CONCORDJava

Controllo e Correzione Dati

Manuale utente

1. INSTALLAZIONE

Installazione in ambiente Windows

ConcordJava richiede l'installazione dell'ambiente JAVA (Java 2 Runtime Environment 6.0 o superiore).

Per installare il software si deve scaricare il file setup_ConcordJava.exe sul proprio PC ed eseguirlo. Non sono necessarie autorizzazioni (per esempio avere una utenza con permessi di amministratore). Se non è possibile scrivere sul disco C è sufficiente cambiare la cartella in cui installare il softwae.
L'applicazione richiama l'editor di Windows notepad.exe quando è necessario l'uso di un editor di testo (per esempio nella scrittura delle regole di Scia). Per modificare questa impostazione e utilizzare un altro editor si deve modificare il settaggio della variabile EDITOR nel file ConcordJava.ini che si trova nella cartella di ConcordJava. Per esempio:

```
EDITOR=C:\Programmi\Notepad++\notepad++.exe
```

E' anche necessario che la variabile d'ambiente PATH, che contiene il percorso di ricerca dei comandi, contenga il riferimento alla cartella Java.
Per reimpostare la variabile di ambiente PATH in ambiente Windows da:

Start -> Impostazione -> Pannello di controllo -> Sistema -> Avanzate -> Variabili d'ambiente

Qui reimpostare la variabile PATH aggiungendo il percorso della cartella che contiene il file java.exe. Per esempio:

```
PATH=C:\Programmi\Java\jre1.6.0_03\bin;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;.
```

2. DEFINIZIONE DEL PROGETTO

Per utilizzare CONCORDJAVA occorre definire un **progetto** caratterizzato da un nome e da una cartella di lavoro. Si deve inoltre scegliere l'approccio che si desidera utilizzare (probabilistico, deterministico, donatore).
Il nome del progetto è usato come tipo per i file necessari alla procedura. Per esempio:

- nome progetto: fdl,
- metodo: probabilistico,
- cartella di lavoro: G:\Indagini\fdl\prova.

In questo caso, tutti i files utilizzati da SCIA avranno come qualificatore fdl: VARDOM.fdl, STRUTT.fdl... e staranno nella cartella G:\Indagini\fdl\prova tranne il file dei dati grezzi che dovrà essere indicato per l'esecuzione del check dei dati.

Per aprire un progetto esistente è sufficiente selezionarlo dal list-box dei *Progetti Recenti* e dare OK.

Il nome del progetto e quello della cartella che lo contiene possono essere diversi.

ATTENZIONE!!! Il nome del progetto non può contenere spazi.

Per riutilizzare i progetti creati con la versione SAS di CONCORDJAVA è sufficiente scegliere come nome del progetto *dat* (estensione dei file generata automaticamente dalla procedura) e come cartella di lavoro la vecchia cartella.

3. INTRODUZIONE AL METODO PROBABILISTICO (SCIA)

L'approccio per l'imputazione probabilistica in ConcordJava si basa sulle funzioni di **SCIA** (**S**istema di **C**ontrollo e **I**mputazione **A**utomatica) sistema per l'editing e l'imputazione automatica di variabili qualitative.
Opera esclusivamente su variabili di tipo qualitativo, o quantitative rapportabili a qualitative perché suddivisibili in classi o con dominio poco numeroso (ad esempio l'età) e consente l'individuazione e la correzione automatica degli errori di tipo casuale presenti nei dati con un metodo interamente basato sulla metodologia di Fellegi-Holt.

Per eseguire i processi di controllo e di imputazione dei valori delle variabili devono essere specificate le elaborazioni da effettuare sui dati e le modalità con cui esse devono essere realizzate riassumibili in 5 passi principali:

1. ***definizione delle variabili***, cioè dei campi del tracciato record dell'indagine con nome, posizione e lunghezza;
2. ***definizione delle regole formali (o strutturali)***, cioè di quelle regole che derivano direttamente dalla struttura del questionario (in particolare, dalle istruzioni di compilazione del questionario stesso). Esse esprimono condizioni di incompatibilità fra variabili, specificano cioè situazioni di non correttezza dei record. In particolare, esse indicano quando la presenza o l'assenza di risposta per una variabile o una lista di variabili risulta incompatibile con i valori assunti da variabili precedenti. Esse vengono inserite direttamente in forma normale;
3. ***definizione delle regole sostanziali,*** cioè di quelle regole che derivano dalle conoscenze a priori sulle relazioni esistenti fra le variabili rilevate. Come le regole formali anch'esse esprimono condizioni di incompatibilità fra variabili, specificano cioè situazioni di non correttezza dei record e vengono inserite direttamente in forma normale;
4. ***creazione dell'insieme completo***: questo passo è eseguito dopo aver inserito le regole originali. E' dalle caratteristiche di questo insieme (completezza, non contraddittorietà, ecc.) che dipende in massima parte la qualità dei risultati finali.
5. ***specificazione dei parametri***, mediante i quali vengono definite modalità generali per la fase di imputazione. ConcordJava prevede che siano specificati:
   - il grado di fissità delle variabili;
   - le variabili da sottoporre a imputazione forzata, quando viene usato il metodo di imputazione basato sulle distribuzioni marginali delle variabili stesse ;
   - i pesi da assegnare alle modalità delle variabili in caso di imputazione mediante distribuzioni marginali;
   - le variabili chiave sulle quali viene ordinato il file di input quando viene usato il metodo di imputazione da donatore: quando questo parametro è specificato, un record errato viene corretto con un donatore avente le stesse chiavi del record errato;
   - criteri per la gestione dell'insieme di record da cui vengono scelti i donatori, e cioè: la dimensione per il *serbatoio* dei record candidati come donatori e il numero massimo di volte che può essere usato uno stesso record donatore.

Una volta inserito l'insieme delle regole formali e sostanziali (insieme *iniziale* delle regole) prima dell'esecuzione della fase di controllo dei dati ed imputazione, il sistema prevede che vengano effettuate alcune elaborazioni distinte:

1. Controllo delle regole e generazione dell'insieme minimale di edit

La generazione dell'insieme minimale viene effettuata al fine di :

a. eliminare eventuali regole *ridondanti*, ossia regole che esprimono condizioni già implicate in altre regole;
b. segnalare regole direttamente *contraddittorie*;
c. aggregare regole che si possono combinare fra loro.

L'insieme minimale è quindi di dimensioni non superiori a quelle dell'insieme originale ed è sufficiente per il controllo dei dati. Il suo utilizzo ai fini dell'imputazione, però, non garantisce né la correttezza dei risultati finali, né la minimalità nel numero di correzioni che verranno effettuate sui dati: solo l'insieme completo fornisce questo tipo di garanzie. Inoltre, i tempi di elaborazione per l'imputazione di ogni record possono essere elevati, in quanto il sistema può dover effettuare un gran numero di tentativi prima di individuare la soluzione per la correzione del record.

Maggiore è il rapporto fra il numero di edit impliciti ed il numero di edit originali, maggiore è la frequenza con cui tali inconvenienti si verificano. Nel caso in cui tale rapporto è molto basso, l'applicazione ai dati dell'insieme minimale dà risultati soddisfacenti: per questo motivo si può decidere di generare ed applicare ai dati solo l'insieme minimale, senza porsi il problema della generabilità dell'insieme completo ed, eventualmente, della suddivisione dell'insieme originale degli edit.

Le necessità di cui ai punti a, b e c fanno sì che l'insieme minimale venga sempre generato, anche nel caso in cui sia possibile generare l'insieme completo. Il ricorso all'insieme minimale per la localizzazione degli errori o la loro imputazione è perseguito nel caso in cui risulti impossibile o eccessivamente onerosa la generazione dell'insieme completo. ConcordJava genera nella cartella di progetto il file "regole_da_minset" con il set minimale di regole che serve al riciclo del passaggio di c*ontrollo delle regole e generazione dell'insieme minimale di edit* per verificare eventuali ulteriori accorpamenti.

2. Generazione dell'insieme completo di edit

Con la generazione dell'insieme completo di edit il sistema individua tutti gli edit implicitamente contenuti nell'insieme iniziale di regole (edit espliciti), combinando fra loro gli edit originali secondo la metodologia di Fellegi-Holt. Il procedimento di generazione consiste nel tentare di combinare gli edit assumendo come campo generatore via via tutte le variabili coinvolte: se questo procedimento produce nuovi edit, esso va ripetuto anche combinando gli edit nuovi con quelli preesistenti, e così via fino a quando nessun nuovo edit viene prodotto.La generazione dell'insieme completo garantisce la creazione di un insieme di edit non contraddittorio, e la correttezza della correzione dei dati rispetto a tali regole.

3. Eventuale suddivisione dell'insieme originale di regole

Nel caso in cui non sia stato possibile generare l'insieme completo di edit a causa dell'eccessiva complessità ed onerosità dell'operazione, occorre procedere a ridurre tale complessità suddividendo l'insieme iniziale di regole in due o più sottogruppi: tali sottogruppi saranno poi sottoposti, separatamente al processo di generazione dell'insieme completo. Poiché, lo ricordiamo, ad ogni

insieme di regole corrisponde un processo di elaborazione dei dati (editing e imputazione), l'operazione di suddivisione dell'insieme originale di edit comporta la generazione di due o più distinti processi di elaborazione. Se gli insiemi di regole generati sono disgiunti, l'ordine di esecuzione è ininfluente. Se invece gli insiemi di regole generati contengono variabili comuni, durante l'esecuzione di uno dei processi di elaborazione sarà necessario tenere fisse tutte le variabili imputate dal/dai processi precedentemente eseguiti. Naturalmente la suddivisione sarà tanto migliore quanto minore è il numero di variabili comuni: il risultato ottimale è quello in cui i vari sottoinsiemi di regole risultano completamente disgiunti

Controllo dei dati:

Il problema dell'individuazione dei record errati consiste nel localizzare, usando la funzione di check, le unità statistiche in cui le variabili rilevate assumono valori tali da attivare uno o più edit del piano di incompatibilità.

Correzione dei dati:

Identificati i record che violano uno o più edit, per ognuno di essi il sistema, tramite la funzione di imputazione, deve individuare l'insieme di variabili da modificare e l'insieme dei valori da assegnare ad esse in modo tale che siano garantite le seguenti proprietà:

- ✓ il numero di correzioni per ogni record sia minimo;
- ✓ restino invariate le distribuzioni originali dei dati;
- ✓ il record risultante soddisfi tutti gli edit.

Il problema di cui al punto i) viene risolto in ConcordJava implementando l'algoritmo proposto da Fellegi-Holt: "*l'insieme minimo di variabili da imputare viene determinato attraverso l'identificazione di quelle variabili che coprono tutti gli edit attivati dal record errato*". L'utente può comunque impedire o rendere meno probabile l'inserimento di una o più variabili nell'insieme minimale, assegnando a ciascuna di esse un grado di fissità (da 1 a 9) dipendente dalla probabilità di errore prevista per tali variabili.

Per quanto riguarda i problemi di cui ai punti ii) e iii), essi trovano soluzione all'interno degli algoritmi implementati in ConcordJava per l'imputazione dei dati.

In particolare, sono possibili tre possibili strategie di correzione:

1. ***imputazione congiunta***;
2. ***imputazione sequenziale*** ;
3. ***imputazione basata sulle distribuzioni marginali* o *imputazione forzata***.

Le prime due sono strategie di correzione del tipo "da donatore", mentre la terza tecnica è basata sull'analisi e sull'utilizzo delle distribuzioni marginali semplici rilevate nell'indagine per le variabili dell'insieme minimale.

Per quanto riguarda la correzione dei dati, in ConcordJava è presente una procedura generale di imputazione che, implementando al suo interno (sequenzialmente) le suddette strategie di correzione, ha una struttura indipendente dalle caratteristiche delle strategie stesse. Le caratteristiche delle diverse tecniche di imputazione agiscono infatti solo all'interno di alcune delle fasi componenti la procedura di correzione stessa.

Tale procedura generale è costituita da due fasi principali:

- costruzione di un "*serbatoio*" di record donatori, costruzione dipendente dalle specifiche assegnate dall'utente tramite i *parametri* e selezione del record errato;
- scelta del donatore e correzione dei record, le cui modalità dipendono dal tipo di algoritmo di correzione utilizzato.

Si tenga presente che, mentre il meccanismo di costruzione del serbatoio può essere controllato dall'utente (appunto attraverso i parametri), la particolare strategia di imputazione che il sistema adotterà per l'effettiva correzione di un certo record dipende quasi esclusivamente da criteri ed elaborazioni interni al sistema stesso.

Come già detto, in ConcordJava sono implementate due metodologie di imputazione da donatore:

- **imputazione congiunta**;
- **imputazione sequenziale**.

La prima tecnica, in particolare, prevede le due seguenti versioni:

1. ***imputazione congiunta ristretta***, in cui, dato un certo record errato, vengono selezionati come possibili donatori quei record che possiedono, per le *variabili di accoppiamento*, valori identici a quelli contenuti nel record errato;
2. ***imputazione congiunta allargata***,in cui, dato un certo record errato, vengono selezionati come possibili donatori quei record che possiedono, per le *variabili di accoppiamento*, valori contenuti nei corrispondenti intervalli (*range*) opportunamente determinati.

Nel caso di *imputazione sequenziale* si procede ad imputare una variabile alla volta: per ciascuna variabile appartenente all'insieme minimo viene calcolato il *range* dei valori ammissibili; per ciascuna di esse viene quindi cercato nel serbatoio e, se esiste, selezionato un record donatore con valore compreso nel corrispondente *range*.

Riassumiamo quindi le varie fasi di cui si compone il processo di correzione dei dati con tecniche da donatore:

I. ***Inviduazione dell'insieme minimale***.In questa fase, dato il record errato r, viene determinato il minimo numero di variabili da correggere tra quelle presenti in tutti gli edit attivati da r.
II. ***Controllo delle variabili marginali***.Prima di procedere alla ricerca del donatore, il sistema verifica la presenza di qualcuna delle variabili dell'insieme minimale all'interno della lista di variabili specificata come marginali. In caso positivo, tali variabili vengono corrette direttamente col metodo dell'imputazione forzata, e si procede alla ricerca del donatore per la correzione delle variabili residue dell'insieme minimale.
III. ***Selezione del donatore***. In questa fase, a seconda della strategia di imputazione adottata, viene selezionato dal serbatoio il record donatore d.

Nel caso in cui, per un certo record errato r, non sia stato possibile individuare un donatore adatto con nessuna delle tecniche da donatore disponibili in ConcordJava, il sistema corregge automaticamente le variabili dell'insieme minimale utilizzando le corrispondenti distribuzioni marginali semplici (*correzione forzata*). La correzione di tali variabili avviene sequenzialmente.

Per la correzione di una o più variabili l'utente può anche decidere di non tentare affatto la

correzione basata sulle tecniche da donatore, può cioè richiedere al sistema di sottoporre direttamente tali variabili al metodo dell'imputazione forzata specificandole come marginali.

La tecnica di correzione forzata è basata su un algoritmo random di estrazione del valore da assegnare alla variabile errata (selezionato tra i valori ammissibili), estrazione guidata da una funzione di probabilità definita sulla base della distribuzione di frequenze che la variabile stessa assume nel file dei dati originari.

L'uso di regole di tipo deterministico è previsto solo nel caso si debbano correggere errori di tipo sistematico: questo tipo di errori derivano, generalmente, da problemi strutturali nel questionario, nell'organizzazione della rilevazione, nella registrazione dei dati. La presenza di errori sistematici nei dati viene generalmente verificata attraverso un'analisi delle imputazioni probabilistiche effettuate dal sistema, analisi condotta possibilmente in fase di test del piano di incompatibilità.

Questa analisi viene condotta sulla base dei report che ConcordJava produce automaticamente al termine del processo di correzione.

3.1 La metodologia Fellegi-Holt

Tre sono i criteri fondamentali per l'imputazione delle variabili qualitative alla base della metodologia proposta da Fellegi e Holt[1]:

1. in ogni record i dati devono soddisfare tutte le regole di validità e incompatibilità, cambiando il meno possibile il valore dei campi;
2. le regole di imputazione devono essere derivate dalle regole di controllo, senza esplicita specificazione;
3. le distribuzioni di frequenza marginali e congiunte devono essere mantenute il più possibile.

Edit in forma normale

Distinguiamo gli edit logici, riguardanti le variabili qualitative, dagli edit aritmetici, riguardanti le variabili quantitative.

DEFINIZIONE: un ***edit logico*** esprime una condizione di inaccettabilità su una data combinazione di valori di due o più variabili

Un edit può essere formalizzato come l'applicazione di una funzione f a sottoinsiemi dei domini di n variabili:

$$f(A_1^0, A_2^0, \dots, A_n^0)$$

dove:

A_i^0 : sottoinsieme del dominio della variabile i-esima

f : funzione logica che connette i vari A_i^0 mediante gli operatori logici di intersezione ($\cap$) e unione

[1] Quanto segue è una sintesi dell'articolo "A Systematic Approach to Automatic Edit and Imputation" di

I.Fellegi e D.Holt pubblicato sul Journal of the American Statistical Association (marzo 1976)

($\cup$)

Un record $\underline{a}$ è errato se:

$$\underline{a} \in f(A_1^0, A_2^0, \ldots, A_n^0)$$

Applicando ripetutamente alla f la legge distributiva otteniamo:

$$f(A_1^0, A_2^0, \ldots, A_n^0) =$$
$$(A_{i_1}^1 \cap A_{i_2}^1 \cap \ldots \cap A_{m_i}^1) \cup (A_{j_1}^2 \cap A_{j_2}^2 \cap \ldots \cap A_{m_j}^2) \cup \ldots \cup (A_{k_1}^r \cap A_{k_2}^r \cap \ldots \cap A_{m_k}^r)$$

Possiamo dire che un record è errato se appartiene ad almeno uno dei termini a secondo membro. Definiamo come "edit in forma normale" ognuno di tali termini.

DEFINIZIONE: un ***edit in forma normale*** è un edit logico in cui l'unico operatore ammesso è quello di intersezione

In simboli:

$$\bigcap_{i \in s} A_i^*$$

Ogni edit logico, di qualsiasi forma, può sempre essere tradotto in una serie di edit in forma normale. Consideriamo, ad esempio, la seguente regola (di compatibilità):

"*Se una persona ha età inferiore a 16 anni, oppure frequenta una scuola elementare, allora non può essere capo-famiglia, ed il suo stato civile deve essere celibe o nubile*".

Questa regola può essere convertita in una serie di edit in forma normale attraverso i seguenti passi:

1. *formalizzazione:*

$$[\ (\text{Età} < 16) \cup (\text{Scuola Elementare})] \rightarrow [\ (\neg\ \text{Capo-famiglia}) \cap (\text{Celibe/Nubile})\ ]$$

2. *traduzione in regola di incompatibilità*:

$$[\ (\text{Età} < 16) \cup (\text{Scuola Elementare})]\ \cap \neg\ [\ (\neg\ \text{Capo-famiglia}) \cap (\text{Celibe/Nubile})\ ] = \text{errore}$$

3. *semplificazione*:

$$[\ (\text{Età} < 16) \cup (\text{Scuola Elementare})]\ \cap [\ (\text{Capo-famiglia}) \cup (\neg\ \text{Celibe/Nubile})\ ] = \text{errore}$$

4. *applicazione della legge distributiva*:

$$[\ (\text{Età} < 16) \cap (\text{Capo-famiglia})] \cup$$
$$[\ (\text{Età} < 16) \cap (\neg\ \text{Celibe/Nubile})] \cup$$
$$[\ (\text{Scuola Elementare}) \cap (\text{Capo-famiglia})] \cup$$
$$[\ (\text{Scuola Elementare}) \cap (\neg\ \text{Celibe/Nubile})] = \text{errore}$$

I quattro termini nell'ultima espressione sono altrettanti edit in forma normale.

L'insieme completo degli edit

DEFINIZIONE: gli edit in forma normale specificati direttamente dallo statistico sono detti ***edit espliciti***.

Un record che non attiva alcun edit esplicito si dice corretto, e non necessita di alcuna modifica. Al contrario, un record che attiva almeno un edit esplicito si dice errato, e necessita della modifica di almeno una variabile.

Mentre gli edit espliciti sono necessari e sufficienti per determinare la correttezza di un record, essi non sono sufficienti per una sua ottimale correzione.

DEFINIZIONE: chiamiamo ***edit implicito*** un edit logicamente contenuto negli edit espliciti. La funzione degli edit impliciti, considerati congiuntamente con gli edit espliciti, è quella di permettere la correzione ottimale di un record errato.

DEFINIZIONE: l'***insieme completo*** degli edit è dato dall'unione degli edit espliciti e di quelli impliciti.

Per eseguire in modo ottimale il passo di scelta delle variabili da imputare, e di determinazione del range di valori imputabili, è necessario preventivamente generare l'insieme completo di edit.

Consideriamo il seguente esempio.

Supponiamo che un record contenga tre variabili, di cui siano definiti i seguenti domini:

VARIABILI	***DOMINI***
ETA'	0-14, 15-99
STATO CIVILE (STACIV)	celibe, coniugato, separato,divorziato,vedovo
RELAZIONE COL CAPO FAMIGLIA (RELCF)	capofamiglia, coniuge, altro

Siano stati definiti i seguenti edit in forma normale espliciti, esprimenti condizioni di incompatibilità:

I. (ETA = 0-14) $\cap$ (STACIV = coniugato, separato,divorziato,vedovo)

II. (STACIV = celibe, separato, divorziato, vedovo) $\cap$ (RELCF = coniuge)

Possiamo riscriverli come condizioni di compatibilità nel seguente modo:

(ETA = 0-14) $\rightarrow$ (STACIV = celibe)

(STACIV = celibe, separato, divorziato, vedovo) $\rightarrow$ (RELCF $\neq$ coniuge)

Poichè la conseguenza della prima implicazione è contenuta nella premessa della seconda, possiamo derivare che

(ETA = 0-14) $\rightarrow$ (RELCF $\neq$ coniuge)

relazione che, opportunamente ritradotta in forma normale, diventa:

III. (ETA = 0-14) $\cap$ (RELCF = coniuge)

Questo terzo edit era implicitamente contenuto nei primi due.

Supponiamo ora di considerare il seguente record:

(ETA = 0-14) $\cap$ (STACIV = coniugato) $\cap$ (RELCF = coniuge)

Questo record attiva gli edit I e III.

Per correggere il record, ricerchiamo l'insieme minimo di variabili che *copra* tutti gli edit attivati (espliciti e impliciti) dal record in questione. Nel nostro caso verifichiamo che la variabile ETA' è presente sia nel primo che nel terzo edit attivato. Per disattivare tali edit è sufficiente assegnare a ETA' un valore interno all'*intersezione dei complementi* dei valori che compaiono negli edit attivati o attivabili:

$$(\neg\ 0\text{-}14) \cap\ (\neg 0\text{-}14) = 15\text{-}99$$

Assegnando il valore 15-99 alla variabile ETA', il record può dirsi corretto, in quanto non attiva alcun edit: nel far ciò abbiamo tenuto conto del principio del minimo cambiamento, in quanto abbiamo modificato una sola variabile.

Se in questo processo di ricerca dell'insieme minimale di variabili da imputare non avessimo tenuto conto dell'edit implicito, avremmo considerato il solo edit I: per disattivarlo, avremmo potuto

scegliere di imputare sia ETA' che STACIV. Se avessimo scelto STACIV, che compare anche nell'edit II, avremmo constatato che l'intersezione del complemento dei relativi valori è l'insieme vuoto $\varnothing$:

$$\neg \text{(coniugato, separato, divorziato, vedovo)} \cap \neg \text{(celibe, separato, divorziato, vedovo)} =$$
$$= \text{celibe} \cap \text{coniugato} = \varnothing$$

L'impossibilità di trovare dei valori imputabili a STACIV tali da correggere il record deriva dal fatto che STACIV non è contenuto nell'edit III, implicito, attivato dai valori delle variabili ETA' e RELCF. La conseguenza di carattere generale è che *la non considerazione degli edit impliciti non permette di definire sempre insiemi minimi di variabili da imputare che siano in grado di riportare il record in una situazione di correttezza.*

LEMMA: dati s edit e_r e n variabili, per ogni arbitraria variabile i, un edit

$$e_i^* : \bigcap_{j=1}^{n} A_j^*$$

si dice generato dagli s edit se e solo se

$$\begin{cases} A_j^* = \bigcap_{r \in s} A_j^r & j = 1,2,\ldots,n \quad i \neq j \\ A_i^* = \bigcup_{r \in s} A_i^r \end{cases}$$

In altri termini, fissata una variabile i (detta *generante*), il corrispondente A_i^* sarà ottenuto come *unione* degli A_i^r, mentre ogni altro A_j^* sarà ottenuto come *intersezione* degli A_j^r.

DEFINIZIONE: Un edit generato si dice ***edit implicito essenzialmente nuovo*** se e solo se:

1. A_i^* coincide col dominio della variabile i;
2. ogni A_i^r è non vuoto ed è un sottoinsieme proprio del dominio della variabile i;

Consideriamo il seguente esempio. Siano dati gli edit:

I. (ETA = 0-14) $\cap$ (RELCF = qualsiasi) $\cap$ (STACIV $\neq$ celibe)
II. (ETA=qualsiasi) $\cap$ (RELCF = coniuge) $\cap$ (STACIV = celibe, separato, divorziato, vedovo)

Se fissiamo ETA' come variabile generante otteniamo:

(ETA=qualsiasi) $\cap$ (RELCF = coniuge) $\cap$ (STACIV = separato, divorziato, vedovo)

che è ridondante rispetto al secondo edit.
Fissando invece RELCF otteniamo:

(ETA=0-14) $\cap$ (RELCF = qualsiasi) $\cap$ (STACIV = separato, divorziato, vedovo)

che è ridondante rispetto al primo edit.
Infine, scegliendo STACIV come variabile generante:

(ETA=0-14) $\cap$ (RELCF = coniuge) $\cap$ (STACIV = qualsiasi)

che è un edit implicito essenzialmente nuovo.
DEFINIZIONE : Un edit generato da due o più edit tra loro contraddittori (inconsistenti) è detto ***edit degenere***
Consideriamo il seguente esempio:

I. (ETA = 0-14) $\cap$ (STACIV $\neq$ celibe)

II. (ETA = 15-99) $\cap$ (STACIV $\neq$ celibe)

Assumendo ETA' come campo generante, otteniamo l'edit esplicito

III. (ETA = qualsiasi valore) $\cap$ (STACIV $\neq$ celibe) = (STACIV $\neq$ celibe)

che ci dice che sono errati tutti i valori di STACIV diversi da celibe, il che chiaramente contraddice la definizione del dominio della variabile STACIV. L'edit III è un edit degenere, ed in quanto tale può essere generato solo da edit tra loro contraddittori.

I seguenti teoremi e corollari assicurano che, *avendo a disposizione l'insieme completo di edit, un qualsiasi record errato è sempre correggibile, e lo è in modo ottimale*.

Sia Ω l'insieme completo di edit, e sia Ω_k un sottoinsieme tale da coinvolgere le prime k variabili (con l'esclusione, quindi, di tutti gli edit in cui compaiano le variabili k+1, k+2, ... , n).

TEOREMA 1: se gli a_i^0 sono possibili valori per le prime k-1 variabili, e se questi valori soddisfano tutti gli edit in Ω_{k-1}, allora esiste un qualche valore a_k^0 tale da soddisfare tutti gli edit in Ω_k.

La ripetuta applicazione del teorema 1 permette di conseguire il seguente

COROLLARIO 1: se un record ha n variabili, di cui le prime k-1 hanno valori a_i^0 (i=1,2,...,k-1) tali che tutti gli edit in Ω_{k-1} sono soddisfatti, allora esistono valori a_i^0 (i=k,k+1,...,n) tali da soddisfare tutti gli edit in Ω.

Ed inoltre:

COROLLARIO 2: se un record ha n variabili, di cui un sottoinsieme s ha la proprietà che almeno uno dei valori a_i ((i$\in$s) compare in ogni edit attivato dal record, allora esistono dei valori a_i^0 ((i$\in$s) tali che, assieme agli a_i (i$\notin$s) fanno si che il record soddisfi tutti gli edit.

Metodi di imputazione

La metodologia prevede, per ogni record errato:

1. l'identificazione dell'*insieme minimo di variabili da modificare*;
2. per ogni variabile rientrante nell'insieme minimo, la *determinazione dell'insieme di valori attribuibili*, e *imputazione* di uno tra questi.

Per quanto riguarda il punto 1, ricordiamo che l'insieme minimo di variabili da imputare è costituito da quell'insieme di variabili che "coprono" tutti gli edit attivati dal record e che risulta essere di dimensione minima.

Per quanto concerne il punto 2, sono proposti due metodi, entrambi di tipo *hot deck*, consistenti nell'imputare in una variabile del record corrente (ricevente) il valore della stessa variabile in un record (donatore) scelto tra quelli esatti. I metodi in questione sono:

- metodo dell'imputazione sequenziale;
- metodo dell'imputazione congiunta.

METODO 1: **Imputazione sequenziale**

Consideriamo un record errato di cui sia già stato identificato un insieme minimo di k variabili da imputare. Il metodo consiste nell'imputare dapprima la k-esima variabile, e poi, sequenzialmente, le variabili k-1,k-2,...,1.
Consideriamo tutti gli M edit in cui

- è presente la variabile k;
- non sono presenti le variabili 1,2,...,k-1.

Tra questi, consideriamo solo gli M' edit in cui non sono presenti gli edit sicuramente disattivati dai valori correnti delle variabili k+1, k+2, ... , n: gli M' edit sono quelli che possono essere attivati o meno in funzione dei valori della sola variabile k. Se vogliamo che il record soddisfi tali edit, il valore da assegnare alla variabile k deve soddisfare la condizione:

$$a_k^0 \in \bigcap_{r=1}^{M'} \overline{A_r^k}$$

cioè deve appartenere all'insieme intersezione dei complementi dei valori indicati per la variabile k in tutti gli M' edit: tale insieme non è mai vuoto per il teorema 1.

Lo stesso procedimento viene iterato per le variabili k-1, k-2, ...1, fino all'esaurimento dell'insieme minimo di variabili da imputare.

Consideriamo il seguente esempio, con 5 variabili:

VARIABILI	***DOMINI***
SESSO	maschio, femmina
ETA	0-14,15-16,17-99
STATO CIVILE (STACIV)	celibe, coniugato, separato, divorziato, vedovo
RELAZIONE COL CAPOFAMIGLIA (RELCF)	moglie, marito, figlio, altro
LIVELLO D'ISTRUZIONE (ISTRUZ)	nessuno,elementare, secondario, post-secondario

L'insieme (completo) degli edit è il seguente:

e_1 : (SESSO=maschio) $\cap$ (RELCF=moglie)

e_2 : (ETA=0-14) $\cap$ (STACIV$\neq$celibe)

e_3 : (STACIV$\neq$coniugato) $\cap$ (RELCF=moglie,marito)

e_4 : (ETA=0-14) $\cap$ (RELCF=moglie,marito)

e_5 : (ETA=0-16) $\cap$ (ISTRUZ=post-secondaria)

Sia dato il seguente record:

VARIABILE	***VALORE***
SESSO	maschio
ETA	12
STACIV	coniugato
RELCF	moglie
ISTRUZ	elementare

Il record attiva gli edit e_1, e_2, e_4. Nessuna singola variabile "copre" i tre edit. Tre coppie di variabili coprono gli edit attivati: (SESSO, ETA'), (ETA', RELCF) e (STACIV, RELCF). Supponiamo di scegliere la coppia (SESSO, ETA'): la dimensione s dell'insieme è pari a 2.

Sia ETA' la variabile k-esima (k=2). Consideriamo tutti gli edit che contengono ETA' ma non SESSO (la variabile k-1=1):

e_2 : (ETA=0-14) $\cap$ (STACIV$\neq$celibe)

e_4 : (ETA=0-14) $\cap$ (RELCF=moglie,marito)

e_5 : (ETA=0-16) $\cap$ (ISTRUZ=post-secondaria)

L'edit e_5 è sempre soddisfatto per qualsiasi valore di ETA' dal momento che nel record il valore di ISTRUZ è "elementare". Per calcolare i valori imputabili ad ETA' dobbiamo quindi considerare solo A_2^2 e A_2^4:

$$a_2^* \in \overline{A_2^2} \cap \overline{A_2^4} \equiv \overline{(0\text{-}14)} \cap \overline{(0\text{-}14)} = (15\text{-}99)$$

cercheremo quindi un record donatore con un valore di ETA' compreso tra 15 e 99: supponiamo 22.

Passiamo ora variabile SESSO (k-1=1). Solo l'edit e_1 la contiene, quindi:

$$a_1^* \in \overline{A_1^1} \equiv \overline{\text{maschio}} = \text{femmina}$$

Essendo unico, il valore "femmina" è direttamente imputato alla variabile SESSO. Il record corretto sarà quindi il seguente:

VARIABILE	***VALORE***
SESSO	femmina
ETA	22
STACIV	coniugato
RELCF	moglie
ISTRUZ	elementare

METODO 2: **Imputazione congiunta.**

Per un dato record errato siano state definite le k variabili da imputare. Si considerino gli M" edit con le k variabili

$$e_r : \bigcap_{i=1}^{n} A_i^r \quad (r=1,2,...,M")$$

dove $a_i^0 \in A_i^r$ (i=k+1,k+2,...,n). Sono gli edit in cui sono presenti le k variabili, e dove le variabili k+1, k+2, ..., n hanno nel record valori interni agli A_i^r: sono cioè gli edit attivabili o meno in funzione dei valori che si danno alle k variabili.

Si considerino gli insiemi

$$A_i^* = \bigcap_{r=1}^{M"} A_i^r \quad (i=k+1, k+2, \ldots, n)$$

Se scegliamo un qualsiasi record, tra quelli esatti, i cui valori delle variabili k+1, k+2, ... ,n siano interni agli insiemi così definiti, i valori di tale record nelle variabili 1,2,...,k sono attribuibili in blocco al record errato corrente, in quan

to costituiscono una combinazione che sicuramente garantisce che tutti gli M'' edit siano soddisfatti (cioè disattivati). Per tale motivo non c'è alcun bisogno di calcolare l'insieme dei valori attribuibili alle k variabili dell'insieme minimo.

Riprendiamo in considerazione l'esempio visto per l'imputazione sequenziale: siano ancora SESSO ed ETA' le variabili dell'insieme minimo: queste due variabili sono presenti negli edit e_1, e_2, e_4 ed e_5. Quest'ultimo è soddisfatto comunque per il valore di ISTRUZ. Restano:

e_1 : (SESSO=maschio) $\cap$ (RELCF=moglie)

e_2 : (ETA=0-14) $\cap$ (STACIV$\neq$celibe)

e_4 : (ETA=0-14) $\cap$ (RELCF=moglie,marito)

E' questo l'insieme M'' di edit.

Si determinano gli insiemi di valori per le variabili k+1, k+2, ..., n, cioè per STACIV (3), RELCF (4) e ISTRUZ (5):

A_3^* = coniugato, separato, divorziato, vedovo

A_4^* = moglie $\cap$ (moglie, marito) = moglie

A_5^* = qualsiasi valore

A questo punto, tra i record esatti viene ricercato un donatore che abbia i valori di STACIV e RELCF interni agli insiemi così determinati, ed i relativi valori di SESSO ed ETA' vengono attribuiti al record errato corrente.

3.2 Definizioni

Definizione delle variabili

Il metodo probabilistico tratto solo variabili qualitative con valori codificati da 0 a 9999 o blank.
Per definire le variabili si deve conoscere il tracciato record del questionario con indicate la posizioni iniziali e la lunghezza di ogni variabile.
E' necessario definire SOLO le variabili che si vogliono controllare e imputare fino ad un massimo di 500 variabili. Il resto del record sarà ricopiato senza modifiche.
Per ogni variabile si devono indicare:

- ✓ NOME, univoco e lungo al massimo 6 caratteri (esempi: ETA, Q12,COL1_6,...); il nome deve essere univoco anche rispetto ai nomi di eventuali LISTE.
- ✓ POSIZIONE, che indica la colonna di nizio della variabile nel record.
- ✓ LUNGHEZZA del campo (massimo 4 caratteri).
- ✓ MISSING se la variabile ammette la non risposta (blank).
- ✓ DOMINIO della variabile. Per dominio di una variabile si intendono i valori per un massimo di 700 che la variabile può assumere. Sono scritti nella forma DA A.

Per l'inserimento/modifica del dominio cliccare sulla casella: si aprirà una nuova finestra per l'inserimento dei valori.

ATTENZIONE!!! Prima di salvare le modifiche USCIRE DALL'ULTIMO CAMPO INSERITO

Il programma controlla che la lunghezza della variabile sia compresa tra 1 e 4, che i valori non superino la grandezza della variabile e che le variabili non si sovrappongano.
Le variabili sono registrate nel file VARDOM.progetto.

Definizione delle regole

Le regole di incompatibilità o edit scritte dall'utente, per un massimo di 2000, sono chiamate insieme minimale e descrivono l'incompatibilità logiche tra le variabili.
Non si devono scrivere le regole per il controllo del dominio di una variabile che è effettuato automaticamente sulla base dei valori del dominio definito per le variabili.
Le regole (EDIT) vanno scritte in formato libero definendo variabili e, tra parentesi, i valori del relativo dominio che, se verificati, attiveranno l'errore.

ATTENZIONE!!! La procedura richiama un editor di testo (per default notepad cioè il blocco note

di Windows, ma può essere modificato indicando un altro editor di testo nel file *concordjava.ini*).

E' indispensabile lasciare una riga vuota come ultima riga del file.

Il programma è CASE SENSITIVE: i nomi delle variabili devono essere MAIUSCOLI.

Operatori

( )	sottodomini *uguali a*
<)	sottodomini *diversi da*
-	intervallo di valori compresi gli estremi
,	separa i valori del sottodominio

I commenti iniziano con un * all'inizio di ogni riga.
Gli edit sono registrati nel file REGOLE.

Esempio:

```
ETA(1,3-10) RELCF(1)
```

che significa se età' ha il valore 1 oppure i valori da 3 a 10, e relazione con il capofamiglia ha valore 1 allora la regola è' verificata e il record è' errato.
Se la parentesi iniziale è **<** invece che **(** si intende diverso da. esempio: ETA(1,3-10) RELCF<2,3,4) che significa: se età ha il valore 1 oppure i valori da 3 a 10, e relazione con il capofamiglia ha valore diverso da 2, da 3 e da 4 allora la regola è verificata e il record è errato.

Definizione delle liste

Le liste di variabili sono utili per semplificare la scrittura delle regole. Sono insiemi di variabili la cui risposta nel questionario può mancare o non mancare in funzione del valore di un'altra variabile. Tipicamente appartengono ad una lista tutte le variabili di una sezione del questionario che deve essere compilata o no sulla base di un quesito filtro precedente, oppure sezioni di questionario che ammettono molte risposte multiple.
Scrivendo una sola regola che indica incompatibilità tra una variabile e la variabile di lista, saranno automaticamente generate, nella fase di controllo, tante regole quante sono le variabili indicate nella lista se la lista è stata definita **OR**, o una regola che comprende tutte le variabili della lista se questa è stata definita **AND**.
Una lista può comprendere al massimo 100 variabili.
Esempio: Definiamo la lista LISTA1 che comprende le variabili COND POSPRO CAROCC e la regola ETA(0-14) LISTA1<).
La funzione di controllo delle regole, se LISTA1 fosse in **OR**, genererà automaticamente i seguenti edit:

ETA(0-14) COND<)
ETA(0-14) POSPRO<)

ETA(0-14) CAROCC<)

Se, invece, LISTA1 fosse definita in **AND** avremmo un solo edit:

ETA(0-14) COND<) POSPRO<) CAROCC<)

Le liste in OR sono registrate sul file STRUTT e quelle in AND su LISTE.

Definizione delle fissità

Si può condizionare l'imputazione delle variabili attraverso alcuni parametri.
Per ogni variabile si possono impostare i seguenti parametri:

- **Fissità** - Valori da 1 a 9 (default 0).
  Vincolano la scelta delle variabili da imputare che sono cercate prima fra le variabili con fissità 0, poi, fra le variabili con fissità 1 finchè non si individui l'insieme minimo.
  Le variabili con fissità 9 non vengono imputate.
- **Match** - Per definire una variabile di match, cioè una variabile che dovrà avere lo stesso valore sia nel record errato che nel record donatore, se il programma di correzione decide di correggere la variabile selezionata.
  Si utilizza solo in caso di imputazione sequenziale.
- **Chiave** - Servono a definire degli strati per i donatori. Il serbatoio dei donatori viene rinnovato quando cambia lo strato. I record devono essere ordinati per le variabili chiave (al massimo 3).
  Per le variabili chiave la fissità deve essere 9.
- **Marginale** - Una variabile definita marginale viene corretta con l'imputazione sequenziale, seguendo la distribuzione marginale delle frequenze dei dati grezzi.

3.3. Funzioni

Controllo delle regole

La fase di controllo delle regole (insieme minimale) verifica gli eventuali errori di sintassi, incongruenze (edit contradditori) e ridondanze delle regole, accorpa regole che hanno le stesse variabili con domini diversi solo per un valore ed esplode le liste.

Esempio

edit contradditori:

STACIV<1) ETA(0-15)
STACIV<1) ETA<16-99)

edit ridondanti:

STACIV<1) ETA(0-15)
STACIV<1) ETA<0-9)

Le modifiche effettuate sono segnalate.
Le regole contraddittorie o errate fermano il processo mentre le regole ridondanti sono segnalate

In questa fase si scrivono i seguenti file:

- TABVARF che contiene, per ogni variabile, il dominio definito in classi derivate dall'insieme delle regole che la trattano.
- MINICE con la matrice derivante dell'insieme minimo in forma di 0/1.
- MINSET con le regole di MINICE in forma leggibile.
- SERICE per il trattamento della matrice nei passi successivi.
- SYSCON con i messaggi del passaggio di controllo.

Le segnalazioni di errore o di avvertimento, riferite al file "MINSET.dat", comprendono nella numerazione anche i commenti.

Derivazione delle regole implicite

Questa fase esegue la derivazione delle regole implicite a partire dalle regole esplicite scritte dall'utente.
Mentre le regole esplicite sono sufficienti per la separazione dei dati in esatti ed errati, le regole implicite sono indispensabili per garantire la correttezza della correzione dei dati.
La derivazione è un programma che per ogni variabile, detta "generatrice", verifica se le n regole che la contengono, raggruppate a 2,3,...a n coprono l'intero dominio della variabile, e in questo caso, se le altre variabili contenute nelle regole generano una nuova regola che viene detta "implicita".
Se questa regola non risulta compresa nelle altre regole dell'insieme minimo è detta "essenzialmente nuova" quindi diventa parte dell'insieme delle regole e rientra nel passo che viene riciclato.
Non è possibile quantificare a priori il tempo, talvolta anche di molte ore, necessario alla derivazione dell'insieme completo.
In questa fase si scrivono i seguenti file:

- MAXICE che contiene la matrice dell'insieme completo in forma binaria
- GENER con la storia di ogni regola nella generazione
- SYSDER con i messaggi del passo di derivazione
- COMPLETE con le regole in forma leggibile
- LISGEN con la descrizione per ogni regola generata della variabile
- LISGEN_progetto.csv integra le informazioni di LISGEN e GENER

Se il passo di derivazione si ferma per "edit degenere" o "edit contraddittori", bisogna riferirsi al file GENER e al file COMPLETO per capire quali regole sono tra loro incompatibili. GENER contiene un record per ogni regola con numeri indicanti:

- la variabile generatrice (primo numero nella riga);
- le regole che hanno contribuito alla generazione della regola in esame (numeri successivi);
- -9999 che chiude l'insieme delle regole generatrici.

i numeri di regola negativi (es.: -34) indicano che la regola è stata cancellata nei cicli di derivazione, e quindi bisogna considerare le regole successive al -9999 che hanno derivato la regola in esame e così via.
Le regole esplicite hanno il record corrispondente su "GENER.dat" con tutti 0.

Risultati derivazione
Sono visualizzate due tabella. La prima contiene gli edit espliciti (definiti dall'utente). La seconda gli edit impliciti generati dal passo di derivazione. Per questi si mostra anche la variabile generatrice e i numeri degli edit coinvolti nella genarazione. I dati sono letti dal file LISGEN_progetto.csv

Insieme completo
Visualizza il file COMPLETO.progetto dove sono elencate tutte le regole esplicite e implicite che formano l'insieme completo degli edit usati per la localizzazione.

Check dei dati

Il check dei dati separa i dati grezzi in esatti ed errati, verificando per i record di input, le incompatibilità descritte negli edit dell'insieme minimo tramite il file esterno MINICE (non leggibile) scritto nella fase di controllo delle regole. Le regole corrispondenti in formato leggibile sono registrate nel file MINSET.
Durante la fase di check, ogni record in esame che violi anche una sola regola di incompatibilità è registrato tra gli errati. In parallelo, per ogni record errato, si scrive un record nel file ERRORI con la lista dei numeri degli edit attivati.
A fine passaggio di check, segnalato da apposito messaggio, vengono mostrati i contatori dei record letti, esatti ed errati e una tabella con elencate le regole violate e gli eventuali dati fuori dominio.
Tutti i riferimenti numerici delle regole vanno fatti con il file MINSET.

In questa fase si scrivono i seguenti file

- SYSCHK contiene i messaggi e i contatori;
- LISCHK con la tabella riassuntiva delle regole verificate. I numeri fanno riferimento alle regole scritte nel MINSET.
- ESATTI con i record esatti

- ERRATI con i record errati
- ERRORI parallelo al file degli errati con i numeri delle regole errate per ogni record
- FREQUEN con la frequenza dei casi per ogni dominio di TABVARF.

Localizzazione

La fase (facoltativa) di localizzazione degli errori individua, per ogni record errato, l'insieme di variabili che dovranno essere imputate affinchè il record soddisfi tutti gli edit utilizzando l'algoritmo del minimo cambiamento di Fellegi-Holt.
I file usati in questa fase sono:

- TABVARF con i domini separati in classi per ogni variabile
- VARFIX con le variabili definite fisse
- MAXICE insieme completo in forma binaria
- SYSCHK con i contatori dei record errati ed esatti
- ERRATI file dei record errati
- DATILOC file dei dati errati con i campi individuati per l'imputazione flaggati da '*'
- SYSLOC contatori delle imputazioni effettuate per tipo di imputazione
- LISTALOC lista delle variabili da imputare per record

Imputazione

La fase di correzione dei dati, o imputazione, trasforma i record errati, output della fase di controllo o check, in record corretti, utilizzando l'algoritmo del minimo cambiamento, una delle basi della metodologia di Fellegi-Holt
Per ogni record errato, tramite l'insieme completo "MAXICE.dat", generato nella fase di derivazione degli edit impliciti, vengono verificate le regole di incompatibilità, e si cerca il numero minimo di variabili che, modificate con i valori presi da un serbatoio di donatori e tentando di prendere sempre il record esatto più somigliante, rendono corretto il record errato in esame.
Il serbatoio dei record donatori è costituito dal numero di record esatti, output del passaggio di check. La sua dimensione può essere modificata da un parametro di imputazione (massimo: 50000 record). Questo serbatoio, che viene rinnovato solo se esistono variabili definite "chiave", è l'unica fonte che fornisce i valori per la correzione delle variabili che fanno parte dell'insieme minimo di variabili da correggere.
I metodi di imputazione sono:

- da donatore: le variabili da imputare sono prese da un record donatore a distanza minima scelto con uno dei seguenti criteri:
    1. congiunta ristretta: si cerca fra gli esatti un record con i valori delle variabili da non modificare uguali a quelle del record da correggere.
    2. congiunta allargata: si cerca un record tra gli esatti con i valori compatibili nelle variabili da non modificare.
    3. sequenziale: si imputa una variabile alla volta: per ogni variabile si calcola il range dei valori ammissibili, quindi si cerca nel serbatoio e, se esiste, si seleziona un record donatore con valore compreso nel corrispondente range.

- marginale - se non si è riusciti a trovare un donatore si passa all'imputazione marginale utilizzando le distribuzioni marginali semplici (imputazione forzata).

Si possono modificare i seguenti parametri:

- Tipo di imputazione - impostato all'esecuzione di tutti e tre i tipi di imputazione. Si può ridurre la sequenza dei tentativi di imputazione alle sole imputazioni allargata e sequenziale oppure alla sola imputazione sequenziale.
- Statistiche - Con NO si elimina la produzione delle statistiche di correzione.
- Numero record serbatoio donatori -impostato a 50000. Può essere diminuito.
- Numero massimo di donazioni - impostato a 99999 (numero di donazioni senza limite). Si può diminuire e serve a limitare le donazioni che un singolo record esatto può effettuare.

I file usati in questa fase sono:

- PARM con i parametri per l'imputazione
- TABVARF con i domini separati in classi per ogni variabile
- VARFIX con le variabili definite fisse
- MAXICE insieme completo in forma binaria
- SYSCHK con i contatori dei record errati ed esatti
- ESATTI file dei record esatti
- ERRATI file dei record errati
- CORRETTI file dei record imputati
- INCORRETTI eventuali record non correggibili
- SYSIMP contatori delle imputazioni effettuate per tipo di imputazione
- STATIS statistiche di correzione
- FREQUEN frequenze dei dati grezzi
- LISTAIMP lista delle variabili imputate per record

4. IMPUTAZIONE DA DONATORE (RIDA)

4.1 La metodologia del donatore

Le funzioni di ConcordJava nell'approccio di correzione tramite donatore sono uguali a quelle di RIDA (Ricostruzione delle Informazioni con Donazione Automatica): la correzione di un file di dati di qualsiasi tipo avviene tramite la tecnica del donatore di distanza minima. Nel seguito sono descritti i principi su cui la tecnica si basa, nonché brevemente i passi che l'utente deve eseguire per rendere operativo il sistema.

Rappresentazione dei dati.

Sia data una matrice di dati X, formata da n unità e k variabili di tipo qualsiasi. Le unità rappresentano i vettori-riga, le variabili i vettori-colonna. Le variabili sono di tipo qualsiasi.

Dal punto di vista della archiviazione elettronica della informazione, la matrice dei dati X è contenuta in un file, costituito da un insieme di record, ognuno rappresentante una unità, e contenente un numero di campi pari al numero di variabili (da ora in poi useremo il termine record o unità come sinonimi).

Dividiamo in due gruppi le k variabili:

variabili affette da errore (in numero di h<k);

variabili esatte (in numero di k-h).

Supponiamo di sottoporre ad un processo di controllo ogni record, in modo che ognuno degli h campi corrispondenti alle variabili affette da errore contenga o un flag di errore o un valore esatto. Il file risulta diviso in due:

insieme dei record totalmente esatti;

insieme dei record che presentano almeno un flag di errore.

Costruzione della metrica delle distanze.

Proponiamoci ora di misurare la distanza tra due unità, rispetto alle variabili esatte. A questo scopo è necessario introdurre una metrica per ogni tipologia di variabile (si veda Abbate, 1996 a questo proposito).Sia quindi d la distanza tra due unità, misurata rispetto ad una variabile:

a) Variabile qualitativa sconnessa.

Si pone d=0 se le unità presentano la stessa modalità,

d=1 se la modalità è diversa.

b) Variabile ordinata con m modalità.

Si pone d=0 se sulle due unità è stata rilevata la stessa modalità,

d=1 se le modalità sono adiacenti,

d=2 se tra di esse ce n'è una sola, e così via fino a

d=m-1, se le due modalità sono agli estremi opposti.

Per normalizzare la distanza d tra 0 ed 1, essa viene divisa per il suo massimo m-1.

c) Variabile qualitativa gerarchica o telescopica.

La distanza è valutata sulla base del numero di cifre differenti a partire dall'ultima.

Si pone d=0 se tutte le cifre sono uguali,

d=1 se è diversa solo l'ultima cifra,

d=2 se sono diverse soltanto l'ultima e la penultima,

d=k se è diversa la prima cifra della variabile dove K è il numero totale delle cifre.

Per normalizzare la distanza d tra 0 ed 1, essa viene divisa per K.

d) Variabile quantitativa.

Sia X1 il valore assunto dalla variabile X nella prima unità, X2 nella seconda. Poniamo d=abs(X1- X2). La distanza è essere resa variabile tra 0 e 1 dividendola per il suo massimo, pari alla differenza tra i valori massimo (Xmax) e minimo (Xmin) della variabile X presenti nel file.

Formalizzazione della funzione di distanza mista ponderata.

Assegnata una matrice di dati, presentante k-h variabili non affette da errore, definiamo distanza mista ponderata D tra due generiche unità una espressione del tipo:

$$D = \sum_{i=1}^{r} W_i D_i$$

dove D_i è la distanza tra le due unità rispetto alla variabile i, misurata con una delle espressioni di cui sopra e W_i è un numero reale positivo che rappresenta l'importanza assegnata alla variabile i nel calcolo della distanza. Le r variabili sono scelte tra le k-h quelle non affette da errore.

Chiamiamo variabili di accoppiamento o di matching le r variabili scelte per il calcolo della distanza.

Scelta dell'unità donatrice.

Data un'unità affetta da errore nella variabile k si vuole trovare l'unità esatta posta alla distanza minima. Essa è detta unità donatrice, perché il valore della variabile k relativo ad essa è "donato" all'unità affetta da errore. L'insieme della unità tra le quali è scelta l'unità donatrice è detto serbatoio dei donatori. Il serbatoio dei donatori può essere costruito in due modi:

selezionando le unità esatte rispetto alla sola variabile k;

selezionando le unità esatte rispetto a tutte le variabili.

Nel primo caso si usa un diverso serbatoio per ogni variabile da errata, nel secondo caso si utilizza un serbatoio unico per tutte le variabili affette da errore. La prima procedura è utile quando si desidera disporre di serbatoi di donatori relativamente numerosi per ogni variabile da correggere.

Questa scelta deve essere effettuata e realizzata prima di utilizzare ConcordJava.

La scelta dell'unità donatrice è ulteriormente affinabile scegliendo, nell'insieme delle variabili non affette da errore e non usate come variabili di accoppiamento, delle variabili dette di strato. Dopo aver formato il serbatoio dei donatori in uno dei due modi di cui sopra, si seleziona l'unità donatrice tra quelle che inoltre, rispetto alle variabili di strato, presentano le stesse modalità dell'unità affetta da errore. L'uso di variabili di strato implica l'accettazione della possibilità di non avere donatori idonei per quell'unità.

Funzione di distanza mista ponderata corretta.

Possiamo introdurre un perfezionamento alla distanza mista ponderata sopra introdotta, per penalizzare l'unità del serbatoio che è già stata utilizzata nella donazione. Ridefiniamo la distanza D come:

$$D = \sum_{i=1}^{r} W_i D_i + kp,$$

dove k è il numero di volte per cui l'unità è stata precedentemente utilizzata, p è un fattore di penalità. Questa espressione più completa è adottata da RIDA, che richiede che p sia un numero intero.

Ponderazione delle variabili di matching.

Sono molte le tecniche possibili di ponderazione delle variabili di matching. Le applicazioni finora realizzate nell'interno dell'istituto hanno utilizzato il criterio del χ^2 (si veda [1]). Esso si applica nel seguente modo:

si misura la connessione tra la variabile affetta da errore e quelle esatte tramite l'indice χ^2. Il valore dell'indice dipende dal numero di celle della tabella di contingenza. Poiché bisogna confrontare il valore dei χ^2 ottenuti, per renderli confrontabili occorre o riclassificare in modo opportuno almeno la variabile da correggere, se di tipo quantitativo, in modo da ottenere tabelle di contingenza di dimensioni omogenee, oppure dividere direttamente il valore del χ^2 per il numero di gradi di

libertà, che è pari al prodotto tra il numero delle righe e delle colonne della tabella di contingenza diminuiti entrambi di uno;

l'utilizzatore del metodo deve esaminare criticamente i valori di χ^2 così ottenuti, eventualmente divisi per il numero dei gradi di libertà: le variabili non affette da errore che presentano il valore più alto sono le migliori candidate ad essere variabili di strato, quelle con valore immediatamente inferiore possono diventare variabili di matching. L'utilizzatore del metodo deve usare i valori come supporto a una decisione che tiene anche conto della sua conoscenza dell'indagine.

La scelta delle variabili di strato deve tener conto anche del fatto che all'aumentare del loro numero, aumenterà la selettività nell'ambito del serbatoio dei donatori, ma aumenterà anche la probabilità di non trovare il donatore.

4.2 Definizione dei parametri

Per la definizione dei parametri la procedura richiama il file DVARDOM.progetto in un editor.

Esempio di schede parametri
Gli strati sono costruiti in base alla provincia e al comune.
Le distanze (variabili di match) fra i record sono calcolate per il cap , l'indirizzo e il codice strada.
La variabile da donare e' la sezione di censimento.

```
.V S01 P=10 L=3 T=X
.V S02 P=13 L=3 T=X
.V M01 P=16 L=50 T=S W=0.3
.V M02 P=56 L=5 T=X W=0.2
.V M03 P=61 L=5 T=X W=0.5
.V A01 P=55 L=7 T=X C=* X=1
.U 2
.G
```

Regole di scrittura delle schede parametro
Le righe significative devono avere come primo carattere diverso da spazio il punto (.) seguito dalla lettera che identifica il tipo di scheda. Tutte le righe che cominciano con un carattere diverso dal punto sono considerate commenti.
I gruppi che identificano i parametri devono essere separati da spazi. • All'interno del gruppo non devono esserci spazi.

Variabili di imputazione

Individuano le variabili che devono essere corrette nel caso dei record errati, o che devono fornire il valore esatto nel caso dei donatori.
Si definisce una variabile A per ogni variabile da correggere.

.V	Identifica il tipo di scheda

ANN	dove NN è un numero intero da 01 a 24 che identifica la variabile di imputazione.
P	Posizione della variabile. Deve essere la stessa su donatore e ricevente.
L	Lunghezza della variabile. Deve essere la stessa su donatore e ricevente. Se il tipo di correzione è 2 (calcolo), le variabili numeriche (tipo N, C) decimali devono essere definite nella forma L=LI.LD in cui LD= numero decimali e LI= lunghezza della parte intera. Se è presente il punto decimale, la lunghezza della parte intera deve comprendere anche la posizione del punto. In output i risultati sono scritti sempre senza il punto decimale esplicito
T	Tipo variabile che può essere • X = per i codici • N = per i numeri • C = variabili classificate Se il tipo è uguale C (variabile classificata) la lunghezza deve essere intera e deve esistere almeno una scheda .C.
C	Carattere di riempimento che identifica il dato errato che si vuole corretto (default spazio). Il carattere deve essere ripetuto per tutta la lunghezza della variabile. Per indicare che la variabile deve essere sempre corretta: *.
X	Tipo di correzione da operare 1. la correzione consiste nello spostamento della corrispondente variabile del record donatore. 2. la correzione è il risultato di un calcolo. Se X=2 devono esistere almeno una scheda .E ANN e la scheda .F ANN. Se X=1 il tipo T deve essere X (alfanumerico) o N (numerico). Se X = 2 T deve essere N.

Esempio:
Si vuole correggere la variabile A01, posizione 30, lunghezza 3 individuata nei record errati dai caratteri BBB, sostituendo il campo con il corrispondente valore del record donatore.
Nel file "dvardom.dat" si deve scrivere una riga con i seguenti parametri:

```
.V A01 P=30 L=3 T=X X=1 C=B
```

Variabili di strato

Le variabili di strato si utilizzano per individuare gruppi di record, relativamente numerosi, che definiscono insiemi non simili fra loro. La ricerca del donatore o dei record simili si esegue all'interno degli strati, limitando il numero dei confronti.

Si scrive una scheda per ogni variabile che identifica uno strato.

.V	Identifica il tipo di scheda
SNN	dove NN è un numero intero da 01 a 24 che identifica la variabile di strato.
P	Posizione.
L	Lunghezza.
T	Tipo variabile che può essere

	• X = per i codici • T = variabili telescopiche (es. codici ateco). • C = variabili classificate Se il tipo è uguale C (variabile classificata)la lunghezza deve essere intera e deve esistere almeno una scheda .C.

Variabili di match

Le variabili di match s'utilizzano per calcolare la funzione di distanza mista minima per tutti i record di uno strato. Il donatore sarà il record più vicino al record errato, in altre parole quello con distanza minima. Si scrive una scheda per ogni variabile utilizzata nel calcolo della distanza

.V	Identifica il tipo di scheda
MNN	dove NN è un numero intero da 01 a 24 che identifica la variabile di match.
P	Posizione.
L	Lunghezza. Le variabili numeriche (tipo N, C) decimali devono essere definite nella forma L=LI.LD in cui LD= numero decimali e LI= lunghezza della parte intera. Se è presente il punto decimale, la lunghezza della parte intera deve comprendere anche la posizione del punto.
T	Tipo variabile che può essere • X = per i codici • T = variabili telescopiche (es. codici ateco). • C = variabili classificate • N = variabili numeriche • S = variabili stringa • J = parola (algoritmo jaro) Se il tipo è uguale C (variabile classificata)la lunghezza deve essere intera e deve esistere almeno una scheda .C.
W	Il peso da utilizzare nel calcolo della distanza. (Default W=1: Variabile non pesata).

Variabili di classificazione

Per ogni variabile SNN, MNN con tipo=C (classificata) devono esistere una o più schede .C, in cui sono riportati gli estremi superiori delle classi.

Gli estremi delle classi devono essere crescenti. L'estremo inferiore delle classi intermedie si calcola come estremo superiore classe precedente più 1.
La prima classe è aperta inferiormente.
L'ultima classe è aperta superiormente.

Ogni scheda .C può contenere al massimo 10 valori.

Variabili per il calcolo

Scheda che indica le variabili da utilizzare per la correzione delle variabili ANN. Deve esisterne almeno una se tipo correzione T=2.

.E	Identifica il tipo di scheda
E/D	Indica se la variabile deve essere presa dal file degli errati (E) o dei donatori (D).
NN	NN è un numero intero da 01 a 24 che identifica la variabile di imputazione.
P	Posizione.
L	Lunghezza. Le variabili numeriche decimali devono essere definite nella forma L=LI.LD in cui LD= numero decimali e LI= lunghezza della parte intera. Se è presente il punto decimale, la lunghezza della parte intera deve comprendere anche la posizione del punto.
T	Tipo variabile che può essere solo N (Numerica).

Formula di calcolo

Individua la formula di calcolo del valore da imputare nella variabile ANN.
La formula di calcolo può contenere gli operatori aritmetici +,-,*,/ e ^ per l'elevazione a potenza.
Gli operandi possono essere costanti o variabili identificate dal nome delle schede .E.
Si possono essere utilizzate le parentesi ( ) per modificare l'ordine di esecuzione delle operazioni.

Altre variabili

.U	Rappresenta il fattore di penalizzazione per record gia' utilizzati. Può essere un numero intero o decimale. Se non esiste allora si assume U=0 ovvero non si penalizzano i record già utilizzati.
.R	Rappresenta il numero massimo di volte in cui uno stesso record puo' essere utilizzato. Deve essere un numero intero. Se non esiste allora si assume R=0 ovvero riutilizzo illimitato.
.L	Rappresenta il limite massimo della distanza tra due record. Può essere un numero intero o decimale. Se non esiste allora si assume L=0 ovvero la distanza può essere grande quanto si vuole.
.G	Se presente indica che si vuole un scelta casuale del donatore fra tutti quelli con valore minimo della funzione di distanza.

4.3 Funzioni

Controllo

Controllo sintattico del file dei parametri.

Selezione dei file di input

I file dati di input necessari per il passo di imputazione sono:

1. Esatti: file da cui prendere i donatori;
2. Errati: file dei record da imputare.

Esecuzione del programma

Dopo la fase di definizione e possibile lanciare il programma. In questa fase girano 5 programmi.

1. dona1 per il controllo dei parametri;
2. dona2 per la creazione di file temporanei ;
3. dona3 per il calcolo delle distanze fra i record e la scelta dei donatori;
4. dona4 per l'imputazione;
5. dona5 per la scrittura di report quantitativi.

5. STRUMENTI

Gestione file piatti

- **Inserimento identificatore**
  Creazione di un identificatore numerico univoco per ogni record.
  I parametri da inserire sono:
  - **Input:** Nome del file di input
  - **Output:** Nome del file di output
  - **Posizione:** Posizione di partenza per l'identificatore. Per default (999999) il campo viene aggiunto alla fine del record
  - **Lunghezza:** Lunghezza dell'identificatore.

- **Ordinamento di un file**
  Per ordinare un file posizionale, Sono permesse al massimo 3 chiavi di sort.
  I parametri da inserire sono:
  - **Input:** Nome del file di input
  - **Output:** Nome del file di output
  - **Posizione:** Posizione di partenza la chiave di sort.
  - **Lunghezza:** Lunghezza della chiave di sort.

- **Accodamento file**
  Per copiare due file esterni in un unico file.
  I parametri da inserire sono:
  - Nome del primo file del input
  - Nome del secondo file del input
  - Nome del file di output.

Conversione file

- **Da formato fisso a delimitato**
  Trasforma un file ascii posizionale in un file delimitato.
  I parametri da inserire sono:
  - **Input:** Nome del file da convertire
  - **Tracciato record:** Nome del file contenente il tracciato record.
    Il file contenente il tracciato deve essere un file delimitato da tabulatore o spazio. Per ogni variabile devono essere indicati il nome, la posizione di inizio e la lunghezza del campo.
    Le righe che iniziano con un asterisco (*) sono considerate commenti.
  - **Output:** Nome del file delimitato
  - **Separatore:** A scelta fra punto e virgola (;), virgola (,) e tabulatore (tab).

- **Da formato delimitato a fisso**
  Trasforma un file ascii da formato delimitato a formato posizionale a campi fissi come richiesto da ConcordJava. I campi numerici allineati a destra e riempiti con zeri.
  I parametri da inserire sono:
  - **Input:** Nome del file da convertire
  - **Tracciato record:**Opzionale. Nome del file contenente il tracciato record.
    Il file contenente il tracciato deve essere un file delimitato da tabulatore o spazio. Per ogni variabile devono essere indicati il nome, la posizione di inizio e la lunghezza del campo.
    Le righe che iniziano con un asterisco (*) sono considerate commenti.
    Se il file non Ã¨ presente viene generato automaticamente con nome uguale al nome del file di input a cui si aggiunge il suffisso *.trk* . La lunghezza delle variabili Ã¨ uguale alla lunghezza massima della variabile nel file
  - **Output:** Nome del file trasformato in formato fisso
  - **Separatore:** A scelta fra punto e virgola (;), virgola (,) e tabulatore (tab).