define that demand for a class is binomially distributed. However, as the number of sample is very large, demand is approximately normally distributed.

\[ X_i \sim \text{Normal}(E[X_i], \text{Var}[X_i]) \]

The mean and the variance for the distribution can be calculated by the given past sales data.

\[ E[X_r] = \frac{1282 \text{Total tickets sold}(r)}{13 \text{weeks}} = 98.6154 \]

\[ E[S_d] = \frac{818 \text{Total tickets sold}(d)}{13 \text{weeks}} = 62.9231 \]

\[ \text{Var}[X_r] = \frac{1}{n-1} \sum_{i=1}^{13} (X_r - \bar{X}_r)^2 = 628.2436 \]

\[ \text{Var}[X_d] = \frac{1}{n-1} \sum_{i=1}^{13} (X_d - \bar{X}_d)^2 = 2338.0900 \]

Therefore demand for each class is defined as follows:

\[ X_r \sim \text{Normal}(98.6154, 628.2436) \]

\[ X_r \sim \text{Normal}(62.9231, 2338.0900) \]
8. The number of tickets sold

If the number of allocated seats on a class is defined as \( A_i \), then we can calculate the expected number of tickets sold on a class by conditioning.

\[
E[S_i | X_i = x] = \begin{cases} 
  A_i & \text{if } A_i < x \\
  x & \text{if } A_i \geq x 
\end{cases}
\]

9. The number of no-show

As we assume that each passenger in a same class has a same probability of no-show, we can define that the number of no-show for a class given the number of tickets sold is binomially distributed.

\[ N_i | S_i \sim \text{Binomial}(S_i, p_{N_i}) \]

Then the expected number of no-show can be calculated as follows:

\[
E[N_i] = E[N_i | S_i] = E[p_{N_i} | S_i] = p_{N_i} \cdot E[S_i]
\]

2.3.2 Formulation and Result

If the number of allocation for regular seat \( A_r \) is fixed, we can calculate the expected total profit for the flight in this way:

\[
E[\text{profit}] = \text{sum of expected revenue from each class} - \text{expected refund amount} - \text{expected cost}
\]

\[
= f_r \cdot E[S_r] + f_d \cdot E[S_d] - r_n \cdot f_r \cdot E[N_r] - c_f \cdot M
\]

\[
= f_r \cdot E[E[S_r | X_r]] + f_d \cdot E[E[S_d | X_d]] - r_n \cdot f_r \cdot p_{N_r} \cdot E[S_r] - c_f \cdot M
\]

where

\[
M = A_r + A_d = 164
\]

\[
E[S_r | X_r = x] = \begin{cases} 
  A_r & \text{if } A_r < x \\
  x & \text{if } A_r \geq x 
\end{cases}
\]

\[
E[S_d | X_d = x] = \begin{cases} 
  (M - A_r) & \text{if } M - A_r < x \\
  x & \text{if } M - A_r \geq x 
\end{cases}
\]
Therefore, the total expected profit can be written as a following equation:

\[
E[profit] = f_r \cdot \left[ \int_0^{A_r} x \cdot f_{X_r}(x) \, dx + \int_{A_r}^{\infty} A_r \cdot f_{X_r}(x) \, dx \right] \\
+ f_d \cdot \left[ \int_0^{164-A_r} x \cdot f_{X_d}(x) \, dx + \int_{164-A_r}^{\infty} (164 - A_r) \cdot f_{X_r}(x) \, dx \right] \\
- r \cdot f_r \cdot p_{N_r} \cdot \left[ \int_0^{A_r} x \cdot f_{X_r}(x) \, dx + \int_{A_r}^{\infty} A_r \cdot f_{X_r}(x) \, dx \right] - c_f \cdot 164
\]

In this model, our goal is to find the optimal value of \( A_r \) which maximizes the expected profit. To find the optimal value of \( A_r \) and the maximized profit, we use numerical method to calculate the model with R program.

As a result, the profit can be maximized as $386,115.20 when the number of allocated seats for regular class \( A_r \) is 111, and \( A_r' \) is 164-111=53.

![Plot of Expected Profit](Figure 1) Graphical output of the model
Figure 2] R script for calculating the model —part I

```
> # evaluation of parameters
> Exp_D=62.923076923076920
> Var_D=6.2824358974359006+02
> Exp_R=2.338089743897444+03
> pr_NoR=0.0468
> fare_R=3000
> refund_R=0.8
> cost_F=1000
>
> # initializing the value of Expected Profit
> Exp_Profit=0
>
> # define functions of x for calculating integrations
> Xpdf_R <- function(x) {x*dnorm(x,Exp_R,sqrt(Var_R))}
> Xpdf_R2 <- function(x) {dnorm(x,Exp_R,sqrt(Var_R))}
> Xpdf_D <- function(x) {x*dnorm(x,Exp_D,sqrt(Var_D))}
> Xpdf_D2 <- function(x) {dnorm(x,Exp_D,sqrt(Var_D))}
```

Figure 3] R script for calculating the model —part II

```
> # calculating the Expected Profit as increasing the value of a
> for (a in 1:164){
+    + # calculating each integrations
+    int1=integrate(Xpdf_R, lower=-Inf, upper=a)
+    int2=integrate(Xpdf_R2, lower=a, upper=Inf)
+    int3=integrate(Xpdf_D, lower=-Inf, upper=164-a)
+    int4=integrate(Xpdf_D2, lower=164-a, upper=Inf)
+    int5=integrate(Xpdf_R, lower=-Inf, upper=a)
+    int6=integrate(Xpdf_R2, lower=a, upper=Inf)
+    + # converting values of integrations to numerical values
+    int1=as.numeric(int1[1])
+    int2=as.numeric(int2[1])
+    int3=as.numeric(int3[1])
+    int4=as.numeric(int4[1])
+    int5=as.numeric(int5[1])
+    int6=as.numeric(int6[1])
+
+    + # calculation of the Expected Profit
+    Exp_Profit[a]=fare_R*(int1+a*int2)+fare_D*(int3+(164-a)*int4)
+    -refund_R*fare_R*
+    (pr_NoR*int5+pr_NoR*a*int6)-cost_F*164
+    }
```
Figure 4] R script for calculating the model – part III

```r
> #ploting the Expected Profit v.s. the allocation
> plot(Exp_Profit, ylab="Expected profit ($)", xlab="The number of seats allocation for $r$
> 
> > max(Exp_Profit)
> [1] 356115.2
```
3. Static Pricing Model with Overbooking

3.1 Assumption

The only change is the assumption about the refund policy (marked by *)

1. The seats should be fully allocated.
2. There is no cancellation ahead of boarding.
3. Each class has independent demand.
4. Each passenger on the same class has the same probability of buying a ticket.
5. Demand for each class is according to random distribution.
6. The ticket fares of all classes and are fixed.
7. Each passenger in a same class has a same probability of no-show.

*8. Only the regular tickets are refunded for no-show.

*9. Overbooked passengers who are not allowed to board will get refund.

*10. The waiting list is needed only for the discounted seats.

Notice that majority of discounted fares are purchased in advance to regular fares, which means that the number of selling tickets for discounted seats are determined earlier. Hence, overbooking can be considered and waiting list may exist on discounted fares. On the other hand, since majority of demand for regular tickets occurs close to when the flight takes off, we do not consider the waiting list in selling regular seats.

![Figure 5] Occurrence of demand for each fare class

Therefore, if there are exceeded total of requests for a flight, then some passengers on waiting lists (discounted seats) are rejected, but regular seats are guaranteed.
3.2 Notation

3.2.1 Sets

\[ i = \{ r, d \} \text{: } r \text{ is a set for regular class and } d \text{ is a set for discounted class} \]

3.2.2 Parameters

The new parameters we add are marked by *.

\[ f_i \text{: ticket fare for a seat of class } i \]
\[ r_i \text{: refund rate for a no – show passenger of class } i \]
\[ c_f \text{: fixed cost for a seat} \]
\[ M \text{: the maximum number of seats} \]
\[ A_i \text{: the number of allocated seats for class } i \]
\[ p_{N_i} \text{: probability that a passenger of class } i \text{ does not show – up} \]
\[ *S_{i_{\text{max}}} \text{: the maximum number of selling tickets on class } i \]
\[ *N_{i_{\text{max}}} \text{: the maximum number of no – show passengers of } i \text{ class} \]
\[ *o_i \text{: penalty rate for an overbooked passenger who is not allowed to board} \]
\[ \text{the airplane on class } i \]

3.2.3 Random Variables

The new variable we add is marked by *.

\[ S_i \text{: the number of tickets sold on class } i \]
\[ X_i \text{: demand of tickets on class } i \]
\[ N_i \text{: the number of no – show passengers on class } i \]
\[ *R \text{: the number of rejected passengers on the waiting list} \]
3.3 Mathematical Model

3.3.1 Evaluation of parameters and random variables

Since many of parameters and variables are the same with those in the previous model, we discuss only the newly introduced parameters and variables in this part.

1. The maximum number of selling tickets

Unlike the previous model, the overbooking policy is allowed in this model, and the maximum number of tickets sold may exceed the number of allocated seats for a class.

\[ S_{r_{\text{max}}} + S_{d_{\text{max}}} \geq M = 164 \]

2. The maximum number of no-show passengers of a class

As we discussed previously, only the discounted class has the waiting list. And therefore, requests on the waiting list would be rejected if the total number of selling tickets exceeds the maximum capacity of the flight which is 164.

Then, we are interested in how to define the number of waiting passengers on discounted fare reasonably. To define this, we use the expected maximum number of no-shows on each fare class with a certain probability level. By using the historical data, we find the following probability:

\[ \Pr[N_t \leq N_{t_{\text{max}}}] \geq 0.95 \]

And then, the maximum number of selling tickets for each class can be defined as follows:

\[ S_{d_{\text{max}}} = A_d + N_{d_{\text{max}}} \]

\[ S_{r_{\text{max}}} = A_r + N_{r_{\text{max}}} = 164 - (A_d - N_{r_{\text{max}}}) \]

where \( N_{t_{\text{max}}} \) is the 95% quantile value of Binomial\( (A_t, p_{N_t}) \)

This means that if the number of requests for seats exceeds the number of allocated seats on a certain class, the company will receive the additional requests up to 95% of maximum number of no-show.
3. The penalty rate for an overbooked passenger who is not allowed to board the airplane on a class

In addition to the original price for the ticket, we set additional 50% of the price for the compensation.

\[ o_t > 1.5 \]

4. The number of rejected passengers on the waiting list.

If there are exceeded total of requests for either discounted seats or regular seats, then some passengers on waiting lists are rejected. We define the number of rejected passengers as follows:

\[ R = \begin{cases} S_d - N_d + S_r - N_r - M & \text{if } R > 0 \\ 0 & \text{if } R < 0 \end{cases} \]

And the expected number of rejected passengers is defined as follows.

\[ E[R] = \begin{cases} E[S_d] - E[N_d] + E[S_r] - E[N_r] - M & \text{if } R > 0 \\ 0 & \text{if } R < 0 \end{cases} \]

3.3.2 Formulation and Result

In formulating the pricing model for this problem, we add the penalty for passengers who are not allowed to board the plane due to overbooking. So, the pricing model for this problem is defined as follows:

\[
E[\text{profit}] = \text{Sum of expected revenue from each class} - \text{expected refund amount} - \text{expected cost} - \text{expected penalty for overbooking} \\
= f_r \cdot E[S_r] + f_d \cdot E[S_d] - r_n \cdot f_r \cdot E[N_r] - c_f \cdot M - o_d \cdot f_d \cdot E[R] \\
= f_r \cdot E[E[S_r|X_r]] + f_d \cdot E[E[S_d|X_d]] - r_n \cdot f_r \cdot p_{N_r} \cdot E[S_r] - c_f \cdot M \\
- o_d \cdot f_d \cdot \max\{0, E[S_d] - E[N_d] + E[S_r] - E[N_r] - M\}
\]

where

\[
E[S_r|X_r = x] = \begin{cases} 164 - (A_d - N_{r_{\max}}) & \text{if } 164 - (A_d - N_{r_{\max}}) < x \\ x & \text{if } 164 - (A_d - N_{r_{\max}}) > x \end{cases}
\]
Therefore, the total expected profit can be written as a following equation:

\[
E[profit] = f_r \left[ \int_0^{164-(A_d-N_{r_{max}})} x \cdot f_{X_r}(x)dx 
+ \int_{164-(A_d-N_{r_{max}})}^{\infty} (164 - (A_d - N_{r_{max}})) \cdot f_{X_r}(x)dx \right] \\
+ f_d \left[ \int_0^{A_d+N_{d_{max}}} x \cdot f_{X_d}(x)dx + \int_{A_d+N_{d_{max}}}^{\infty} (A_d + N_{d_{max}}) \cdot f_{X_r}(x)dx \right] \\
- r_n \cdot f_r \left[ \int_0^{164-(A_d-N_{r_{max}})} x \cdot p_{N_r} \cdot f_{X_r}(x)dx \right] \\
+ \left[ \int_{164-(A_d-N_{r_{max}})}^{\infty} (164 - (A_d - N_{r_{max}})) \cdot p_{N_r} \cdot f_{X_r}(x)dx \right] \\
- c_r \cdot 164 - o_r \cdot f_r \cdot \max\{0, E[E[S_d|X_d]] - p_{N_d} \cdot E[S_d] + E[E[S_r|X_r]]\} \\
- p_{N_r} \cdot E[S_r] - 164 \right]
\]

In this model, our goal is to find an optimal number of \(A_d\) which maximize the total expected profit. To find the optimal value of ‘\(A_d\)’ and the maximized expected profit, we use numerical method to calculate the model with R program.

As a result, the profit can be maximized as \$415,662.66 when the number of allocated seats for each class is ‘\(A_d = 57, A_r = 107\)’. The numbers of selling tickets are \(S_d = 60, S_r = 116\), and the total number of selling tickets is 116+60=176.
[Figure 6] Graphical output of the model

![Plot of Expected Profit](image)

The number of seats allocation for discounted class

[Figure 7] R script for calculating the model – part I

```r
# evaluation of parameters
> Exp_D=62.923076923076920
> Exp_R=98.615384615384610
> Var_D=6.282433897435900e+02
> Var_R=2.338039743589744e+03
> pr_NoR=0.0466
> pr_NoD=0.0196
> fare_D=3000
> fare_R=5000
> refund_R=0.8
> cost_F=1000
> cost_Ov=1.5
>
> # initializing the value of Expected Profit
> Exp_Profit=0
```

> # initializing the value of capacity of selling tickets and its parameters
> s_R=0
> s_D=0
> max_NoD=0
> max_NoR=0
> # initializing the value of capacity of selling tickets and its parameters
> s_R=0
> s_D=0
> max_NoD=0
> max_NoR=0
> 
> # initializing the value of parameters of overbooking
> ovi_Ov=0
> Ov=0
> 
> # define functions of x for calculating integrations
> Xpdf_R <- function(x) {x*dnorm(x,Exp_R,sqrt(Var_R))}
> Xpdf_R2 <- function(x) {dnorm(x,Exp_R,sqrt(Var_R))}
> Xpdf_D <- function(x) {x*dnorm(x,Exp_D,sqrt(Var_D))}
> Xpdf_D2 <- function(x) {dnorm(x,Exp_D,sqrt(Var_D))}

[Figure 8] R script for calculating the model – part II
> # calculating the Expected Profit as increasing the value of a
> for (a in 1:164) {
>   +
>   + # calculating capacity of selling tickets (range parameters) for integration.
>   + max_NoD[a]=ceiling(qbinom(0.95,a,pr_NoD))
>   + max_NoR[a]=ceiling(qbinom(0.95,164-a,pr_NoR))
>   + s_D[a]=s+max_NoD[a]
>   + s_R[a]=164-s+max_NoR[a]
>   +
>   + # calculating each integrations
>   + int1=integrate(Xpdf_R, lower=-Inf, upper=s_R[a])
>   + int2=integrate(Xpdf_D2, lower=s_R[a], upper=Inf)
>   + int3=integrate(Xpdf_D, lower=-Inf, upper=s_D[a])
>   + int4=integrate(Xpdf_D2, lower=s_D[a], upper=Inf)
>   +
>   + # converting values of integrations to numerical values
>   + int1=as.numeric(int1[1])
>   + int2=as.numeric(int2[1])
>   + int3=as.numeric(int3[1])
>   + int4=as.numeric(int4[1])
>   +
>   + # calculating cri_Ov for determining maximum overbooking
>   + cri_Ov[a]=ceiling(((int1+s_R[a])*int2+(int3+s_D[a])*int4)
>   + -(pr_NoR*int1+pr_NoR*s_R[a]*int2)-(pr_NoD*int3+pr_NoD*s_D[a]*int4)-164)
>   +
>   + # calculating the number of overbooking for each class
>   + if (cri_Ov[a] > 0) {Ov[a]=cri_Ov[a]} else Ov[a]=0
>   +
>   + # calculation of the Expected Profit
>   + Exp_Profit[a]=fare_R*(int1+int2)+fare_D*(int3+s_D[a]*int4)
>   + -(refund_R*fare_R*pr_NoR*int1+pr_NoR*s_R[a]*int2)-cost_F*164-cost_Ov*fare_D*Ov[a]
>   +}

[Figure 9] R script for calculating the model – part III
```r
> # rounding up values of the Expected Profit
> round(Exp_Profit[,2])
```

[Figure 10] R script for calculating the model – part III
4. Dynamic Price Decision Model

Now, we will analysis dynamic price decision for a special case of price discrimination based on different rates for a given good or service depending on the time, day, and month and so on. Dynamic price decision model will provide us to optimize revenue by adjusting time changes. To define this model we have to set up a new index time $t$, which will be added to these and other time dependent decision variables.

4.1 Assumption

1. There is no cancellation ahead on boarding.
2. Demand of R and D are independent.
3. Demand for each class is according to a certain probability distribution respectively.
4. Each customer of both classes depends on each no-show probability.

*5. The ticket fares of discount seats are changes as time goes by, that is, the more purchase the discount ticket, the more discount it's cost, whereas regular seats revenue are fixed.

*6. The highest ticket fares of D cheaper than those of R.

7. An event of refund occurs only when a passenger on regular class doesn’t show-up (no-show).
8. Overbooked passengers who are not allowed to board will get refund higher money than ticket price.
9. The waiting list is needed only for the discounted seats.

4.2 Notation

4.2.1 Indices

$i = \{r, d\}$: $r$ is a indice for regural class and $d$ is a set for discounted class

4.2.2 Parameters

$f_r$: ticket fare for a regular class

$f_{at}$: ticket fare for a discount seat at time $t$

$r_i$: refund rate for a no show passenger on class $i$
4.2.3 Random Variables

\(X_{it}\): demand of tickets on class i before time t

\(Y(W)\): total passengers showing up to take a flight among W buyers

\(S_t\): the number of available seats before time t

\(N_i\): the number of no-show passengers on class i

4.3 Mathematical Model

4.3.1. State transition

This problem can be solved by using dynamic programming. First state transition should be considered before finding recursive equation. In this model, \(S_t\) plays role of state variable.

\[
S_{t-1} = [S_t - X_{it}]^+ = -X_{dt} \wedge u_t
\]

where

\[
[X]^+ = \begin{cases} x & \text{if } x \leq 0 \\ 0 & \text{if } x > 0 \end{cases}
\]

\(X_{dt} \wedge u_t = \min\{X_{dt}, u_t\}\)
4.3.2 Recursive equation

Here, we can derive the recursive equation for the optimal solution. \( u_t^* \) is a function including 0. First step of dynamic programming represents as follows:

\[
\prod_{t=1}^{n} t^*(s_t) = \max_{0 \leq u_t \leq [s_t - x_{rt}]^+} \prod_{t=1}^{t}(u_t | s_t)
\]

\[
= E_{D_{d1}}E_{Y(M)}[f_{rt}[X_{d1} \wedge S_t] + f_{d1}[X_{d1} \wedge S_t] - \alpha o[Y(M) - \bar{S}]^+}
\]

The feasibility region of \( u_t \) is included in the remaining seats after allocation of regular seats. Therefore, recursive equation can be shown as follows:

\[
\prod_{t=1}^{n} t^*(s_t) = \max_{0 \leq u_t \leq [s_t - x_{rt}]^+} \prod_{t}
\]

We can find optimal solution in \( n_{th} \) step, that is,

\[
\max_{s} \pi_{n}^*(s)
\]

Here, \( \pi_{n}^*(s) \) is optimal amount of ticket sale including overbooking.

5. Conclusion

We handled with this problem to find maximize expected profit when operating an airplane in specific case. To solve these problems, we found that given data follows normal probability distribution. Also, we mainly use conditional probability function and change parameter and variable, such as ticket fare, seats allocation and so on, to find various solutions. In addition, we use “R” program to earn precise conclusion. When we suggest dynamic price decision model, we had read various papers to propose reasonable formulation. Thus, we can make an appropriate formulation by using dynamic programming. We can apply these formulations in real situation if we more consider other constraints.
6. Reference