

Mario Alai
Università di Urbino “Carlo Bo”
mario.alai@libero.it

INFORMATICA E DIDATTICA DELLA FILOSOFIA

SOMMARIO

1. INTRODUZIONE	3
2. ESISTE UNA LOGICA DELLA SCOPERTA SCIENTIFICA?	4
3. È POSSIBILE LA FORMAZIONE DEI CONCETTI PER ASTRAZIONE?	6

ABSTRACT

Computer programs may be used to model human intelligent practices in flexible and detailed ways. This is one of the reasons why computing can be very useful to philosophers both in research and in teaching.

Two examples are suggested here, one drawn from philosophy of science and the other from epistemology. The former concerns the debate on the 'logic of discovery': after a short review of the basic terms and present state of the question, instances are given of how this topic might be taught by simultaneously introducing the students to conceptual thinking, history of science, history of Artificial Intelligence and programming languages.

The latter example concerns the classical problem of innate ideas: the application of computing to this question is mainly discussed theoretically, while only hints are given to possible classroom presentations

1. Introduzione

I possibili utilizzi didattici dei *prodotti* dell'informatica sono sotto gli occhi di tutti: ipertestualità, banche dati, reti informatiche, posta elettronica, ecc. Quali contributi può offrire invece all'insegnamento delle altre discipline la conoscenza e la pratica *dell'informatica stessa*? Poiché lo specifico dell'informatica consiste nel formalizzare l'informazione per permetterne, secondo l'etimologia del termine, il trattamento automatizzato, le sue applicazioni più naturali e immediate si trovano in quelle aree del sapere che sono già di per sé stesse formalizzate: le scienze logico-matematiche e le scienze naturali, a loro volta più o meno pervasivamente matematizzate. L'utilizzo di programmi informatici per la soluzione di ogni genere di problemi, infatti, sta diventando parte integrante dell'insegnamento di queste discipline non solo nell'università e nella scuola media superiore, ma anche nei precedenti ordini di scuole. Ma proprio perché in questi campi il passo richiesto per l'informatizzazione è così breve, anche i risultati sono in un certo senso meno innovativi e significativi. Invece, la sfida più interessante, quella che esalta maggiormente il potenziale di rinnovamento dell'informatica, sta forse nella sua applicazione alle discipline umanistiche.

Un primo possibile obiettivo, in questo senso, è la filosofia: in quanto si caratterizza per un metodo e dei contenuti eminentemente razionali, anche se non puramente formali e non esplicitamente formalizzati, essa può costituire un buon punto d'equilibrio tra la ricerca di terreni nuovi e inesplorati e l'esigenza di non partire subito dai più difficili.

La filosofia, del resto, può fare buon uso di una tipica prerogativa dell'informatica, quella di prestarsi alla costruzione di modelli flessibili a piacere e in un certo senso molto concreti. È addirittura proverbiale che la maggior difficoltà nell'apprendimento della filosofia viene dalla sua natura astratta; e se il problema in quanto tale non si può eliminare, se ne può facilitare di molto il superamento per mezzo di esempi, vale a dire di modelli concreti: infatti, come ben sanno gli insegnanti, una volta compreso un problema nell'applicazione particolare, risulta assai più facile afferrarlo anche nella sua dimensione generale e astratta. E questo non vale solo per la didattica, ma anche per la ricerca, se come ha detto Wittgenstein una malattia filosofica è generalmente indice di una dieta povera di esempi. Ora, non solo l'informatica può in generale fornire dei modelli espliciti ed esatti di moltissimi tipi di fenomeni; ma una sua branca, l'intelligenza artificiale, si dedica precisamente alla ricostruzione delle attività intelligenti dell'uomo, che costituiscono il cuore degli interessi filosofici: la conoscenza (sia comune che scientifica, nei suoi aspetti e modalità), la razionalità pratica, l'etica, la politica, tutte sono o possono diventare oggetto di studio dell'intelligenza artificiale¹.

E se il miglior apprendimento è quello che si ottiene *facendo*, o come insegna Vico *verum et factum convertuntur*, dovrebbe essere un buon investimento quello di familiarizzare docenti e studenti con i concetti, i linguaggi e le tecniche della programmazione, fino a permetter loro di cimentarsi, almeno a livello elementare, nella ricostruzione di processi cognitivi di particolare interesse. In questa sede desidero considerare come possibili esempi due classiche problematiche dell'epistemologia e della gnoseologia: quella della logica della scoperta e quella dei concetti innati.

¹ Devo alcune delle idee fin qui espresse, così come il principale impulso alla stesura di questo contributo, al prof. Giorgio Casadei, del Dipartimento di Scienze dell'Informazione dell'Università di Bologna.

2. Esiste una logica della scoperta scientifica?

Se la grandezza delle scoperte di Galileo, Newton, o Einstein, non è solamente frutto d'intuito, chiarezza o fortuna, il processo della scoperta scientifica dev'essere *razionale*. Ma nessuno è riuscito finora ad esporre i principi o le regole di tale razionalità in modo da fornire a chi se ne voglia servire un metodo di sicuro successo nella soluzione dei misteri che la natura tuttora ci riserva. Ecco perché importanti epistemologi (i neopositivisti², Popper³, Kuhn⁴, e Feyerabend⁵) negano che esista una *logica della scoperta scientifica*, mentre altri, convinti che *razionalità* significhi anche un metodo codificabile, l'affermano (ad esempio, Bacone⁶, Mill⁷, Hanson⁸, e Lakatos⁹). Ora, è certo che se il processo della scoperta potesse venir ripercorso, simulato, o addirittura anticipato da un computer, si tratterebbe di un processo logico, dato che il computer è appunto una macchina logica. Inoltre l'analisi dei procedimenti meccanici permetterebbe di comprendere meglio la struttura di quelli umani. Di fatto, i programmi della serie BACON di Simon (da BACON₁ A BACON₅, ed in più GLAUBER, STAHL e DALTON) hanno riprodotto la scoperta di un gran numero di leggi famose (di accelerazione uniforme, dei gas ideali, terza legge di Keplero, legge di gravitazione universale, leggi di Coulomb, di Ohm, di Snell, di Black), e sono in grado di utilizzare i principi della chimica per riscoprire pesi atomici degli elementi, classi di elementi, strutture delle reazioni, struttura molecolare degli elementi¹⁰. Tuttavia non sono stati (e certamente non saranno) in grado di compiere scoperte nuove e applicabili a problemi pratici attuali, né di simulare la scoperta di veri e propri modelli teorici, come il modello eliocentrico del sistema solare, il modello atomico della materia, ecc. I sistemi esperti elaborati dagli studiosi che si ricollegano alla tradizione di Turing, come il programma ID3 di J. R. Quinlan, o il programma GOLEM, di S. Muggleton e C. Feng, producono risultati applicabili a problemi di ordine pratico, ed hanno anche scoperto nuove 'leggi naturali'¹¹. Tuttavia si tratta di generalizzazioni empiriche poco profonde e generali, alquanto lontane dal carattere teorico delle leggi che hanno segnato gli sviluppi scientifici più rilevanti¹². Promettono forse meglio approcci recenti basati su programmi connessionistici, come P.I. di Holland, Holyjoke, Nisbett e Thagard¹³, che dei ricercatori umani non imita specifiche strategie induttive, ma la flessibilità e la capacità di sfruttare il feed-back dell'ambiente per autodirigersi alla scoperta di soluzioni originali ai problemi. Tuttavia tali approcci si possono ancora considerare in fase iniziale. Questa problematica potrebbe venir impostata in chiave didattica in quattro momenti. Anzitutto, familiarizzando insegnanti e studenti col concetto di *macchina di Turing*. Come è noto, una macchina di Turing è un dispositivo ideale composto da un nastro di

² Ad esempio Hempel in Hempel (1968).

³ Popper (1970). L'apparente paradosso di un'opera intitolata *Logica della scoperta scientifica* in cui si nega l'esistenza di una tale logica è più che altro conseguenza della fuorviante traduzione italiana: il titolo originale, *Logik der Forschung*, parla solo di una logica della ricerca.

⁴ Kuhn (1978).

⁵ Feyerabend (1979).

⁶ Bacone (1942).

⁷ Mill (1988).

⁸ Hanson (1978).

⁹ Lakatos (1976).

¹⁰ Vedi Langley et al. (1987).

¹¹ Vedi Gillies (1998), cap. 2.

¹² Su questi problemi vedi anche Alai (2000).

¹³ In Holland et al. (1986).

infinita lunghezza suddiviso in caselle e da una testina-cursore che scorre su di esso. La testina può (a) assumere un numero finito di stati interni, (b) leggere i caratteri iscritti nella casella su cui attualmente si trova, e a seconda di quel che legge e dello stato in cui si trova (c) cambiare o meno il carattere iscritto, (d) spostarsi sulla casella di destra o di sinistra, ed (e) cambiare stato interno. In una versione appena un po' più complessa, (f) gli stati interni dipendono anche da *inputs* esterni e (g) possono produrre *outputs* verso l'esterno. La generalità di questo concetto è tale che, in primo luogo, qualsiasi programma informatico si può considerare come una macchina di Turing; secondo, un computer non è altro che un dispositivo in grado di realizzare ogni possibile macchina di Turing; e infine, si può motivatamente avanzare l'ipotesi che anche la mente umana sia precisamente un dispositivo di tal genere. Quest'ultima ipotesi, tra l'altro, costituisce già in sé stessa un importante banco di prova per gran parte delle discussioni filosofiche sulla natura della mente e sui suoi rapporti col corpo. Credo che capire il funzionamento della macchina di Turing, un compito non difficilissimo, consenta però già di farsi un'idea esatta di cosa sia e come funzioni un programma. È anche possibile vedere una macchina di Turing all'opera nella soluzione di problemi elementari, come il calcolo di semplici funzioni matematiche.

Il secondo momento didattico potrebbe consistere nell'apprendimento dei rudimenti di un linguaggio di programmazione, come ad esempio il Prolog¹⁴. È questo un linguaggio costruito proprio in vista delle ricerche di intelligenza artificiale; inoltre possiede strutture inferenziali vicine a quelle del ragionamento comune e alla loro codificazione nella logica tradizionale, che gli studenti hanno pure occasione di conoscere nello studio della filosofia. Apprese le basi del Prolog, lo si può vedere applicato in semplici programmi costruiti per risolvere piccoli problemi matematici o d'ordine pratico (compilare una lista, consultare una tabella, ecc.). Con sufficiente tempo a disposizione, e se opportunamente seguiti, gli studenti possono anche costruire da sé alcuni di questi elementari programmi, e costruendoli comprendere come un programma risolve i problemi, e in che senso un procedimento razionale è un procedimento esplicitabile e codificabile¹⁵.

In un terzo momento si potrebbe iniziare a studiare l'informatica in quanto applicata alla scoperta scientifica. Per la sua semplicità si presta assai bene il programma COPYCAT di Hofstadter e del suo gruppo¹⁶, pensato per rispondere a domande come: «Si supponga che la stringa *aabc* venga modificata in *aabd*: che cosa diventerebbe *ijkk* se la si trasformasse nello "stesso modo"?» Nonostante ogni apparenza, si tratta di questioni tutt'altro che banali, affini al problema wittgensteiniano del 'seguire una regola'¹⁷, e vertenti sulle strutture più fondamentali del pensiero analogico e dei processi induttivi. L'artificialità e la semplicità di questo genere di problemi sono dunque le stesse delle simulazioni di laboratorio, utili a mettere a fuoco con precisione problemi di ben maggiore portata e complessità.

Col quarto momento, infine, si esplorerebbero le potenzialità dei programmi informatici rispetto ai casi reali di scoperta scientifica. Si potrebbero scegliere come esempi tre

¹⁴ Utilizzando Bratko (1987) o altri manuali pure disponibili.

¹⁵ Fin qui il programma che stiamo delineando è stato realizzato, con apprezzabili risultati, in una sperimentazione congiunta cui partecipavano il Liceo Scientifico 'Curbastro' di Lugo, il Liceo Scientifico 'Righi' di Cesena, l'I.T.C. 'Ginanni' di Ravenna, la ditta EQUITEC di Lugo e il Dipartimento di Scienze dell'Informazione dell'Università di Bologna. Vedi *Intelligenza Artificiale* (2001).

¹⁶ Vedi Hofstadter (1996).

¹⁷ Vedi Wittgenstein (1967), §§ 143-242.

scoperte, una di quelle riprodotte dai programmi della serie BACON (la terza legge di Keplero, o la legge dei gas ideali, o l'analisi delle strutture molecolari), una di quelle ottenute coi sistemi esperti della tradizione di Turing (p. es. la struttura secondaria delle proteine), ed un caso rappresentativo tra quelli che né gli uni né gli altri sono in grado di trattare. Questi casi andrebbero anzitutto studiati storicamente (tramite alcuni degli accurati *case studies* offerti dalla letteratura) per rendersi conto della complessità e della raffinatezza dei percorsi razionali che conducono alla scoperta. Dopo di ciò, si analizzerebbe il funzionamento dei programmi rispetto a ciascun caso: per i programmi di Simon bisognerebbe chiedersi che dati, conoscenze di sfondo, ipotesi, principi metodologici, ecc. essi presuppongano, e se tra questi ve ne siano che non fossero precedentemente noti allo scopritore originale; in tal caso, si dovrebbe anzitutto analizzare le diverse intuizioni, valutazioni, scelte, ecc., che hanno condotto lo scopritore al successo, e in cui si esplica l'intrinseca razionalità della scoperta¹⁸, e poi cercar di capire quali aspetti di tale razionalità non siano stati adeguatamente formalizzati nel programma, quali potrebbero esserlo migliorandone le strategie di *problem solving*, le euristiche e le funzioni di valutazione, e quali eventualmente non potrebbero esser formalizzati in alcun modo. Sia per i programmi di Simon che per quelli della tradizione di Turing bisognerebbe poi chiedersi quali caratteristiche del terzo esemplare di scoperta ne precludano loro l'affronto, e se tali problemi non potrebbero, in prospettiva, venir superati da programmi diversi, del tipo del sopra menzionato P.I. È chiaro comunque che far tutto questo significherebbe già muoversi sul fronte avanzato della ricerca, in un'area di confine tra intelligenza artificiale ed epistemologia; in sede didattica, anche qualche piccolo passo compiuto in questa direzione costituirebbe già un buon successo¹⁹.

3. È possibile la formazione dei concetti per astrazione?

Un altro esempio delle potenzialità dell'informatica nella didattica della filosofia ci viene da una delle più classiche controversie gnoseologiche, quella sulle idee innate, o sull'origine delle idee²⁰. Si tratta in altri termini del contrasto tra razionalismo ed empirismo, che vede schierati su un versante Platone, Leibniz, Kant, Whewell, Chomsky, e sull'altro Aristotele, i filosofi stoici, Locke, Mill, gli empiristi logici e Quine, per non citarne che alcuni dei principali protagonisti.

I razionalisti sostengono che se almeno alcuni concetti generali non fossero innati in noi, non saremmo mai in grado di ricavarli dall'esperienza, e senza di essi è pacifico che nessuna conoscenza sarebbe possibile. Si tratta anzitutto di concetti di tipo logico-matematico, ma per alcuni anche di concetti morali, e secondo i più radicali anche di concetti empirici, come quelli di bianco, di letto, di simile, ecc. Secondo Platone, ad esempio, riconoscere che qualcosa è un letto significa confrontare quell'oggetto con il concetto (che dev'essere in me innato) di letto, e trovare che gli somiglia; riconoscere che due oggetti sono simili significa confrontare la coppia da essi formata con il

¹⁸ Vedi su questi problemi Shapere (1984), pp.219, 265 e passim; Antiseri (1996), pp.266-271; Parrini (1995), pp.205-209; Pera (1982), pp. 27-29.

¹⁹ Il percorso di questo quarto momento è stato in parte seguito dallo scrivente nei corsi di epistemologia tenuti presso l'Università di Urbino negli anni accademici 1999-2000 e 2000-2001.

²⁰ Alai (1992).

concetto di similitudine (anch'esso innato) e rendersi conto che le rassomiglia, cioè che ne costituisce un'istanza. Non posso dunque formarmi il concetto di bianco, di letto o di similitudine semplicemente osservando oggetti o coppie di oggetti e notando che essi posseggono certe caratteristiche, poiché solo confrontandoli coi corrispondenti concetti innati riesco ad accorgermi che hanno quelle caratteristiche. Se quei concetti non fossero innati in me potrei anche osservare migliaia di oggetti bianchi, di letti o di coppie simili, senza accorgermi che essi sono tali²¹.

Secondo Mill, al contrario,

lo stesso atto di confronto che culmina nella connessione, da parte nostra, dei fatti per mezzo del concetto, può esser la fonte da cui deriviamo il concetto stesso. Se non avessimo mai visto oggetti bianchi o non avessimo mai visto un animale con lo zoccolo fesso, acquisteremmo l'idea nello stesso istante e con lo stesso atto mentale con cui l'impieghiamo per collegare i fenomeni osservati²²,

e ciò grazie alla capacità di astrazione.

Se si potesse costruire un programma capace non solo di classificare degli esemplari sotto un concetto o classe, ma di costruirsi da sé il concetto o la classe partendo dai particolari, si dimostrerebbe che Mill ha ragione e Platone ha torto. Effettivamente, esistono programmi di apprendimento induttivo dei concetti²³: essi utilizzano come dati degli insiemi di oggetti indicati come esempi positivi del concetto, ed analizzandone le proprietà, costruiscono il concetto inteso come specificazione delle proprietà necessarie e/o sufficienti. In altri termini, partono dall'*estensione* del concetto, per costruirne l'*intensione*. Ad esempio, data una serie di mattinate di cui si sa (a) se sono adatte all'arrampicata oppure no, e (b) se sono fresche, ventose, nuvolose, piovose, oppure no, si ricava il concetto di *mattina adatta all'arrampicata* (che potrebbe ad es. risultare quello di una mattina *non piovosa, non fredda o non ventosa, nuvolosa o fresca*)²⁴. Tali programmi vengono classificati come *top-down* (o *model-driven*) se partono dalla più ampia generalizzazione compatibile coi primi esempi considerati, per poi restringerla mano a mano a fronte degli esempi successivi, e *bottom-up* (o *data-driven*) se seguono la direzione opposta. In ogni caso, tuttavia, essi presentano due limiti, che li rendono inadatti a risolvere il problema che ci siamo proposti: in primo luogo, non *costruiscono* il concetto, che, come si è visto, viene loro fornito nella sua estensione, cioè tramite gli esempi positivi: essi non risolvono il problema di come riconoscere se un caso particolare sia sussumibile o no al concetto (nell'esempio, se una particolare mattina sia adatta all'arrampicata o meno), che è il problema sollevato da Platone: la soluzione a questo problema viene loro fornita come dato. Essi partono dunque dalla conoscenza dell'estensione per giungere all'intensione (l'insieme delle proprietà che definiscono il concetto). In secondo luogo, anche per ricavare l'intensione, tali programmi devono *possedere già* una serie di concetti, quelli delle proprietà elementari (fresca, piovosa, ventosa ...) costitutive della proprietà complessa (adatta all'arrampicata); pertanto il problema si porrebbe nuovamente per l'acquisizione di tali proprietà. Di conseguenza, piuttosto che smentire l'argomento di Platone sulla necessità di concetti innati, essi sembrerebbero semmai confermarlo.

²¹ Vedi Platone (1971), 73c-77a.

²² Mill (1988), vol. II, p. 869.

²³ Come ID3 di Quinlan, gli algoritmi di Winston, di Michalsky, di Mitchell, ecc.: vedi Gillies (1996), pp.39. sgg., e Scaruffi (1991), pp.287-288.

²⁴ Quinlan (1986), pp.86-87.

Che significa affermare, come Mill, che la mente è in grado di ricavare il concetto di una proprietà come la bianchezza direttamente dal confronto tra alcuni oggetti, senza possederlo o possederne altri in precedenza? Come minimo, significherà che essa è in grado di distinguere, registrare e conservare la memoria di tale proprietà quando la percepisce, e in seguito, quando la percepisca di nuovo, distinguerla e riconoscerla come *la stessa proprietà*. E in che consiste il riconoscere una proprietà come la stessa percepita in precedenza? La questione è complessa, ma la contemporanea filosofia della mente, a partire da Wittgenstein²⁵, ha mostrato che sebbene tale riconoscimento possa essere contrassegnato da particolari esperienze o sensazioni psichiche (ad es., una sensazione di *déjà vu*, o una sensazione di familiarità, di soddisfazione, ecc.), tali sensazioni non sono necessarie né sufficienti a garantire il riconoscimento. Per poter parlare di riconoscimento, come minimo, bisognerà che il soggetto dimostri di reagire alla proprietà data nello stesso modo ogni volta che la incontra (almeno a parità di condizioni), o con le stesse reazioni esteriori, o associandola ad altre proprietà, ma ogni volta le stesse. Supponiamo ad esempio che un soggetto dotato di elementari facoltà mentali provi dolore in una serie di circostanze in cui percepisce anche il colore rosso (ad esempio in presenza del fuoco, di una ferita sanguinante, ecc.). Il fatto che di fronte a una successiva presentazione del rosso (magari di un papavero) esso reagisca con disagio, timore, impulso a fuggire, ecc., dimostra che ha riconosciuto la sensazione presente come caratterizzata dalla medesima proprietà (il colore rosso) che aveva percepito nelle precedenti occasioni dolorose.

Può una macchina far qualcosa del genere? Parrebbe di sì: immaginiamo una macchina che possa ricevere contemporaneamente due tipi di *input*: [P, D], e [R, V, G] (che rappresentano ciò che per noi sarebbero [piacevole, doloroso], e [rosso, verde, giallo]). Immaginiamo ancora che essa possa annotare nella memoria di lavoro un'iscrizione diversa per ciascuno di questi *inputs*: rispettivamente, *p*, *d*, e *r*, *v*, *g*, e che possa annotare almeno due iscrizioni contemporaneamente. Supponiamo infine che la macchina sia programmata così:

- a) Quando due iscrizioni compaiono insieme nella memoria di lavoro, esse vengono trascritte unitamente nella memoria permanente (oppure, che dopo esser comparse insieme per un certo numero di volte nella memoria di lavoro, vengono trascritte unitamente nella memoria permanente).
- b) Quando nella memoria di lavoro si trova iscritto *d* l'*output* è la fuga, quando vi si trova *p* è l'avvicinamento (questa regola rappresenta dunque una sorta di riflesso immediato, innato o condizionato, del soggetto).
- c) Quando nella memoria di lavoro compare un'iscrizione, ad es. *r*, la macchina controlla se essa compare anche nella memoria permanente, e se vi compare unitamente ad un'altra (ad es., *d*). Se questo è il caso, la macchina iscrive anche quest'ultima nella memoria di lavoro.

Come risultato, se la macchina si trova per un certo periodo di fronte a *inputs* 'dolorosi' e 'rossi' insieme, in seguito reagirà a nuovi *inputs* 'rossi' nello stesso modo, evocando la 'sensazione di dolore', e con l'*output* 'fuga'.

In conclusione, una tale macchina saprebbe riconoscere il rosso in quanto (a) annota la medesima iscrizione *r* ogni volta che riceve una 'sensazione' di rosso, e (b) lo dimostra anche esteriormente con la sua capacità a farsi 'condizionare' a reagire con la fuga ad ogni nuova 'sensazione' di rosso. Dal momento in cui *r* viene iscritta nella memoria

²⁵ Wittgenstein (1967).

permanente e può esser confrontata e trovata identica con un'iscrizione contenuta nella memoria di lavoro, la macchina ha acquistato un concetto che prima non possedeva, traendolo dagli stimoli dell'esperienza.

Si potrebbe obiettare che anche questa macchina incontra la stessa difficoltà che si è messa in luce per i sistemi di apprendimento induttivo: poiché dev'esser capace di scrivere *r* tutte le volte che riceve lo stesso stimolo, e poiché dev'esser capace di determinare se l'iscrizione presente in una memoria e quella presente nell'altra sono identiche o meno, non presuppone essa già la conoscenza di un concetto? E per esistere e funzionare, non necessita forse di venir programmata da qualcuno che possiede già il concetto in questione? Se è così, non può dirci nulla sulla possibilità della mente umana di costruire per astrazione il concetto di rosso, in quanto la mente umana non è stata programmata da nessuno.

Da un lato, tuttavia, si noti che quel che troviamo inizialmente nella macchina non è il *concetto* di rosso, ma semplicemente una *capacità*, quella di riconoscere le identità. Essa è di tipo generale, e permette l'apprendimento non solo del concetto di rosso, ma di qualunque altro. In secondo luogo, per costruire una macchina del genere non serve in realtà un programmatore cosciente, ma basterebbero dei meccanismi causali che di fatto la determinino ad agire secondo le regole sopra indicate, annotando cioè la medesima iscrizione quando posta di fronte alla medesima proprietà, ecc. Ora, dobbiamo presupporre che dei meccanismi del genere esistano, e sono precisamente quelli posti in atto dalla *selezione naturale*. Il più semplice prototipo della macchina descritta è l'organismo unicellulare che è sensibile a due soli *inputs*, presenza o assenza di corpuscoli nutrienti ai suoi bordi, e produce due sole reazioni, l'inglobare o il non inglobare i corpuscoli 'percepiti'. Quegli organismi che a contatto dei corpuscoli esterni avevano la reazione giusta (inglobarli se nutrienti, altrimenti no) e *sempre la stessa* sono sopravvissuti, trasmettendo tale capacità alla loro discendenza tramite il patrimonio genetico; gli altri, che non ne erano dotati, si sono semplicemente estinti. Forse, allora, l'essere umano è proprio un organismo di quel tipo, solo molto più complesso.

Tutto quel che dobbiamo presupporre, allora, è che esistano degli universali in natura, come la proprietà di esser rosso, nutriente, o di essere *simile* o *identico* (è questa la tesi dell'esistenza degli universali *in re*, la forma di realismo moderato sostenuta da Aristotele), e che la natura, nel suo operare causale (incluso il processo della selezione naturale), segua le linee di distinzione tracciate da questi universali. Con tali presupposti sembra possibile che una macchina, e quindi anche la mente, si formi dei concetti per astrazione. Questo confuterebbe dunque l'argomento platonico in favore della versione più forte di innatismo (l'esistenza di concetti innati di tipo empirico), anche se nulla si è detto fin qui sulla necessità di ammettere l'esistenza innata di concetti logici, matematici, morali, o di forme categoriali o di altro genere di strutture.

È ovvio che abbiamo lavorato con idee estremamente semplificate, e che sia la nozione di *concetto* che il progetto di una macchina capace di astrarre sono considerevolmente più complessi. Per fare un solo esempio, abbiamo considerato il rosso come una proprietà semplice sempre identica, mentre in realtà include un'intera gamma di gradazioni cromatiche. Alla nostra macchina bisognerebbe quindi aggiungere un meccanismo che le permetta di trattare allo stesso modo gli stimoli di tutta la gamma di gradazioni compresa tra certi limiti più o meno esattamente definiti. Ciò la complicherebbe alquanto, ma non credo richiederebbe né la conoscenza innata di concetti, né meccanismi di base che non si possano trovare in natura.

Per quanto riguarda la traduzione didattica di questa discussione, una volta acquisite le basi di un linguaggio di programmazione come indicato nel paragrafo precedente, gli studenti potrebbero, con la guida di un esperto, costruire un programma che operi come la macchina appena descritta. Alla luce del suo funzionamento, poi, potrebbero esser condotti a rileggere e discutere i passi più rilevanti dei filosofi menzionati all'inizio del paragrafo.

Bibliografia

Alai, M., 1992, «L'origine dei concetti: posizioni e problemi», in Pollini, F. e Tarozzi, G. (a cura di), *I concetti della fisica*, Accademia Nazionale di Scienze, Lettere e Arti, Modena, Mucchi Editore, pp. 217-244.

Alai, M., 2000, «Informatica e logica della scoperta», in Andronico, A., Casadei, G., Sacerdoti, G. (a cura di), *Didattica 2000. Informatica per la didattica*, Cesena, Il Ponte Vecchio, vol. 1, pp. 13-19.

Antiseri, D. 1996, *Trattato di metodologia delle scienze sociali*, Torino, UTET.

Bacone, F., 1942, *Novum Organum*, trad. it. di A. Saloni, Firenze, La Nuova Italia (1620).

Bratko, I., 1987, *Programmare in Prolog per l'intelligenza artificiale*, trad. it. di D. Fum, Milano, Masson (1986).

Feyerabend, P., 1979, *Contro il metodo*, trad. it. di L. Sosio, Milano, Feltrinelli (1975).

Gillies, D., 1998, *Intelligenza artificiale e metodo scientifico*, trad. it. di P. Poggi, Milano, Cortina (1996).

Hanson, N. R., 1978, *I modelli della scoperta scientifica. Ricerca sui fondamenti concettuali della scienza*, trad. it. di L. Sosio, Milano, Feltrinelli (1958).

Hempel, C. G., 1968, *Filosofia delle scienze naturali*, trad. it. di A. Berra, Bologna, il Mulino (1966).

Hofstadter, D. R., 1996, *Concetti fluidi e analogie creative*, trad. it. di M. Corbò, I. Gilberti, M. Codogno, Milano, Adelphi (1995).

Holland, J. et al., 1986, *Induction. Processes of Inference, Learning and Discovery*, Cambridge (Mass.), MIT Press.

Intelligenza Artificiale, 2001, ipertesto all'URL www.racine.ra.it/curba/set/privato, Progetto Speciale per l'Educazione Scientifica e Tecnologica del M.I.U.R.

Kuhn, T., 1978, *La struttura delle rivoluzioni scientifiche*, trad. it. di A. Carugo, Torino, Einaudi (1962).

Lakatos, I., 1976, «La falsificazione e la metodologia dei programmi di ricerca scientifici», in Lakatos, I. e Musgrave, A. (a cura di), *Critica e crescita della conoscenza*, trad. it. di G. Giorello, Milano, Feltrinelli (1970), pp. 164-276.

Langley, P. et al., 1987 *Scientific Discovery. Computational Explorations of the Creative Processes*, MIT Press, Cambridge (Mass.).

Mill, J. S., 1988, *Sistema di logica induttiva e deduttiva*, trad. it. a cura di M. Trincherò, Torino, UTET (1848).

Parrini, P., 1995, *Conoscenza e realtà. Saggio di filosofia positiva*, Bari, Laterza.

Pera, M., 1982, *Apologia del metodo*, Bari, Laterza.

Platone, 1971, *Fedone*, trad. it. di M. Valgimigli, in *Opere complete*, vol. 1, Bari, Laterza, pp. 99-189.

Popper, K. R., 1970, *Logica della scoperta scientifica*, trad. it di M. Trincherò, Einaudi, Torino (1934).

Quinlan, J. R., 1986, «Induction of Decision Trees», in *Machine Learning*, 1, pp.81-106.

Scaruffi, P., 1991, *La mente artificiale. Realtà e prospettive della 'macchina pensante'*, Milano, Franco Angeli.

Shapere, D., 1984, *Reason and the Search for Knowledge*, Dordrecht, Reidel.

Wittgenstein, L., 1967, *Ricerche filosofiche*, trad. it. di R. Piovesan e M. Trincherò, Torino, Einaudi (1953).