E-TIVITY 2 – VINCERÒ LA PARTITA DI TENNIS?

Con questo lavoro mi sono posto l'obiettivo di creare un programma che, in funzione di tre parametri definiti dall'utente, riesca a restituire come output una percentuale di vittoria o sconfitta di un ipotetico match di tennis.

Al tal fine, ho preso come dataset di partenza i risultati dell'AO maschile del 2013 messo a disposizione dal portale UC Irvine Machine Learning Repository[1] e, prima di passare all'elaborazione su vscode, ho effettuato un'analisi e creazione della rete su Excel[2] così da prendere confidenza sia con i calcoli probabilistici ed il test chi-quadro, che con la definizione stessa di "rete bayesiana".

Il punto di partenza, o per meglio dire "l'ipotesi iniziale", è stato chiedersi che relazione ci fosse tra il numero di ace fatti da un giocatore in una partita con il risultato finale (Vittoria = Si | Vittoria = No), alla quale, poi, ne sono susseguite delle altre che, in funzione dell'effettiva relazione statistica significativa o meno, sono state inserite o scartate nell'elaborato finale in Python. Infine, grazie alla libreria "pgmpy" è stata creata una rete bayesiana, riferita al modello definito precedentemente, in grado di rispondere alla domanda dell'utente.

Di seguito, sono riportate nel dettaglio le operazioni effettuate:

1. **Scelta del dataset**

   Un focus particolare merita la modellazione del dataset attuata su Excel per una corretta analisi delle variabili scelte. Infatti, sono state dapprima create due tabelle distinte per ciascun giocatore iscritto al torneo, con i relativi dati d'interesse, per poi effettuare un'unione delle due in una definitiva per i relativi calcoli statistici. Nota Bene, la differenza dei valori individuali tra Excel/Python risiede esclusivamente nel conteggio dei dati[3] ma non nel risultato statistico finale.

2. **Formulazione di un'ipotesi iniziale**

   Sono partito da una **prima Ip_0**: "*Chi realizza un numero maggiore di ace ha più probabilità di vincere la partita?*". A Questa, sono susseguite, in ordine, le seguenti:

[1] https://archive.ics.uci.edu/dataset/300/tennis+major+tournament+match+statistics

[2] Allegato anch'esso alla consegna.

[3] In Python sono stati accorpati per giocatore, con un totale di 256, in Excel, invece, sono state riportate tutte le singole partite disputate dai giocatori, anche nei turni successivi, per un numero complessivo di 2458 record.

- *"Chi ha un % di first serve[4] più alto vince più partite? C'è una relazione tra aces fatti e FirstServe%?"*
- *"Chi fa più vincenti al servizio, a prescindere che sia una prima o una seconda palla, ha più probabilità di vincere?"*
- *"Chi crea più Break Point ha maggiore possibilità di vincere la partita? La probabilità varia in funzione della capacità del giocatore di trasformarli in Break Point Vincenti?"*

3. **Costruzione delle tabelle di contingenza**

   Prendendo come punto di partenza il conteggio del valore massimo, minimo e della media delle singole variabili, sono stati definiti dei "range" di appartenenza per ognuna (['Basso', 'Medio', 'Alto']), divenendo il punto di partenza per la generazione delle tabelle dei dati osservati, del valore neutro (test/attesi) e per il successivo calcolo del chi-quadro.

4. **Test del chi-quadro**

   Infine, con il test del chi-quadro sono state definite le seguenti relazioni che, a seconda della valenza statistica o meno, sono state inserite/escluse dalla rete.

RELAZIONE	**RISULTATO STATISTICO**
Aces → Vittoria	**Significativa**: inserita nella rete
FirstServe% → Vittoria	Non significativa: esclusa
FirstServe% → Aces	Non significativa: esclusa
ServeWon → Vittoria	Non significativa: esclusa
BreakPointCreati → BreakPointVinti	**Significativa**: inserita nella rete
BreakPointVinti → Vittoria	**Significativa**: inserita nella rete
BreakPointCreati → Vittoria	**Significativa**: ma assorbita nel legame causale precedente

Aces → **Vittoria** ← *BPW* ← BPC

5. **Implementazione su Python**

   La differenza sostanziale riscontrata tra Excel e Python è la semplicità attraverso cui ho ottenuto i medesimi risultati di Excel tramite l'utilizzo di specifiche librerie (in particolare **scipy.stats** che permette di calcolare automaticamente le tabelle dei valori attesi, il chi-quadro, in funzione dei gradi di libertà, e il p-value). Inoltre, con un'altra libreria, denominata **pgmpy**, ho implementato la possibilità di far inserire all'utente

[4] Nel tennis, la percentuale di prime di servizio vincenti indica il numero di punti realizzati dal giocatore attraverso la prima palla di servizio. A differenza dell'ace che è un punto diretto, senza che l'avversario tocchi la palla, un punto attraverso un servizio vincente è conteggiato anche a seguito di un tentativo di risposta del giocatore avversario.

finale i 3 parametri di riferimento per ottenere una risposta personalizzata in funzione della richiesta.

Conclusione

Dall'analisi effettuata si è riscontrata un'alta probabilità di vincita della partita nel caso in cui un giocatore dovesse avere un buon range di ace ma, ciò che maggiormente è risaltato è che la chiave per portarsi a casa il risultato risiede nell'abilità del giocatore di trasformare le palle break create in game vincenti.