

Anche i bibliotecari (talvolta) devono prendere decisioni

Con questo intervento si conclude il ciclo dedicato alla "ricerca operativa"

di Mario Sebastiani

Avevamo visto, nel primo di questi interventi dedicati alla ricerca operativa¹ che, in definitiva, questa disciplina è un *corpus* di tecniche e metodi utili per mettere a punto modelli concettuali della realtà che favoriscano l'adozione di decisioni razionali. Non c'è dubbio che nelle biblioteche, soprattutto in seguito alla diffusione dell'automazione, cresce la necessità di una scienza della *sana amministrazione*. Questa scienza — se per scienza intendiamo la capacità di analizzare i problemi e formulare previsioni attendibili soprattutto sotto il profilo quantitativo — esiste ed ha un nome: *ricerca operativa*.

La conoscenza dei metodi compresi in questa disciplina può quindi favorire il perseguimento di una gestione efficiente della biblioteca.

Ma gestire significa risolvere problemi e per risolvere i problemi occorrono risposte, cioè *informa-*



zioni utili. Un problema di gestione è spesso un problema di scelta tra diverse opzioni possibili. Per effettuare una scelta razionale occorrono informazioni e queste devono essere desunte dai dati di cui disponiamo mediante opportune procedure di *analisi dei dati*,

procedure che fanno parte del vasto campo della statistica.

Nel secondo intervento² abbiamo visto come applicare concretamente la più canonica di queste procedure, vale a dire il test del *chi-quadrato* e del *coefficiente di contingenza*. Sempre in questo intervento abbiamo avuto modo di constatare come l'applicazione di queste metodologie richiede una certa competenza non solo in campo matematico e statistico ma anche in campo informatico: la capacità di programmare infatti consente, mediante un personal computer, di effettuare rapidamente calcoli molto complessi. Scrivere piccoli software ad hoc per i nostri calcoli ci consente di elaborare grandi quantità di dati,

di verificare come cambiano i risultati al variare dei dati in input, di memorizzare e riutilizzare in un secondo momento sia i dati, sia i risultati, sia il software. Per illustrare in che modo la capacità di programmare estende e migliora le nostre possibilità di analizzare e comprendere i problemi, nel secondo intervento — per l'esempio di applicazione del test del chi-quadrato — ci si è avvalsi di un piccolo programma Basic sviluppato appositamente.

Se l'analisi dei dati e la conseguente elaborazione di informazioni utili sul sistema oggetto di studio costituiscono la fase preliminare di ogni applicazione della ricerca operativa, la fase successiva è rappre-

sentata dall'elaborazione di un modello del sistema. Nella ricerca operativa il modello di un sistema non è altro che un insieme di equazioni matematiche e di vincoli su tali equazioni che esprimono i legami quantitativi che intercorrono tra le variabili che compongono

il sistema e che consentono quindi di prevederne il comportamento. Il modello naturalmente è solo un'approssimazione della realtà. Una volta elaborato il modello, si assume, al fine di effettuare previsioni e prendere decisioni, che il sistema si comporti effettivamente come è stabilito nel modello. SCOPO ultimo della ricerca operativa è allora l'elaborazione di modelli sempre più rifiniti del sistema oggetto di studio. Dal punto di vista dell'approssimazione alla realtà, i modelli si dividono in *lineari* e *non lineari*. I modelli lineari sono quelli composti solo da equazioni e disequazioni di primo grado. I modelli non lineari sono quelli che non rispettano questa limitazione. I modelli lineari costituiscono, chiaramente, un'approssimazione più grossolana alla realtà, ma sono i modelli con i quali è più facile studiare il comportamento del sistema e prendere decisioni in merito.

Nel terzo intervento³ abbiamo avuto modo di vedere un esempio di *programmazione lineare* (in questo caso il termine programmazione non ha il consueto significato di codificazione di un programma software) costituito dall'applicazione di procedure di *ottimizzazione matematica* ad un modello lineare rappresentante gli acquisti di una biblioteca. In questa occasione abbiamo avuto modo di verificare come, tramite i fogli di calcolo elettronici più avanzati, è possibile applicare le procedure di ottimizzazione matematica senza conoscere il formalismo matematico che ne sta alla base e senza avere nozioni di programmazione. Per applicare queste procedure è sufficiente conoscere, a livello elementare, il funzionamento dei fogli di calcolo.

Nel quarto intervento⁴ invece abbiamo visto un esempio di *programmazione non lineare*, vale a dire un esempio di applicazione

della teoria delle code al prestito bibliotecario. In questa occasione, data la complessità nel formalismo matematico, per sviluppare l'applicazione concreta dell'esempio, è stato codificato nuovamente un piccolo programma Basic.

In questo quinto ed ultimo intervento infine entriamo nel merito di quello che è lo scopo ultimo per cui elaboriamo modelli formali, siano essi lineari o non lineari, di un dato sistema. Questo scopo ultimo non è altro che quello di procedere, sulla base delle previsioni che possiamo trarre dal modello, ad una riorganizzazione del sistema al fine di migliorarne il rendimento complessivo. Ma per riorganizzare un sistema occorre necessariamente individuare, tra diverse alternative progettuali, che sono il risultato di differenti analisi e differenti programmazioni, l'alternativa "migliore". L'elaborazione di un modello infatti non fornisce automaticamente la soluzione giusta per migliorare il sistema. Lo studio del sistema, reso possibile dal modello, conduce sempre all'individuazione di un numero più o meno grande di alternative progettuali. Alternative cioè che costituiscono differenti ipotesi di lavoro su come procedere alla riorganizzazione del sistema.

Scegliere la migliore tra varie alternative possibili è un problema di decisione e il bibliotecario — all'interno dei propri margini di autonomia decisionale — condivide questo problema con altre figure professionali quali il direttore di un ospedale, il preside di una scuola, il responsabile di una rete di trasporti pubblici, ecc. Nella forma i metodi di decisione sono gli stessi, indipendentemente dal settore sociale dove vengono applicati e sono uguali anche le questioni di ordine più generale implicite in una metodologia di decisione rispetto ad un'altra.

È opportuno quindi, prima di de-

scrivere metodi utili per prendere "decisioni razionali", soffermarci a riflettere su che cos'è una decisione razionale e soprattutto su quando può dirsi razionale una decisione amministrativa, intendendo con questo una decisione adottata all'interno di un sistema sociale pubblico.

Fatti e valori condizionano le nostre scelte

Nei sistemi pubblici le alternative progettuali di ampio respiro, quelle che incidono profondamente sull'organizzazione complessiva del sistema, sia esso una biblioteca, una scuola, un'ospedale ecc. possono essere ordinate tra loro — da un livello di massima ad un livello di minima preferibilità — solo con grande difficoltà. Nei sistemi sociali pubblici infatti non è possibile definire un'unica funzione di rendimento che consenta di determinare esplicitamente quanto sia apprezzabile un'alternativa progettuale rispetto ad un'altra. Perché? Rispondere a questa domanda ci conduce necessariamente ad un terreno di indagine più vasto tanto della biblioteconomia che della ricerca operativa.

Sebbene non esaustive, le riflessioni sulla scienza dell'amministrazione sviluppate dal grande filosofo americano Herbert Simon costituiscono un punto di riferimento fondamentale nel grande dibattito sulla teoria delle decisioni. L'assunto di partenza di Simon — esplicitato per la prima volta nell'opera del 1947 *Il comportamento amministrativo* — è che ogni decisione amministrativa contiene tanto elementi di fatto che elementi di valore.⁵ Elementi di fatto, vale a dire premesse empiriche della decisione, sono quelle affermazioni suscettibili di essere verificate come "vere" o "false" in base al confronto con la realtà. ➤

Viceversa elementi di valore, cioè premesse etiche della decisione, sono affermazioni delle quali non si può sostenere la verità o la falsità tramite il confronto con la realtà. Ne consegue che definire "corretta" una decisione amministrativa è una questione alquanto complessa che non può essere risolta tramite una mera verifica empirica delle conseguenze della decisione.

Nella misura in cui la biblioteca si configura come un sistema sociale pubblico, anche il bibliotecario è chiamato a prendere decisioni che contengono elementi di fatto ed elementi di valore. Anche in biblioteca allora, quando occorre esprimere una scelta tra diverse alternative progettuali di ampio respiro — ad esempio per l'automazione della biblioteca, per la riorganizzazione dei servizi, per il trasferimento in un'altra sede, per l'adesione a sistemi bibliotecari locali, nazionali, internazionali, ecc. — tale scelta non può essere considerata una mera decisione "tecnica". Come tutte le decisioni amministrative, si sottrae, per via della sua componente etica, esplicita o implicita che sia, ad una mera valutazione empirica dei risultati. Prendere decisioni razionali è un problema molto più semplice da risolvere nelle organizzazioni private volte al profitto che non nei sistemi sociali. Anche in un'organizzazione privata occorre prendere decisioni che contengono elementi di valore, ma si tratta di valori propri dello spirito imprenditoriale che sono facilmente sottoponibili a misurazioni e calcoli. Come mette in evidenza Simon: "La semplicità del criterio di efficienza nelle organizzazioni commerciali o industriali è in gran parte dovuta al fatto che il denaro costituisce un denominatore comune per la misurazione della produzione e del ricavo consentendo un loro confronto diretto. Il

Tab. 1

B1:	{	accettae	"Codice di riconoscimento generale: ";k1}~
B2:	{	accettae	"Nome del recensore: ";k2}~
B3:	{	accettae	"Nome del programma: ";k3}~
B4:	{	accettae	"Release (numero o data): ";k4}~
B5:	{	accettae	"Attrezzatura necessaria; (enter) ";k5}~
B6:	{	accettae	"Computer: ";k6}~
B7:	{	accettae	"Disk Drives necessari: ";k7}~
B8:	{	accettae	"Monitor richiesto: ";k8}~
B9:	{	accettae	"Periferiche richieste: ";k9}~
B10:	{	accettae	"Assegnare un valore compreso tra 0 e 5; (enter) ";k10}~
B11:	{	accettan	"Metodo di avviamento (0-5): ";k11}~
B12:	{	accettae	"Tempi operativi; (enter) ";k12}~
B13:	{	accettan	"Tempo di avviamento (0-5): ";k13}~
B14:	{	accettan	"Tempo operativo (0-5): ";k14}~
B15:	{	accettae	"Materiali supplementari forniti; (enter) ";k15}~
B16:	{	accettan	"Documentazione (0-5): ";k16}~
B17:	{	accettan	"Guida alla biblioteca (0-5): ";k17}~
B18:	{	accettae	"Soluzioni per la gestione dei problemi; (enter) ";k18}~
B19:	{	accettan	"Disponibilità di una hotline (0-5): ";k19}~
B20:	{	accettan	"Possibilità di sostituzione (0-5): ";k20}~
B21:	{	accettae	"Assegnare un valore compreso tra 0 e 5; (enter) ";k21}~
B22:	{	accettae	"Contenuto del programma; (enter) ";k22}~
B23:	{	accettan	"Scopi e obiettivi esplicitati (0-5): ";k23}~
B24:	{	accettan	"Scopi e obiettivi chiari (0-5): ";k24}~
B25:	{	accettan	"Scopi e obiettivi hanno valenza educativa (0-5): ";k25}~
B26:	{	accettan	"Scopi e obiettivi sono realistici (0-5): ";k26}~
B27:	{	accettan	"Scopi e obiettivi coprono le necessità della biblioteca (0-5): ";k27}~
B28:	{	accettan	"Scopi e obiettivi coprono i curriculum (0-5): ";k28}~
B29:	{	accettan	"Metodologia di insegnamento congruente con il programma (0-5): ";k29}~
B30:	{	accettan	"Appropriate competenze pregresse necessarie (0-5): ";k30}~
B31:	{	accettan	"Appropriate conoscenze pregresse necessarie (0-5): ";k31}~
B32:	{	accettan	"Appropriata competenza linguistica necessaria (0-5): ";k32}~
B33:	{	accettan	"Utilizzabilità in differenti contesti (0-5): ";k33}~
B34:	{	accettan	"Utilizzazione di metodi educativi consolidati (0-5): ";k34}~
B35:	{	accettae	"Assegnare un valore compreso tra 0 e 5; (enter) ";k35}~
B36:	{	accettan	"Contenuti realistici (0-5): ";k36}~
B37:	{	accettan	"Esente da circoli viziosi e stereotipi (0-5): ";k37}~
B38:	{	accettan	"Conoscenza acquisita trasferibile (0-5): ";k38}~
B39:	{	accettae	"Presentazione; (enter) ";k39}~
B40:	{	accettan	"Personalizzazione (0-5): ";k40}~
B41:	{	accettan	"Indicazioni chiare (0-5): ";k41}~
B42:	{	accettan	"Indicazioni sullo schermo o accessibili (0-5): ";k42}~
B43:	{	accettan	"Formato chiaro e coerente (0-5): ";k43}~
B44:	{	accettae	"Schermate; (enter) ";k44}~
B45:	{	accettan	"Schermate efficaci (0-5): ";k45}~
B46:	{	accettan	"Spaziatura gradevole (0-5): ";k46}~
B47:	{	accettan	"Testo chiaro ed attraente (0-5): ";k47}~
B48:	{	accettan	"Caratteri di dimensioni e formato adeguato (0-5): ";k48}~
B49:	{	accettan	"Colori usati efficacemente (0-5): ";k49}~
B50:	{	accettan	"Animazione usata efficacemente (0-5): ";k50}~
B51:	{	accettan	"Grafica usata efficacemente (0-5): ";k51}~
B52:	{	accettan	"Suono usato efficacemente (0-5): ";k52}~
B53:	{	accettan	"Organizzazione logica e coerente del materiale (0-5): ";k53}~
B54:	{	accettae	"Impressioni nell'uso del programma; (enter) ";k54}~
B55:	{	accettan	"Impressioni piacevoli e divertenti (0-5): ";k55}~
B56:	{	accettan	"Impressioni interessanti (0-5): ";k56}~
B57:	{	accettan	"Impressioni stimolanti (0-5): ";k57}~

```

B58: ' {accettan "Stimola la creatività (0-5): ";k58}~
B59: ' {accettae "Uso e interazione con il computer; (enter) ";k59}~
B60: ' {accettan "User friendly (0-5): ";k60}~
B61: ' {accettan "Uso coerente dei tasti di controllo (0-5): ";k61}~
B62: ' {accettan "L'utente può modulare la velocità (0-5): ";k62}~
B63: ' {accettan "L'utente può inserirsi a differenti livelli e
posizioni (0-5): ";k63}~
B64: ' {accettan "L'utente può salvare i risultati e ripartire (0-
5): ";k64}~
B65: ' {accettan "L'utente può controllare i percorsi (0-5): ";k65}~

B66: ' {accettan "Attivo coinvolgimento dell'utente (0-5): ";k66}~
B67: ' {accettae "Risposte; (enter) ";k67}~
B68: ' {accettan "Input necessari segnalati (0-5): ";k68}~
B69: ' {accettan "Tentativi di accesso limitati per numero o per
tempo (0-5): ";k69}~
B70: ' {accettan "Accetta varie tipologie di input (0-5): ";k70}~
B71: ' {accettan "Accetta abbreviazioni (0-5): ";k71}~
B72: ' {accettan "Accetta sinonimi (0-5): ";k72}~
B73: ' {accettan "Consente risposte incomplete (0-5): ";k73}~
B74: ' {accettan "Feedback positivo (0-5): ";k74}~
B75: ' {accettan "Rinforzo positivo (0-5): ";k75}~
B76: ' {accettan "Possibilità di diramazioni (0-5): ";k76}~
B77: ' {accettan "Disponibilità dell'Help (0-5): ";k77}~
B78: ' {accettan "Disponibilità di spiegazioni e definizioni (0-5):
";k78}~
B79: ' {accettan "Facili uscita e restart (0-5): ";k79}~
B80: ' {accettan "Indipendenza dagli interventi del bibliotecario
(0-5): ";k80}~
B81: ' {accettae "Utilizzazione del bibliotecario: (enter) ";k81}~
B82: ' {accettan "Modificabile e/o espandibile (0-5): ";k82}~
B83: ' {accettan "Possibilità di registrare dati (0-5): ";k83}~
B84: ' {accettan "Possibilità di stampare dati (0-5): ";k84}~

```

concetto va però allargato, se lo si vuol applicare a processi di decisione in cui sono in gioco fattori non direttamente misurabili in termini monetari. Tali fattori saranno certamente presenti in organismi non industriali o commerciali, dove la valutazione della produzione in termini monetari è abitualmente priva di significato se non impossibile".⁶

Per ovviare ai problemi cui va incontro ogni tentativo di definire una nozione di efficienza nei sistemi pubblici paragonabile a quella delle organizzazioni commerciali o industriali, si è andato formando, nel mezzo secolo trascorso, "un corpus impressionante di elaborazioni teoriche formali, formulate da studiosi di statistica ed economisti allo scopo di aiutare la nostra ragione alle prese con queste materie, senza peraltro introdurre un nuovo tipo di logica. L'idea fondamentale di questo corpus teorico è costringere tutti i valori in un'unica funzione, la funzione di utilità, risolvendo in

tal modo il problema di come comparare valori fra loro disomogenei".⁷

Questa teoria si chiama teoria dell'utilità soggettiva prevista e la sua elaborazione "costituisce una delle più impressionanti imprese intellettuali della prima metà del ventesimo secolo ed essa è un meccanismo altamente sofisticato per applicare la ragione ai problemi di scelta".⁸

Il modello di utilità soggettiva attesa presuppone che il soggetto decisionale possa valutare l'intera gamma delle alternative a disposizione, con un panorama completo delle conseguenze future di ciascuna scelta in modo da poter conciliare o quantomeno equilibrare i valori in conflitto e sintetizzarli in un'unica funzione di utilità.

Ma si tratta di un modello che può vivere solo nel cielo delle idee di Platone. Nella realtà gli individui, osserva Simon, prendono decisioni secondo uno schema comportamentale che potremmo

definire di razionalità limitata. Secondo tale schema le decisioni sono generalmente inerenti ad argomenti ristretti e raramente vengono presi in considerazione dettagliati scenari futuri, magari corretti da un calcolo delle probabilità e delle alternative previste. È improbabile che un'unica onnicomprensiva funzione di utilità possa coprire l'intera gamma delle decisioni possibili.⁹

Analisi costi-benefici e analisi multicriteri

Nella ricerca operativa la procedura di decisione più calzante con il modello dell'utilità soggettiva attesa, è la cosiddetta analisi costi-benefici.

Questa tecnica di valutazione degli investimenti si articola in alcune equazioni la cui applicazione è abbastanza semplice ma che si basano tutte sul presupposto che i benefici conseguenti ad un investimento siano esprimibili tramite un'unica funzione di rendimento monetaria. Ad esempio, se definiamo con:

C_0 il capitale iniziale;

C_n il capitale risultante al termine del periodo di investimento;

n il numero di unità di tempo (giorno, settimana, mese, ecc.) che compongono il periodo di investimento;

i il tasso di interesse, cioè l'interesse che un capitale matura in una delle unità di tempo considerate;

R la rata di ammortamento che si deve restituire, supponendo di operare in regime di capitalizzazione composta (regime nel quale l'interesse maturato viene sommato al capitale al termine di ciascun periodo costituendo il montante per il calcolo dell'interesse nel periodo successivo);

avremo che varranno le seguenti equazioni: ➤

$$C_0 = R * \frac{(1 + i)^n - 1}{(1 + i)^n i}$$

$$C_0 = \frac{C_n}{(1 + i)^n}$$

Queste espressioni individuano il metodo di analisi costi-benefici detto del *valore attuale netto*. Con questo metodo è possibile valutare più alternative di investimento e scegliere quella caratterizzata dal maggior valore positivo. Altri metodi per l'analisi costi-benefici sono il metodo dei costi-annui equivalenti, il metodo del tasso di redditività, il metodo del periodo di recupero.¹⁰ Tutti hanno in comune il fatto di confinare l'analisi esclusivamente al rendimento monetario di un investimento. Per applicare l'analisi costi-benefici ad un progetto pubblico occorre allora garantire che tutti i benefici di quel progetto siano esprimibili in termini monetari. Ma abbiamo visto, nel paragrafo precedente, le profonde difficoltà inerenti in questo approccio, quando l'analisi è orientata allo studio di sistemi pubblici.

Per ovviare ai limiti dell'analisi costi-benefici, sono stati elaborati, nell'ambito della ricerca operativa, metodi di valutazione dei progetti che prescindono dalla definizione di un'unica funzione monetaria di rendimento. Tali metodi, detti *multicriteri*, consentono di individuare, dentro un insieme più o meno vasto di alternative progettuali, un ordinamento di preferenza che, prescindendo dal calcolo di una funzione di rendimento univoca, fa riferimento solo ad una molteplicità più o meno ampia di criteri di valutazione quantitativa e/o qualitativa.

Le metodologie multicriteri sviluppate dagli studiosi di ricerca operativa sono alquanto complesse e la loro applicazione rende necessario scrivere programmi lunghi e complicati. In questa sede non esamineremo un "vero" metodo

multicriteri ma, invece, vedremo come, utilizzando le potenzialità offerte dagli odierni fogli di calcolo, si possono elaborare semplici ma efficaci procedure di valutazione multicriteri.

Scelta multicriteri con il foglio di calcolo

Tutti i fogli di calcolo prevedono oggi la possibilità di utilizzare le funzioni @ (chiamate così perché sono sempre precedute dal simbolo @) che eseguono calcoli avanzati restituendo i relativi valori, nonché la possibilità di creare le cosiddette macro, vale a dire sequenze di comandi del foglio elettronico e di funzioni @ che, scritte in determinate zone del foglio elettronico, costituiscono veri e propri programmi in grado di svolgere automaticamente sequenze complesse di attività e di calcoli. Nelle tabelle da 1 a 6 è riportato un programma macro scritto con il foglio di calcolo Quattro Pro — ma compatibile con il foglio Lotus 1-2-3 — che consente di mettere in opera una procedura di valutazione multicriteri per l'analisi di software per l'insegnamento.¹¹

Per scrivere una macro occorre scegliere una zona del foglio di calcolo che non verrà utilizzata per svolgere operazioni. In sequenza, verticalmente, si procede poi ad inserire in caselle distinte, secondo le modalità previste dal foglio di calcolo, i vari comandi in cui è articolata la macro.

Naturalmente per scrivere la macro occorre conoscere la terminologia con cui devono essere espressi i vari comandi e le funzioni @. Questa terminologia, insieme alle regole che ne governano l'uso, costituiscono un vero e proprio linguaggio di programmazione il cui dizionario e la cui sintassi sono riportati nei manuali

che corredano il foglio di calcolo. Oggi, anche un normale word processing consente di scrivere delle macro e quindi, di fatto, include un linguaggio di programmazione.

Per scrivere i comandi si possono utilizzare, indifferentemente, caratteri maiuscoli o minuscoli. Un comando macro deve essere sempre preceduto dal carattere «'» e racchiuso tra parentesi graffe «{,}». Al termine del comando occorre scrivere il carattere tilde «~». Questi caratteri non compaiono sulla tastiera italiana e quindi, per scriverli, è necessario premere il tasto ALT e digitare sul tastierino numerico il relativo codice ASCII. Per il carattere «{» il codice è 123; per il carattere «}» il codice è 125; per il carattere «~» il codice è 126. Attenzione a non utilizzare i numeri della tastiera alfanumerica ma solo quelli del tastierino numerico!

Una volta chiariti questi preliminari si può procedere a digitare sul foglio elettronico il programma macro riportato nelle Tabelle da 1 a 6. Occorrerà avere l'accortezza di digitarlo esattamente in quelle caselle (B1, B2, B3, ecc.) che precedono, con i due punti, i comandi. La macro infatti viene eseguita sequenzialmente: prima il comando della casella B1, poi il comando della casella B2, ecc. Determinati comandi però permettono di "saltare" il rigido ordine sequenziale e di affidare, per così dire, la prosecuzione della macro, a comandi contenuti in caselle differenti da quella immediatamente successiva (un comando di questo genere è quello contenuto nella casella B87 della Tabella 2). Se la macro non dovesse essere digitata nelle caselle predefinite, verrebbero persi questi riferimenti e la macro non potrebbe più operare correttamente.

Nella Tabella 1 sono riportati i comandi utilizzati dalla macro per l'inserimento dei dati, cioè le va-

Tab. 2

```

B85: '{posiziona}b240~
B86: '{poni m85;l85+1;}~
B87: '{se l85>5}{salto b167}~
B88: '{se l85=2}{b}~
B89: '{se l85=3}{b}{b}~
B90: '{se l85=4}{b}{b}{b}~
B91: '{se l85=5}{b}{b}{b}{b}~
B92: '{poni @cursore("indirizzo");k1}~
B93: '{d}{poni @cursore("indirizzo");k2}~
B94: '{d}{poni @cursore("indirizzo");k3}~
B95: '{d}{poni @cursore("indirizzo");k4}~
B96: '{d}{poni @cursore("indirizzo");k6}~
B97: '{d}{poni @cursore("indirizzo");k7}~
B98: '{d}{poni @cursore("indirizzo");k8}~
B99: '{d}{poni @cursore("indirizzo");k9}~
B100: '{d}{poni @cursore("indirizzo");k11}~
B101: '{d}{poni @cursore("indirizzo");k13}~
B102: '{d}{poni @cursore("indirizzo");k14}~
B103: '{d}{poni @cursore("indirizzo");k16}~
B104: '{d}{poni @cursore("indirizzo");k17}~
B105: '{d}{poni @cursore("indirizzo");k19}~
B106: '{d}{poni @cursore("indirizzo");k20}~
B107: '{d}{poni @cursore("indirizzo");k23}~
B108: '{d}{poni @cursore("indirizzo");k24}~
B109: '{d}{poni @cursore("indirizzo");k25}~
B110: '{d}{poni @cursore("indirizzo");k26}~
B111: '{d}{poni @cursore("indirizzo");k27}~
B112: '{d}{poni @cursore("indirizzo");k28}~
B113: '{d}{poni @cursore("indirizzo");k29}~
B114: '{d}{poni @cursore("indirizzo");k30}~
B115: '{d}{poni @cursore("indirizzo");k31}~
B116: '{d}{poni @cursore("indirizzo");k32}~
B117: '{d}{poni @cursore("indirizzo");k33}~
B118: '{d}{poni @cursore("indirizzo");k34}~
B119: '{d}{poni @cursore("indirizzo");k36}~
B120: '{d}{poni @cursore("indirizzo");k37}~
B121: '{d}{poni @cursore("indirizzo");k38}~
B122: '{d}{poni @cursore("indirizzo");k40}~
B123: '{d}{poni @cursore("indirizzo");k41}~
B124: '{d}{poni @cursore("indirizzo");k42}~
B125: '{d}{poni @cursore("indirizzo");k43}~
B126: '{d}{poni @cursore("indirizzo");k45}~
B127: '{d}{poni @cursore("indirizzo");k46}~
B128: '{d}{poni @cursore("indirizzo");k47}~
B129: '{d}{poni @cursore("indirizzo");k48}~
B130: '{d}{poni @cursore("indirizzo");k49}~
B131: '{d}{poni @cursore("indirizzo");k50}~
B132: '{d}{poni @cursore("indirizzo");k51}~
B133: '{d}{poni @cursore("indirizzo");k52}~
B134: '{d}{poni @cursore("indirizzo");k53}~
B135: '{d}{poni @cursore("indirizzo");k55}~
B136: '{d}{poni @cursore("indirizzo");k56}~
B137: '{d}{poni @cursore("indirizzo");k57}~
B138: '{d}{poni @cursore("indirizzo");k58}~
B139: '{d}{poni @cursore("indirizzo");k60}~
B140: '{d}{poni @cursore("indirizzo");k61}~
B141: '{d}{poni @cursore("indirizzo");k62}~
B142: '{d}{poni @cursore("indirizzo");k63}~
B143: '{d}{poni @cursore("indirizzo");k64}~
B144: '{d}{poni @cursore("indirizzo");k65}~
B145: '{d}{poni @cursore("indirizzo");k66}~
B146: '{d}{poni @cursore("indirizzo");k68}~
B147: '{d}{poni @cursore("indirizzo");k69}~
B148: '{d}{poni @cursore("indirizzo");k70}~
B149: '{d}{poni @cursore("indirizzo");k71}~
B150: '{d}{poni @cursore("indirizzo");k72}~
B151: '{d}{poni @cursore("indirizzo");k73}~
B152: '{d}{poni @cursore("indirizzo");k74}~
B153: '{d}{poni @cursore("indirizzo");k75}~
B154: '{d}{poni @cursore("indirizzo");k76}~
B155: '{d}{poni @cursore("indirizzo");k77}~
B156: '{d}{poni @cursore("indirizzo");k78}~
B157: '{d}{poni @cursore("indirizzo");k79}~
B158: '{d}{poni @cursore("indirizzo");k80}~
B159: '{d}{poni @cursore("indirizzo");k82}~
B160: '{d}{poni @cursore("indirizzo");k83}~
B161: '{d}{poni @cursore("indirizzo");k84}~
B162: '{poni m85;l85}~

```

lutazioni — con un numero compreso tra 0 e 5 — delle caratteristiche tecniche dei differenti software (il programma consente di confrontare fino a 5 software differenti). Il comando «accettae» indica che viene inserito un dato di testo (questo comando è utilizzato anche per intercalare ai comandi per l'inserimento dei dati, righe con una funzione discorsiva ed esplicativa). Il comando «accettan» indica che deve essere inserito un dato numerico. La macro viene avviata posizionando il cursore sopra la casella B1 e attivando il comandi di avviamento nel menu a tendina (Varie/Macro/Esegui in Quattro Pro; Foglio/Macro/Esegui in Lotus 1-2-3). All'avviamento della macro compariranno in sequenza le diciture racchiuse tra virgolette e il foglio si troverà predisposto per acquisire il testo o il numero che verranno digitati. Nel

corso dell'esecuzione di questa parte della macro, i dati saranno registrati nelle caselle del foglio di calcolo indicate nella parte terminale dei comandi, dopo il punto e virgola (K1, K2, K3, ecc.)

Prima di avviare la macro, salvate il foglio su cui avete lavorato in modo da poter sempre recuperare una versione "pulita" della macro cioè una versione scritta su un foglio completamente privo di dati. Al termine della prima parte della macro troviamo una serie di comandi, riportati nella Tabella 2, che hanno lo scopo di scrivere su una riga del foglio, a partire dalla posizione B240, i dati dei vari software. Più precisamente: i dati del primo software verranno scritti su una riga orizzontale a partire dalla posizione B240; i dati del secondo software verranno scritti su una riga a partire dalla posizione B241; così via fino ai dati del

quinto software che verranno scritti su una riga a partire dalla posizione B244.

La macro, nel caso della Tabella 2, sposta il cursore e scrive i dati così come faremmo noi se dovessimo eseguire questa operazione manualmente. Il comando B85 posiziona il cursore del foglio di calcolo sulla casella B240. Il comando B86 incrementa di 1 il valore della casella L85 (questo valore serve a "contare" quanti software sono stati esaminati). Il comando B87 trasferisce l'esecuzione alla casella B167 nel caso siano stati già esaminati 5 software. I comandi B88..B91 trasferiscono in basso di 1, 2, 3, 4 caselle il cursore a seconda che i dati si riferiscano al secondo, al terzo software, ecc. Il comando B92 scrive nella casella in cui si trova il cursore il dato contenuto nella casella K1. I comandi dalla riga B93 in poi ➤

spostano via via il cursore a destra di una casella (mediante il comando {d}) e scrivono i dati contenuti nelle caselle K2, K3, ecc. Al termine il comando B162 memorizza nella casella M85 il contenuto della casella L85.

Dopo questa operazione di riscrittura dei dati, la sequenza di comandi della Tabella 1 può essere utilizzata nuovamente per acquisire i dati di un ulteriore software. Le caselle K1, K2, ecc. possono essere utilizzate per registrarvi, momentaneamente, i dati di un altro software. Nella Tabella 3 compaiono i comandi che servono a trasferire nuovamente l'esecuzione della macro alla riga B1. Il comando B163 fa comparire una riga interrogativa nella quale si chiede se vi è un altro software

dati di un software ma per assegnare i "pesi" alle differenti voci di valutazione (sempre con un valore compreso tra 0 e 5). Assegneremo un valore basso alle voci meno importanti e un valore alto a quelle che devono "pesare" di più nella valutazione dei software. Terminata la sequenza di comandi della Tabella 1, la macro prosegue con i comandi della Tabella 2. Viene eseguito il comando B85 normalmente e viene eseguito anche il comando B86 (a questo punto il valore contenuto nella casella L85 è pari a 11). Ma quando la macro esegue il comando B87, dato che il contenuto di L85 è superiore a 5 (di fatto è 11) l'esecuzione viene trasferita alla casella B167.

A partire dalla casella B167 entrano in opera i comandi della Ta-

sieme dei numeri registrati dalla posizione J240 alla posizione BS244 (cioè i dati numerici relativi ai software che vengono confrontati) costituiscono una matrice di numeri composta da 5 righe e 62 colonne (nel caso che i software siano meno di 5 le ultime righe saranno vuote). Invece l'insieme di numeri registrato dalla posizione A250 alla posizione A311 costituisce una matrice di 1 colonna e 62 righe (altrimenti detta vettore colonna).

Matrici e vettori possono essere moltiplicati tra loro. Come, naturalmente, è una questione un po' lunga da spiegare. Ciò che ci interessa evidenziare, in questa sede, è che se moltiplichiamo una matrice di n righe e m colonne per un vettore di m righe e 1 colonna il risultato sarà un vettore di n righe e 1 colonna. Il primo di questi numeri sarà costituito dalla somma del numero della prima colonna della prima riga della matrice, moltiplicato per il numero della prima riga del vettore *più* il numero della seconda colonna della prima riga della matrice, moltiplicato per il numero della seconda riga del vettore, e così via. Il secondo risultato sarà costituito dalla somma del numero della prima colonna della seconda riga della matrice, moltiplicato per il numero della prima riga del vettore *più* il numero della seconda colonna (sempre della seconda riga della matrice), moltiplicato per il numero della seconda riga del vettore, più ecc. ecc.

In pratica, data la matrice [A] di 2 righe e 3 colonne:

2 4 7

3 2 5

e il vettore [B] di 3 righe e 1 colonna:

2

3

1

Il risultato della moltiplicazione di [A] per [B] sarà il vettore [C] di 2

Tab. 3

```
B163: '{accettae "Un altro sw da valutare (s/n)? ";1163}~
B164: '{se 1163="s"}{salto b1}~
```

da valutare. Successivamente, se la risposta è stata affermativa, il comando B164 riconduce l'esecuzione della macro alla casella B1. Se invece si risponde negativamente alla richiesta del comando

bella 5. Questi comandi operano in maniera analoga ai comandi della tabella 2 con la differenza però che i dati contenuti nelle caselle K11, K13, K14, ecc. non vengono copiati lungo una riga orizz-

Tab. 4

```
B165: '{accettae "Assegnare i pesi; (enter)";1169}~
B166: '{poni 185;10}{salto b11}~
```

B163, l'esecuzione della macro prosegue normalmente (oltrepassando la casella B164 e la B165 che ha solo valore esplicativo) fino alla casella B166 (nella Tabella 4). Il comando di questa casella effettua due azioni: pone il valore 10 nella casella L85 e trasferisce l'esecuzione della macro alla casella B11.

Da questo momento la sequenza dei comandi della Tabella 1 viene ripercorsa — a partire dalla casella B11 — non più per inserire i

zontale, ma in verticale. Il cursore cioè, comando dopo comando, viene spostato in basso di una casella, a partire dalla posizione A250, mediante il comando {b}.

A questo punto, terminata la registrazione dei dati e dei pesi, possiamo utilizzare una delle funzionalità più avanzate disponibili sugli odierni fogli di calcolo: la moltiplicazione tra matrici. Lo spazio naturalmente non è sufficiente per illustrare le basi del calcolo matriciale. Possiamo dire però che l'in-

righe e 1 colonna i cui elementi saranno pari a:

$$23 = 2*2 + 4*3 + 7*1$$

$$17 = 3*2 + 2*3 + 5*1$$

Se osserviamo ora la zona del foglio di calcolo dove sono stati registrati i dati relativi ai software da valutare, potremo constatare che la zona compresa tra la posizione J240 e la posizione BS244 costituisce una matrice di numeri di 5 righe e 62 colonne (eventualmente, se i software sono meno di 5, le ultime righe saranno vuote). Invece la zona del foglio di calcolo compresa tra la posizione A250 e la posizione A311, dove sono registrati i "pesi" dei criteri di valutazione dei software, costituisce un vettore di 62 righe e 1 colonna. Moltiplicando la matrice per il vettore otterremo allora una colonnina di 5 numeri ognuno dei quali rappresenta la valutazione complessiva e "pesata" di uno dei vari software.

Ottenere questo risultato, con la macro, è abbastanza semplice. Nella Tabella 6, alla riga B230, viene attivato il comando previsto dal foglio di calcolo per la moltiplicazione tra matrici. Alle righe B231..B234 viene individuata la dimensione in righe della matrice di partenza a seconda che i software da valutare siano 2, 3, 4 o 5. Alla riga B236 viene individuato il vettore per il quale deve essere moltiplicata la matrice (il vettore è quello contenuto dalla posizione A250 alla posizione A311). Alla riga B238 viene individuata la casella a partire dalla quale devono essere emessi i risultati (casella A240).

Questa è la casella immediatamente a sinistra della riga contenente i dati dei singoli software. Scegliendo questa posizione per l'emissione dei dati otteniamo che, al termine dell'esecuzione della macro, avremo, a partire dalla posizione A240 fino alla posizione BS244, i dati dei singoli

Tab. 5

B167:	'	{	posiziona	}	a250~	
B168:	'	poni	@cursore	("	indirizzo	")
B169:	'	b	{	poni	@cursore	("
B170:	'	b	{	poni	@cursore	("
B171:	'	b	{	poni	@cursore	("
B172:	'	b	{	poni	@cursore	("
B173:	'	b	{	poni	@cursore	("
B174:	'	b	{	poni	@cursore	("
B175:	'	b	{	poni	@cursore	("
B176:	'	b	{	poni	@cursore	("
B177:	'	b	{	poni	@cursore	("
B178:	'	b	{	poni	@cursore	("
B179:	'	b	{	poni	@cursore	("
B180:	'	b	{	poni	@cursore	("
B181:	'	b	{	poni	@cursore	("
B182:	'	b	{	poni	@cursore	("
B183:	'	b	{	poni	@cursore	("
B184:	'	b	{	poni	@cursore	("
B185:	'	b	{	poni	@cursore	("
B186:	'	b	{	poni	@cursore	("
B187:	'	b	{	poni	@cursore	("
B188:	'	b	{	poni	@cursore	("
B189:	'	b	{	poni	@cursore	("
B190:	'	b	{	poni	@cursore	("
B191:	'	b	{	poni	@cursore	("
B192:	'	b	{	poni	@cursore	("
B193:	'	b	{	poni	@cursore	("
B194:	'	b	{	poni	@cursore	("
B195:	'	b	{	poni	@cursore	("
B196:	'	b	{	poni	@cursore	("
B197:	'	b	{	poni	@cursore	("
B198:	'	b	{	poni	@cursore	("
B199:	'	b	{	poni	@cursore	("
B200:	'	b	{	poni	@cursore	("
B201:	'	b	{	poni	@cursore	("
B202:	'	b	{	poni	@cursore	("
B203:	'	b	{	poni	@cursore	("
B204:	'	b	{	poni	@cursore	("
B205:	'	b	{	poni	@cursore	("
B206:	'	b	{	poni	@cursore	("
B207:	'	b	{	poni	@cursore	("
B208:	'	b	{	poni	@cursore	("
B209:	'	b	{	poni	@cursore	("
B210:	'	b	{	poni	@cursore	("
B211:	'	b	{	poni	@cursore	("
B212:	'	b	{	poni	@cursore	("
B213:	'	b	{	poni	@cursore	("
B214:	'	b	{	poni	@cursore	("
B215:	'	b	{	poni	@cursore	("
B216:	'	b	{	poni	@cursore	("
B217:	'	b	{	poni	@cursore	("
B218:	'	b	{	poni	@cursore	("
B219:	'	b	{	poni	@cursore	("
B220:	'	b	{	poni	@cursore	("
B221:	'	b	{	poni	@cursore	("
B222:	'	b	{	poni	@cursore	("
B223:	'	b	{	poni	@cursore	("
B224:	'	b	{	poni	@cursore	("
B225:	'	b	{	poni	@cursore	("
B226:	'	b	{	poni	@cursore	("
B227:	'	b	{	poni	@cursore	("
B228:	'	b	{	poni	@cursore	("
B229:	'	b	{	poni	@cursore	("

software corredati dalla loro valutazione complessiva "pesata".

Il valore più alto tra questi individuerà il software "migliore". Software però che risulterà essere il migliore non dal punto di vista di una funzione assoluta di rendimento, ma in base ad una analisi

multicriteri nella quale il confronto tra i diversi software avviene sulla base di valutazioni allo stesso tempo quantitative e qualitative.

Memorizzando il foglio di calcolo con un nome diverso da quello utilizzato per memorizzare la ➤

Tab. 6

```

B230: '{ / operazioni; moltiplicamatrice}
B231: '{ se m85=2 } j240..bs241
B232: '{ se m85=3 } j240..bs242
B233: '{ se m85=4 } j240..bs243
B234: '{ se m85=5 } j240..bs244
B235: '~
B236: '{ a250..a311
B237: '~
B238: '{ a240
B239: '~

```

macro, potremo riutilizzare in un secondo tempo tanto i dati che i risultati. Come si vede il foglio di calcolo ci permette di eseguire operazioni molto complesse, come la moltiplicazione di matrici, con pochi semplici comandi. Ma per sfruttare pienamente le possibilità offerte da un foglio di calcolo, cioè per creare delle macro, occorre avere una qualche consuetudine con la logica della programmazione software.

L'applicazione dei metodi e dei procedimenti della ricerca operativa, sia che avvenga tramite un linguaggio procedurale come il Basic, sia che avvenga tramite un foglio di calcolo, non può prescindere, oltre che dalla conoscenza di determinate nozioni di matematica e statistica, anche da una certa padronanza della logica della programmazione software. Sarebbe auspicabile quindi che alcuni dei procedimenti della ricerca operativa, insieme alle nozioni matematiche e statistiche necessarie e insieme ad una certa padronanza della programmazione, rappresentassero una parte di una certa consistenza nel bagaglio professionale di tutti quei bibliotecari interessati a specializzarsi professionalmente nell'analisi sistemica della biblioteca e dei sistemi di biblioteche. ■

Note

¹ M. SEBASTIANI, *Dall'analisi del sistema alla biblioteca efficace*, "Biblioteche oggi", 11 (1993), 10.

² M. SEBASTIANI, *L'analisi dei dati in biblioteca*, "Biblioteche oggi", 12 (1994), 2.

³ M. SEBASTIANI, *Decidere con l'ottimizzazione matematica*, "Biblioteche oggi", 12 (1994), 4.

⁴ M. SEBASTIANI, *Analizzare il prestito con la teoria delle code*, "Biblioteche Oggi", 12 (1994), 10.

⁵ H. SIMON, *Il comportamento amministrativo*, Il Mulino 1967.

⁶ H. SIMON, *cit.*, p. 264.

⁷ H. SIMON, *La ragione nelle vicende umane*, Il Mulino 1984, p. 43.

⁸ H. SIMON, *cit.*, p. 43.

⁹ H. SIMON, *cit.*, p. 50-51.

¹⁰ A. LEGGIO, *Information technology e*

management. Indicazioni tecnico gestionali ad uso delle pubbliche amministrazioni, Milano, Il Sole 24 ore libri, 1993, p. 255-259. Anche in questo testo viene messa in evidenza la difficoltà di applicare l'analisi costi-benefici nei sistemi pubblici, là dove si afferma: "È evidente che, nel caso della pubblica amministrazione, il calcolo dei benefici non può essere espresso in termini di profitto, ma bensì in funzione dell'incremento del livello di servizio offerto al cittadino e in funzione degli eventuali risparmi in termini di personale" (p. 259).

¹¹ Lo spunto per questo programma è tratto da ROBERT MACHALOW, *Using Lotus 1-2-3. A how-to-do-it manual for library applications*, New York - London, Neal Schuman Publishers, 1989. L'elenco delle voci per l'analisi di software per l'insegnamento di cui alla Tabella 1, costituisce una traduzione pressoché integrale del programma riportato alle p. 131-134 di questo testo. È originale invece la modalità di elaborazione dei dati secondo quanto previsto nelle Tabelle 2, e successive.



G. ORECCHIA