

UNIVERSITÀ DEGLI STUDI DI  
MILANO-BICOCCA

DECISION MODELS

FINAL PROJECT

---

CASINO FLOOR MIX OPTIMIZATION  
AT LUCKY DUCK ENTERTAINMENT  
A STUDENT SOLUTION

---

*Authors:*

Beatrice Fumagalli - 784549 - b.fumagalli9@campus.unimib.it

Matteo Porcino - 748876 - m.porcino1@campus.unimib.it

Pierluigi Tagliabue - 835211 - p.tagliabue7@campus.unimib.it

17 giugno 2018



## Sommario

*Casino floor mix optimization at Lucky Duck Entertainment - a student solution* vuole suggerire un approccio originale e non eccessivamente complicato, da parte degli studenti, per fornire una soluzione ed individuare il migliore mix di slot machines all'interno delle sale dei casinò *Lucky Duck Entertainment*. I dati utilizzati all'interno del progetto, forniti sotto forma di tabella *excel* sono stati analizzati, manipolati e compresi per poter supportare la formulazione del problema di programmazione lineare intera che massimizzasse il profitto di ogni sala dei casinò.

## 1 Introduzione

Questo progetto suggerisce un approccio e una soluzione per trovare la miglior combinazione di *Slot Machines* che massimizzi il profitto di due casinò in Nevada. Entrambi i casinò sono suddivisi in quattro sezioni, rispettivamente chiamate *Entrance*, *Interior*, *Boundary*, *RestaurantPlaza*, ognuna delle quali ha un numero differente di *Slot Machines*. L'obiettivo di questa analisi è quello di determinare il miglior mix di *Slot Machines*, in ognuno dei casinò in oggetto, che sia in grado di massimizzarne i profitti. La totale mancanza di dati che facessero riferimento alla superficie disponibile all'interno delle sale dei casinò, allo spazio occupato da ogni singola slot machine e i dati parziali sulla disposizione mensile di queste ha portato alla formulazione di un'assunzione che ha reso possibile lo svolgimento di questo progetto. L'assunzione principale formulata è che **il numero di slot machines presenti all'interno di ogni casinò sia costante durante il periodo di osservazione**. Durante il processo di formulazione del problema sono state fatte ulteriori assunzioni che verranno presentate in seguito nella sezione 2.2 dedicata all'analisi dei dati presenti all'interno del dataset poichè sono una diretta conseguenza di queste ultime. Contestualmente alle operazioni effettuate sui dati a disposizione è stata fatta una ricerca online per poter capire quali potessero essere gli approcci suggeriti per la formulazione e risoluzione del problema. Sono stati individuati un totale di due articoli che trattassero lo stesso case study fornito da **SAS** o un case study affine. Il primo è una tesi di dottorato a cura di *Kasra Christopher Ghaharian* dell'Università del Nevada [1], mentre il secondo è uno *Statement of Work* a cura di *Shin Woong Sung*[2] basato sulla challenge proposta da **SAS**. Dopo aver correttamente manipolato i dati e formulato il problema di programmazione lineare è stata usata

la libreria Python Pulp[3] per poter implementare il modello ed ottenerne la soluzione.

## 2 Dataset

Il dataset iniziale fornito all'interno del file "Lucky Duck Entertainment revenue 2013.xls" presenta un totale di 4149 osservazioni. Per ogni record nel dataset sono disponibili i seguenti attributi:

- *MachineName*: Nome della *Slot Machine* presente all'interno dei casinò
- *Casino*: rappresenta il casinò in cui è posizionata la slot machine, sono presenti 3007 osservazioni per il valore **Aries** e 1142 osservazioni per il valore **Libra**
- *Section*: rappresenta le quattro sezioni in cui i casinò sono suddivisi. Sono Boundary, Entrance, Interior, RestaurantPlaza
- *Denomination*: corrisponde alla somma minima necessaria per ogni giocata
- *Machine Type*: *reel* , *video* o *vpoker*
- *Manufacturer*: azienda fabbricante della *Slot Machine*, sono 7 nell'intero dataset
- *Model*: modello della *Slot Machine*, in totale 54 valori
- *Month* : Timestamp del mese a cui si riferisce ogni record, vanno da '2011-09-01' a '2012-08-01'
- *Revenue* lorda e per *unit machine*
- *NoMachines*: numero di *Slot Machine* dello stesso tipo
- *Plays* totali e per *unit machine*

## 2.1 Data Exploration

La prima operazione effettuata è stata quella di esplorare i dati in possesso per riuscire ad avere un quadro generale di quelle che erano le informazioni a disposizione.

Si è notato fin da subito l'assenza di un identificatore univoco, ad esempio un numero seriale, che individuasse le diverse slot machines presenti nei casinò per questo si è scelto di utilizzare come identificatore univoco la tupla formata da ***MachineName, Manufacturer, Model e Denomination***. La scelta di includere anche la giocata minima per partita di una slot machine nella tupla identificativa è dovuta alla presenza di record che si riferiscono alla medesima slot machine ma con il valore di *Denomination* differente e conseguentemente differenti valori legati ai profitti.

Osservando diversi campioni del dataset è stato possibile notare la presenza di alcuni record pressochè identici tra di loro eccetto che per l'attributo *No-Machines*. È stata effettuata un'operazione di `groupby` per poter aggregare questi record in un'unica osservazione.

Analizzando i dati sulla base dell'attributo *Month*, aggregandoli per sezione e sommando il numero di slot machine presenti al loro interno, la situazione che si presentava era la seguente:

	Boundary	Entrance	Interior	RestaurantPlaza		Boundary	Entrance	Interior	RestaurantPlaza
Month					Month				
2011-09-01	195	219	248	187	2011-09-01	53	56	39	82
2011-10-01	109	114	97	130	2011-10-01	22	19	32	26
2011-11-01	85	114	108	109	2011-11-01	28	40	37	24
2011-12-01	125	120	104	94	2011-12-01	30	25	21	41
2012-01-01	145	97	135	118	2012-01-01	30	25	27	24
2012-02-01	98	110	154	125	2012-02-01	36	23	25	26
2012-03-01	89	104	91	120	2012-03-01	30	20	30	27
2012-04-01	99	146	110	119	2012-04-01	22	21	27	31
2012-05-01	144	112	107	135	2012-05-01	28	37	30	21
2012-06-01	119	111	100	148	2012-06-01	30	35	26	33
2012-07-01	87	118	119	114	2012-07-01	35	37	30	22
2012-08-01	100	125	132	107	2012-08-01	34	32	33	33

Figura 1: Capienza delle sezioni del Casinò Aries

Figura 2: Capienza delle sezioni del Casinò Libra

Si può vedere come le osservazioni riferite ai mesi successivi a '2011-09-01'

includano un numero molto inferiore di dati riferiti alle slot machines. Dopo aver controllato che all'interno dei record in questione non fossero presenti tuple non osservate a '2011-09-01' si è potuto concludere che la capienza totale delle sale dei casinò, e dei casinò stessi, sia quella individuata dai record con attributo *Month* pari a '2011-09-01'.

Possiamo quindi riassumere le assunzioni emerse in questa fase nelle seguenti:

- Ogni singola slot machine è univocamente identificata dalla tupla:  $\{ \textit{MachineName}, \textit{Manufacturer}, \textit{Model}, \textit{Denomination} \}$
- Alla data di *Settembre 2011* sono presenti tutte le slot machines presenti all'interno dei casinò per il periodo di osservazione contenuto nel dataset
- La capienza totale delle sale dei casinò, e dei casinò stessi, è data dal numero di singole slot machines presenti all'interno delle sale alla data di *Settembre 2011*

## 2.2 Data Manipulation e Cleaning

Il processo di *Data Cleaning*, nonostante il grande numero di record mancanti per i diversi mesi di osservazione, si è limitato all'eliminazione dal dataset di quegli attributi per cui purtroppo non si è riusciti né a trovare una descrizione del dato che rappresentassero né a trovare una relazione che legasse questi con il resto degli attributi.

Al contrario, il processo di *Data Manipulation* sul dataset è stato fondamentale per poter adattare i dati in possesso in una forma tale da permettere la creazione del modello di programmazione lineare.

### 2.2.1 Inventario Slot Machines

Per ogni casinò è stato creato un *DataFrame*, mediante l'utilizzo della libreria *Pandas*[4] con i seguenti attributi: *MachineName*, *Manufacturer*, *Model*, *MachineType*, *Denomination*, *NoMachines*.

In questi *DataFrame* sono contenute tutte le diverse slot machines presenti all'interno di ogni casinò con la quantità di slot per ogni configurazione diversa di slot (intendiamo con configurazione la tupla identificativa della slot). Si ottengono così:

- Per il Casinò *Aries*: 424 configurazioni diverse per un totale di 849 slot

- Per il Casinò *Libra*: 169 configurazioni diverse per un totale di 230 slot

### 2.2.2 DataFrame dei profitti

Come per l'inventario delle slot machine è stato necessario creare un **DataFrame** per ogni casino al cui interno fossero presenti per ogni configurazione di slot machine il valore del profitto ad essa associato. Per calcolare il profitto di ogni slot machine si è partiti dagli attributi *GrossRevenuePerMachine* e *PlaysPerMachine* disponibili all'interno del dataset.

Questi, essendo dati con granularità mensile, sono stati divisi per 30 per ottenere delle quantità che potessero essere considerate come dei dati giornalieri rispettivamente del guadagno medio e delle giocate medie. Andando a moltiplicare riga per riga queste nuove quantità calcolate si ottiene un valore che andiamo ad indicare con l'attributo **Daily\_profitability**.

Con questo nuovo attributo a disposizione è stata effettuata una **groupby** sugli attributi delle tuple identificative delle slot machines e sull'attributo *Section*, creando così il **DataFrame** dei profitti. In questa tabella è possibile trovare i dati della redditività di ogni configurazione di slot machine rispetto ad ogni *Section* in cui è stata osservata quella configurazione.

Facendo questo tipo di considerazione è stato possibile inoltre gestire il problema dell'assenza di osservazioni per determinate configurazioni nell'arco di tempo coperto dal dataset.

## 3 The Methodological Approach

In questa sezione si andrà ad illustrare come è stato formulato ed implementato il modello di programmazione lineare per la risoluzione di questo problema, facendo particolare attenzione a come sono state definite le variabili di decisione, la funzione obiettivo e i vincoli del problema. Prima di entrare nel merito del modello andiamo a mettere insieme tutte le assunzioni che sono state formulate per la creazione e risoluzione dello stesso.

### 3.1 Assunzioni del Modello

- Alla data '2011-09-01' sono già presenti tutte le slot machines di entrambi i casinò

- La capienza dei singoli ambienti del casinò è data dal numero di slot machines presenti alla data '2011-09-01' ed è costante nel tempo
- Le configurazioni di slot machines presenti alla data '2011-09-01' sono costanti, pertanto non è possibile modificare il valore dell'attributo *Denomination*.
- Il profitto di una configurazione di slot machine varia al variare dell'ambiente in cui è posizionata
- Il valore della redditività di ogni configurazione è dato dalla media delle singole redditività registrate mensilmente, è pertanto un valore indicativo e non veritiero

## 3.2 Variabili Decisionali

Come variabili di decisione per il modello sono state definite un totale di:

- 3396 variabili per il Casinò Aries (849 slot machine per 4 sezioni)
- 920 variabili per il Casinò Libra (230 slot machine per 4 sezioni)

Ogni variabile di decisione è una variabile di decisione binaria così definita:

$$X_{j,i} = \begin{cases} 1 & \text{se la slot } i \text{ è presente nella sezione } j \\ 0 & \text{altrimenti} \end{cases} \quad (1)$$

$$j \in Section, i \in \{\# \text{ di slot machines nel casinò}\}$$

Per definire queste variabili è stato necessario utilizzare la funzione resa disponibile all'interno della libreria Pulp, `pulp.LpVariable.dicts`.

```
problem_variables = pulp.LpVariable.dicts("is there",
((j,i) for j in section
  for i in decision_variables),
0,1,LpBinary)
```

Questa funzione è in grado di creare un dizionario utilizzando come chiavi la combinazione degli indici passati come argomento, ciò è possibile grazie alla `list comprehension` propria di Python. Per semplicità di lettura del codice ad ogni singola slot machine è stata assegnata una variabile nella forma `x_i` ed è stata creata una tabella per conservare la corrispondenza tra variabile di decisione e corrispondente slot machine.

### 3.3 Matrice dei Profitti

Per poter utilizzare i dati relativi alla redditività precedentemente calcolati è stato necessario creare una matrice dei profitti per ogni casinò.

Questa matrice è formata da un numero di righe pari al numero di slot machines presenti all'interno dei casinò (849 per *Aries* e 230 per *Libra*) ed un numero di colonne pari agli ambienti presenti di questi (*Boundary*, *Entrance*, *Interior*, *RestaurantPlaza*), è stato possibile ottenere così una tabella con un numero di celle pari al numero di variabili di decisione del problema. Al fine di utilizzare queste matrici all'interno del modello si è resa necessaria la loro trasformazione in due **DataFrame** aventi come attributi le diverse *Section* presenti in ogni casinò e come indice il nome delle variabili di decisione `x_i`, così facendo è possibile per la libreria **Pulp** accedervi mediante gli stessi indici `j` ed `i` con la quale accede al dizionario creato nella sezione 3.2.

### 3.4 Modello

Attraverso l'utilizzo della libreria PuLP viene definito il problema di programmazione lineare, specificando che si tratta di un problema di massimizzazione con il comando `pulp.LpMaximize`.

La **funzione obiettivo** del problema è espressa nella formula:

$$\sum_{j \in Section} \sum_{i \in Decision\_variables} X_{j,i} P_{j,i} \quad (2)$$

Dove  $X_{j,i}$  sono le variabili binarie definite nella sezione 3.2 e  $P_{j,i}$  i coefficienti di profitto definiti nella sezione 3.3.

Possiamo osservare nell'immagine 3 sottostante come è avvenuta l'implementazione in codice della **funzione obiettivo 2**.

```
prob += (
    pulp.lpSum(
        [profit_coeff[j][i] * is_machine_var[(j,i)]
         for j in section for i in decision_variables]
    )
)
```

Figura 3: implementazione della funzione obiettivo nella sintassi della libreria Pulp

Notare come la modalità di accesso alle variabili `profit_coeff` e `is_machine_var` sia differente ma avvenga tramite gli stessi indici `j` ed `i`.



### 3.5 Vincoli

Per il modello di ottimizzazione creato i vincoli da definire sono i seguenti:

- vincolo sulla capienza di ogni section del casinò

$$\sum_{j \in Section} X_{j,i} = \text{capienza della sala } j, \forall i \quad (3)$$

- ogni slot machine può comparire in al più una sezione del casinò

$$\sum_{i \in Decision\_variables} X_{j,i} = 1, \forall j \quad (4)$$

Vediamo nella figura 4 l'implementazione dei vincoli per il *Casino Aries*.

```
prob += pulp.lpSum([is_machine_var['Boundary', i] for i in decision_variables]) == 195 #Boundary
prob += pulp.lpSum([is_machine_var['Entrance', i] for i in decision_variables]) == 219 #Entrance
prob += pulp.lpSum([is_machine_var['Interior', i] for i in decision_variables]) == 248 #Interior
prob += pulp.lpSum([is_machine_var['RestaurantPlaza', i] for i in decision_variables]) == 187 #RestaurantPlaza
#La somma dei valori di Xji nelle 4 sezioni deve essere pari a 1
for i in decision_variables:
    prob += is_machine_var['Boundary',i] + is_machine_var['Entrance',i] +
            + is_machine_var['Interior',i] + is_machine_var['RestaurantPlaza',i] == 1
```

Figura 4: implementazione dei vincoli nella sintassi della libreria Pulp

## 4 Risultati

Calcolato il modello per i due casinò e sulla base delle assunzioni fatte precedentemente, la soluzione ottima per il *Casino Aries* è pari a 1.896.884.720,64 \$ mentre per il *Casino Libra* è 512.223.405,79 \$, rispetto ai dati iniziali riferiti a *Settembre 2011* che sono pari a 1.176.521.023,72 \$ per il primo e 310.458.333,79 \$ per il secondo.

Successivamente alla risoluzione, per entrambi i modelli, si è eseguito un controllo sulla capienza delle singole sezioni che è risultata corretta e sul posizionamento di ogni slot machine all'interno di al più una sezione, anch'esso corretto, prova che i vincoli utilizzati sono stati rispettati durante l'esecuzione del modello. Nella parte sottostante è possibile vedere nelle figure 5 e 6 i

grafici riferiti alla redditività mensile di ogni casinò ripartiti per i diversi ambienti. L'ultimo bin in entrambi i grafici, etichettato con data '2012-09-01' rappresenta la redditività della disposizione ottimale calcolata.

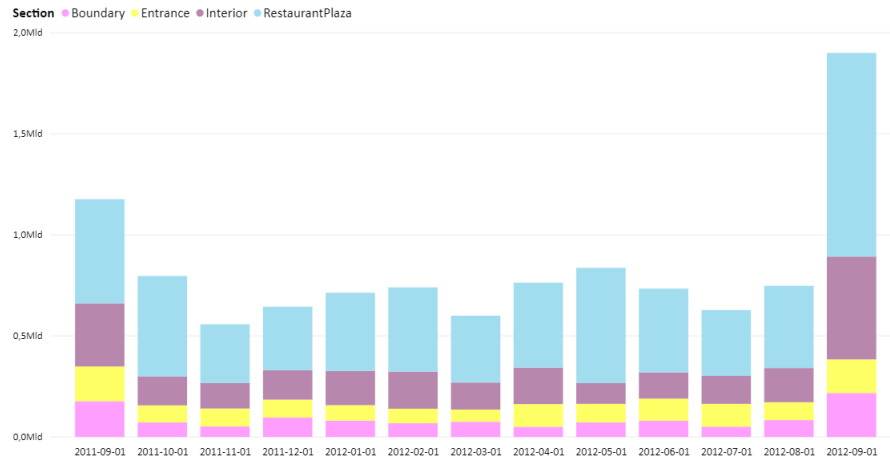


Figura 5: Redditività del *Casinò Aries* suddivisa per mesi e section

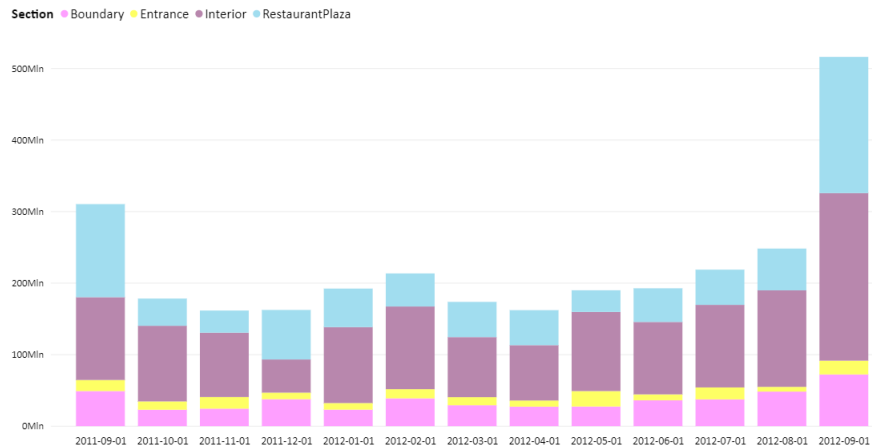


Figura 6: Redditività del *Casinò Libra* suddivisa per mesi e section

Sono mostrate ora attraverso le figure 7 e 8 le disposizioni delle slot machine riferite a *Settembre 2011* e alla disposizione calcolata con il modello. Lo scopo delle immagini è di mostrare come effettivamente la disposizione delle slot machine sia stata cambiata.

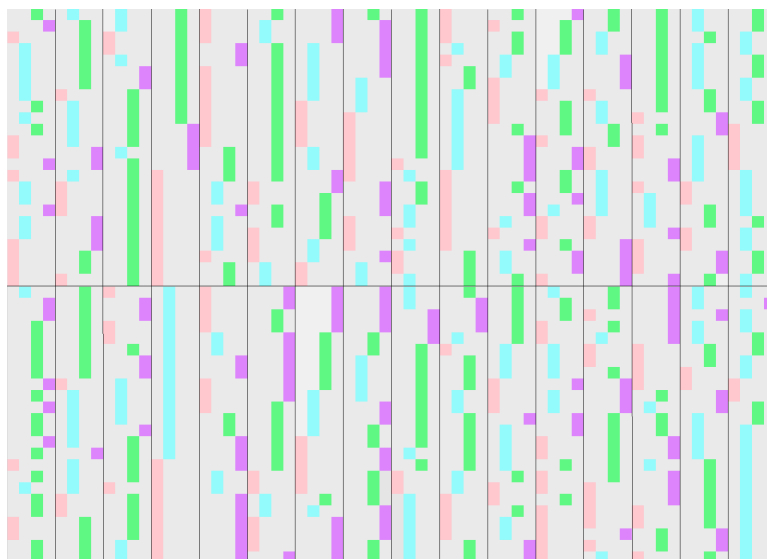


Figura 7: *Casinò Aries*, nella parte alta vediamo la disposizione a *Settembre 2011*, in basso la disposizione ottimale calcolata

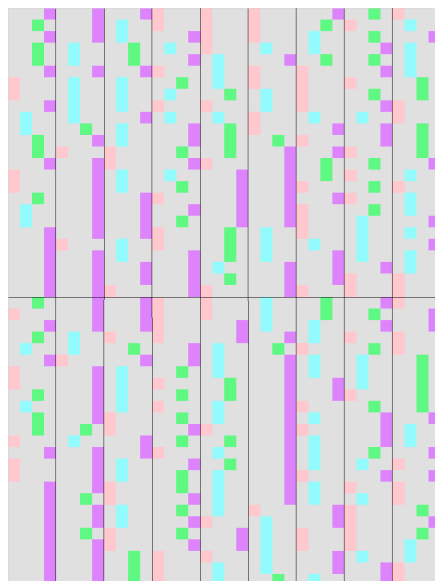


Figura 8: *Casinò Libra*, nella parte alta vediamo la disposizione a *Settembre 2011*, in basso la disposizione ottimale calcolata

## 5 Discussione

I risultati dei modelli di ottimizzazione precedentemente descritti contengono il miglior mix di macchine per casinò capace di massimizzarne il profitto. I

grafici sotto riportati possono dare un'idea concreta al lettore della differenza tra disposizioni, da un lato abbiamo un grafico rappresentate la disposizione iniziale delle slot dall'altro la loro disposizione ottimale calcolata grazie modello. È tuttavia doveroso specificare che, nonostante gli ottimi risultati ottenuti, i dati a disposizione erano molto scarni oltre che mancanti e questo ha inevitabilmente influito sulla verosimiglianza dei dati calcolati.

Nonostante questo fondamentale problema, lo scopo del progetto non riguardava il calcolo preciso della profitability delle macchine, ma la ricerca del mix ottimo capace di massimizzare il profitto. Obiettivo poi tradotto nei grafici precedentemente descritti che rappresentano appunto la differenza tra la disposizione di settembre 2011 e la disposizione ottimale delle singole macchinette.

Infine è bene dire che se ci fossero stati a disposizione altri tipi di dati, primi fra tutti quelli spaziali riguardanti le dimensioni delle varie sezioni dei casinò e quelle riguardanti le macchinette si sarebbe potuto anche massimizzare l'utilizzo dello spazio per ogni settore.

## 6 Conclusioni

In conclusione, il lavoro svolto è riuscito nel suo intento, ossia trovare la disposizione ottimale delle *Slot Machines* all'interno dei due casinò. Nonostante i risultati possano esser definiti come definitivi, è tuttavia doveroso ribadire che la mancanza di dati combinata alla scarsità di informazioni a disposizione non hanno consentito un'analisi più approfondita e completa di quella effettuata.

Al fine di una migliore analisi sarebbe stato utile aver avuto a disposizioni dati più precisi e completi, associati alle singole slot oltre ai dati dimensionali già precedentemente citati.

## Riferimenti bibliografici

- [1] K. C. Ghaharian, “A mathematical approach for optimizing the casino slot floor: A linear programming application,” Ph.D. dissertation, University of Nevada, Las Vegas, 2010.
- [2] S. W. Sung, “Statement of work,” p. 5.
- [3] S. Mitchell, A. Kean, A. Mason, M. O’Sullivan, and A. Phillips, “PuLP: A linear programming toolkit for Python,” 2011–. [Online]. Available: <https://www.coin-or.org/PuLP/>
- [4] W. McKinney, “Data structures for statistical computing in python,” pp. 51 – 56, 2010.