

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

Scuola di Scienze
Corso di Laurea Magistrale in Fisica

**SCRIPTING AUTOMATION E MODELLI BAYESIANI:
APPLICAZIONI CLINICHE IN RADIOTERAPIA E
SVILUPPO DI TECNICHE INNOVATIVE PER
ADAPTIVE RADIATION THERAPY**

Relatore:

Prof. GIUSEPPE BALDAZZI

Presentata da:

CLAUDIO VECCHI

Correlatore:

Dott. GABRIELE GUIDI

Sessione III

Anno Accademico 2012/2013

Alla mia famiglia e agli amici

Abstract

La presente ricerca consiste nel validare ed automatizzare metodiche di *Adaptive Radiation Therapy* (ART), che hanno come obiettivo la personalizzazione continua del piano di trattamento radioterapico in base alle variazioni anatomiche e dosimetriche del paziente. Tali variazioni (casuali e/o sistematiche) sono identificabili mediante l'utilizzo dell'imaging diagnostico.

Il lavoro svolto presso la struttura di Fisica Medica dell'Azienda Ospedaliera Universitaria del Policlinico di Modena, si inserisce in un progetto del Ministero della Salute del bando Giovani Ricercatori dal titolo: "*Dose warping methods for IGRT and ADAPTIVERT: dose accumulation based on organ motion and anatomical variations of the patients during radiation therapy treatments*".

Questa metodica si sta affermando sempre più come nuova opportunità di trattamento e, per tale motivo, nasce l'esigenza di studiare e automatizzare processi realizzabili nella pratica clinica, con un utilizzo limitato di risorse.

Si sono sviluppati script che hanno permesso l'automazione delle operazioni di *Adaptive* e deformazioni, raccogliendo i dati di 51 pazienti sottoposti a terapia mediante Tomotherapy.

L'analisi delle co-registrazioni deformabili delle strutture e delle dosi distribuite, ha evidenziato criticità del software che hanno reso necessario lo sviluppo di sistemi di controllo dei risultati, per facilitare l'utente nella revisione quotidiana dei casi clinici.

La letteratura riporta un numero piuttosto limitato di esperienze sulla validazione e utilizzo su larga scala di questi *tools*, per tale motivo, si è condotto un esame approfondito della qualità degli algoritmi elastici e la valutazione clinica in collaborazione di fisici medici e medici radioterapisti.

Sono inoltre stati sviluppati principi di strutturazione di reti Bayesiane, che consentono di predire la qualità delle deformazioni in diversi ambiti clinici (H&N, Prostata, Polmoni) e coordinare il lavoro quotidiano dei professionisti, identificando i pazienti, per i quali sono apprezzabili variazioni morfo-dosimetriche significative.

Da notare come tale attività venga sviluppata automaticamente durante le ore notturne, sfruttando l'*automation* come strumento avanzato e indipendente dall'operatore.

Infine, il forte sviluppo, negli ultimi anni della biomeccanica applicata al movimento degli organi (dimostrato dalla numerosa letteratura al riguardo), ha avuto come effetto lo sviluppo, la valutazione e l'introduzione di algoritmi di deformazione efficaci. In questa direzione, nel presente lavoro, si sono analizzate quantitativamente le variazioni e gli spostamenti delle parotidi, rispetto all'inizio del trattamento, gettando le basi per una proficua linea di ricerca in ambito radioterapico.

Sommario

Introduzione	1
1 Automazione informatica nella moderna radioterapia	3
1.1 Radioterapia e tecniche moderne	3
1.2 Cenni di fisica delle radiazioni	7
1.3 La Tomotherapy in breve	12
1.4 TPS (Treatment Planning System)	13
1.5 Imaging biomedicale	14
1.6 Lo standard DICOM RT	17
1.7 Progetto e metodo di lavoro	19
2 Deformazioni e scripting	24
2.1 I volumi di interesse	24
2.2 Rigid registration	26
2.3 Deformable registration	35
2.4 Mapping structures: misure sperimentali	45
2.5 Dose deformation	48
2.6 Estrazione dei dati statistici	52
2.7 Scripting automation	57
2.7.1 Script ART	60
2.7.2 Script di controllo	62
2.7.3 Script generali	63
3 Risultati e criticità dell'automazione	64
3.1 Tempi a confronto	64
3.2 Analisi dei risultati	70
3.3 Limiti dell'automazione	74
4 Prime valutazioni biomeccaniche basate su deformazioni d'organo	78
4.1 Deformazioni delle parotidi	79
4.2 Analisi dei risultati	84
5 Valutazione Bayesiana nella pratica clinica delle deformazioni d'organo	88
5.1 Probabilità: concetti chiave	88

5.2	Le reti Bayesiane	90
5.3	Applicazione pratica.....	93
5.3.1	H&N.....	99
5.3.2	ADK prostatico	103
5.3.3	SBRT polmonare.....	107
6	Conclusioni	111
	Legenda	112
	Bibliografia.....	114
	Sitografia	118
	Appendice A	121
	Appendice B	140
B.1	Head & Neck.....	141
B.2	ADK prostatico.....	149
B.3	SBRT polmonare.....	157

Ringraziamenti

Il relatore, Prof. Giuseppe Baldazzi, per avermi dato l'opportunità di svolgere la tesi presso l'Azienda Ospedaliera Universitaria del Policlinico di Modena.

Il servizio di Fisica Sanitaria: il direttore Dott.ssa Tiziana Costi; il responsabile nonché correlatore della tesi Dott. Gabriele Guidi, per il supporto costante, la disponibilità di tempo e per la sua esperienza risultata veramente utile nello svolgimento di questa tesi.

Dott. Giovanni Gottardi per il tempo dedicato alla correzione della tesi e tutti i componenti del reparto per l'accoglienza, la gentilezza e l'assistenza in questa esperienza.

Il reparto di Radioterapia, diretto dal Dott. Filippo Bertoni e tutti i medici che hanno contribuito direttamente o indirettamente al raggiungimento degli obiettivi della tesi.

I colleghi di Tecnologie Avanzate S.r.l.: Dott. Guido Catolla, Dott. Ing. Giancarlo Cangialosi, Dott.ssa Eleonora Lanzi.

I compagni di studi, in particolare Nicola Maffei, per la cooperazione nel progetto di tesi e di ricerca.

Famiglia e amici che da sempre mi hanno incoraggiato, capito e motivato nel proseguimento degli studi.

Introduzione

Ricerca scientifica ed evoluzione tecnologica hanno consentito l'impiego di strumenti informatici sempre più avanzati in Radioterapia Oncologica. Le possibilità di *automation* che tali strumenti offrono, hanno permesso di semplificare le attività di raccolta, elaborazione e trasmissione di dati e immagini, riducendo notevolmente i tempi necessari, rispetto all'approccio prevalentemente manuale. Congiuntamente, l'utilizzo di computer e algoritmi software avanzati ha incrementato l'accuratezza e la qualità complessiva dei trattamenti radioterapici, fino allo sviluppo di tecniche moderne che sfruttano l'*Image Guided Radiation Therapy* (IGRT).

Il presente lavoro ha come obiettivo l'automatizzazione della tecnica ART mediante script e lo sviluppo di una rete Bayesiana predittiva elementare, applicabile nella pratica clinica, per valutare la qualità dei risultati ottenuti via software, a fronte degli standard ottenibili dall'esperienza clinica degli operatori.

Lo studio della nuova metodica di *Adaptive* viene effettuato mediante *Treatment Planning System* (TPS), consentendo una personalizzazione dei trattamenti sempre più efficace. Quest'ultima, costituisce anche un'area di ricerca fondamentale per il futuro, per le sue prospettive di applicazione pratica nella tecnica nel processo radioterapico, verificando efficacia ed efficienza nel campo delle deformazioni d'immagini, degli organi e delle dosi erogate mediante la tecnica di Tomotherapy.

Un ulteriore obiettivo è la validazione di algoritmi di deformazioni mediante fantocci dinamici, per definire la tecnica utilizzabile nella raccolta e post-elaborazione dei trattamenti sui pazienti. La necessità di rendere clinica tale metodica, rende fondamentale l'automatizzazione per operazioni, atte a ridurre drasticamente i tempi di elaborazione. La validazione dei processi e delle deformazioni mediante reti Bayesiane predittive può infatti fornire al medico informazioni, a priori o durante la terapia, su possibili variazioni anomale nell'anatomia dei pazienti e identificare in modo probabilistico la consistenza dei risultati ottenibili in modalità manuale.

Le variazioni morfologiche possono essere analizzate e simulate attraverso metodiche di biomeccanica, disciplina che sta assumendo un ruolo sempre più centrale nella radioterapia, per la soluzione di problemi riguardanti la meccanica del corpo umano. In

particolare, la creazione di modelli realistici del movimento di organi come le parotidi, i polmoni, il retto, la vescica ecc., unitamente allo sviluppo della robotica medica, può costituire uno degli obiettivi in vista del prossimo futuro, nell'ambito del raggiungimento dell'ART in tempo reale (*real time ART*).

1 Automazione informatica nella moderna radioterapia

1.1 Radioterapia e tecniche moderne

La radioterapia è una delle discipline mediche che impiega radiazioni ionizzanti per il trattamento di neoplasie. Le moderne tecnologie (LINAC, Tomotherapy ecc.) rendono possibile l'utilizzo delle radiazioni ionizzanti per la cura dei tumori.

Grazie all'evoluzione tecnologica, ai progressi scientifici nel campo delle radiazioni e alle loro applicazioni mediche, nonché grazie all'avvento negli ultimi decenni delle tecnologie informatiche si può parlare oggi di "*Information Technology*" (IT) in radioterapia.

Le tecniche oggi utilizzate sono:

- **3D-Conformal Radiation Therapy (3D-CRT):** costituisce oggi lo standard nei trattamenti radioterapici. La dose viene distribuita in modo "conformato" al volume definito dal Medico Radioterapista (MR), mediante immagini di tomografia computerizzata (TC) o risonanza magnetica (RMN) o tomografia a emissione di positroni (PET). Questa tecnica fa uso di lamelle schermanti (MLC) posizionate indipendentemente fra di loro in maniera automatica e sotto controllo informatico (Record&Verify - R&V). Si può, quindi, modificare il profilo del fascio radiante, in modo da salvaguardare i tessuti sani circostanti e concentrare maggior dose sui target con una conseguente riduzione degli effetti collaterali.
- **Intensity-Modulated Radiation Therapy (IMRT):** è un'evoluzione delle tecniche conformazionali e si basa sull'utilizzo dei collimatori multilamellari con lo scopo di erogare dosi variabili su volumi irregolari. I campi, con conformità diversa, variano dinamicamente per modulare l'intensità della dose erogata al volume da trattare. I vantaggi di questa tecnica sono elevati nel caso della presenza di volumi irregolari (es. concavi) e complessi in stretta contiguità con organi critici.

Solitamente vengono selezionati i parametri desiderati, relativi alla distribuzione di dose nel volume bersaglio e ai limiti di dose negli organi a rischio (mediante

software di inverse treatment planning) che determinano la distribuzione di dosi prescritte dal MR.

- Volumetric Modulated Arc Therapy (VMAT): è una tecnica analoga all'IMRT ma con movimentazione rotazionale del gantry che consente di ottenere distribuzioni analoghe di dose e tempi ridotti di terapia.
- Stereotactic radiation therapy: permette di somministrare un'elevata dose di radiazioni ad un volume circoscritto, in una singola seduta o in poche sedute, con risparmio dei tessuti sani circostanti. Con l'ausilio di apparati di immobilizzazione può essere eseguita con acceleratori lineari su bersagli sia cranici che extracranici. Ne esistono di 2 tipi:
 1. Stereotactic radiosurgery (RS): si riferisce ad un singolo trattamento o più trattamenti stereostatici dell'area encefalica.
 2. Stereotactic body radiation therapy (SBRT): si riferisce ad un singolo trattamento o più trattamenti stereostatici del corpo.
- Intraoperative radiation therapy (IORT): questa tecnica eroga un fascio di elettroni (nel letto tumorale dopo la rimozione chirurgica del tumore) attraverso un particolare acceleratore lineare dedicato, posto in sala operatoria.
- Image guided radiation therapy (IGRT): è una tecnica radioterapica guidata dalle immagini, cioè la lesione tumorale e gli OARs (Organ At Risks) sono individuati e ricostruiti nello spazio, tridimensionalmente. Grazie all'imaging diagnostico, è possibile (risparmiando gli organi a rischio) erogare fasci di radiazioni che depositano dose nei target e valutare la morfologia del paziente giorno per giorno e in tempo reale (es. CBCT (Cone Beam Computed Tomography), MVCT (Mega Voltage Computed Tomography), US (Ultra Sound)).
- Tomotherapy: rappresenta una tecnologia recente, che integra l'imaging diagnostico (MVCT), la pianificazione (mediante sistema di pianificazione inverso) e l'erogazione della terapia in un unico processo e in un'unica apparecchiatura. Utilizza una tecnica IMRT con erogazione elicoidale della dose, analogamente a quanto avviene in ambito diagnostico nell'esecuzione di una TC spirale.
- Adrotherapy: è una tecnica radioterapica che utilizza radiazioni prodotte da protoni/ioni carbonio, cioè particelle ad alto LET. Per la maggiore efficacia biologica e le modalità particolari di cessione di energia, tali trattamenti hanno indicazioni cliniche molto selettive. L'adroterapia richiede l'utilizzo di ciclotroni e

sincrotroni e attualmente vi è un grande interesse per tali terapie, specie in ambito di sviluppo tecnologico.

Gli ultimi anni sono stati contraddistinti dallo sviluppo di strumenti informatici dedicati all'utilizzo in Radioterapia. L'impiego di questi strumenti, ha portato ai seguenti vantaggi:

- Le attività risultano semplificate.
- I tempi sono ridotti, nonostante la complessità delle tecniche.
- La precisione/accuratezza risulta migliorata.

L'integrazione dei processi di trattamento radioterapico con tecniche di acquisizione di immagini rende possibile la visualizzazione dell'anatomia del paziente subito prima di somministrare ogni singola frazione di dose, ottenendo quindi in tempo reale una precisa localizzazione del volume bersaglio e degli organi critici.

Questa modalità terapeutica, definita sopra come IGRT, ha il potenziale di assicurare che, per ciascuna frazione erogata, la posizione del target sia esattamente la stessa di quanto previsto nella pianificazione del trattamento.

Lo scopo di questa tecnica è infatti procedere on-line ad un'eventuale intervento correttivo di ri-centratura e quindi ridurre la dose agli organi a rischio, concentrando alti gradienti di dose nelle regione neoplastiche.

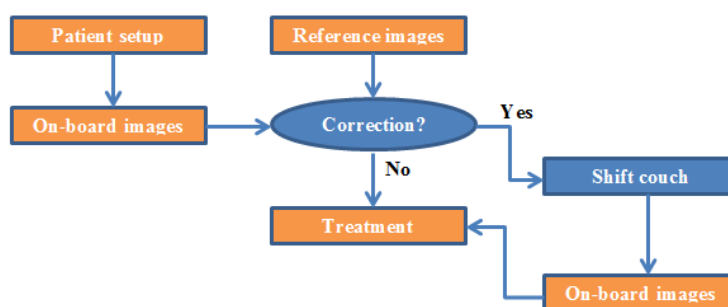


Figura 1.1 - Workflow IGRT.

La conoscenza in tempo reale dell'anatomia della regione trattata durante il trattamento rende possibili interventi "adattativi", in relazione al riscontro giornaliero di eventuali modificazioni della lesione neoplastica quali conseguenza di risposta al trattamento. Questi interventi adattativi, fanno parte delle tecniche conosciute come ART, le cui caratteristiche principali sono la possibilità di verifica, riposizionamento del paziente e l'eventuale modifica del piano di trattamento.

Il flusso di processo o *workflow* per un trattamento radioterapico convenzionale si può schematizzare:

1. *Treatment simulation*: definizione del target e degli OAR sulle CT del paziente.
2. *Treatment planning*: esecuzione del piano di trattamento mediante TPS.
3. *Treatment delivery*: erogazione del trattamento, assumendo che il corpo umano sia un sistema statico.

Il corpo umano è un sistema dinamico e soggetto a variazioni principalmente identificate da:

1. Cambiamenti temporali nell'anatomia del paziente (es. contrazione/crescita del tumore, perdita di peso o movimento interno degli organi e cambiamenti di forma).
2. Cambiamenti nella biologia/funzione del tumore (es. ipossia).

Da queste considerazioni nasce l'idea e la possibile necessità di applicare la tecnica ART, grazie alle informazioni che si ricavano dalle immagini, consentendo di calcolare un nuovo piano di trattamento per il paziente ogni qual volta venga rilevata una delle due condizioni citate sopra. Questa ripianificazione può avvenire in 3 differenti momenti:

1. Offline tra le frazioni.
2. Online immediatamente prima della frazione erogata.
3. Real time durante l'erogazione della frazione.

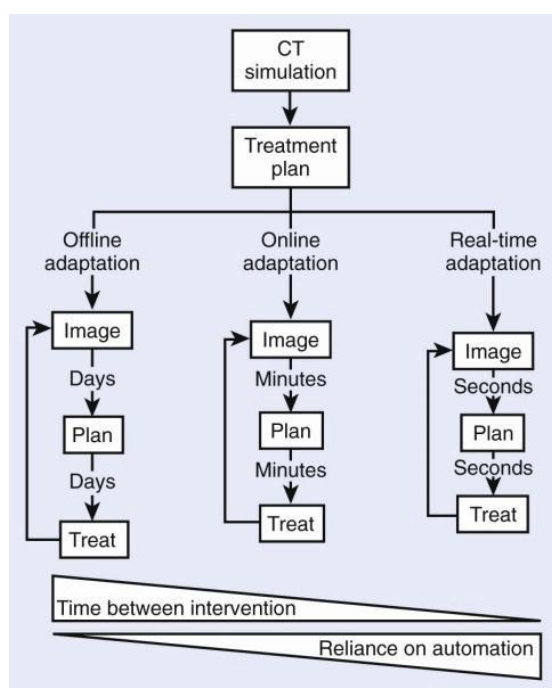


Figura 1.2 - Ripianificazione Offline, Online e in Real time.

1.2 Cenni di fisica delle radiazioni

Le radiazioni ionizzanti impiegate in radioterapia oncologica agiscono sulla materia vivente attraverso una serie di fenomeni complessi, producendo ionizzazioni o eccitazioni degli atomi che la compongono.

Le radiazioni si distinguono in due grandi categorie in base alla carica delle particelle e alla loro energia:

1. Direttamente ionizzanti: elettroni, protoni, particelle α , muoni.
2. Indirettamente ionizzanti: neutroni, fotoni X e γ (Energia > 10 eV).

La grandezza fondamentale nel settore delle radiazioni ionizzanti è la dose assorbita, definita come l'energia assorbita per unità di massa, ed è misurata in Gray (Gy) pari a 1 Joule/Kg.

Si possono riassumere in una tabella gli effetti delle radiazioni sulla materia vivente nel tempo.

FASE	TEMPO	EFFETTO
FISICA	10^{-13} s	Ionizzazione o eccitazione
FISICO-CHIMICA	$10^{-9}/10^{-6}$ s	Formazione di radicali liberi
BIOCHIMICA	Frazioni di secondi, settimane	Inattivazione di enzimi e organuli cellulari
BIOLOGICA	Giorni, mesi, anni	Inattivazione o morte cellulare
CLINICA	Giorni, mesi, anni	Manifestazione clinica a carico dell'organismo

Tabella 1 - Riassunto degli effetti delle radiazioni sui tessuti umani.

Per raggiungere gli obiettivi della moderna radioterapia sono necessari fasci di radiazioni con energie differenti e campi di dimensioni diverse.

Solitamente vengono utilizzati fasci con le seguenti energie:

1. Superficial (30 kV to 80 kV).
2. Orthovoltage (100 kV to 300 kV).
3. Megavoltage (Co-60 to 25 MV).

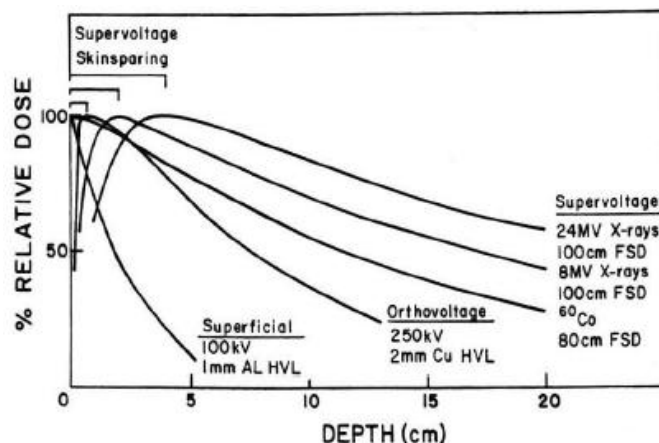


Figura 1.3 - Percentuale di dose rilasciata in relazione alla profondità di penetrazione per differenti energie¹.

Esistono due parametri che quantificano la penetrazione del fascio di fotoni:

1. Normalized depth dose (NDD).
2. Tissue phantom ratio (TPR).

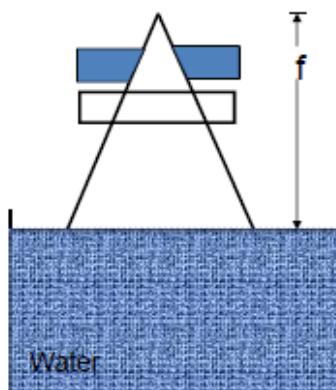


Figura 1.4 - Fantoccio omogeneo².

Il primo viene definito nel seguente modo:

$$NDD(d, s, f, Q) = \frac{D_d}{D_{dref}} \quad (1)$$

Dove d è la profondità di misura nell'asse centrale del phantom, s è la grandezza del campo alla superficie del phantom, f è la distanza tra la sorgente e la superficie del

¹ Immagine tratta da: http://www-naweb.iaea.org/nahu/DMRP/documents/slides/Chapter_07_Treatment_planning.pdf,

² Immagine tratta da: <http://www.port.ac.uk/research/iir/research/currentresearch/dataanalysis/planningtreatment/>

phantom, Q è la qualità del fascio di fotoni, D_d e D_{dref} sono rispettivamente le dosi alla profondità d e d_{ref} .

Il secondo parametro, può essere determinato dal primo mediante la formula:

$$TPR(d, s_s, Q) = NDD(d, s, f, Q) \times \left[\frac{(f + d)}{f + d_{ref}} \right]^2 \cdot \frac{S_p(s_{dref})}{S_p(s_d)} \quad (2)$$

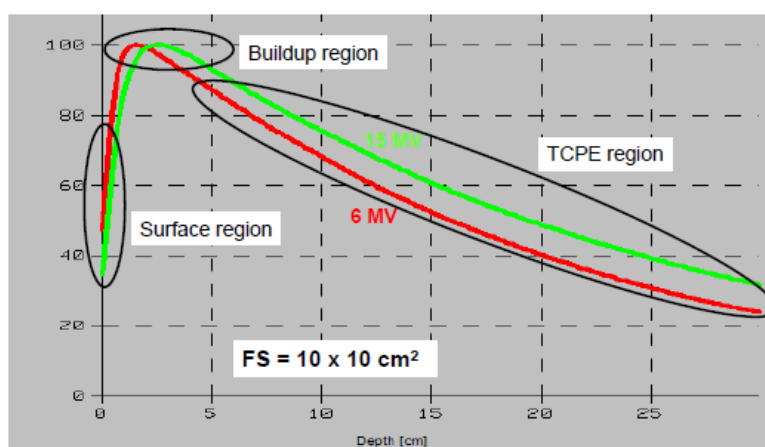


Figura 1.5 - Dose normalizzata in funzione della profondità. Dipendenza dall'energia³.

Uno dei concetti più importanti che riassume le informazioni contenute in un piano di trattamento è il Dose Volume Histograms (DVH).

Queste informazioni consistono nei dati relativi alla distribuzione di dose in una matrice 3D di punti sull'anatomia del paziente.

I DVH sono molto utili per valutare quantitativamente il piano di trattamento e ne esistono di 2 tipi:

1. Diretto o differenziale: il volume risultante (o la percentuale del volume d'organo totale) è graficato in funzione della dose entro un range specifico. Viene costruito tramite somma del numero di voxel coperti da un range specifico di dose per l'organo considerato. È utile nel fornire informazioni riguardanti le variazioni di dose entro la struttura considerata.
2. Cumulativo o integrale: è il grafico del volume relativo ad una data struttura che riceve almeno una certa dose. Viene calcolato come integrale del DVH differenziale e data la sua (maggiore) intuitività è più utilizzato rispetto al DVH differenziale.

³ Immagine tratta da: <http://www.port.ac.uk/research/iir/research/currentresearch/dataanalysis/planningtreatment/>

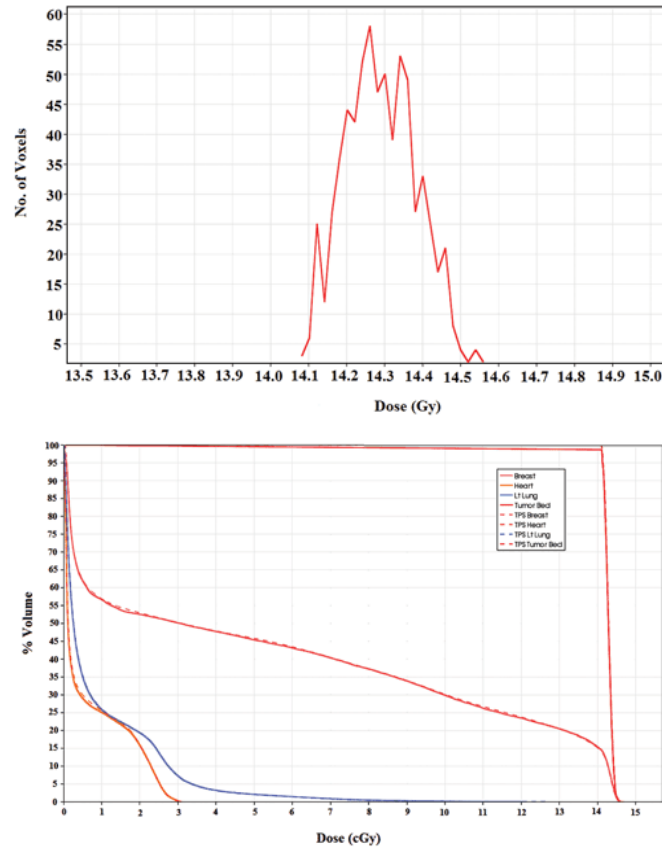


Figura 1.6 - Differenze tra DVH differenziale (in alto) e DVH integrale⁴(in basso).

La dose integrale o “*integral dose*” (*ID*) ad un organo *j* suddiviso in *m* voxel è data dall’equazione seguente:

$$ID = \sum_{k=1}^m v_{kj} D_{kj} \rho_{kj} \quad (3)$$

Dove v_{kj} , D_{kj} , ρ_{kj} sono rispettivamente il volume, la dose e la densità del voxel *k* nell’organo *j*. Se i voxel hanno la stessa dimensione e l’organo ha una densità uniforme, l’equazione (3) può essere ridotta a:

$$ID = V_j \bar{D}_j \rho_j \quad (4)$$

Dove \bar{D}_j è la dose media all’organo considerato.

⁴ <http://www.kitware.com/source/home/post/82>

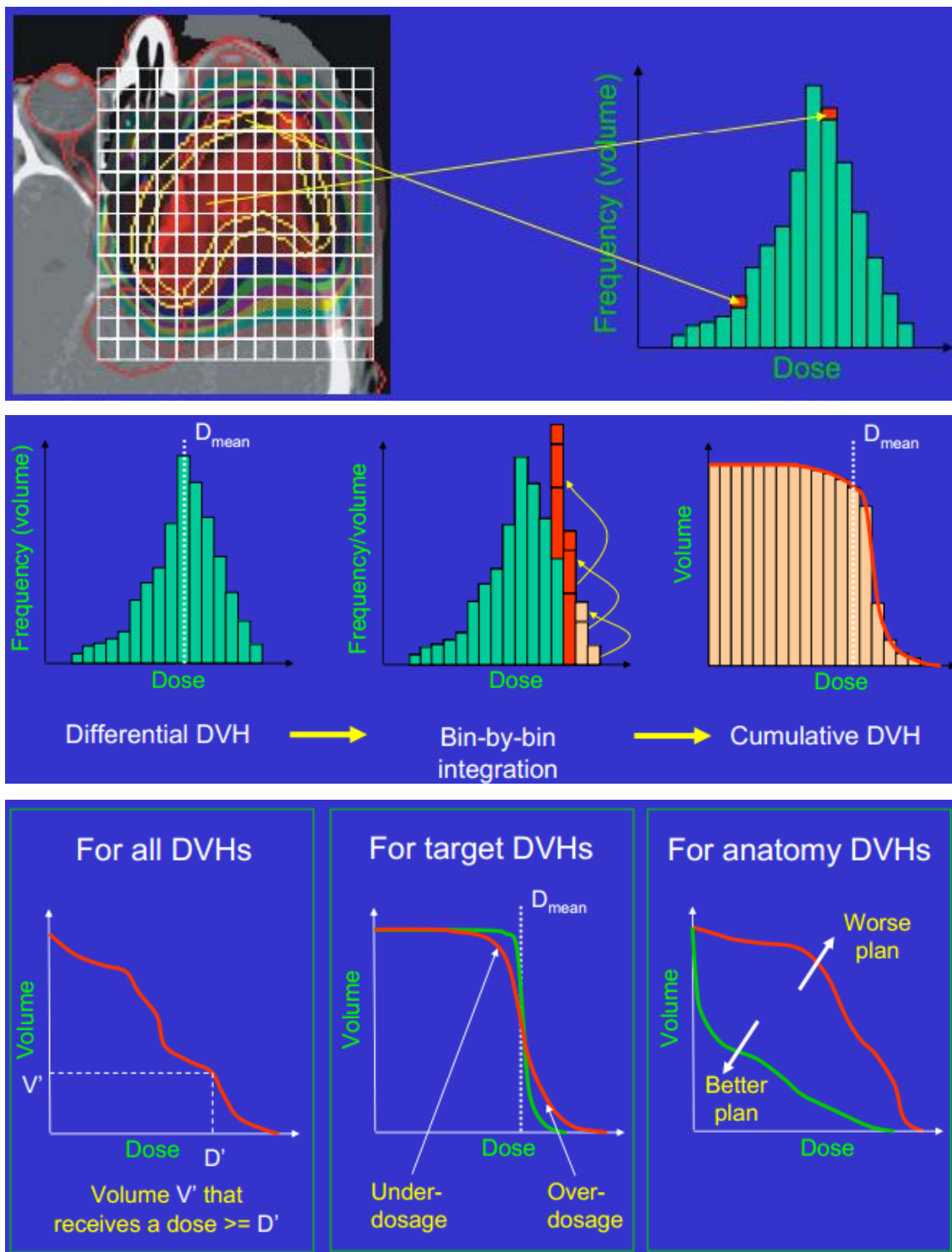


Figura 1.7 - Calcolo ed interpretazione del DVH integrale⁵.

⁵ = immagini tratte da http://www.sasro.ch/SRO/TPS_Evaluation.pdf

1.3 La Tomotherapy in breve

La “Tomotherapy” è una tecnica avanzata di IMRT in cui la radiazione viene emessa in modalità elicoidale.

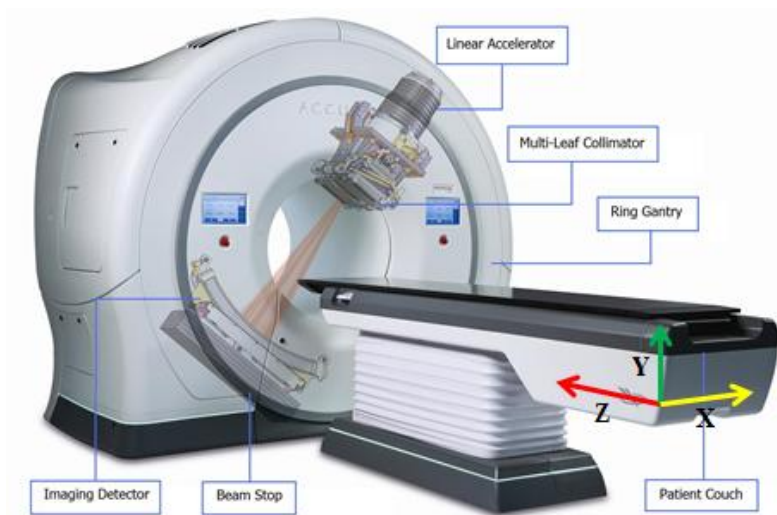


Figura 1.8 - Tomotherapy.

L'apparecchiatura è costituita da un gantry circolare (simile a quello di una TC), che racchiude un LINAC rotante con detector frontali per la ricostruzione MVCT o il monitoraggio della dose erogata (Transit dosimetry).

Il LINAC da 6 MeV ruota in modo continuo attorno al paziente eseguendo una scansione elicoidale. Il paziente adagiato su un lettino di trattamento, può muoversi contemporaneamente nelle direzioni consentendo il riposizionamento sulle coordinate (x, y, z).

La combinazione fra le varie posizioni angolari dei singoli fasci di radiazioni con i movimenti lineari del lettino del paziente, permette di erogare un trattamento continuo di tipo rotazionale, equivalente ad un numero elevatissimo di trattamenti statici convenzionali di tipo IMRT.

Sull'uscita radiante dell'acceleratore è posto un MLC che modula e scompone il fascio di radiazione. Dal lato opposto è presente un sistema di rivelazione che ricostruisce un'immagine in 3D (MVCT) della regione irradiata, utilizzando però un fascio con energia nominale ridotta pari a 3.5 MV. Lo spessore nominale delle slice può essere 2, 4 e 6 mm.

1.4 TPS (Treatment Planning System)

Con il termine TPS ci si riferisce a un insieme di calcolatori e di strumenti utilizzati per lo studio e l'elaborazione del piano di trattamento radiante, che consentono di acquisire e calcolare distribuzioni di dosi su immagini del paziente (CT, RMN, PET, US ecc.).

Questi sistemi permettono di stabilire inoltre:

- Numero di campi di irradiazione, le loro dimensioni e la loro posizione.
- Posizione del gantry o angoli di rotazione.
- Tempo di erogazione della radioterapia.
- Dose di radiazioni che sarà erogata al tumore ma anche a tutti gli OARs.

Esistono molti TPS in commercio, ma veramente pochi integrano al loro interno la possibilità di creare *scripting automation*, cioè automatismi utili nel velocizzare operazioni altrimenti svolte ripetitivamente.

In particolare, il TPS adottato all'Azienda Ospedaliera Universitaria del Policlinico di Modena (RayStation), svolge nell'ambito della ricerca un ruolo fondamentale per il raggiungimento degli obiettivi del progetto di tesi.

Permette, infatti, di seguire le variazioni avvenute nel paziente da seduta a seduta attraverso deformazioni d'immagini, strutture e dosi e la creazione di script: programmi complessi e integrabili all'interno del software e utilizzabili sia nella pratica clinica giornaliera che per finalità di ricerca vera e propria.

I vantaggi offerti dagli script sono:

- Automatizzare il lavoro manuale (molte azioni manuali sono rimpiazzate da un singolo click del mouse).
- Riduzione degli errori umani ed incremento di efficienza nella ri-pianificazione del trattamento.
- Analisi ed estrapolazione dei dati.
- Predisposizione alle nuove attività dai dati e predittività del processo di terapia.

Nei TPS, le immagini dei pazienti, sono organizzate in 3 tipi di visuale a seconda dei piani anatomici che attraversano il corpo umano.

I piani anatomici, sono delle linee immaginarie disegnate attraverso il corpo che permettono di descrivere oggettivamente i movimenti e le posizioni. Sono descritti di seguito:

1. I-S (y +/-) = Inferior-Superior o trasversale: è un piano che divide il corpo in due metà, superiore e inferiore. È situato perpendicolarmente al piano sagittale e coronale.
2. L-R (x +/-) = Left-Right o saggittale: è un piano verticale immaginario che non passa necessariamente per il centro del corpo.
3. P-A (z +/-) = Posterior-Anterior o coronale: è un piano verticale parallelo alla fronte e perpendicolare al piano sagittale. Divide il corpo in parte anteriore e parte posteriore.

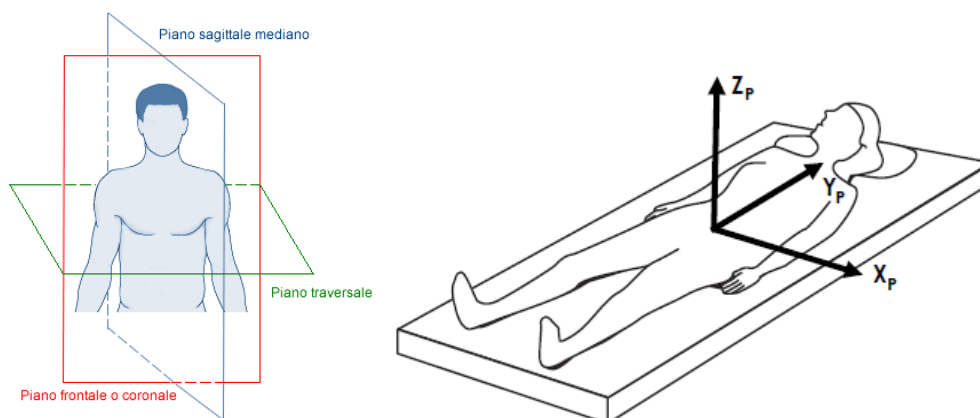


Figura 1.9 - A destra: visuale dei piani anatomici del corpo umano. A sinistra: sistema di coordinate del paziente⁶.

1.5 Imaging biomedicale

Negli ultimi anni, si è assistito ad una vera e propria esplosione di nuove tecniche di acquisizione di immagini, rappresentanti i più diversi aspetti dell'anatomia umana e le possibili patologie ad essi collegate.

Alcune di queste tecniche sono state ampiamente validate e sono divenute di uso comune, mentre altre sono ancora in fase di validazione clinico/diagnostica e necessitano di messa a punto per la produzione su larga scala a costi contenuti.

Generalmente, le modalità di acquisizione delle immagini mediche si possono suddividere in due categorie principali:

1. Immagini anatomiche: usate per descrivere la morfologia anatomica (es. CT, RMN, US).

⁶ Immagini tratte da: <http://www.my-personaltrainer.it/anatomia/assi-e-piani-corpo-umano.htm>, RSL-D-61-146-IFU-EN-4.0-2013-07-05-RayStation 4.0 Instructions for use.pdf

2. Immagini funzionali: per studiare la funzionalità e fisiologia associata alle strutture anatomiche (es. PET, SPECT o fMRI).

L'imaging biomedicale fornisce mezzi diagnostici, per avere informazioni sulle parti interne dei corpi, sfruttando in generale il diverso comportamento degli atomi quando questi assorbono energia.

In questa tesi verranno analizzate le immagini di 51 pazienti, per un totale all'incirca di 1400 immagini MVCT, suddivise per patologia:

- 23 pazienti sottoposti a trattamento H&N.
- 8 pazienti del distretto polmonare sottoposti a SBRT.
- 20 pazienti con ADK (Adenocarcinoma) prostatico sottoposti a hypofrazionamento con tecnica Simultaneous Integrated Boost (SIB).

Le immagini analizzate e ottenute da Tomotherapy, sono di 2 tipi e presentano entrambe le caratteristiche peculiari descritte di seguito:

1. kVCT o kilovoltage CT: immagini di diagnostica acquisite con energie dei fasci nell'ordine di 120 kV. Possiedono una qualità migliore (rumore minore) rispetto alle MVCT e vengono impiegate per il contornamento degli organi e per la pianificazione.
2. MVCT o megavoltage CT: vengono solitamente utilizzate per la verifica del piano di trattamento, cioè per verificare la posizione del paziente e l'anatomia prima della radioterapia. Forniscono anche informazioni accurate sui parametri di attenuazione dei tessuti per il ri-calcolo della dose in caso di deformazioni degli organi interni.

Le immagini MVCT e kVCT hanno caratteristiche differenti in base alla patologia, come si può vedere nella Tabella 2.

	kVCT			MVCT		
	N° di slices	Dim. x, y (n°pixel)	Dim. x, y, z (cm)	N° di slices	Dim. x, y (n°pixel)	Dim. x, y, z (cm)
H&N	180	272 x 272	53 x 53 x 54	30	512 x 512	38.9 x 38.9 x 12
ADK prostatico	112	272 x 272	53 x 53 x 33.6	35	512 x 512	38.9 x 38.9 x 14
SBRT polmonare	39	272 x 272	53 x 53 x 7.8	15	512 x 512	38.9 x 38.9 x 3

Tabella 2 - Dati medi relativi alle dimensioni delle immagini kVCT e MVCT, la coordinata z fornisce indicazioni sul numero di slice.

Le dimensioni dei pixel, così come le dimensioni possono essere variabili, ma in media nelle kVCT sono:

- H&N: (0.215, 0.215, 0.30) cm.
- ADK prostatico: (0.195, 0.195, 0.30) cm.
- SBRT polmonare: (0.195, 0.195, 0.20) cm.

mentre nelle MVCT tipicamente sono (0.076, 0.076, 0.40) cm per H&N e ADK prostatico, mentre per SBRT polmonare sono (0.076, 0.076, 0.20) cm.

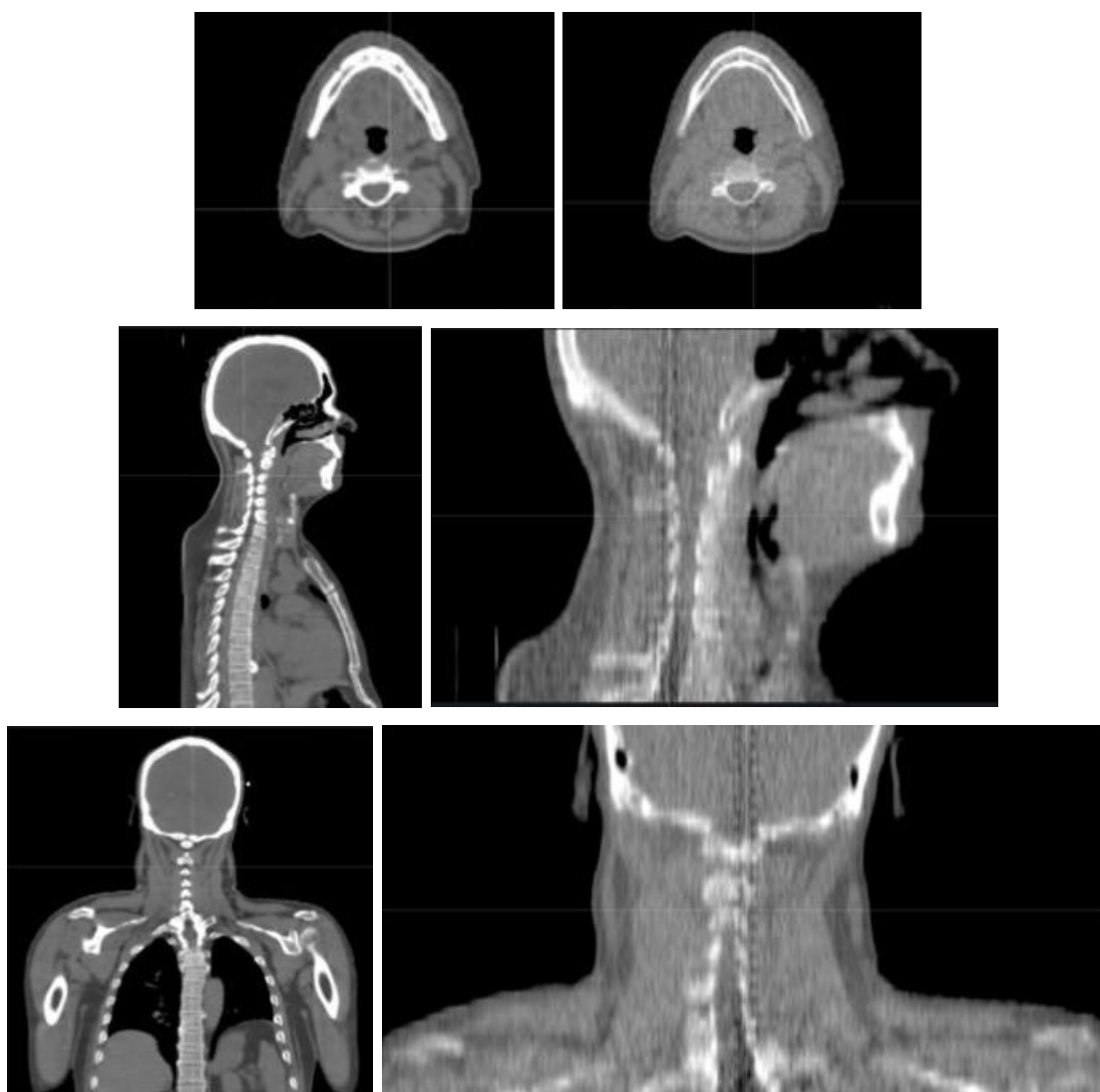


Figura 1.10 - Immagini kVCT a sinistra e MVCT a destra per un paziente H&N. A partire dall'alto: I-S, R-L, A-P.

Nei prossimi capitoli si farà spesso riferimento molte volte alle immagini dei pazienti con nomi differenti, quali “sedute”, “CT” o “esami”.

1.6 Lo standard DICOM RT

DICOM (Digital Imaging and Communications in Medicine) nasce da una collaborazione tra la NEMA (National Electrical Manufacturer Association) e l'ACR (American College of Radiology) finalizzata verso la ricerca di uno standard in grado di superare le incompatibilità esistenti fra le apparecchiature medicali fornite da vari costruttori.

Dalla prima versione 1.0, risalente al 1985, fino ad oggi (versione 3.0) lo standard ha mantenuto immutate le specifiche inerenti al formato delle immagini, aggiungendo però numerosi servizi ed implementato i protocolli di rete TCP/IP e OSI.

Attualmente è lo standard che definisce i criteri per la comunicazione, la visualizzazione, l'archiviazione e la stampa di informazioni di tipo biomedico quali, ad esempio, le immagini radiologiche.

DICOM è uno standard industriale, e non uno standard ISO (universale): ciò comporta una certa tolleranza nell'implementazione delle specifiche, al punto che attualmente forse non esistono apparecchiature che possano definirsi pienamente DICOM conformi nel senso rigoroso che la definizione di uno standard imporrebbe.

Nella maggior parte dei casi, infatti, un'apparecchiatura risulta conforme ad una parte dello standard (ad esempio la modalità di archiviazione delle immagini), mentre adotta tecnologie proprietarie per altre funzionalità (ad esempio la gestione delle liste pazienti).

I dati medici rappresentabili come immagini o le immagini vere e proprie che vengono archiviate secondo lo standard DICOM sotto forma di file vengono comunemente chiamate "immagini DICOM" e l'errore più comune è confondere lo standard con un formato di compressione delle immagini (es. JPEG, GIF, ecc.).

Il DICOM, applicato alla codifica dei file, non è nient'altro che un metodo per incapsulare i dati e per definire come questi debbano essere codificati o interpretati, ma non definisce alcun nuovo algoritmo di compressione. Le specifiche DICOM definiscono sostanzialmente tre cose:

1. Lo standard di interconnessione.
2. Il contenuto degli oggetti gestiti: immagini, dati del paziente, ecc.
3. I servizi che consentono di operare sugli oggetti.

Grazie all'utilizzo di questo standard sono stati risolti molti problemi legati all'integrazione informatica all'interno degli ospedali; in particolare, nei reparti di

radioterapia quando, nel 1997, fu presentata l'estensione DICOM RT appositamente creata per il reparto in questione.

Gli oggetti DICOM RT sono:

- RT Image: contiene tutte le immagini RT, DRR, portal images, simulator images ecc.
- RT Structure Set: contiene le strutture del paziente o *Volumes of Interest* (VOI) e gli OARs.
- RT Plan: contiene i dati geometrici e dosimetrici del piano trattamento.
- RT Dose: contiene la distribuzione totale di dose dal planning system; matrici di dose, dose points, curve di isodose e DVH.

DICOM DICOM-RT

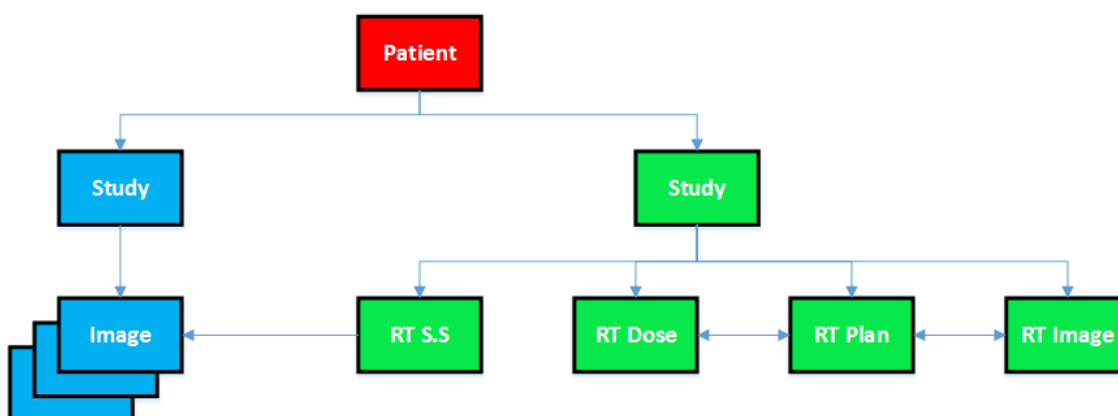


Figura 1.11 - Schema riassuntivo standard DICOM.

Name	Date modified	Type	Size
RTSTRUCT1.2.826.0.1.3680043.2.200.1505236766.544.25237.763.dcm	10/12/2013 15:31	DCM File	321 KB
RTDOSE1.2.826.0.1.3680043.2.200.1523724844.200.5988.1806.dcm	10/12/2013 15:31	DCM File	8.318 KB
CT1.2.826.0.1.3680043.2.200.1505251985.559.30919.445.59.dcm	10/12/2013 15:30	DCM File	146 KB
CT1.2.826.0.1.3680043.2.200.1505251985.559.30919.445.60.dcm	10/12/2013 15:30	DCM File	146 KB
CT1.2.826.0.1.3680043.2.200.1505251985.559.30919.445.61.dcm	10/12/2013 15:30	DCM File	146 KB
CT1.2.826.0.1.3680043.2.200.1505251985.559.30919.445.26.dcm	10/12/2013 15:30	DCM File	146 KB
CT1.2.826.0.1.3680043.2.200.1505251985.559.30919.445.27.dcm	10/12/2013 15:30	DCM File	146 KB
CT1.2.826.0.1.3680043.2.200.1505251985.559.30919.445.28.dcm	10/12/2013 15:30	DCM File	146 KB
CT1.2.826.0.1.3680043.2.200.1505251985.559.30919.445.29.dcm	10/12/2013 15:30	DCM File	146 KB
CT1.2.826.0.1.3680043.2.200.1505251985.559.30919.445.30.dcm	10/12/2013 15:30	DCM File	146 KB
CT1.2.826.0.1.3680043.2.200.1505251985.559.30919.445.31.dcm	10/12/2013 15:30	DCM File	146 KB

Figura 1.12 - File DICOM.

I principali vantaggi nell'adozione di questo standard sono:

1. Miglioramento della compatibilità tra applicazioni e strumenti provenienti da diversi fornitori.
2. Gli oggetti standard, i protocolli e i servizi sono ben definiti.
3. Archiviazione congiunta sia delle immagini che dei dati del paziente, relative informazioni geometriche ecc.
4. In ogni sistema, tutte le immagini, le strutture e i piani sono univocamente identificabili.
5. La tecnologia su cui si basa lo standard è provata e verificata.

Gli svantaggi sono:

1. Difficoltà nel determinare se due prodotti sono veramente compatibili.
2. Sistemi differenti potrebbero implementare alcune caratteristiche in modo diverso.
3. I formati interni utilizzati dalle case produttrici di apparati medicali non sono necessariamente in DICOM.
4. Si tratta di uno standard complesso e di portata molto ampia (potrebbe essere difficile interpretare alcuni requisiti).

1.7 Progetto e metodo di lavoro

Il progetto in corso all'Azienda Ospedaliera Universitaria del Policlinico di Modena, che si è citato nel capitolo introduttivo, ha come obiettivo quello di riuscire a seguire quotidianamente le variazioni anatomiche dei pazienti in radioterapia e di rendere il tutto realizzabile praticamente, valutando efficacia ed efficienza delle deformazioni in ambito *dose warping* per Tecniche IGRT e ADAPTIVERT.

Dato l'impiego notevole di risorse di cui la tecnica ART necessita dal punto di vista clinico-operativo, il presente lavoro sfrutta l'automazione come mezzo per ridurre i tempi in gioco ed abbattere conseguentemente i costi relativi di utilizzo.

Utilizzando RayStation e gli script, si può rendere automatico quasi tutto il processo di ART, articolato nelle operazioni:

1. Import DICOM pazienti (kVCT, MVCT, RT-STRUCT, RT-DOSE) in RayStation.
2. Co-registrazione rigida immagini (MVCT e kVCT).
3. Copia dei contorni dalla kVCT alla MVCT.
4. Co-registrazione deformabile con algoritmo ibrido tra MVCT e kVCT.

5. Mapping delle ROI nella direzione inversa al campo di deformazione.
6. Ricalcolo della dose deformata, considerando le deformazioni d'organo.
7. Estrapolazione dei dati statistici di dose e volume.
8. Segnalazione all'utente di eventuali errori occorsi nel processo e della necessità di intervenire, ripianificando il trattamento.
9. Calcolo della dose accumulata o *dose accumulation* per il calcolo della distribuzione di dose effettivamente erogata.

Attualmente il Policlinico di Modena dispone dei seguenti macchinari per radioterapia:

- N°1 Tomotherapy.
- N°1 LINAC Precise.
- N°1 LINAC Synergy.
- N°1 LINAC Primus.
- N°1 LINAC Synergy (nella sede distaccata di Carpi).

Ai fini della presente tesi, si è deciso di focalizzarsi sulle immagini MVCT acquisite in Tomotherapy. Questa scelta ha consentito una valutazione approfondita delle problematiche, delle difficoltà e dei vantaggi che possono essere sinergicamente traslati alle immagini CBCT provenienti dai LINAC.

L'idea alla base del progetto sarebbe quella di inviare automaticamente le immagini MVCT acquisite in una cartella di sistema in rete, chiamata DICOM.

Il TPS (sfruttando le ore notturne), interroga la cartella DICOM alla ricerca di nuovi esami dei pazienti: se ci sono nuovi esami, questi vengono aggiunti al paziente e vengono avviate tutte le operazioni di Adaptive.

L'utente, il mattino successivo, controlla i risultati in modo da rendersi conto di eventuali errori del software e se è necessaria una ripianificazione (Vedi Figura 1.13).

Ad oggi, i limiti principali di queste operazioni sono:

1. L'apertura di diverse sessioni, pari al numero di pazienti da aggiornare richiede l'intervento dell'utente.
2. Per ogni sessione è necessario l'intervento manuale dell'utente, per avviare lo script che esegue tutti gli altri.
3. Non possono essere avviati tutti gli script con un solo comando. Infatti, dopo l'import della MVCT nuova per ogni paziente e la successiva co-registrazione, è necessario selezionare le ROI di interesse ed eliminare le ROI che potrebbero

rendere più lungo il processo e generare errori (es. Cristallino, Ring, dummy ecc.).
Infine, avviare gli altri script per finire le operazioni di ART.

4. Intervento dell'utente, necessario in fase di deformazione delle dosi.
5. Nel software non è stata implementata la possibilità di importare più di un trattamento per paziente (problema nel caso di pazienti con ri-trattamenti dovuti a recidive).

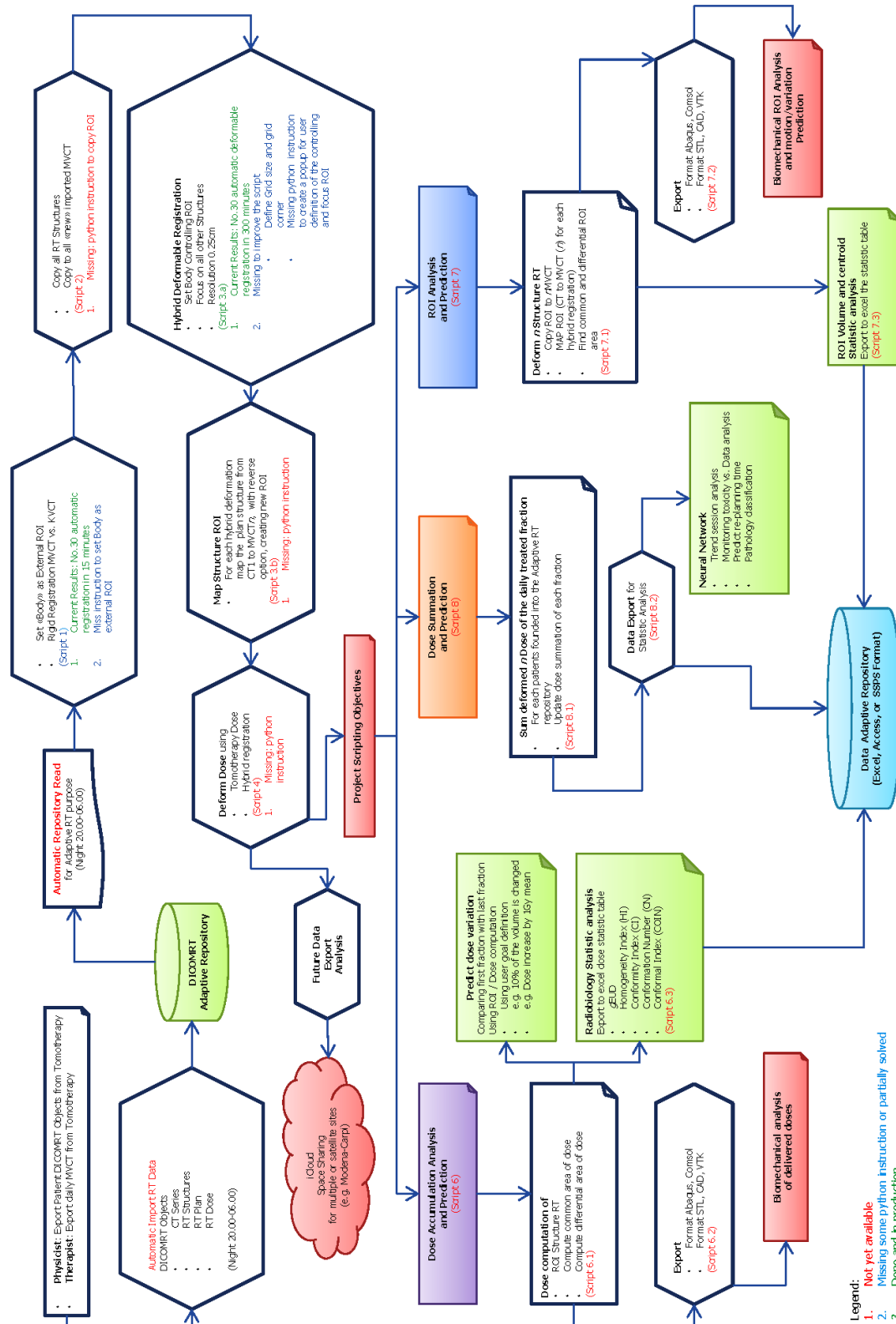


Figura 1.13 - Schema del progetto.

Per raggiungere gli obiettivi prefissati nei tempi previsti, è stato necessario organizzare il lavoro stabilendo fin da subito i tempi a disposizione per ogni attività.

Il diagramma sotto riportato (del tipo detto “di Gantt”) visibile sotto ha aiutato a questo scopo:

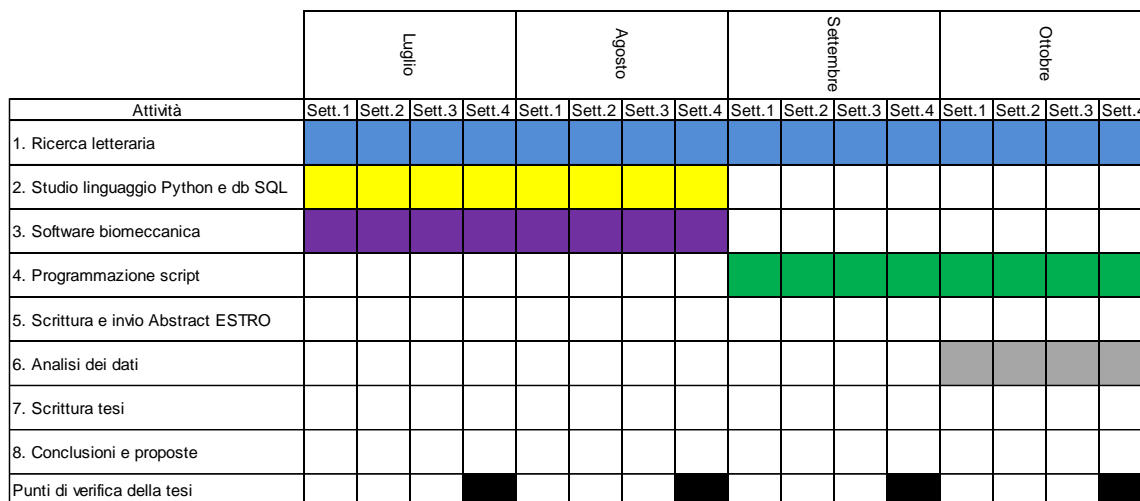


Figura 1.14 - Diagramma di Gantt 1, attività pianificate nei mesi Luglio 2013-Ottobre 2013.

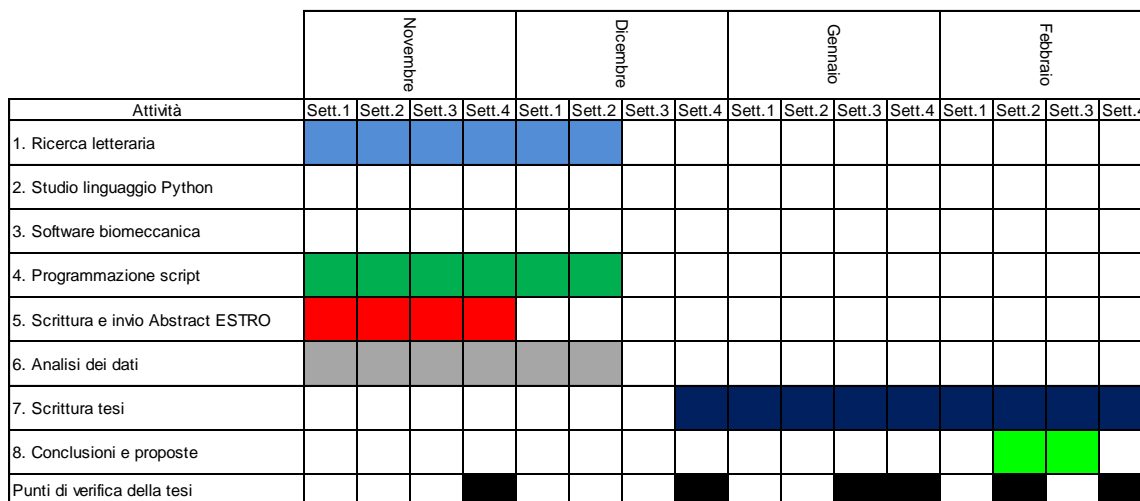


Figura 1.15 - Diagramma di Gantt 2, attività pianificate nei mesi Novembre 2013-Febbraio 2014.

2 Deformazioni e scripting

Le azioni solitamente compiute, nel processo ART descritto nel capitolo precedente (sezione 1.7), sono frutto di uno studio accurato che si pone come obiettivo la ricerca del metodo migliore per eseguire ART con l'utilizzo del TPS RayStation.

Questo metodo ART è stato validato attraverso diversi test, compiuti su fantocci o *phantom* (es. Quasar), messi a disposizione dal reparto di fisica medica dell'Azienda Ospedaliera Universitaria del Policlinico di Modena.



Figura 2.1 - Fantoccio Quasar.

Verranno quindi descritte di seguito tutte le operazioni compiute sui pazienti, corredate anche dalle misure effettuate per la validazione delle stesse.

2.1 I volumi di interesse

Prima di iniziare la trattazione riguardante le operazioni di ART, è necessario definire i volumi di interesse (PTV, ITV, GTV, ecc.) in radioterapia, dato che verranno citati molto spesso perché utilizzati nei piani di trattamento per ogni patologia specifica analizzata.

Il primo passo della pianificazione consiste nell'acquisizione delle immagini del paziente e nell'individuazione e contornamento su queste dei volumi bersaglio o VOI, definiti dall'ICRU (International Commission on Radiation Units and Measurements).

Questi volumi, definiti in modo chiaro e conciso, sono riferiti alle dosi prescritte e indipendenti dalla tecnica usata.

In particolare, si possono definire i seguenti volumi:

- GTV (Gross Tumor Volume): rappresenta il volume macroscopico di malattia, ossia quello palpabile e visibile mediante le varie modalità di imaging.
- CTV (Clinical Target Volume): è il volume di tessuto che contiene il GTV e aree limitrofe o linfonodali sedi di probabile interessamento microscopico di malattia, eventualmente anche solo sospette e che vanno rimosse. Nel corso del trattamento si rende spesso necessaria la ridefinizione del CTV, poichè la terapia ha spesso l'effetto di modificare in forma e dimensioni il CTV.
- ITV (Internal Target Volume): CTV + IM (Internal Margin, margine attorno al CTV che tiene conto delle variazioni di ampiezza, forma e posizione di quest'ultimo).
- PTV (Planning Target Volume): CTV + somma dell'IM e del SM (Set-up Margin, margine attorno al CTV+IM che tiene conto delle incertezze di posizionamento).
- OARs (Organ at Risks): sono gli organi a rischio, cioè organi la cui sensibilità alle radiazioni è tale che la dose erogata dal piano di trattamento potrebbe essere significativa rispetto alle loro tolleranze. Solitamente si cerca di cambiare la disposizione dei fasci o cambiare la dose, in modo da risparmiare questi organi.
- PRV (Planning Organ at Risk Volume Landmarks): margine attorno agli organi a rischio, che tiene in considerazione le variazioni di forma e volume ed il movimento degli organi in tutte le direzioni dello spazio.

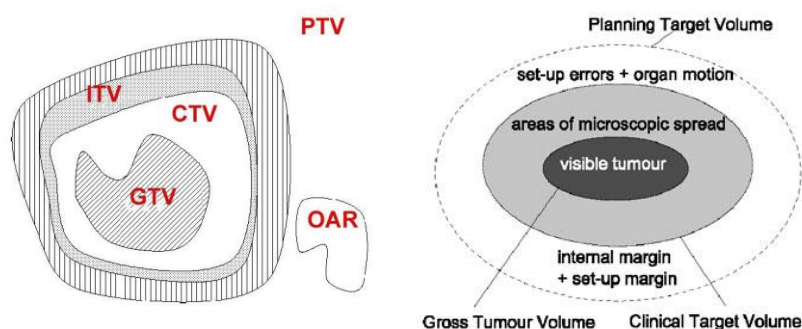


Figura 2.2 - Principali volumi di interesse in radioterapia⁷.

Solitamente si usa inserire accanto a CTV o PTV un numero che indica il limite di dose in Gy.

⁷ Immagini tratte da:
http://www-naweb.iaea.org/nahu/DMRP/documents/slides/Chapter_07_Treatment_planning.pdf,
<http://www.port.ac.uk/research/iir/research/currentresearch/dataanalysis/planningtreatment/>

2.2 Rigid registration

L'acquisizione delle immagini MVCT prima del trattamento ha come scopo controllare e correggere la posizione del paziente per il trattamento da erogare. Questa operazione consiste nel co-registrare le immagini MVCT con l'immagine kVCT del giorno in cui è avvenuta la pianificazione.

La registrazione delle immagini, o co-registrazione, è quel processo di stima di una trasformazione spaziale ottimale, che permetta di sovrapporre due o più immagini della stessa scena prese in tempi diversi, da diversi punti di vista o con diversi sensori [Zitova and Flusser, 2003]. Lo scopo è quindi l'utilizzo di algoritmi di co-registrazione che permettano, basandosi su una trasformazione spaziale, di mappare gli stessi punti anatomici presenti in un'immagine (*source image* o *reference image*) nei relativi punti dell'immagine che vogliamo co-registrare (*target image* o *floating image*).

Questo processo è importante nel campo dell'imaging medico perché permette di allineare le immagini, in modo da procedere successivamente con l'analisi quantitativa o con un confronto qualitativo.

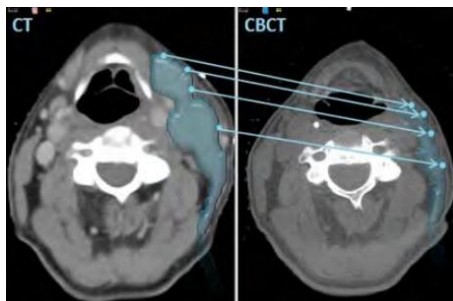


Figura 2.3 - Esempio di co-registrazione⁸.

La registrazione delle immagini consiste nei seguenti step generali:

1. *Feature detection*: oggetti salienti e distinti (regioni, bordi, contorni, intersezioni di linee, angoli, ecc.) vengono individuati sia nella *reference image* che nella *floating image*.
2. *Feature matching*: viene stabilita la corrispondenza tra le caratteristiche delle due immagini.

⁸ Immagine tratta da: <http://www.aapm.org/meetings/amos2/pdf/49-14512-46489-282.pdf>

3. *Transform model estimation*: vengono stimati il tipo e i parametri delle funzioni di mapping che allineano le due immagini.
4. *Image resampling and transformation*: la *floating image* è trasformata in base alle funzioni di mapping.

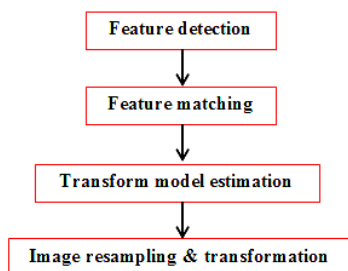


Figura 2.4 - Processo di "Image registration".

Mediante RayStation, la procedura di co-registrazione può essere eseguita sia in modo automatico che manuale e permette solo le registrazioni di tipo rigido, cioè registrazioni in cui viene ignorata la deformazione di organi e tessuti e si considerano solo rototraslazioni.

Il tool automatico esegue co-registrazioni impiegando tre differenti tecniche:

1. *Gray level based registration*.
2. *ROI based registration*.
3. *POI based registration*.

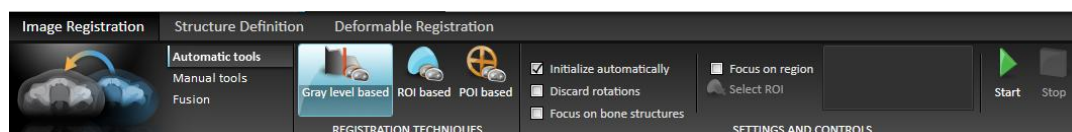


Figura 2.5 - Modulo Image registration.

Per conseguire il nostro scopo, è stata effettuata per ogni paziente la co-registrazione tra MVCT e kVCT, basandosi su un tipo di algoritmo che tiene in considerazione i livelli di grigio delle immagini da co-registrare.

Gli step seguiti sono stati:

1. Impostare la ROI *Body* come esterna.
2. Impostare come *target image* o *secondary* l'immagine della prima seduta (kVCT).

3. Impostare come *source image* o *primary* di volta in volta le immagini delle sedute successive (MVCT).
4. Selezione della tecnica “*Gray level based*”.
5. Selezionare le opzioni più consone alla modalità necessaria:
 - *Initialize automatically*: il sistema cerca il punto migliore da cui partire prima che inizi la co-registrazione.
 - *Focus on bone structures*: il sistema si focalizza nelle strutture ossee delle immagini (utile nel caso di immagini in cui sono presenti ossa e quindi strutture fisse: si cerca di evitare nel caso di pazienti in cui è prevalente la massa grassa).

In Figura 2.6, si può vedere la co-registrazione tra l’immagine della seduta 1 e la seduta 10 per un paziente H&N.

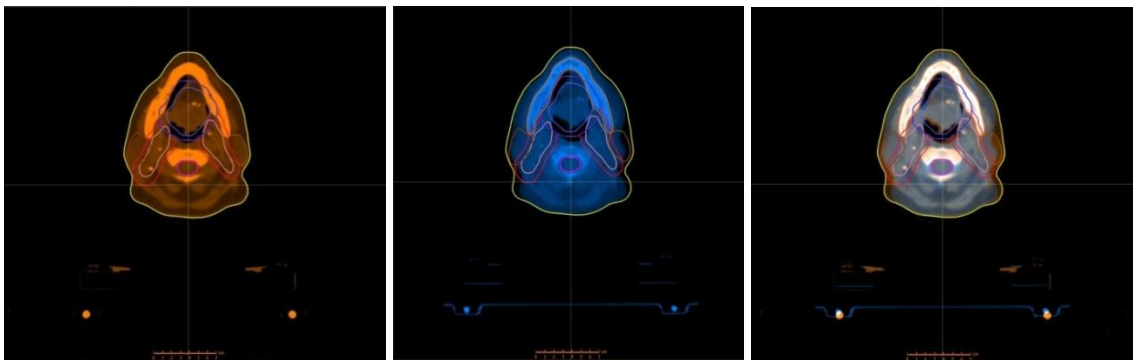


Figura 2.6 - Da sinistra a destra: CT 1 (secondary), CT 10 (primary) e fusione o co-registrazione rigida tra le due immagini.

Le trasformazioni del corpo rigido e quelle affini definiscono le trasformazioni rigide, in cui le coordinate trasformate sono le trasformazioni lineari delle coordinate originali.

Una trasformazione rigida può essere descritta con traslazioni e rotazioni; nel caso 3D avremo 6 parametri possibili: 3 rotazioni e 3 traslazioni. Sono rappresentate con matrici del tipo:

$$T_{lineare}(x) = Rx + t \quad (5)$$

Dove R è la matrice 3x3 di rotazione e t il vettore di traslazione $\in \mathbb{R}^3$ in 3D.

Per le trasformazioni rigide rotazionali, R ha solo 3 parametri. Nel nostro caso, viene usata una matrice singola con coordinate omogenee:

$$T_{lineare}(x) = Mx \quad (6)$$

dove x è il punto in coordinate omogenee e M combina sia R che t .

In 3D, avremo $x = (x, y, z, 1)$ e la matrice sarà:

$$M = \begin{pmatrix} R_{00} & R_{01} & R_{02} & t_0 \\ R_{10} & R_{11} & R_{12} & t_1 \\ R_{20} & R_{21} & R_{22} & t_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (7)$$

Alla fine di ogni registrazione, sono disponibili nell'interfaccia i dati relativi ad ogni co-registrazione (Vedi Figura 2.7), cioè la matrice di trasformazione e i valori della trasformazione rigida nelle coordinate x, y, z :

- Traslazione [cm].
- Assi di rotazione [cm].
- Angolo di rotazione [deg].

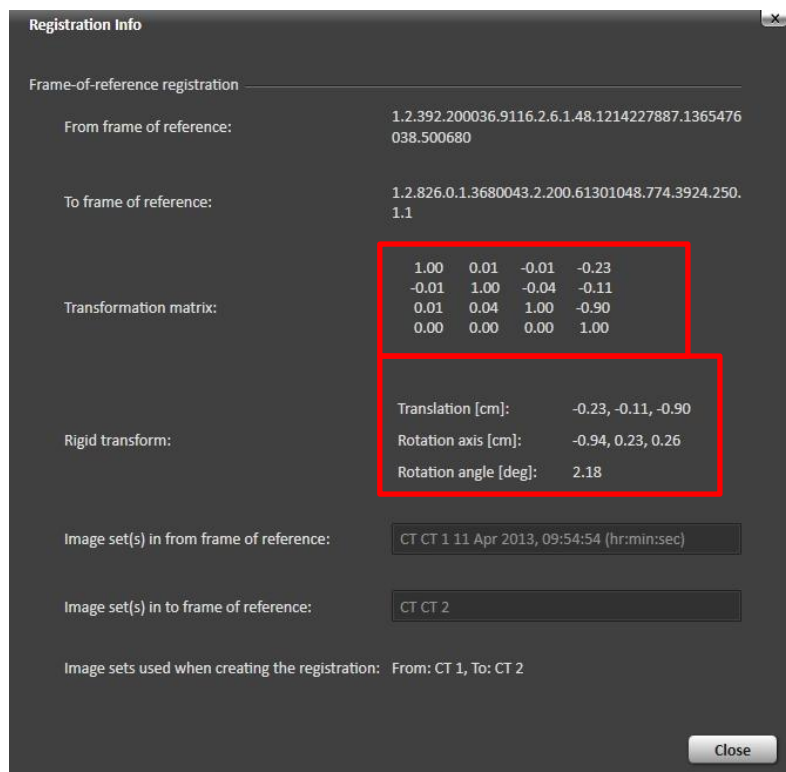


Figura 2.7 - Dati di co-registrazione.

La qualità inferiore delle immagini MVCT e il ristretto numero di slice di ognuna di esse porta a difficoltà nella co-registrazione delle immagini, che conseguentemente si traducono in errori. Per limitare questi errori sono stati messi a punto alcuni script di controllo, che segnalino alla fine delle co-registrazioni rigide eventuali errori del software.

In particolare, lo script “*RotationValue.py*”, accennato nella sezione 2.7.2 successiva, permette di calcolare l’angolo di rotazione tra le immagini co-registrate basandosi sulla matrice di trasformazione.

Prendendo come esempio la matrice di trasformazione M seguente:

$$M = \begin{pmatrix} 0.9999 & -0.0087 & -0.0099 & 0.2325 \\ 0.0091 & 0.9993 & 0.0356 & -0.8908 \\ 0.0095 & -0.0357 & 0.9993 & 0.1412 \\ 0.0000 & 0.0000 & 0.0000 & 1.0000 \end{pmatrix} \quad (8)$$

la parte superiore 3x3 individua la matrice di rotazione R :

$$R = \begin{pmatrix} 0.9999 & -0.0087 & -0.0099 \\ 0.0091 & 0.9993 & 0.0356 \\ 0.0095 & -0.0357 & 0.9993 \end{pmatrix} \quad (9)$$

R è un matrice ortogonale, cioè $RR^T = I$ e la traccia (somma degli elementi sulla diagonale) di questa è indipendente dalle coordinate usate nel sistema. In altre parole, la traccia è indipendente dagli assi di rotazione e dipende solamente dall’angolo di rotazione. Sappiamo che la somma degli autovalori è uguale alla traccia della matrice:

$$tr(R) = 1 + e^{i\varphi} + e^{-i\varphi} \quad (10)$$

$$tr(R) = 1 + \cos \varphi + i \sin \varphi + \cos \varphi - i \sin \varphi = 1 + 2 \cos \varphi \quad (11)$$

$$\cos \varphi = \frac{tr(R) - 1}{2} \quad (12)$$

Per cui l’angolo di rotazione (in radianti) sarà:

$$\varphi = \cos^{-1}\left(\frac{\text{tr}(R) - 1}{2}\right) \quad (13)$$

Nel nostro caso avremo:

$$\text{tr}(R) = 2.998 \quad (14)$$

$$\varphi_{rad} = \cos^{-1}\left(\frac{2.998 - 1}{2}\right) = 0.045 \quad (15)$$

$$\varphi_{deg} = \frac{0.038 \cdot 180}{\pi} \cong 2.18 \quad (16)$$

Sono stati raccolti i dati relativi agli angoli di rotazione nelle co-registrazioni rigide ed analizzati suddividendoli per patologia.

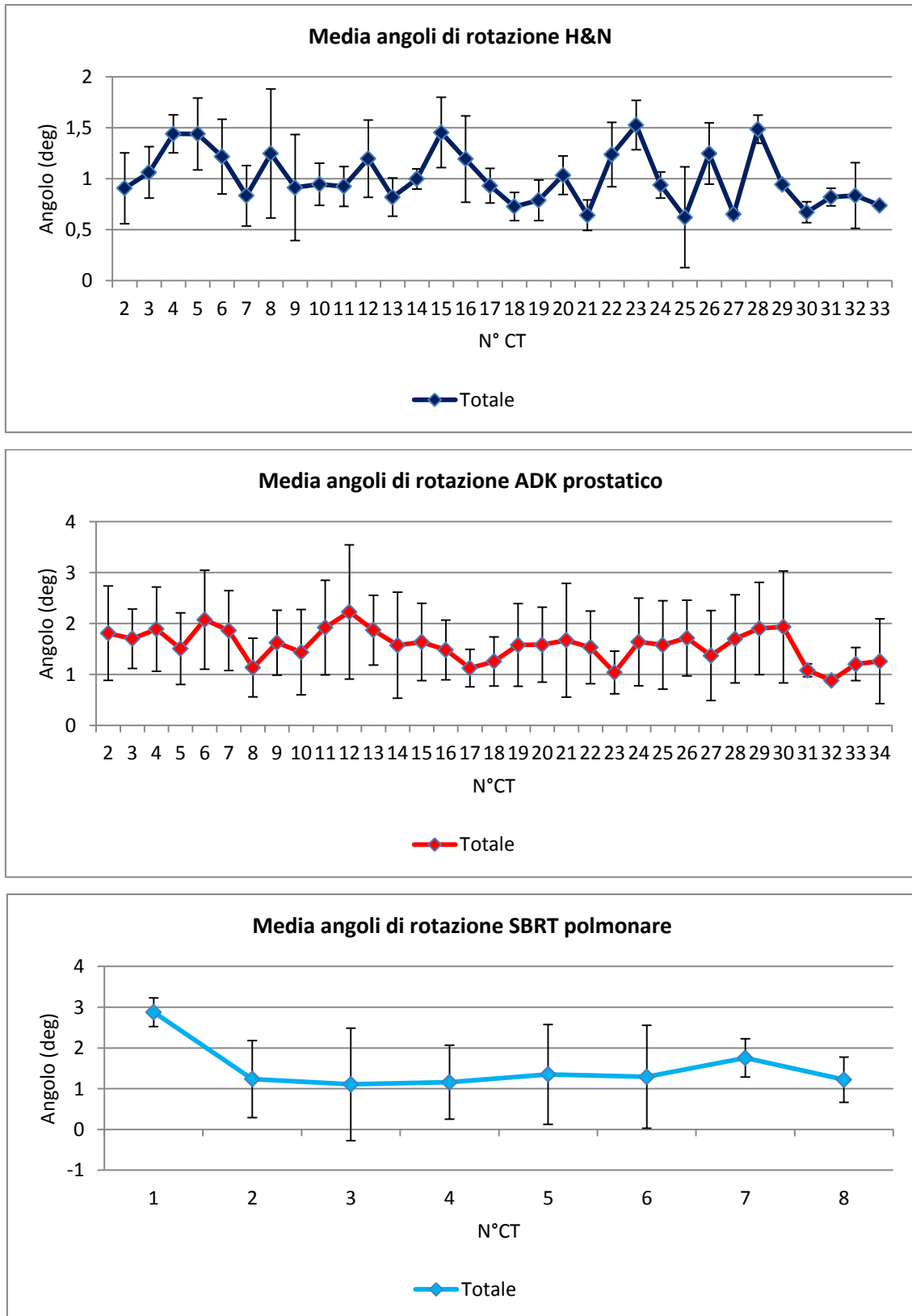


Figura 2.8 - Media degli angoli di rotazione nelle co-registrazioni rigide per seduta e per patologia.

Per ogni patologia è stato ipotizzato un valore limite di angolo di rotazione, considerando gli intervalli di incertezza ottenuti, oltre al quale dovrebbe essere ricontrollata la co-registrazione rigida.

Questi valori sono:

1. H&N: 2°.
2. ADK prostatico: 3.5°.
3. SBRT polmonare: 4°.

I valori degli angoli di rotazione e i messaggi di allarme (“Check”) delle co-registrazioni rigide vengono scritti in un file di testo (“Cognome^Nome_RotationAngle.txt”).

Il file creato viene salvato in una cartella di sistema, in modo che sia accessibile facilmente dall’utente per una successiva valutazione (Vedi Figura 2.9).

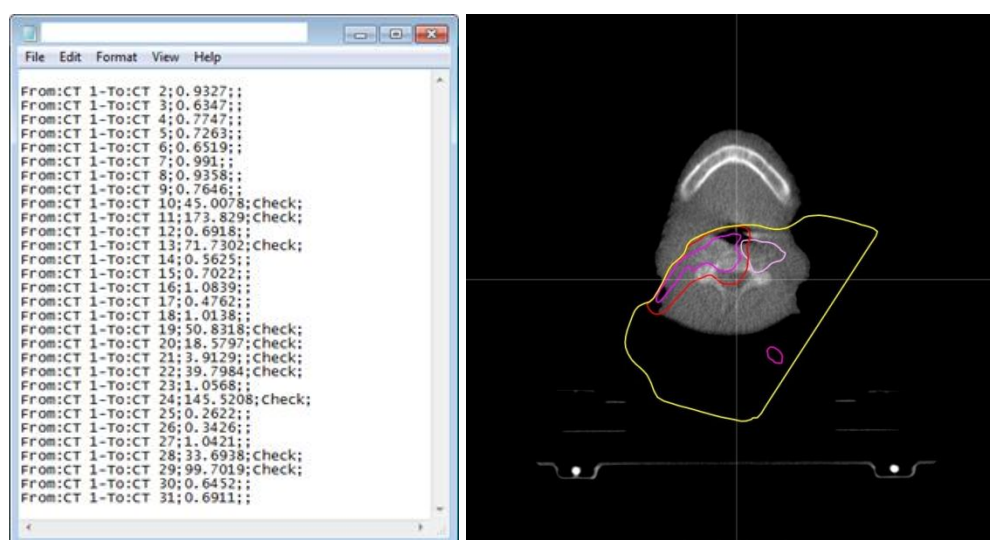


Figura 2.9 - Patologia H&N. A sinistra: file di testo contenente gli angoli di rotazione. A destra: CT 10 (angolo di rotazione di 45°).

Un altro problema, si è presentato in alcuni casi dopo l’operazione di co-registrazione, quando vengono copiate le ROI dalla kVCT alle MVCT. Può capitare, infatti, che le ROI siano copiate fuori dai bordi delle MVCT, per cui si è sviluppato uno script che controlli questa particolare condizione.

Lo script è chiamato “ROI_position.py” ed esegue un controllo delle coordinate assolute (x, y, z) del centro della ROI “Body” rispetto alle coordinate DICOM assolute (x, y, z) dei bordi delle immagini MVCT. Lo script scrive, infine, in un file di testo

(“Cognome^Nome_ROI_Position.txt”) il numero dell’esame e a fianco la stringa “Check ROI Position”, nel caso in cui sia necessario un controllo da parte dell’utente.

È stata scelta la ROI “Body” perché è quella che rappresenta la struttura esterna del paziente e solitamente le ROI relative agli altri organi risiedono al suo interno. Per cui se la ROI “Body” risulta esterna ai bordi DICOM della MVCT nei rapporti previsti, è chiaro che anche le altre ROI, che si trovano al suo interno, saranno anch’esse fuori dai limiti imposti.

In particolare, si rende necessario un controllo, nel caso in cui una delle seguenti condizioni risulti vera:

1. Coordinata x del centro della ROI “Body” > Coordinata x/3 del bordo della MVCT.
2. Coordinata y del centro della ROI “Body” > Coordinata y/3 del bordo della MVCT.
3. Coordinata z del centro della ROI “Body” > Coordinata z*2 del bordo della MVCT.

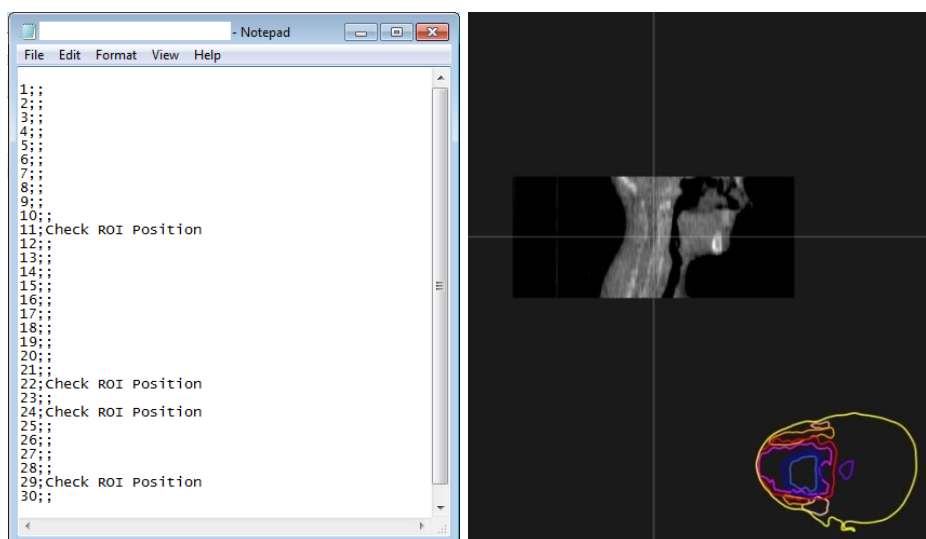


Figura 2.10 - A sinistra: file di testo contenente i numeri delle CT e, a fianco, la stringa di controllo nel caso in cui sia necessario ricontrollare la copiatura delle ROI. A destra: CT 24 da rivedere.

Nel caso si verifichi una delle suddette situazioni di errore, il software mette a disposizione una serie di *tool* manuali di correzione. Spesso vengono utilizzati quelli evidenziati in rosso nella Figura 2.11.

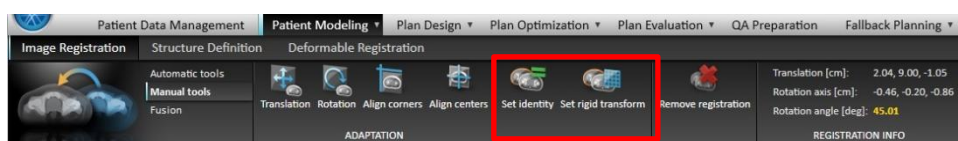


Figura 2.11 - “Manual tools”: Align centers e Set identity

Il risultato della correzione, per la Figura 2.9, è visualizzato nella Figura 2.12.

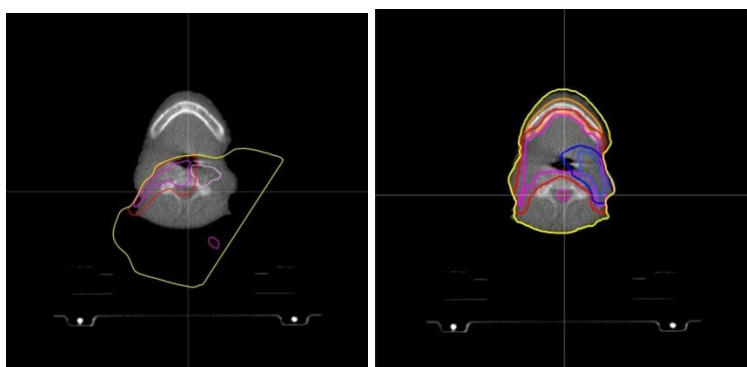


Figura 2.12 - A sinistra: co-registrazione rigida errata. A destra: co-registrazione corretta.

2.3 Deformable registration

Le co-registrazioni deformabili o co-registrazioni elastiche tra le immagini di ogni seduta giornaliera (MVCT) e la prima immagine relativa alla pianificazione (kVCT) sono una componente chiave del processo ART.

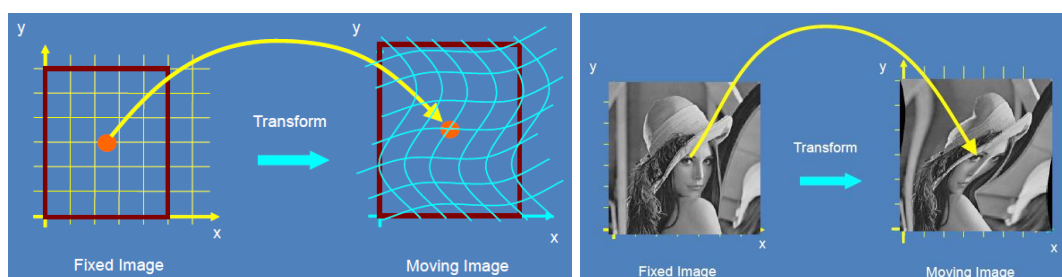


Figura 2.13 - Co-registrazione deformabile⁹.

Dopo avere effettuato la co-registrazione rigida tra le immagini è necessario, prima di proseguire con il secondo step, copiare le ROI dalla prima seduta alle altre sedute.

⁹ Immagini tratte da: http://appuntamentoiafisica.to.infn.it/2012/1/doc/Riccardo_Ragona.pdf

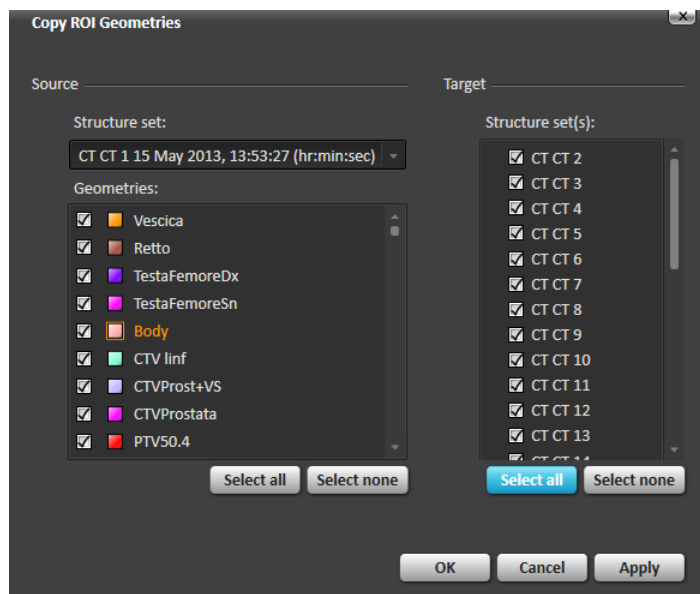


Figura 2.14 - Selezione delle ROI da copiare.

Successivamente, si procede con il deformare le strutture (ROI) di ogni esame, considerando le eventuali deformazioni d'organo o spostamenti.

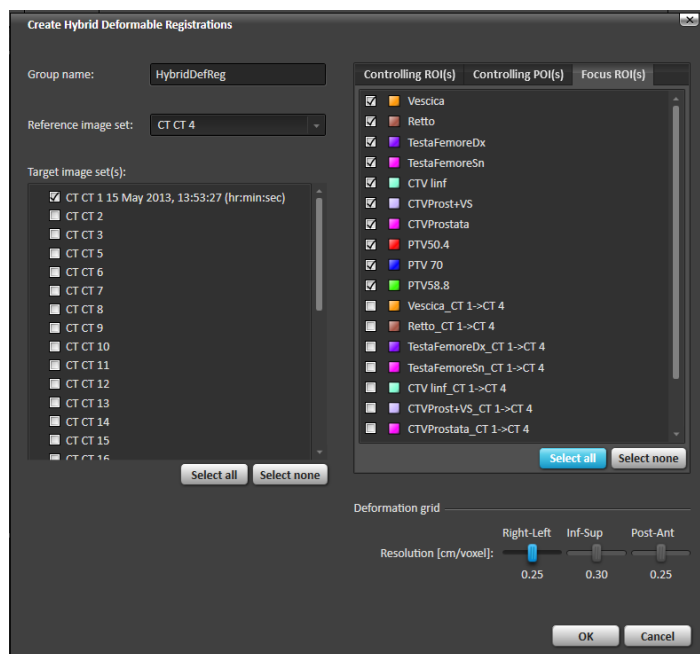


Figura 2.15 - Schermata di scelta delle impostazioni delle co-registrazioni deformabili.

Il risultato di un algoritmo di deformazione è un campo di spostamento dalla *reference image* alla *target image*. Il campo di spostamento può essere usato per deformare oltre alle strutture anatomiche (ROI) anche le dosi tra i set di immagini.

In RayStation, è stato utilizzato un algoritmo di deformazione ibrido, che si basa sia sui livelli di intensità delle due immagini che sulle informazioni relative alle strutture (ROI e POI).

L'algoritmo, sviluppato dalla RaySearch Laboratories per le deformazioni ibride, è una combinazione di un algoritmo basato sulle immagini e un algoritmo che utilizza informazioni anatomiche provenienti dai datasets contornati. Questo migliora la capacità dell'algoritmo di mantenere l'integrità delle strutture anatomiche anche in aree delle immagini dove le informazioni sono meno chiare, mentre l'algoritmo che si basa solamente sulle immagini può solo regolarizzare il campo di deformazione applicando un certo smoothing.

È anche possibile specificare su quale ROI focalizzare la registrazione in modo che la funzione di ottimizzazione si concentri sulla regione selezionata.

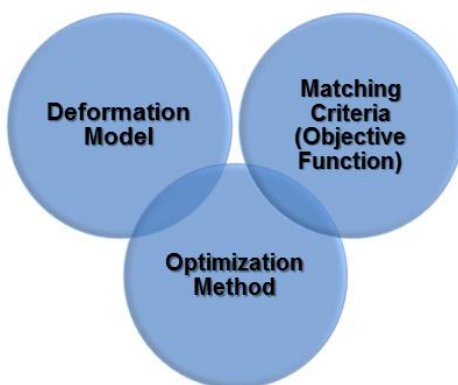


Figura 2.16 - Algoritmo di deformazione.

La formulazione generale della co-registrazione deformabile è la minimizzazione dell'energia o di una funzione costo. La funzione costo, cioè la funzione obiettivo dell'ottimizzatore, è definita come:

$$cost = \int_{volume} deformazione - \int_{volume} similarità \quad (17)$$

Il termine di similarità si comporta come una forza esterna, che massimizza la similarità tra le immagini e può essere una delle metriche che analizzeremo in seguito (CC, MI o CC mixed MI).

Il termine di deformazione tiene conto del movimento dell'oggetto che deve essere coregistrato.

Il movimento dipende dalle proprietà fisiche dell'oggetto; può essere una deformazione elastica lineare, una deformazione tipo fluido viscoso o avere altre forme più complicate.

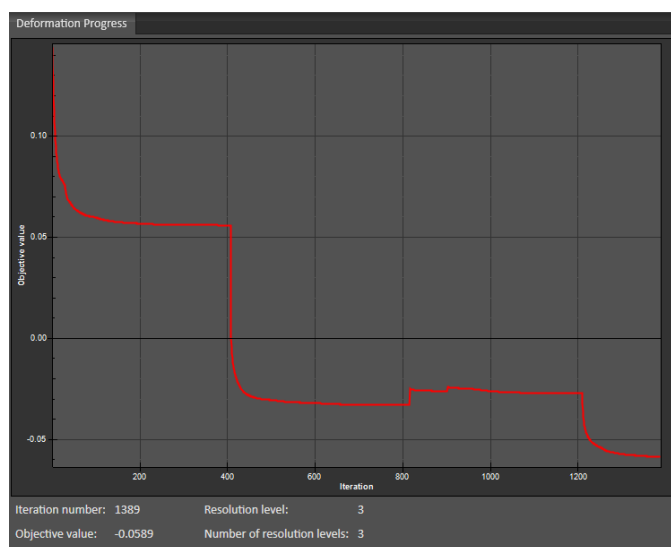


Figura 2.17 - Grafico generato automaticamente durante la co-registrazione deformabile. Ad ogni livello di risoluzione corrispondono un certo numero di iterazioni per raggiungere l' "Objective value", cioè il valore della funzione obiettivo.

Nell'algoritmo di deformazione ibrido, il problema della registrazione è considerato come un problema di ottimizzazione non lineare e risolto con uno strumento sviluppato ad "hoc". La funzione obiettivo è la combinazione lineare di 4 termini non lineari:

1. Image similarity.
2. Regolarizzazione della griglia, che ha come obiettivo quello di mantenere la griglia di deformazione dell'immagine invertibile e con smoothing.
3. Regolarizzazione di forma della griglia, che lavora per mantenere le deformazioni anatomicamente ragionevoli, quando le ROI sono presente nella reference image.
4. Un termine di penalità che viene aggiunto al problema di ottimizzazione, quando sono usate le strutture di controllo (ROI e POI), con lo scopo di deformare le strutture selezionate nella reference image in base alle corrispondenti strutture nella target image.

Una volta che è stata creata la registrazione deformabile, RayStation fornisce un certo numero di tool per modificare il risultato.

1. La co-registrazione elastica può essere visualizzata come griglia di deformazione, oppure come campo con vettori di spostamento, o come fusione tra la reference image e l'immagine target deformata nella stessa schermata.
2. La registrazione deformabile può essere comparata visivamente con ogni altra deformazione o co-registrazione rigida nello stesso set di immagini, attraverso una schermata chiamata side-by-side view (fusione/campo di spostamento).
3. Lo strumento "Point evaluation" permette di visualizzare lo spostamento di uno o più punti dal set di riferimento a quello target.
4. È presente la statistica relativa a ROI e POI della deformazione. La tabella "ROI/POI geometry statistics" è utile per valutare lo spostamento di ROI e POI individuali. Vengono visualizzate informazioni relative a volume, numero di punti mesh, centroide dei punti mesh, traslazione, assi di rotazione, angoli di rotazione, distanza tra i punti mesh, differenza dei punti mesh per le ROI e posizione, distanza, spostamento e differenza per i POI.

Le strutture possono infine essere mappate da un data set ad un altro, usando il campo di deformazione che è stato creato. Questo fornisce all'utente un valido strumento per controllare la qualità del processo di deformazione.

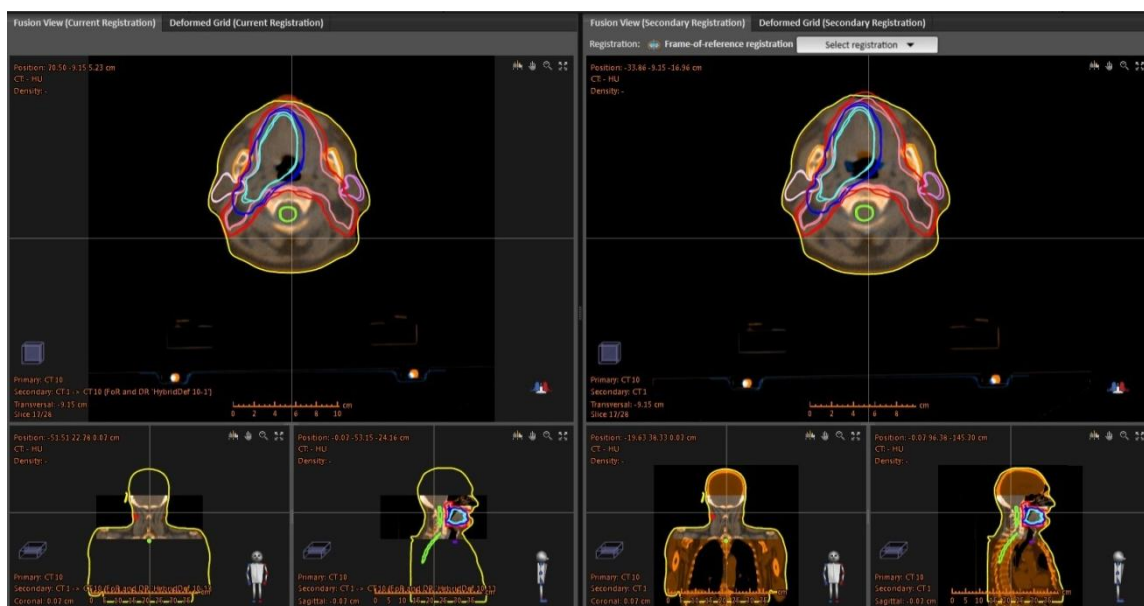


Figura 2.18 - Side by side view: confronto tra co-registrazione deformabile (sinistra) e co-registrazione rigida (destra).

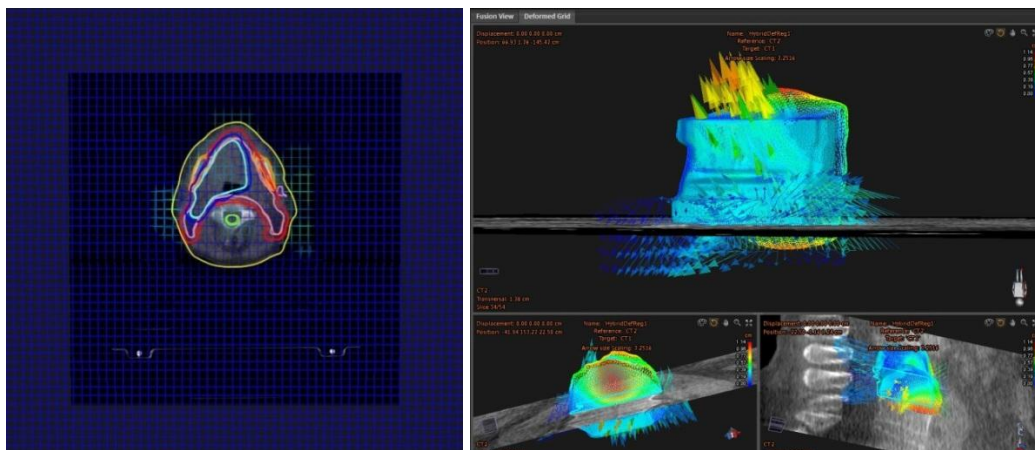


Figura 2.19 - A sinistra: griglia di deformazione. A destra: vettori del campo di deformazione.

Attraverso alcuni parametri, è possibile decidere:

- *Reference Image Set*: MVCT.
- *Target Image Set*: kVCT.
- *Controlling ROI/POI*: si possono includere ROI o POI di controllo nella deformazione.
- *Focus on ROI*: si può focalizzare l'algoritmo di deformazione su specifiche regioni dell'immagine, in modo tale che la funzione di ottimizzazione sia calcolata solo all'interno di questa. Il tempo impiegato per portare a termine una deformazione è indipendente dal numero di ROI incluse nel focus.
- *Resolution level*: si può selezionare il livello di risoluzione della griglia di deformazione in 0.1, 0.25 o 0.5 cm (0.5 cm si dimostra utile solo per valutazioni preliminari, che possono ritenersi approssimative. Infatti voxel maggiori di 0.5 cm non sono mai utilizzati nelle scansioni cliniche poiché molti volumi (es. parotidi, chiasma e midollo) sono significativamente più piccoli).

Le registrazioni deformabili possono essere usate per:

1. Contornamento: si possono mappare le strutture già, definite per un'immagine, in un'altra immagine, o contornare le strutture dell'immagine fusione.
2. Deformazioni di dose: si può mappare ogni dose calcolata o importata da un altro sistema, in un image set. È supportato l'export delle DICOM relativo alle dosi.
3. Dose tracking: dopo avere calcolato le dosi in un image set (ogni image set rappresenta la CT quotidiana del paziente, utilizzata per il repositioning), queste possono essere deformate indietro sull'immagine diagnostica sulla quale viene fatto

il piano di trattamento (kVCT) e accumulate per un'analisi nel DVH. Questi risultati possono essere estrapolati fino all'ultima frazione, per aiutare il clinico ad assicurarsi che gli obiettivi del trattamento originale siano stati raggiunti.

4. Adaptive Planning: il piano di trattamento può essere adattato in base ai risultati del Dose Tracker, l'utente può lanciare il tool di Adaptive Planning che automatizza il processo IMRT e compensa per i problemi di *dose coverage* o *goals*.

Utilizzando gli script è possibile impostare dei parametri aggiuntivi nelle deformazioni. Per capire come e su quali parametri agire, sono stati condotti vari test su un tipico paziente H&N.

Da questi test sono scaturite le configurazioni riassunte nella Tabella 3.

Parametri	Descrizione	Valori
<i>DeformationGridVoxelSize</i>	Vettore con i valori di grandezza dei voxel della griglia di deformazione in coordinate DICOM.	{ 'x': 0.25, 'y': 0.25, 'z': 0.25 }
<i>GaussianSmoothingSigma</i>	Lista con la deviazione standard del filtro di smoothing gaussiano che viene applicato ad ogni immagine.	[[0.333,0.333,0.333]]
<i>ResolutionLevels</i>	L'algoritmo deforma la <i>reference image</i> iterativamente nei livelli di deformazione dati.	[[{ 'x': 1, 'y': 1, 'z': 1 }, { 'x': 0.5, 'y': 0.5, 'z': 0.5 }, { 'x': 0.25, 'y': 0.25, 'z': 0.25 }]]
<i>GridSmoothingWeight</i>	Lista con i pesi di regolarizzazione della griglia di deformazione.	[200 - 1000]
<i>ShapeConstrainingWeight</i>	Peso delle ROI definite nella <i>reference image</i> impiegate implicitamente come vincoli di forma per le deformazioni.	1
<i>LandmarkConstrainingWeight</i>	Peso dei <i>landmark</i> se vengono utilizzati.	10
<i>ControllingRoiWeight</i>	Peso delle ROI di controllo.	10
<i>MaxNumberOfIterationsAtResolutionLevel</i>	Numero massimo di iterazioni per ogni livello di risoluzione.	[[1000,1000,1000]]
<i>ImageSimilarityMeasures</i>	L'algoritmo utilizzato per <i>image similarity</i> .	[["CC", "CC", "CC"]]

Tabella 3 - Riassunto parametri istruzione di deformazione tramite script.

Il metodo di co-registrazione elastica si basa sul presupposto che sia possibile definire una metrica tra le immagini. La metrica è una generica misura quantitativa che dice quanto bene le immagini, reference e target, sono allineate.

Nel nostro caso la metrica si può basare:

1. Sulle differenze tra le intensità delle immagini: cross correlation (CC).
2. Su informazioni statistiche: mutual information (MI).
3. Su tutte e due le informazioni (CC_MIXED_MI).

In particolare, il *CC* viene utilizzato per le co-registrazioni intra-modality. Date due immagini *A* e *B*, il coefficiente di *CC* è dato dalla seguente espressione:

$$CC(A, B) = -1 \times \frac{\sum_{i=1}^N ((A_i - \bar{A}) \cdot (B_i^T - \bar{B}))}{\sqrt{\sum_{i=1}^N (A_i - \bar{A})^2 \cdot \sum_{i=1}^N (B_i^T - \bar{B})^2}} \quad (18)$$

dove A_i è l'*i*-esimo pixel dell'immagine *A*, B_i è l'*i*-esimo pixel dell'immagine *B*, \bar{A} e \bar{B} sono i valori medi delle intensità delle immagini *A* e *B* rispettivamente, *N* è il numero di pixel dell'immagine *A* e *T* è la trasformazione.

La metrica di *MI*, invece, calcola le informazioni in comune tra due immagini *A* e *B*. Misura l'informazione che una variabile casuale (es. intensità dell'immagine reference) esprime rispetto ad un'altra variabile casuale (es. intensità dell'immagine target).

Il vantaggio maggiore che si ha nell'usare l'algoritmo di *MI* è che non c'è bisogno di specificare la forma di dipendenza tra le variabili e ciò rende il *MI* adatto alla co-registrazione multi-modale.

La metrica *MI* è definita in termini di entropia. Sia:

$$H(A) = - \int p_A(a) \log p_A(a) da \quad (19)$$

L'entropia di una variabile casuale *A* e $H(B)$ l'analogo entropia di una variabile casuale *B* e sia:

$$H(A, B) = - \int p_{AB}(a, b) \log p_{AB}(a, b) da db \quad (20)$$

l'entropia congiunta di A e B . Se A e B sono indipendenti allora:

$$p_{AB}(a, b) = p_A(a)p_B(b) \quad (21)$$

$$H(A, B) = H(A) + H(B) \quad (22)$$

Se tra A e B c'è una certa dipendenza si avrà invece:

$$H(A, B) < H(A) + H(B) \quad (23)$$

La differenza è chiamata MI :

$$I(A, B) = H(A) + H(B) - H(A, B) \quad (24)$$

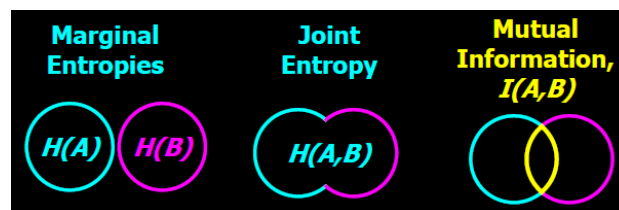


Figura 2.20 - Entropia e MI.

Se le immagini reference e target sono allineate la MI raggiunge il valore massimo [Mario Esposito, Tesi di Dottorato].

La selezione del tipo di metrica da utilizzare dipende fortemente dal tipo di co-registrazione deformabile che deve essere compiuta. Alcune metriche sono adatte per immagini acquisite con la stessa modalità, altre per acquisizioni con diverse modalità.

Non esiste, in definitiva, una regola ben definita per decidere la metrica da utilizzare.

Il software, procedendo con le co-registrazioni deformabili manualmente, adotta la metrica CC e sarà questa ad essere impiegata negli script per l'analisi dei pazienti.

2.4 Mapping structures: misure sperimentali

Di ogni co-registrazione deformabile, sono state mappate le ROI nella direzione inversa al campo di spostamento, basandosi sulle prove effettuate sul fantoccio visibile nella Figura 2.21.



Figura 2.21 - A partire da sinistra: fantoccio posizionato in Tac tramite laser. A destra: fantoccio pronto per test, con inserto motore. A fianco: zoom della scala graduata del motore.

Il fantoccio è costituito da 3 tipi di inserti graduati. All'interno di questi inserti, removibili, sono presenti diverse forme: 1 cubo, 3 sfere di diverse dimensioni e 1 cono.

Il fantoccio ha forma ovale ed è costituito da un materiale avente lo stesso coefficiente di attenuazione del tessuto umano, con gli inserti in 3 diverse posizioni.

Sono state riassunte di seguito, le operazioni eseguite nel test, che hanno portato al confronto di 2 approcci diversi di deformazione delle immagini:

1. Il fantoccio è stato posizionato sul lettino della TAC e allineato con i laser.
2. È stato posizionato l'inserto contenente il cubo.
3. Si è inserito il motore, che ha la funzione di far traslare con una certa frequenza l'inserto al quale viene collegato. Tale possibilità consente lo sviluppo e lo studio cinematico dei tumori (4D-CT).
4. Sono state effettuate n°5 TAC. In ognuna di esse il fantoccio è stato mantenuto nella stessa posizione, cambiando solamente la posizione dell'inserto, tramite la scala graduata presente nel motore. Le posizioni sono: +0.5, +1.5, 0, -0.5 e -1.5 cm rispetto al laser di centratura.

5. Le immagini acquisite sono state successivamente inviate a RayStation, sul quale sono state importate, contornate, co-registrate e deformate utilizzando l'algoritmo ibrido ed infine è stata eseguita la mappatura delle ROI.

Nelle immagini di Figura 2.24 e Figura 2.25 si può notare il confronto tra i due approcci. Per un'esposizione migliore le immagini del fantoccio sono state tagliate, riportando solo la parte interessata da movimento.

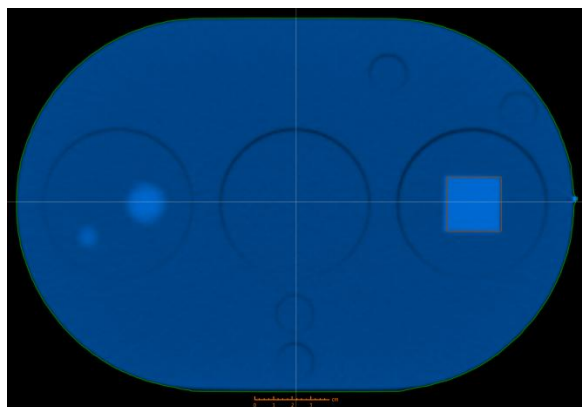


Figura 2.22 - CT 1 in cui è stato contornato (ROI marrone) il cubo interno all'inserto a destra. Visuale I-S.

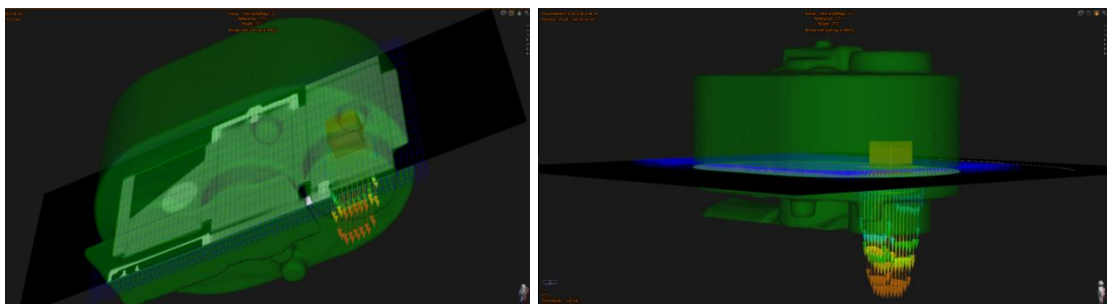


Figura 2.23 - A sinistra: visuale 3D della deformazione ibrida della CT 3. A destra: visuale 3D del fantoccio con griglia e campo di deformazione.

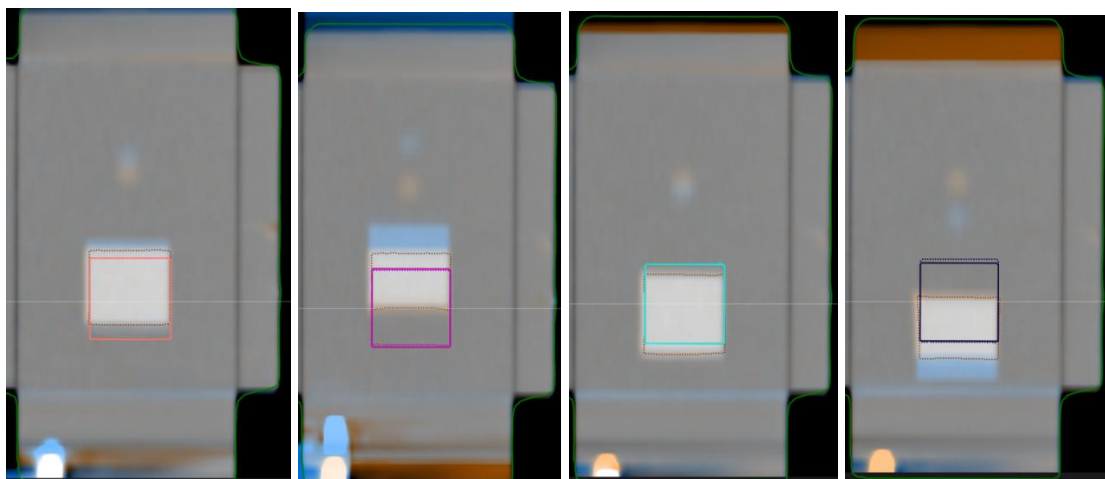


Figura 2.24 - Da sinistra a destra: sezioni del fantoccio in movimento rispettivamente di +0.5 e +1.5, -0.5 e -1.5 cm nelle CT 2, CT 3, CT 4, CT 5.

Nella Figura 2.24 e Figura 2.25, le ROI di colore marrone rappresentano le ROI iniziali contornate sulla CT 1. Le ROI colorate sono quelle create utilizzando il comando Map ROI nella direzione diretta nel caso di Figura 2.24 e nella direzione inversa nel caso di Figura 2.25.

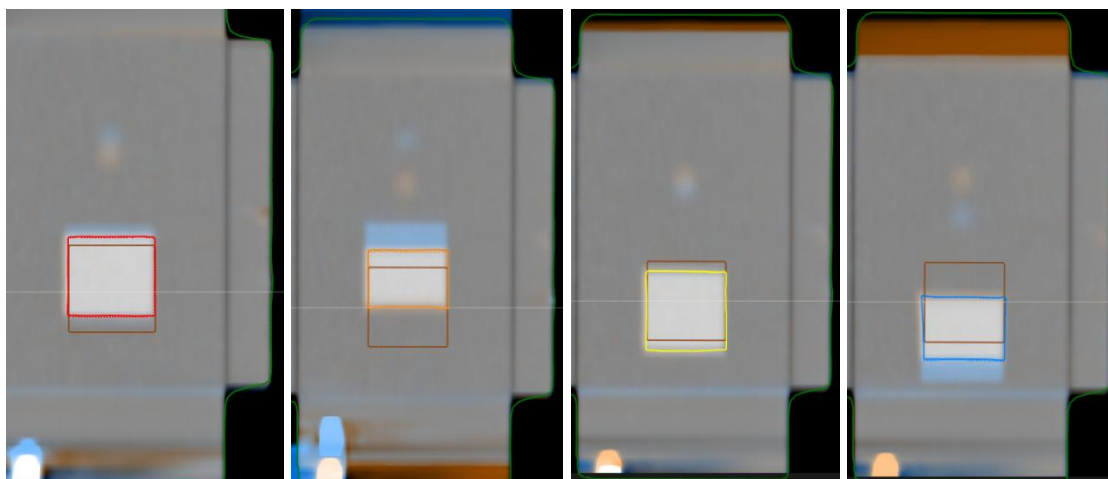


Figura 2.25 - Da sinistra a destra: sezioni del fantoccio in movimento rispettivamente di +0.5 e +1.5, -0.5 e -1.5 cm nelle CT 2, CT 3, CT 4, CT 5.

Si può notare come il secondo approccio di Figura 2.25 sia il migliore: infatti, attraverso l'operazione che mappa le ROI in direzione inversa al campo di spostamento della deformazione, si riesce a seguire il movimento dell'oggetto interno all'inserto.

Questa operazione verrà quindi eseguita su tutti i pazienti che si analizzeranno successivamente e creerà, per ogni CT e per tutte le strutture selezionate, delle nuove ROI nelle immagini fusione, tenendo conto delle deformazioni avvenute. Il tempo impiegato dipende dal numero delle ROI per ogni seduta e dal numero di esami.

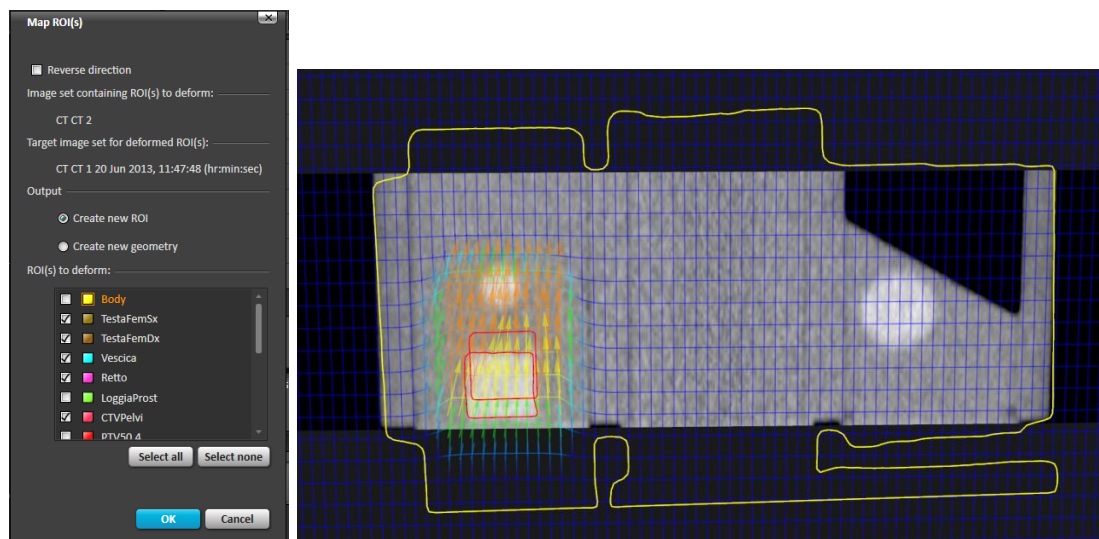


Figura 2.26 - A sinistra: schermata di settaggio del “Map ROI”. A destra: esempio di deformazione eseguita correttamente.

Come si può notare in Figura 2.26, la mappatura della ROI rossa sul fantoccio è avvenuta correttamente, seguendo la forma dell’oggetto, e in direzione inversa al campo di spostamento. In colore blu si vede la griglia di deformazione, mentre le frecce indicano l’intensità e la direzione del campo di deformazione.

2.5 Dose deformation

Il risultato delle co-registrazioni deformabili sono dei campi di deformazione che mappano i voxel delle MVCT nella kVCT di pianificazione. I campi di deformazione vengono utilizzati anche nelle operazioni di *dose deformation*, per mappare la dose relativa alla seduta giornaliera della MVCT con la dose pianificata nella kVCT, attraverso una procedura di interpolazione.

La distribuzione di dose è formattata in una matrice, nella quale ogni elemento rappresenta la dose calcolata in ogni voxel dell’immagine, per ogni frazione del trattamento.

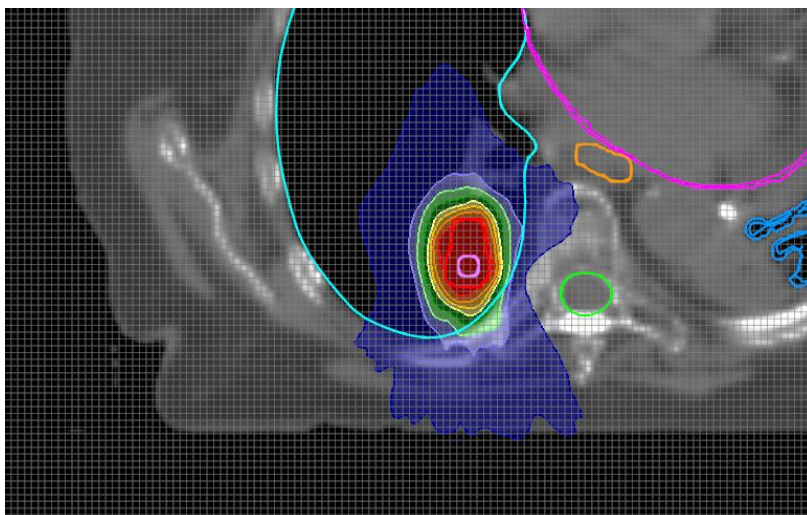


Figura 2.27 - Esempio di matrice di dose per un paziente SBRT polmonare.

L'obiettivo di questa operazione è riuscire in futuro a sommare le dosi deformate (tramite somma delle matrici di dose deformate) quotidianamente, in modo da ottenere la dose cumulativa o *dose accumulation* (totale erogata) per il paziente. Infine si potrebbe in futuro comparare:

- Dose pianificata.
- Dose erogata giornaliera.
- Dose accumulation.
- DVH.

In questa tesi ci si è limitati a confrontare visivamente e quantitativamente (tramite l'estrapolazione delle tabelle “*Dose Statistics*”, vedi sezione 2.6), le dosi relative alla prima seduta con le dosi della CT n-esima per controllare eventuali variazioni significative.

Le dosi sono visualizzabili in diversi *workspace layout*, cioè in differenti modalità di visualizzazione, che servono per rendere più agevole la comparazione tra diversi piani di trattamento (Vedi Figura 2.28).



Figura 2.28 - Workspace layout disponibili.

Una volta deformate le dosi, viene selezionato il secondo workspace layout che permette di confrontare la dose iniziale sulla CT 1 con la dose deformata della CT n-esima selezionata, facendone la differenza. Spostando il cursore sull'immagine differenza, si possono leggere i valori di dose, densità e le HU (Hounsfield Units) dei tessuti di tutte le 3 immagini. A lato di ogni immagine è presente una scala graduata e colorata in modo da distinguere le regioni, in base alle percentuali di dose sul totale di dose erogata.

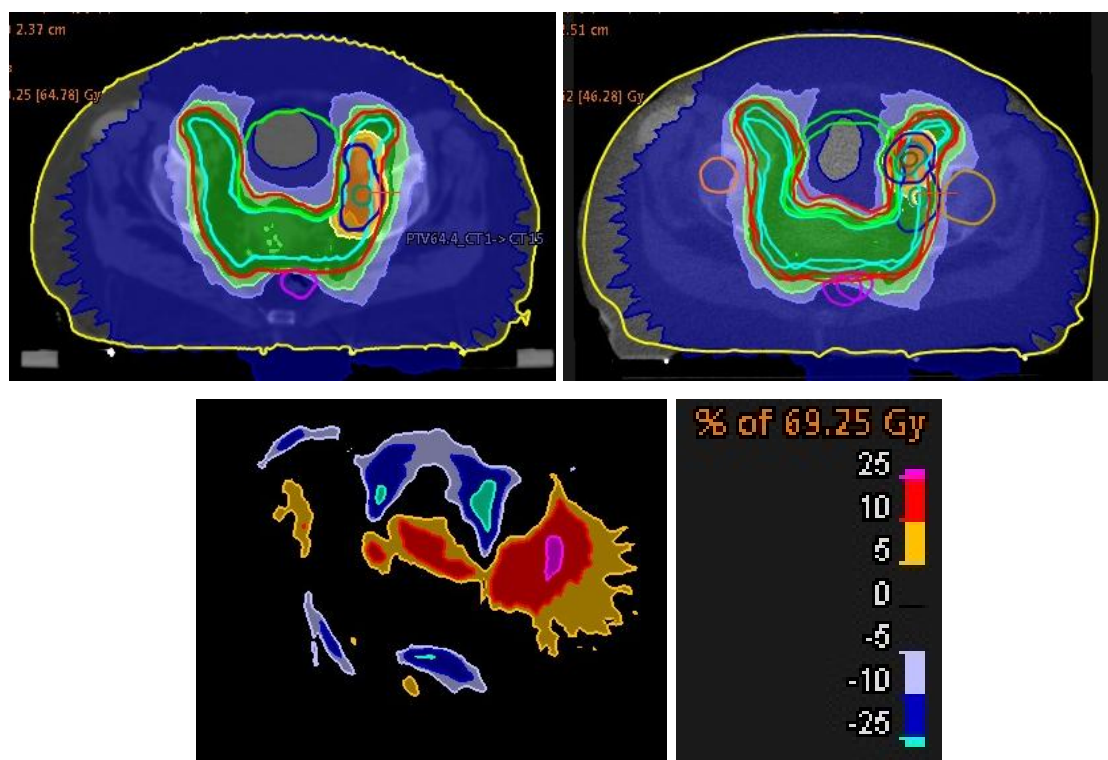


Figura 2.29 - In alto a sinistra: dose pianificata sulla CT 1. In alto a destra: dose deformata sulla CT 15. In basso: dose differenza delle due CT nel secondo workspace layout.

Nella Figura 2.29, in alto a sinistra, è riportata l'immagine delle dosi pianificate sulla kVCT nel piano trasversale per una patologia di ADK prostatico, dove sono presenti anche le strutture; a destra, l'immagine delle dosi deformate sulla CT 15 dello stesso paziente e, in basso a sinistra, la differenza tra le dosi della CT 1 e le dosi della CT 15 ($D_{CT1} - D_{CT20}$) con a fianco la scala colori.

Analizzando i colori dell'immagine differenza e spostando il cursore su di essa, si può capire dove si verificano eventuali variazioni significative di dose, rispetto alla CT 1 iniziale.

La presenza di gradienti di dose, si può verificare leggendo la scala dei colori:

- Punti gialli e rossi, nei quali la differenza di dose si attesta rispettivamente a $5\div 10\%$ di 69.25 Gy; cioè, la distribuzione di dose nei punti considerati nella CT 1 è maggiore rispetto alla distribuzione di dose negli stessi punti della CT 20.
- Punti neri, nei quali non c'è differenza di dose.
- Punti grigi e blu, nei quali si ha una differenza di dose rispettivamente di $-10\div -25\%$ di 69.25 Gy; cioè, la distribuzione di dose nei punti considerati nella CT 1 è minore rispetto alla distribuzione di dose negli stessi punti della CT 20.
- Punti azzurri, in cui la differenza è minore di -25% .

Queste differenze di dose sono all'incirca nulle per le prime sedute in cui ci si aspettano poche variazioni anatomiche del paziente e, conseguentemente campi di deformazione più piccoli, che portano a dosi deformate più simili alle dosi pianificate.

Con l'aumentare del numero di sedute si riscontra invece un aumento delle variazioni di dose.

Per limiti del software non è stato possibile implementare script per l'automazione di queste operazioni di *Dose deformation*, che pertanto sono state effettuate sempre manualmente.

2.6 Estrazione dei dati statistici

Oltre alle deformazioni di dose, è possibile visionare la tabella “*Dose statistics*” sottostante riguardante le statistiche di dose e volume per le ROI deformate.

Dose	ROI	ROI vol. [cm ³]	Dose [Gy]							
			D99	D98	D95	Average	D50	D2	D1	
Plan dose: Empty plan (CT 1)	Body	12332.82	0.31	0.36	0.45	14.31	4.38	59.13	63.54	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	Body	12323.77	0.31	0.35	0.45	14.37	4.42	59.13	63.81	
Plan dose: Empty plan (CT 1)	CTV54	394.58	51.67	51.85	52.12	57.29	55.29	67.05	67.44	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	CTV54	391.41	51.69	51.85	52.14	57.11	55.01	66.86	67.18	
Plan dose: Empty plan (CT 1)	CTV54_CT 1->CT 2	391.11	51.67	51.84	52.11	57.27	55.21	67.05	67.44	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	CTV54_CT 1->CT 2	402.03	51.75	51.90	52.17	57.38	55.28	66.97	67.30	
Plan dose: Empty plan (CT 1)	CTV66	40.6	64.21	64.40	64.66	66.10	66.12	67.93	68.24	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	CTV66	40.3	64.40	64.53	64.83	66.00	66.00	67.47	67.81	
Plan dose: Empty plan (CT 1)	CTV66_CT 1->CT 2	40.27	64.21	64.40	64.66	66.08	66.11	67.84	68.17	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	CTV66_CT 1->CT 2	43.17	64.43	64.55	64.82	66.04	66.01	67.73	67.97	
Plan dose: Empty plan (CT 1)	mandibola	89.76	10.50	11.18	13.55	43.35	43.66	66.94	67.63	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	mandibola	87.59	10.47	11.20	13.66	43.82	44.44	66.94	67.82	
Plan dose: Empty plan (CT 1)	mandibola_CT 1->CT 2	87.46	10.56	11.24	13.63	43.46	43.77	66.93	67.64	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	mandibola_CT 1->CT 2	77.07	10.47	11.14	13.24	42.73	42.62	66.70	67.48	
Plan dose: Empty plan (CT 1)	midollo	36.63	0.13	0.13	0.15	8.16	2.21	30.37	31.12	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	midollo	32	0.13	0.13	0.15	8.12	2.18	30.49	31.18	
Plan dose: Empty plan (CT 1)	midollo_CT 1->CT 2	31.81	0.13	0.13	0.15	8.03	2.19	30.33	31.09	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	midollo_CT 1->CT 2	34.18	0.13	0.14	0.16	8.54	5.72	30.49	31.14	
Plan dose: Empty plan (CT 1)	parotidedx	31.28	8.80	9.45	10.74	23.42	19.32	54.56	58.49	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	parotidedx	30.71	8.55	9.23	10.57	23.46	19.71	53.84	57.92	
Plan dose: Empty plan (CT 1)	parotidedx_CT 1->CT 2	30.69	8.83	9.55	10.76	23.23	19.22	53.33	57.18	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	parotidedx_CT 1->CT 2	28.84	8.70	9.36	10.63	23.28	19.45	53.84	57.41	
Plan dose: Empty plan (CT 1)	parotidesx	31.78	5.98	6.15	6.55	18.28	10.59	60.17	61.85	
Evaluation dose: Empty plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1	parotidesx	31.09	6.00	6.17	6.55	19.67	11.05	61.40	62.36	
Plan dose: Empty plan (CT 1)	parotidesx_CT 1->CT 2	31.05	5.99	6.13	6.55	18.20	10.54	60.15	61.83	

Figura 2.30 - Tabella “*Dose statistics*”.

Questa tabella riporta 4 variabili:

1. Dose: nome della dose.
2. ROI: nome della ROI e delle strutture ottenute mediante mapping nella direzione inversa che hanno il suffisso “_CT 1->CT N”.
3. ROI vol. [cm³]: volume delle ROI in cm³.
4. Dose [Gy] delle singole ROI suddivise da sottocategorie dosimetriche:
 - D99: valore di dose in Gy erogata al 99% del volume (equiparabile a D_{min}).
 - D98: valore di dose in Gy erogata al 98% del volume.
 - D95: valore di dose in Gy erogata al 95% del volume.
 - Average: valore di dose medio in Gy.
 - D50: valore di dose in Gy erogata al 50% del volume.
 - D2: valore di dose in Gy erogata al 2% del volume.
 - D1: valore di dose in Gy erogata al 1% del volume (equiparabile a D_{max}).

Solitamente, per ogni ROI (tranne il “*Body*”) si hanno 4 righe con nomi differenti. Per esempio, se confrontiamo la CT 1 e la CT 2 abbiamo per ogni ROI le seguenti righe:

1. Plan dose: plan (CT1) → Dose pianificata sul volume della ROI iniziale nella CT 1.
2. Evaluation dose: plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1 → Dose deformata sul volume della ROI iniziale nella CT 1.
3. Plan dose: plan (CT 1) → Dose pianificata sul volume della ROI deformata nella CT 2.
4. Evaluation dose: plan (CT 1) deformed to: CT 2 using: HybridDeformation 2-1 → Dose deformata sul volume della ROI deformata nella CT 2 .

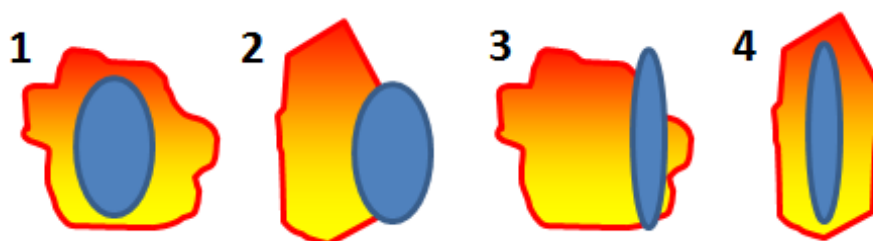


Figura 2.31 - Casi relativi alle righe 1,2,3,4. In blu è rappresentato il volume della ROI. In rosso/giallo la dose.

Ai fini della nostra analisi, le righe che ci interessano sono:

- 1 → dose e volume originali relativi alla pianificazione.
- 3 → dose pianificata e volume deformato, cioè quello che succede adesso senza ART.
- 4 → dose che si vorrebbe erogare, che tiene conto delle deformazioni.

Questi dati venivano inizialmente estrapolati tramite screenshot, salvati in una memoria esterna, importati in un software OCR che li convertiva in dati Excel e infine sistemati in base alle nostre esigenze.

Questa operazione veniva ripetuta per ogni seduta del paziente e richiedeva molto tempo sia per la conversione dei dati (per un paziente di 25 CT all’incirca 50 immagini) sia per la loro correzione dovuti ad errori del software utilizzato.

L’automazione creata con lo script specifico “*GetDoseStatistics_db.py*”, ha permesso di ridurre notevolmente i tempi e di estrapolare, oltre ai dati della tabella, anche altri valori utili ai fini della statistica.

Questi valori, sono stati scritti in un file di testo .txt (diverso per ogni paziente), importati in Excel e infine analizzati tramite grafici pivot.

Una parte, di esempio, del database scaturito, è visibile in Tabella 4 e Tabella 5:

Patologia	Paziente	ROI	CT	Deformation	Vol.	V% (rif. kVCT)	V% (rif. MVCT)
H&N	*****	CTV54	1	1 - ROI on kVCT	166,81	100,00	
H&N	*****	CTV54	2	2 - ROI on MVCT	165,12	98,99	
H&N	*****	CTV54	2	3 - Def ROI on kVCT	164,83	98,81	
H&N	*****	CTV54	2	4 - Def ROI on MVCT	162,73	97,55	100,00
H&N	*****	CTV60	1	1 - ROI on kVCT	107,90	100,00	
H&N	*****	CTV60	2	2 - ROI on MVCT	106,74	98,92	
H&N	*****	CTV60	2	3 - Def ROI on kVCT	106,69	98,88	

Tabella 4 - Prima parte dati statistici di volume

D99	D99%	D98	D98%	D95	Aver.	Dose Av%	D50	D2	D2%	D1	D1%	HI
52,2	100,0	52,4	100,0	52,7	55,4	100,0	53,9	66,0	100,0	66,4	100,0	0,5
40,9	78,2	45,2	86,1	50,7	55,0	99,1	54,0	65,6	99,4	66,0	99,5	0,3
52,2	100,0	52,4	100,0	52,7	55,4	99,9	53,9	66,0	99,9	66,4	99,9	0,2
52,3	100,1	52,5	100,1	52,8	55,6	100,2	54,0	65,9	99,8	66,3	99,9	0,2
58,5	100,0	58,7	100,0	58,9	61,1	100,0	59,8	66,3	100,0	66,4	100,0	0,1
58,7	100,2	58,8	100,2	59,0	61,1	99,9	59,8	66,2	99,8	66,3	99,8	0,1
58,6	100,0	58,7	100,0	58,9	61,1	100,0	59,8	66,3	100,0	66,4	99,9	0,1

Tabella 5 - Seconda parte dati statistici di dose

In radioterapia, l'obiettivo è, da sempre, erogare la dose massima al volume target in modo omogeneo (linee guida ICRU 83), cercando di erogare meno dose possibile agli organi attorno.

La distribuzione di dose può essere visualizzata nel DVH e nelle linee di isodose, per definire parametri come la dose massima, minima, media e mediana erogate ad ogni volume di interesse.

Esiste però un parametro, chiamato Homogeneity Index (HI), che permette di valutare quantitativamente la qualità del piano di trattamento ed è calcolato nella sua formula originale come rapporto tra la massima e la minima dose nel volume target.

Sostanzialmente, si tratta di un parametro fisico per analizzare l'uniformità della distribuzione di dose nel volume target (PTV o CTV), ed un valore basso del rapporto indica una distribuzione di dose più omogenea entro il volume considerato.

In letteratura l'HI è calcolato in diversi modi, tra cui il seguente, adottato in questo lavoro:

$$HI = \frac{D_2}{D_{98}} \quad (25)$$

Dove D_2 e D_{98} sono rispettivamente la dose massima al 2% del volume target e la dose minima al 98% del volume target. Per il calcolo, i valori di dose sono stati rapportati alla prima seduta, in modo tale che per essa risulta il rapporto pari a 1.

Se il rapporto tende a 1, si può dire che la dose assorbita è all'incirca omogenea nel volume di interesse, anche se a volte non descrive così accuratamente l'omogeneità ed è necessario considerare l'intero DVH.

L'HI è stato analizzato per il solo PTV perché per gli organi a rischio, che confinano o si sovrappongono a volte con il PTV, può capitare che si abbia una dose elevata in un certo punto e molto bassa nel punto vicino, creando una grande inomogeneità di dose e quindi un HI di molto superiore o inferiore a 1.

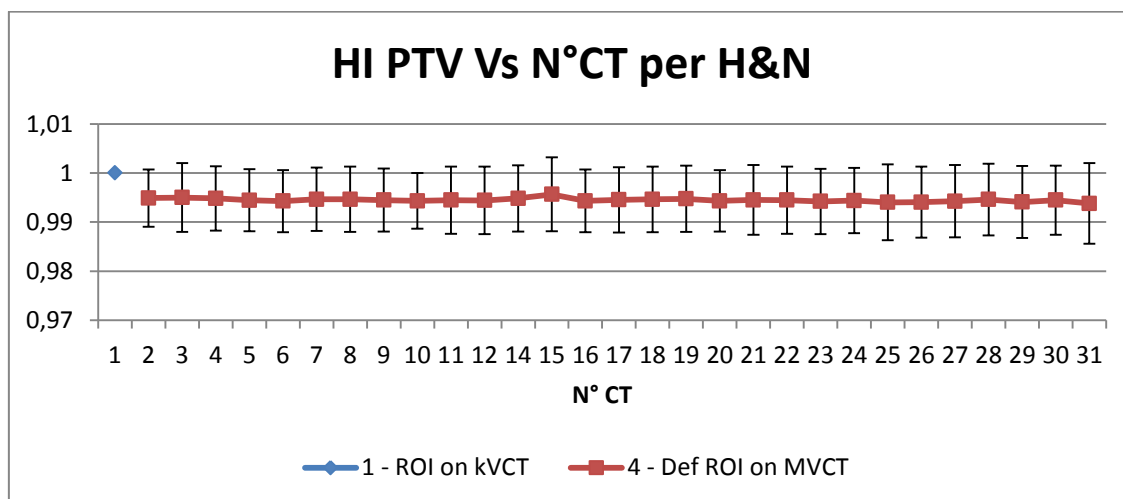


Figura 2.32 - Andamento medio di HI per PTV nelle sedute per trattamento H&N.

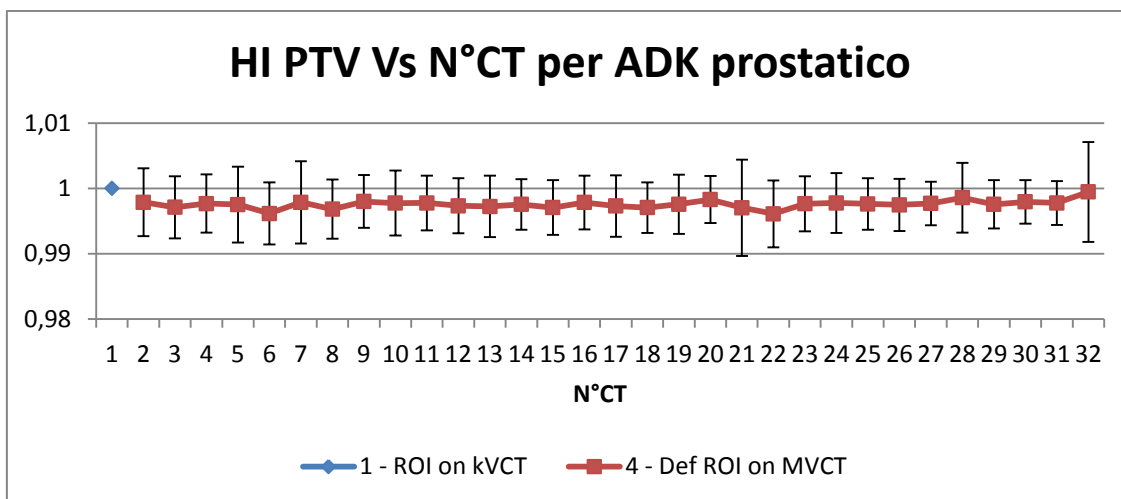


Figura 2.33 - Andamento medio di HI per PTV nelle sedute per trattamento ADK prostatico

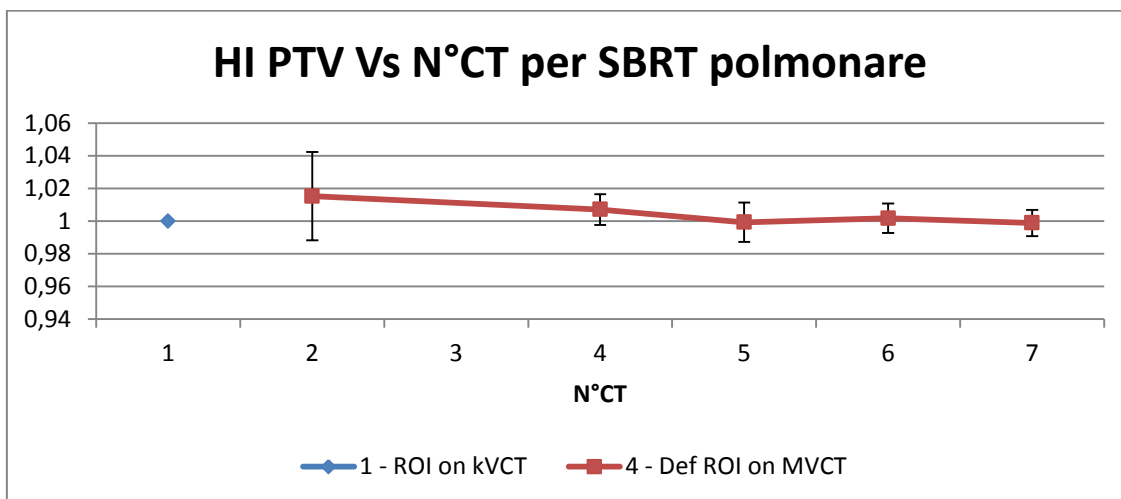


Figura 2.34 - Andamento medio di HI per PTV nelle sedute per trattamento SBRT polmonare

Dai grafici risulta che l'HI della riga 1 della tabella dose statistics (dose relativa alla prima seduta) è all'incirca uguale all'HI della riga 4 (dose deformata nella MVCT) entro gli intervalli di confidenza. Ci si aspetta, infatti, che l'HI per il PTV alla prima seduta sia omogeneo e che venga rispettata l'omogeneità delle dosi anche nelle sedute successive.

2.7 Scripting automation

Tutti i processi descritti sopra, che portano alla realizzazione dell'ART, non sarebbero possibili nella pratica clinica giornaliera senza l'utilizzo degli script.

Sarebbero necessarie, infatti, una o più persone dedite a quest'attività giorno dopo giorno, con problematiche relative ai costi aggiuntivi del personale e ai possibili errori che questi potrebbero compiere nell'eseguire il più delle volte le stesse azioni in forma ripetitiva.

I vantaggi degli script sono riassumibili nei punti seguenti:

1. Automazione: ogni giorno nella pratica clinica, ci sono spesso numerosi compiti o procedure che devono essere eseguite manualmente. Queste variano da clinica a clinica e non possono perciò essere pienamente supportate dal TPS nativo. Attraverso gli script, possono invece essere automatizzate:
 - Procedure cliniche specifiche (es. controllo della qualità di un piano prima dell'approvazione ecc.).
 - Creazione dei piani (es. definizione delle strutture, *beam placement*, definizione degli obiettivi e *clinical goals* attraverso i protocolli di pianificazione ecc.).
 - Operazioni ripetitive (es. import esami dei pazienti ecc.).
2. Connettività: anche se molti sistemi interagiscono attraverso lo standard DICOM, si possono verificare incompatibilità tra software (es. calcolo indipendente delle dosi e verifica attraverso i sistemi di controllo qualità). Gli script possono far interagire questi sistemi.
3. Flessibilità: gli script possono essere usati per creare nuove funzioni che non sono disponibili (es. può essere integrato un sistema di rilevamento automatico dei markers). Possono essere impiegati per avere accesso ad opzioni che non sono disponibili nella *Graphical User Interface* (GUI) (es. export delle immagini relative ai piani di dose e immagini di tutti i punti di controllo).

Il linguaggio usato per gli script, IronPython, è la versione .NET di Python e possiede le seguenti caratteristiche:

1. Linguaggio completo di programmazione (semplice e documentato), open-source e di alto livello.
2. Essendo integrato con Microsoft .NET Framework e Windows Presentation Foundation (WPF), permette di creare e visualizzare finestre con componenti grafici.

3. Offre accesso a tutte le capacità del S.O., così come alle altre applicazioni (es. creare report pdf, scrivere file di testo e interfacciarsi direttamente con altre applicazioni *scriptable* come Microsoft Excel, comunicare con altri computer ecc.).
4. È un linguaggio dinamico: compie cioè molte operazioni durante il tempo di esecuzione, piuttosto che durante il tempo di compilazione, risultando più veloce rispetto ai linguaggi statici come il C.

Durante questo lavoro, sono stati perciò sviluppati una serie di programmi, in grado di automatizzare quasi tutte le operazioni di ART all'interno del TPS, in quanto non è risultato possibile per limiti intrinseci del TPS implementare uno script per il *Dose deformation*.

L'attività di programmazione è iniziata utilizzando la funzione di registrazione degli script all'interno del TPS (funzione particolare che permette di registrare le proprie azioni sottoforma di codici IronPython), ma questa funziona solo per alcuni moduli del TPS e per i nostri scopi è risultata quasi inutile.

Grazie ad alcuni *white paper* su RayStation presenti in internet, è stato possibile capire la sintassi di alcune istruzioni base ma, la ricerca delle istruzioni necessarie per le operazioni di ART è stata dettata soprattutto dall'intuizione, fino a quando non è stato possibile leggere (tramite il database SQL con cui comunica il programma) il contenuto di una .dll del software. Questa .dll, recava le informazioni sulle azioni possibili (alcune tramite script, altre no) all'interno del TPS; e anche se la lettura dei parametri delle istruzioni non è stata agevole, si è arrivati dopo 2 mesi alla scrittura dei primi programmi, fino ad arrivare dopo 5 mesi ad un totale di 13 script, riassunti nei paragrafi seguenti.

Alle difficoltà sopra descritte, si aggiungono anche le problematiche relative agli aggiornamenti di versione del TPS. Infatti, durante il periodo di tesi, si sono susseguite le seguenti versioni:

1. RayStation 3.5.0.21 "Not for clinical use".
2. RayStation 3.99.0.7 "Not for clinical use".
3. RayStation 4.0.1.4 "For clinical use".

Ognuna di esse ha proprie caratteristiche e peculiarità. I primi script sono stati sviluppati inizialmente nella versione 3.5.0.21. A seguito dell'aggiornamento alla versione 3.99.0.7 di RayStation, è stato necessario riscrivere in molti casi i codici contenuti negli script, a causa di variazioni nelle sintassi e nei parametri delle istruzioni stesse.

Con la versione 4.0.1.4, alcuni degli script sono stati poi esportati dalla 3.99.0.7 e adattati a quest'ultima versione, che presenta però alcuni svantaggi.

Mancano, infatti, due istruzioni utili ai fini di questa ricerca e che verranno implementate successivamente nella versione 4.5. Queste istruzioni sono:

- *GetDeformationGrid*: permette di calcolare, prima che venga effettuata la deformazione ibrida, i valori della griglia di deformazione, in modo tale da inserirli nell'istruzione successiva che esegue la deformazione delle immagini. Questi valori sono essenziali, altrimenti lo step successivo di *Deform dose* potrebbe in molti casi non essere possibile.
- *MapRoiGeometriesDeformably*: esegue il mapping delle ROI in verso opposto al campo di deformazione. Risulta necessaria per ridurre notevolmente i tempi di questa operazione.

Gli script sono stati sviluppati all'interno dell'interfaccia di RayStation con un tool dedicato visibile nella Figura 2.35.

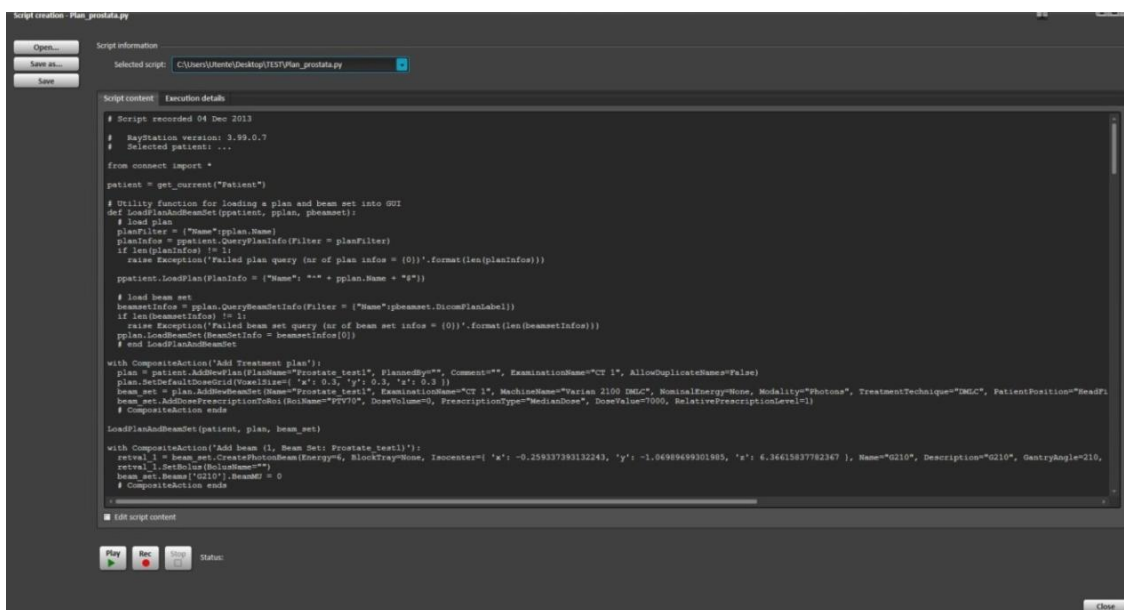


Figura 2.35 - Finestra per la creazione degli script, con uno script aperto

Questo tool permette di scrivere gli script, modificarli e salvarli in una cartella di sistema in modo che siano richiamabili all'occorrenza. Permette inoltre di validare gli script, cioè di renderli disponibili a tutti gli utenti nell'interfaccia grafica di RayStation, attraverso la creazione di pulsanti specifici. Oltre a questo tool, si è utilizzato un

software gratuito chiamato Notepad++, che permetteva l'editing degli script in ambiente esterno.

2.7.1 Script ART

Questa serie di script è stata realizzata per automatizzare le operazioni di ART sui pazienti.

Script	Descrizione
1. <i>ImportPatient_WithoutRTPLAN.py</i>	Importa in ordine di data tutte le serie d'immagini del paziente in base al filtro impostato (es. PatientID), tranne il file RTPLAN.
2. <i>RigidRegistration.py</i>	Esegue la co-registrazione rigida tra gli esami del paziente, impostando la CT 1 come target image set e la CT n-esima come source image set. Inoltre, nella co-registrazione sono permesse le rotazioni e l'algoritmo si focalizza sulle ossa nelle immagini coinvolte.
2.1. <i>RigidRegistration_NoRotations.py</i>	Esegue la co-registrazione rigida, bloccando le possibili rotazioni delle immagini coinvolte.
2.2. <i>RigidRegistration_NoRotations_NoFocusOnBone.py</i>	Esegue la co-registrazione bloccando le possibili rotazioni e senza focalizzarsi sulle strutture ossee.
3. <i>CopyROI.py</i>	Copia le ROI dalla prima seduta a tutte le altre.
4. <i>HybridDeformation_CC.py</i>	Esegue la deformazione ibrida, con algoritmo di cross-correlazione, tra gli esami del paziente.
4.1. <i>HybridDeformation_CCmixedMI.py</i>	Esegue la deformazione ibrida, con

	algoritmo misto di cross-correlazione e mutual information, tra gli esami del paziente.
4.2. <i>HybridDeformation_MI.py</i>	Esegue la deformazione ibrida, con algoritmo di mutual information, tra gli esami del paziente.
5. <i>MapRoiReverse.py</i>	Esegue il Map ROI inverso al campo di deformazione, tra le CT della deformazione ibrida.
5.1. <i>MapRoiReverse_NoAbort.py</i>	Esegue il Map ROI inverso e non interrompe il processo, nel caso in cui ci sia qualche errore nel campo di deformazione.
6. <i>GetDoseStatistics_db.py</i>	Estrapola in un file .txt le dosi e i volumi (dati relativi al secondo workspace layout) della tabella Dose Statistics del software e aggiunge alcune colonne necessarie per la statistica.
6.1. <i>GetDoseStatistics_solo_dosi.py</i>	Estrapola solo le dosi contenute nella tabella Dose Statistics scrivendole in un file di testo.
6.2. <i>GetDoseStatistics_dosi_volumi.py</i>	Estrapola dosi e volumi contenuti nella tabella Dose Statistics scrivendoli in un file di testo.
7. <i>ImportScriptInFolder.py</i>	Importa in successione gli script indicati all'interno del listato del programma. Inoltre, scrive un file di testo con il tempo totale di esecuzione.

Tabella 6 - Script per le operazioni di ART

2.7.2 Script di controllo

Questa serie di script, è stata realizzata per controllare il risultato degli script citati sopra per l'ART sui pazienti.

Script	Descrizione
8. <i>RotationValue.py</i>	Calcola l'angolo di rotazione tra le immagini co-registrate, attraverso la matrice di rotazione delle stesse. I valori degli angoli di rotazioni vengono scritti in un file di testo per il successivo controllo.
9. <i>ROI_position.py</i>	Controlla se le coordinate assolute del centro della ROI "Body" (x,y,z) sono maggiori delle coordinate assolute (x,y,z) dei bordi dell'immagine DICOM nella seduta considerata. In caso affermativo, registra in un file di testo la necessità di un controllo da parte dell'utente nell'esame considerato.
10. <i>GeometryCheck.py</i>	Crea un POI in ogni seduta nel punto (0,0,0) e calcola, scrivendo in un file di testo i seguenti valori: <ul style="list-style-type: none"> • Coordinate x, y, z del centro della ROI indicate (es. Parotide Dx e Parotide Sx) • Modulo dei vettori che congiungono il centro delle ROI con il centro di ogni POI • Angolo tra i vettori • Area del triangolo creato con i vettori • Versore dei vettori • Distanza tra le ROI es. (Parotide Dx e Parotide Sx)

Tabella 7 - Script di controllo.

2.7.3 Script generali

Questa serie di script, è stata realizzata per altri scopi, legati soprattutto alla comodità.

Script	Descrizione
11. <i>Delete_all_doses.py</i>	Elimina tutte le dosi deformate calcolate nel modulo “ <i>Dose evaluation</i> ”.
12. <i>New_console.py</i>	Apri una nuova sessione del TPS.
13. <i>GetPatientName_Id.py</i>	Interroga il database dei pazienti e restituisce in output i nomi e gli ID dei pazienti.

Tabella 8 - Script generali.

3 Risultati e criticità dell'automazione

3.1 Tempi a confronto

Sono stati raccolti i dati per un totale di 51 pazienti e raggruppati successivamente in base alla patologia, per un'analisi dei tempi più corretta. Le patologie trattate sono state:

- 23 Testa e collo o Head&Neck.
- 8 Polmoni o SBRT polmonare.
- 20 Prostata o ADK prostatico.

I tempi graficati dipendono da:

1. Velocità dell'hardware sul quale è installato il software RayStation. Si riepilogano di seguito i dati della Workstation:
 - CPU Intel Xeon X5650 2.67GHz, 6 core, 12 processori.
 - RAM 24 GB.
 - Sistema operativo 64 bit.
 - Scheda grafica NVIDIA Quadro 2000D 1 GB GDDR5.
 - Due unità disco Adaptec Array SCSI da 136 GB e 930 GB.
2. Multitasking e RAM: per raggiungere una certa quantità di pazienti è stato necessario eseguire tre o più sessioni di RayStation contemporaneamente, cercando di aumentare la statistica, ma allo stesso tempo, aumentando relativamente il tempo impiegato per concludere ognuno di essi e rendendo i tempi fortemente variabili anche per pazienti aventi lo stesso numero di sedute nel trattamento radioterapico.
Il sistema operativo, inoltre, alloca per ogni sessione di RayStation aperta una certa quantità di memoria (RAM + file di paging virtuale) che è variabile da 7GB a 13GB, in base alla quantità di dati da visualizzare ed elaborare. L'apertura di tre o più sessioni comporta un sovraccarico importante per tutto il sistema, generando a sua volta rallentamenti evidenti e, in alcuni casi, il blocco del sistema.
Per aggirare questo ostacolo, dopo aver elaborato 3 pazienti sarebbe necessario spegnere la workstation, in modo tale da pulire al riavvio successivo la RAM e liberare spazio utile per le successive sessioni di lavoro.
3. Ottimizzazione del codice degli script.

4. Patologia interessata: i tempi possono variare a causa della diversa complessità degli organi.

Nella Tabella 9, sono stati inseriti i valori medi dei tempi per le operazioni di ART indicate, introducendo anche il guadagno in termini percentuali e differenziando i due approcci: manuale (M) e automatico (A).

H&N	A (min)	M (min)	Time gain (%) ((M-A)/M)*100
CT/MVCT import - Rigid registration	30	50	40
CopyROI - Ibryd deformation - Mapping ROI	180	240	25
Statistical data extraction	60	350	83
Totale	270	640	148
ADK prostatico	A (min)	M (min)	Time gain (%) ((M-A)/M)*100
CT/MVCT import - Rigid registration	43	55	22
CopyROI - Ibryd deformation - Mapping ROI	140	240	42
Statistical data extraction	60	360	83
Totale	243	655	147
SBRT polmonare	A (min)	M (min)	Time gain (%) ((M-A)/M)*100
CT/MVCT import - Rigid registration	7	10	30
CopyROI - Ibryd deformation - Mapping ROI	15	50	70
Statistical data extraction	3	70	96
Totale	25	130	196
Media totale			164

Tabella 9 - Riassunto dei tempi medi (min) per effettuare Adaptive su un paziente di ogni patologia (trattamento completo), sia mediante script che manualmente, corredati da percentuali di guadagno.

Il guadagno di tempo medio in termini percentuali, su tutto il processo ART con l'utilizzo degli script è di circa il 164%. Il risparmio di tempo maggiore è nelle operazioni di estrazione statistica dei dati, che prima dell'implementazione degli script veniva eseguita manualmente.

Per le patologie più complesse e i trattamenti più lunghi, come quelli di H&N e ADK prostatico i tempi sia per le operazioni manuali che automatiche si equivalgono e sono molto maggiori (circa 9 volte per A e 5 volte per M) rispetto alle patologie più semplici e con un numero minore di sedute e ROI come SBRT polmonare. Il tempo per eseguire

L'ART manualmente nel SBRT polmonare è proporzionale alle altre patologie, se si pensa al numero di CT medie per ogni patologia. Infatti, considerando all'incirca 6 CT in media per SBRT polmonare e 30 CT in media per ADK prostatico e H&N, risulta evidente un fattore 5 come moltiplicatore sia del numero di CT che dei tempi. Discorso differente per i tempi relativi all'approccio automatico che, nel caso SBRT polmonare, risultano essere 9 volte minori; imputabile al fatto che si tratta di una patologia più semplice che il calcolatore riesce ad elaborare più velocemente.

In clinica si vorrebbe implementare l'ART quotidianamente sia per trattamenti con Tomotherapy che con LINAC equipaggiati con cone-beam.

Nella realtà il TPS dovrebbe analizzare automaticamente (con alcuni passaggi manuali) solamente una CT al giorno per ogni paziente in trattamento. L'ART per singola CT e per paziente, considerando anche le operazioni manuali avrebbe i seguenti tempi suddivisi per patologia:

- H&N = 30 minuti.
- ADK prostatico = 20 minuti.
- SBRT polmonare = 15 minuti.

Svolgendo le operazioni di ART in modo automatico durante le ore notturne, con 12 ore a disposizione all'incirca, il carico di lavoro sostenibile sarebbe quantificabile in:

- H&N \approx 24 pazienti.
- ADK prostatico \approx 36 pazienti.
- SBRT polmonare \approx 48 pazienti.

Di seguito, suddivisi per patologia, si sono graficati i tempi necessari per compiere Adaptive RT sui pazienti.

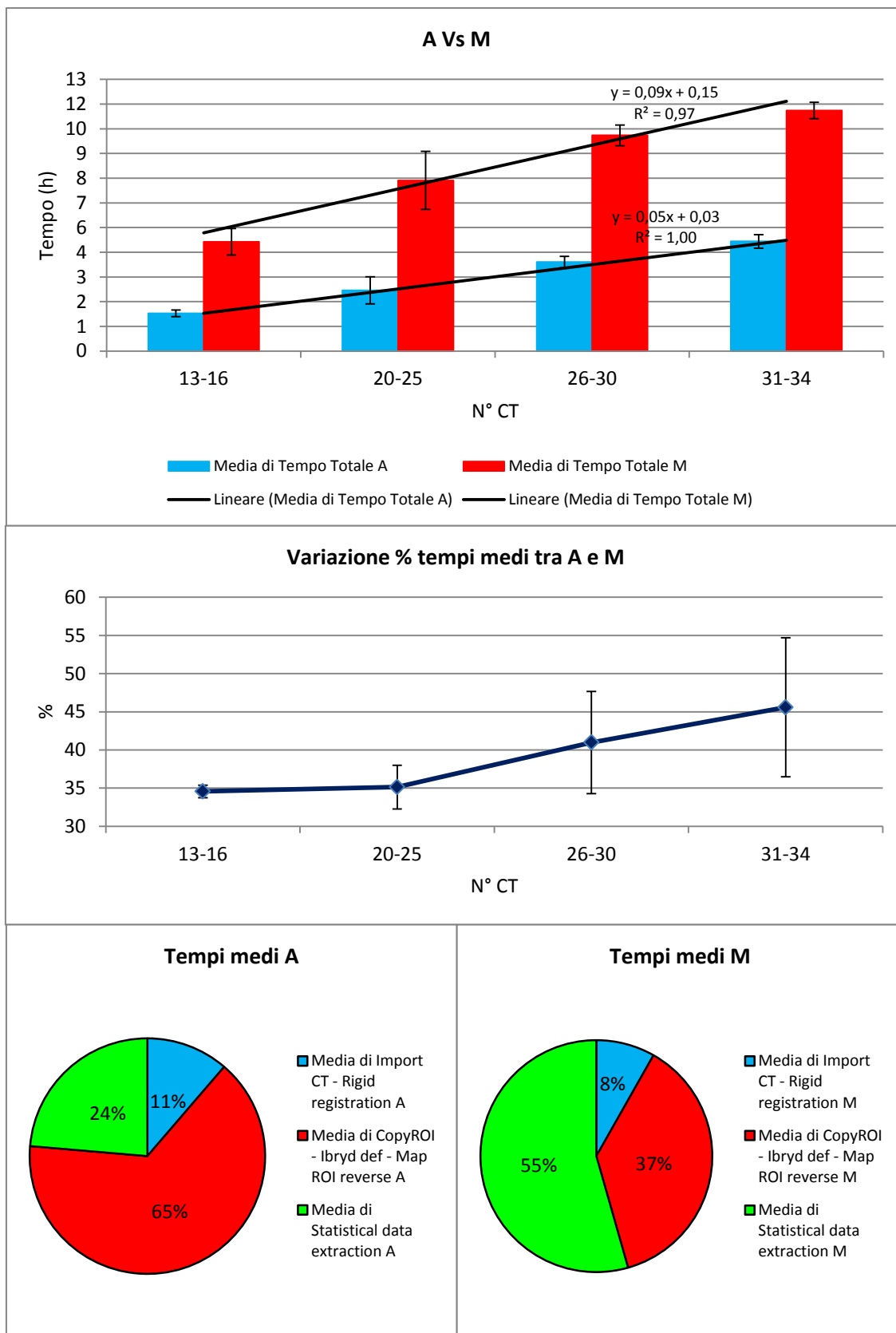


Figura 3.1 - Patologia H&N. A partire dall'alto: confronto dei tempi medi impiegati manualmente e automaticamente; variazione percentuale media tra i tempi automatici e manuali; percentuali di tempo medie impiegate sul totale da ogni script per le operazioni indicate.

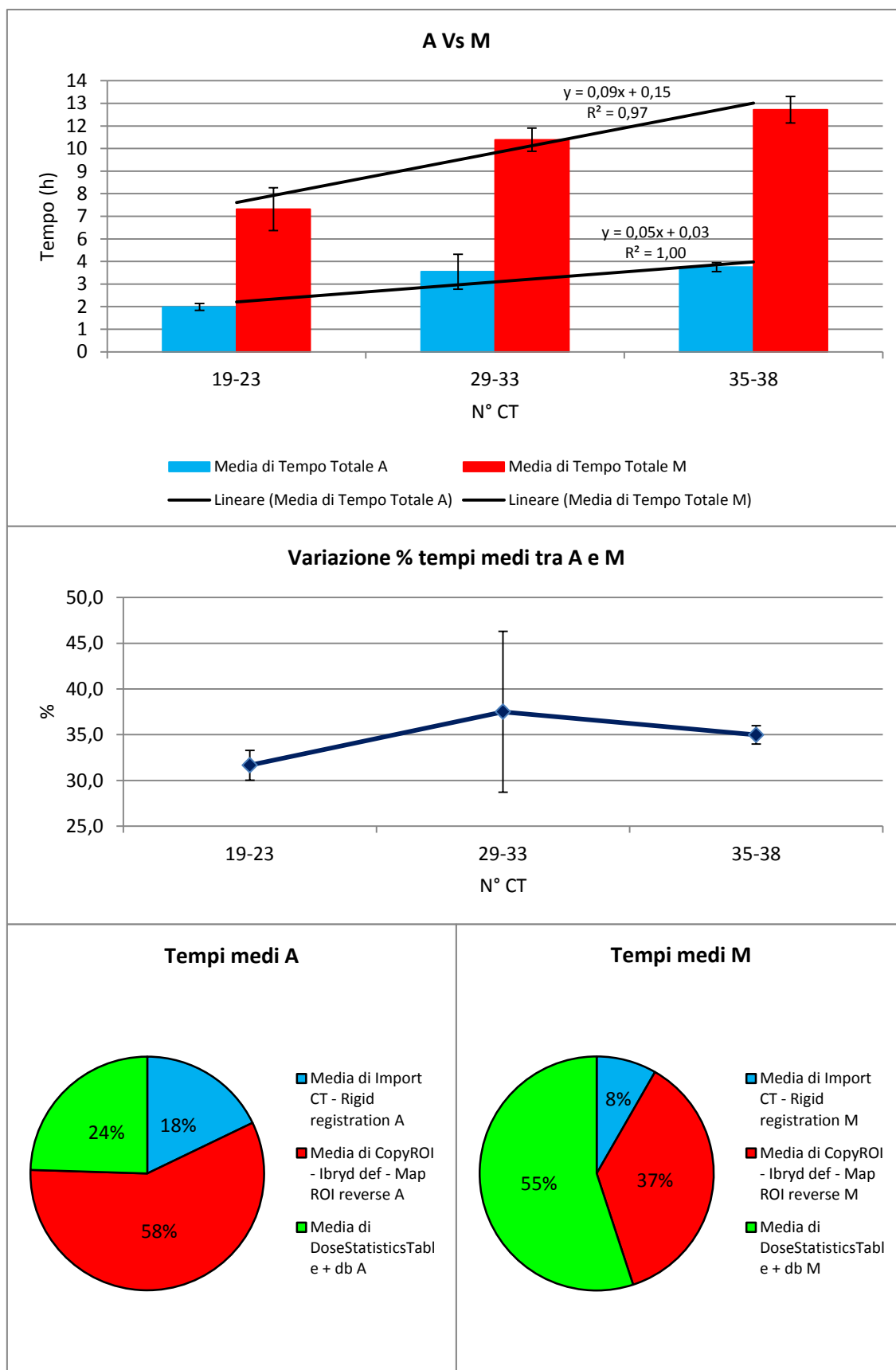


Figura 3.2 - Patologia ADK prostatico. A partire dall'alto: confronto dei tempi medi impiegati manualmente e automaticamente; variazione percentuale media tra i tempi automatici e manuali; percentuali di tempo medie impiegate sul totale da ogni script per le operazioni indicate.

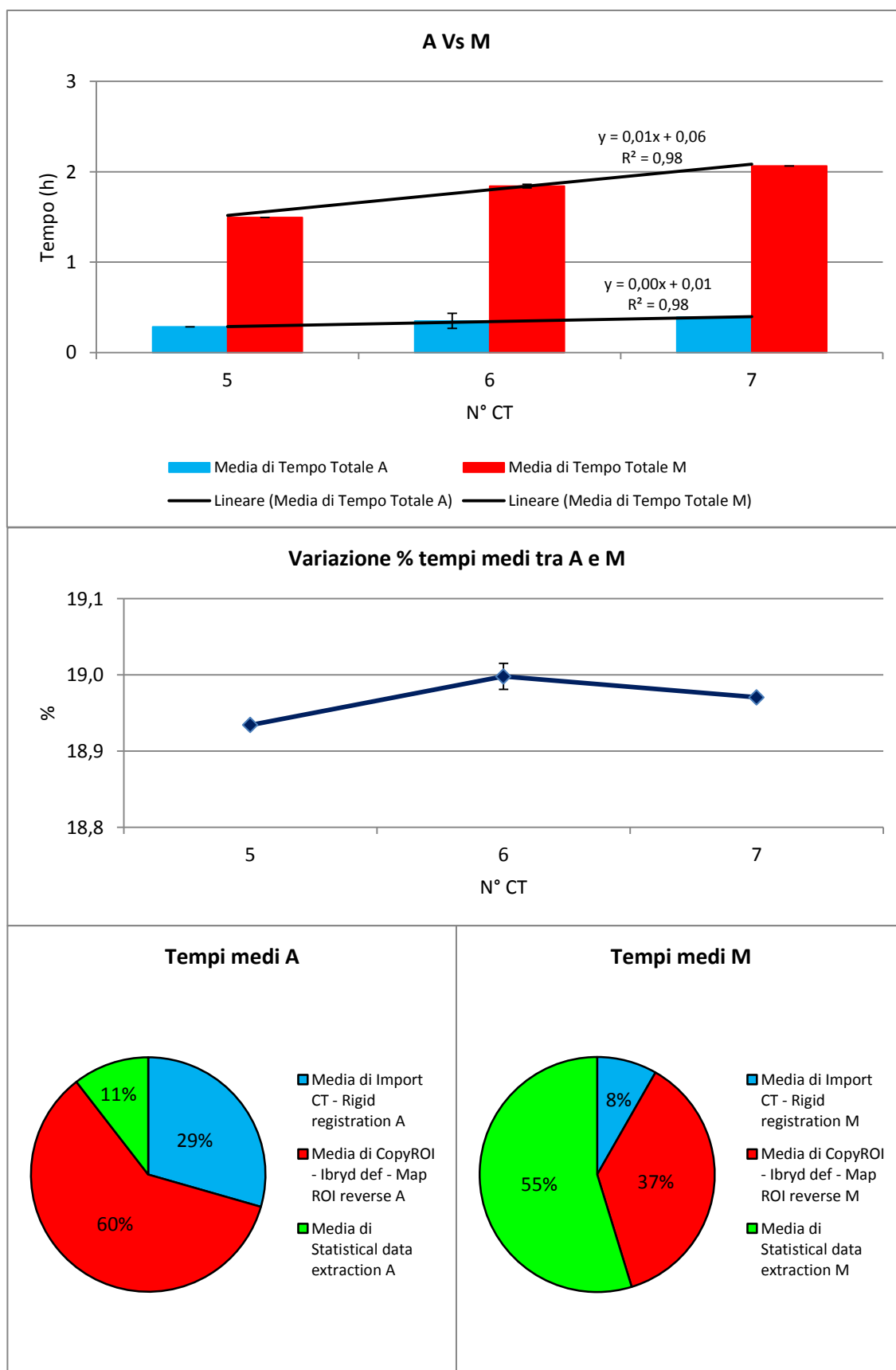


Figura 3.3 - Patologia SBRT polmonare. A partire dall'alto: confronto dei tempi medi impiegati manualmente e automaticamente; variazione percentuale media tra i tempi automatici e manuali; percentuali di tempo medie impiegate sul totale da ogni script per le operazioni indicate.

3.2 Analisi dei risultati

I tempi del primo grafico, di ogni patologia, sono stati graficati in funzione del numero di CT suddivise a loro volta in gruppi, distinti in base al periodo di trattamento.

Nella Tabella 10 per ognuno di questi, si può notare la differenza evidente, in termini di tempo, tra l'approccio manuale (colore rosso) e l'approccio automatico (colore azzurro), quantificabile in termini percentuali nel secondo grafico di ogni patologia.

H&N			ADK prostatico			SBRT polmonare		
Sedute	N° sett.	%	Sedute	N° sett.	%	Sedute	N° sett.	%
13-16	3	35	19-23	4-5	32	5	1	19
20-25	4-5	35	29-33	6	37	6	1	19
26-30	5-6	41	35-38	6-7	35	7	2	19
31-34	6-7	46						

Tabella 10 - Riassunto percentuali di variazione tra approccio manuale e automatico per le tre diverse patologie.

La crescita dei tempi, col numero di CT ha un andamento lineare per tutte le patologie per entrambe le serie di dati entro gli intervalli d'incertezza, con coefficienti:

- H&N: $R^2 = 0.97$ per M e $R^2 = 1$ per A.
- ADK prostatico: $R^2 = 0.97$ per M e $R^2 = 1$ per A.
- SBRT polmonare: $R^2 = 0.98$ per M e $R^2 = 0.98$ per A.

L'intercetta fornisce indicazioni sui tempi latenti nei quali non vengono effettuate operazioni (es. apertura RayStation, cambio di schermata ecc.):

- H&N: intercetta a 180 minuti per M e 36 minuti per A.
- ADK prostatico: intercetta a 180 minuti per M e 36 minuti per A.
- SBRT polmonare: intercetta a 72 minuti per M e 12 minuti per A.

Nel caso di SBRT polmonare, la percentuale di variazione tra A e M si attesta al 19% per tutte le sedute, come ci si aspetta, trattandosi di trattamenti con un range di 5-7 sedute.

Per gli altri trattamenti, invece, lo scarto percentuale tra i primi gruppi rispecchia la variazione del numero di CT, mentre per gli ultimi gruppi (con più di 30 sedute) si ha

uno scarto percentuale che diminuisce, perché i tempi di elaborazione del processore e la potenza di calcolo influiscono di meno sulle operazioni.

Il terzo grafico riassume invece in che percentuale i tempi di ogni operazione, automatica e manuale, contribuiscono al tempo totale per il completamento di un paziente H&N, ADK prostatico e SBRT polmonare con, rispettivamente, 29 CT in media/trattamento, 31 CT in media/trattamento, 6 CT in media/trattamento.

In tutte le patologie, il miglioramento più grande in termini di tempo è stato ottenuto con l'estrazione automatica dei dati statistici; seguono, in ordine, le operazioni di copiatura delle ROI, la deformazione delle strutture, la mappatura ed infine l'import degli esami del paziente.

Infatti, le operazioni di estrazione dei dati e copiatura, deformazione e mappatura delle ROI sono operazioni che dipendono fortemente dal numero di esami e di ROI dei pazienti.

Prendendo in esame i grafici a torta delle operazioni automatiche di scripting, è evidente per ogni patologia la prevalenza nel processo ART delle operazioni di copiatura, deformazione e mappatura delle ROI.

In futuro si potrà lavorare per ridurre i tempi di queste operazioni, tramite l'ottimizzazione degli algoritmi e dei codici Python.

Di seguito vengono riportate alcune immagini dei risultati, ottenuti mediante l'utilizzo degli script.

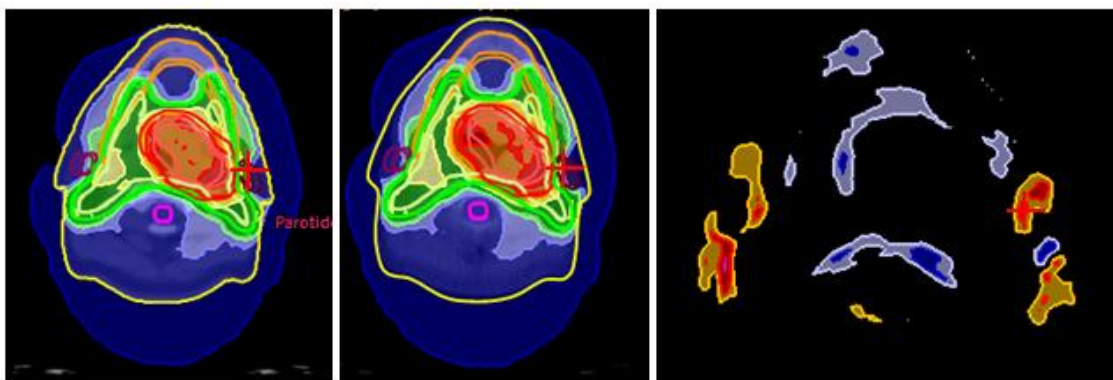


Figura 3.4 - Patologia H&N. A sinistra: dose pianificata nella CT 1. Al centro: deformazione dosi nella CT 20. A destra: dose differenza CT 1 - CT 20.

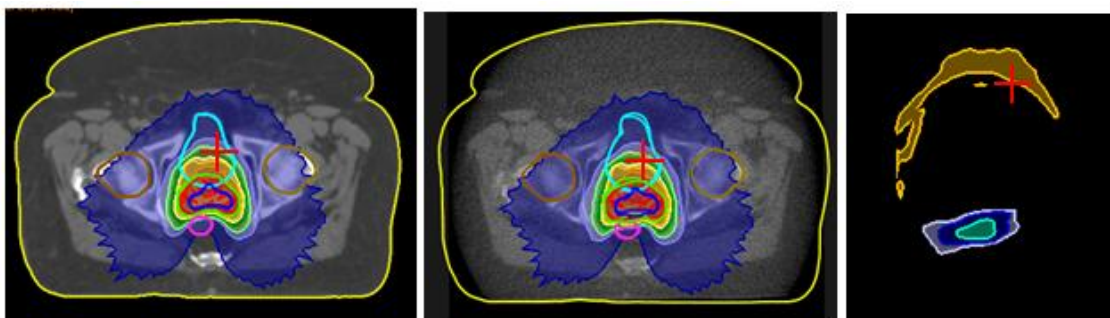


Figura 3.5 - Patologia ADK prostatico. A sinistra: dose pianificata nella CT 1. Al centro: deformazione dosi nella CT 20. A destra: dose differenza CT 1 - CT 20.

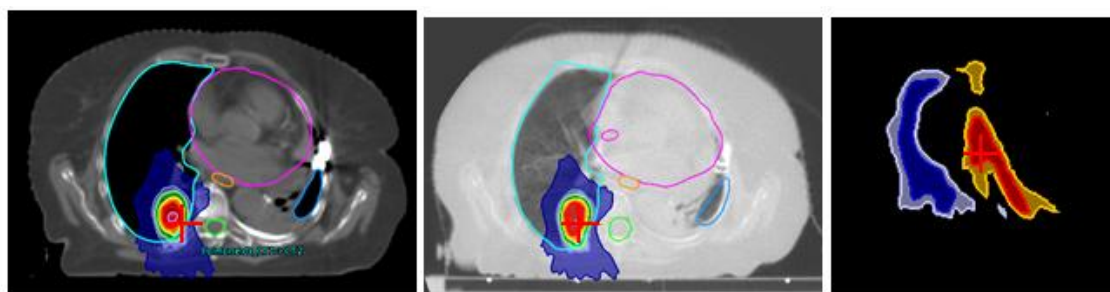


Figura 3.6 - Patologia SBRT polmonare. A sinistra: dose pianificata nella CT 1. Al centro: deformazione dosi nella CT 6. A destra: dose differenza CT 1 - CT 6.

Level [%]	Color	Opacity
25	Magenta	255
10	Red	255
5	Yellow	255
0	Black	0
-5	Light Blue	255
-10	Dark Blue	255
-25	Cyan	255

Figura 3.7 - Codice colori delle immagini con le differenze di dose

In Figura 3.4, Figura 3.5 e Figura 3.6: risultato finale delle operazioni di deformazione di dose, che consiste nella valutazione visiva e quantitativa (Vedi codici colore di Figura 3.7) della dose differenza tra la CT n-esima e la CT di pianificazione.

Questa operazione di ricalcolo della dose è fondamentale per poi valutare l'eventuale ripianificazione del paziente, nel caso in cui le differenze di dose risultino non accettabili.

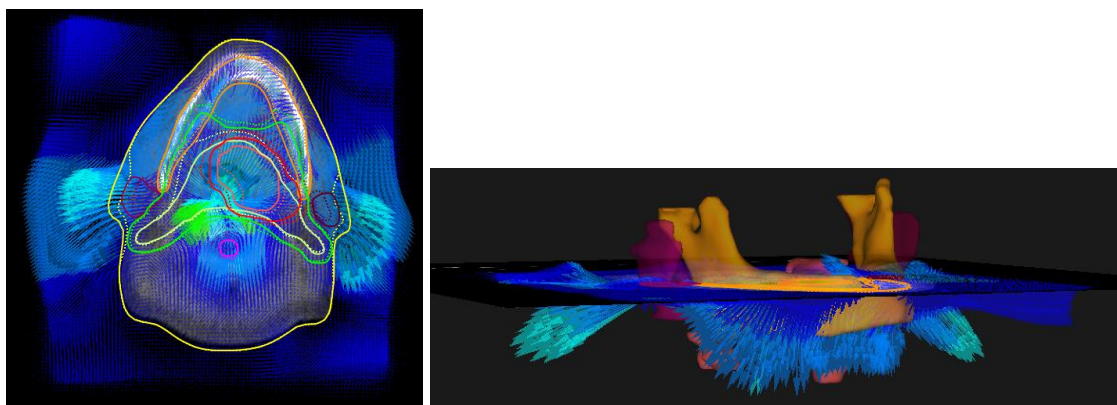


Figura 3.8 - H&N: deformazione ibrida delle ROI, griglia di deformazione e campo di deformazione.

In Figura 3.8, a sinistra l'immagine 2D mentre a destra l'immagine 3D della deformazione ibrida delle strutture per un paziente H&N: è visibile anche la griglia e il campo di deformazione.

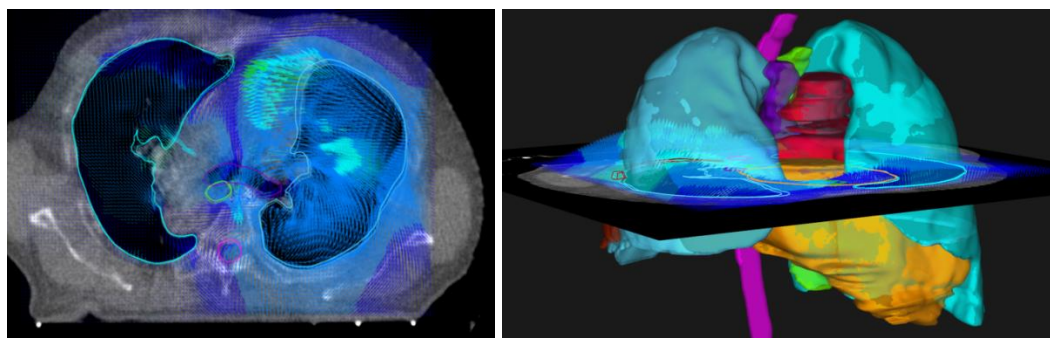


Figura 3.9 - SBRT polmonare: deformazione ibrida delle ROI, griglia di deformazione e campo di deformazione.

In Figura 3.9, il caso di un paziente con patologia SBRT polmonare: nell'immagine di sinistra si possono vedere le deformazioni ibride di alcune ROI, con la griglia e il campo di deformazione. A destra, la stessa immagine in 3D.

3.3 Limiti dell'automazione

Grazie agli script è stato possibile analizzare una grande quantità di pazienti ma, accanto ai vantaggi dell'automazione, è necessario riportare anche alcuni svantaggi o criticità, come si può vedere nelle immagini sotto.

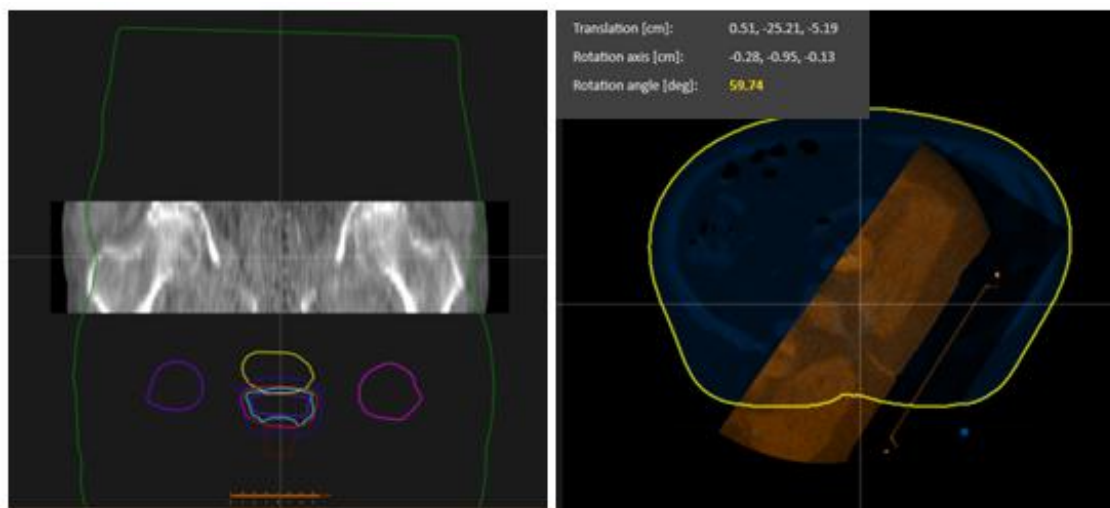


Figura 3.10 - A sinistra: esempio di copiatura delle ROI errata. A destra: co-registrazione rigida tra CT 1 e CT 12 con angolo di rotazione di 59,74°

In Figura 3.10 a sinistra, è riportato un esempio di copiatura delle ROI errata dopo le co-registrazioni rigide. Le ROI in questo caso sono state copiate all'esterno della CT: se si prosegue con le altre operazioni di ART senza nessun controllo, si raggiungono risultati completamente sbagliati.

A destra, invece, un esempio di co-registrazione rigida tra due immagini, in cui erroneamente il software ruota le due immagini di un angolo pari a 59,74°, che risulta praticamente impossibile.

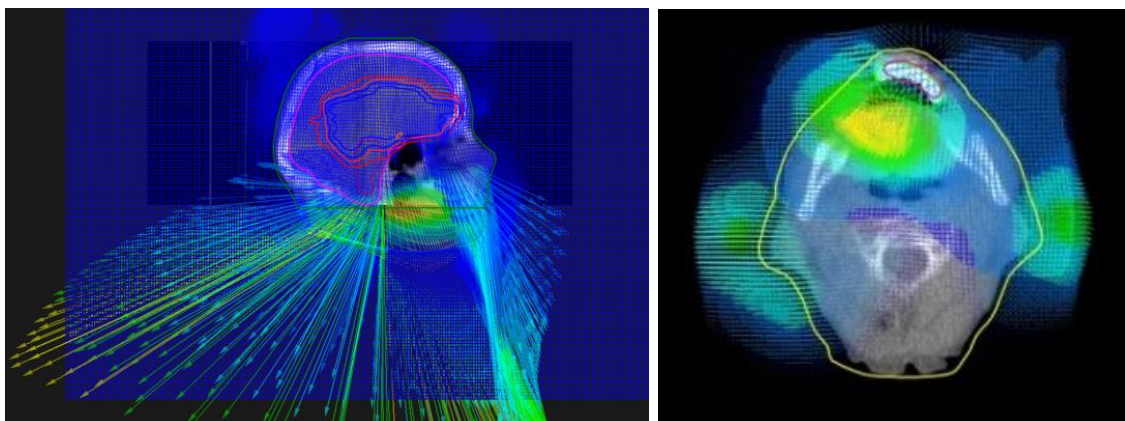


Figura 3.11 - Patologia H&N. A sinistra: campo di deformazione relativo alla mandibola. A destra: deformazione di una protesi dentale

In Figura 3.11 a sinistra, risulta deformata la mandibola, che teoricamente dovrebbe mantenere la propria anatomia durante il corso delle sedute del trattamento essendo un osso. In realtà si potrebbero verificare alcuni spostamenti, ma non così elevati da giustificare un campo di deformazione così grande (nella figura, i colori più tenui delle frecce indicano uno spostamento più lieve, che si intensifica con i colori più caldi).

In Figura 3.11 a destra, è visibile lo stesso tipo di artefatto della mandibola per la deformazione di una protesi dentaria, che teoricamente e praticamente dovrebbe rimanere identica nel corso del trattamento, a meno di alcuni piccoli spostamenti possibili.

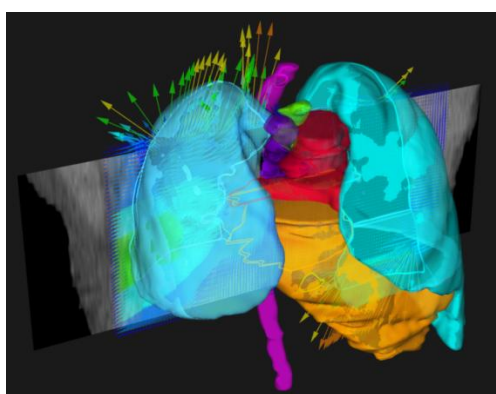


Figura 3.12 - Deformazione polmoni fuori dall'immagine

Nell'immagine di Figura 3.12, viene riportato un esempio di deformazione ibrida dei polmoni, in cui il software rileva un gradiente nei livelli di grigio ai bordi dell'immagine e segnala la deformazione delle ROI al di fuori della CT di appartenenza.

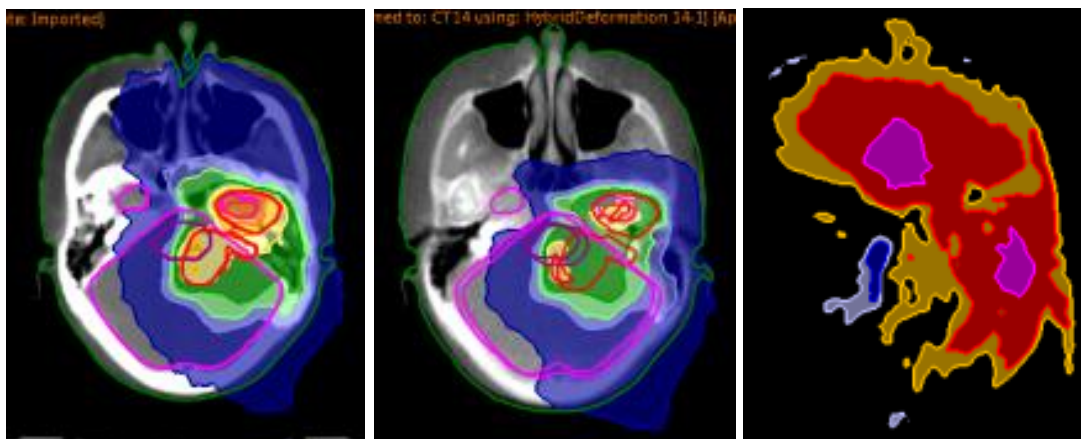


Figura 3.13 - A sinistra: dosi relative alla CT 1. Al centro: dosi relative alla CT 14. A destra: dose differenza tra le CT.

In Figura 3.13, si possono notare variazioni di dose troppo elevate nell'immagine differenza tra la CT 1 e la CT 14. In particolare, si hanno variazioni di dose comprese tra -25% e +25%, a causa di una non corretta co-registrazione iniziale tra le due immagini, che si è ripercossa in tutto il processo ART fino al ricalcolo delle dosi.

Nella Figura 3.14, invece, è indicata una variazione di densità di -847 HU della CT 25 rispetto alla CT 1. Il software, dopo aver eseguito correttamente le operazioni di ART, ricalcola correttamente la dose deformata in base alle variazioni strutturali e anatomiche avvenute nel paziente, ma non prende in considerazione eventuali variazioni densitometriche, che sono altrettanto importanti clinicamente.

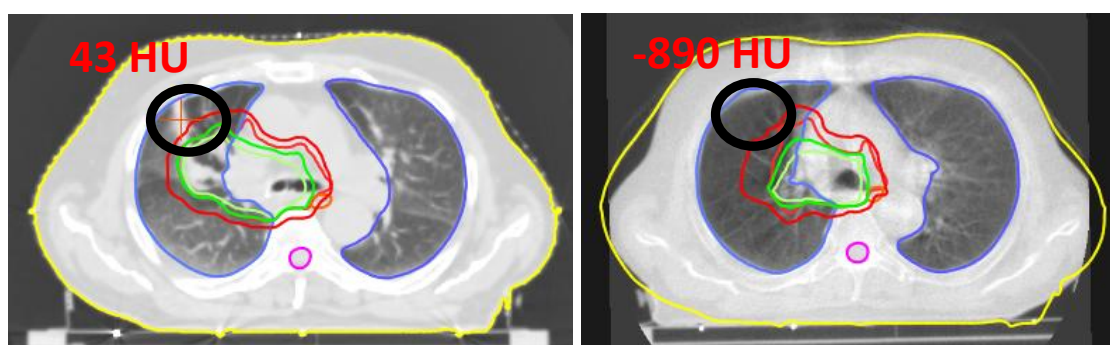


Figura 3.14 - Variazione densitometrica in patologia SBRT polmonare non rilevata.

Dalle immagini di esempio, si evince che occorre un'attenta valutazione dei risultati, per non incorrere in errori anche abbastanza gravi, dovuti a "difetti" del software.

L'automazione, quindi, non esime il personale addetto alla radioterapia dal controllo dei risultati, anche se sono stati messi a punto alcuni script per il controllo degli errori del TPS nel processo di ART, come descritto nel Capitolo 2 precedente.

Data la non completa affidabilità del software, si è pensato di far valutare ad ogni medico i risultati delle deformazioni di ogni ROI, di ogni paziente analizzato in ogni seduta.

Si sono ottenute quindi diverse tabelle, contententi i voti dei medici radioterapisti da 1 a 10. Lo scopo del lavoro è riuscire a creare una rete Bayesiana predittiva, che possa fornire al fisico medico o al medico radioterapista un parametro quantitativo della qualità delle operazioni di deformazione sul paziente (per i dettagli si rimanda al capitolo 5).

4 Prime valutazioni biomeccaniche basate su deformazioni d'organo

L'analisi biomeccanica si affianca ai processi ART, permettendo di simulare e predire tramite software FEM, il comportamento/movimento degli organi che si vogliono analizzare.

I vantaggi delle simulazioni biomeccaniche sono:

- Impatto visivo immediato e quasi-realistico.
- Modello unico e predittivo dell'organo analizzato.

Nel nostro caso, si vogliono valutare le deformazioni di alcuni organi tipici seduta per seduta.

Per far questo è necessario cercare un metodo che estrapola i contorni delle ROI da RayStation, per poi poterli elaborare quantificando la superficie che hanno in comune.

I passaggi che portano dal TPS al software FEM sono:

1. Estrapolazione dei volumi da Raystation come DICOM.
2. Import in 3D-Slicer (software in grado di aprire e manipolare i file DICOM-RT; offre anche uno strumento di sviluppo programmi in linguaggio Python, per automatizzare le operazioni).
3. Conversione delle ROI in una superficie.
4. Export delle superfici in .stl o .vtk.
5. Import e analisi in software FEM (Abaqus, Comsol Multiphysics):
 - Identificare le aree principali di deformazione (tramite somme e differenze delle superfici), cioè le aree di intersezione e di non intersezione ed estrapolarle.
 - Sviluppo di un modello unico di deformazione mediato sui pazienti, che possa simulare come varia l'area di non intersezione nelle sedute e ricerca della relazione matematica che regola l'espansione o la contrazione delle superfici degli organi analizzati.

Questi step, sono stati riportati per completezza e per un'eventuale intenzione futura da parte dell'Azienda Ospedaliera Universitaria del Policlinico di Modena di proseguire questa tipologia di studio.

La strada percorsa, invece, nella presente tesi consiste in una prima valutazione dal punto di vista biomeccanico delle deformazioni delle parotidi nelle patologie H&N attraverso gli script.

Questo tipo di approccio potrà, successivamente, essere esteso ad altri organi critici.

4.1 Deformazioni delle parotidi

Nel caso di pazienti H&N, le parotidi possono variare molto, sia in dimensioni che come posizione durante il corso del trattamento.

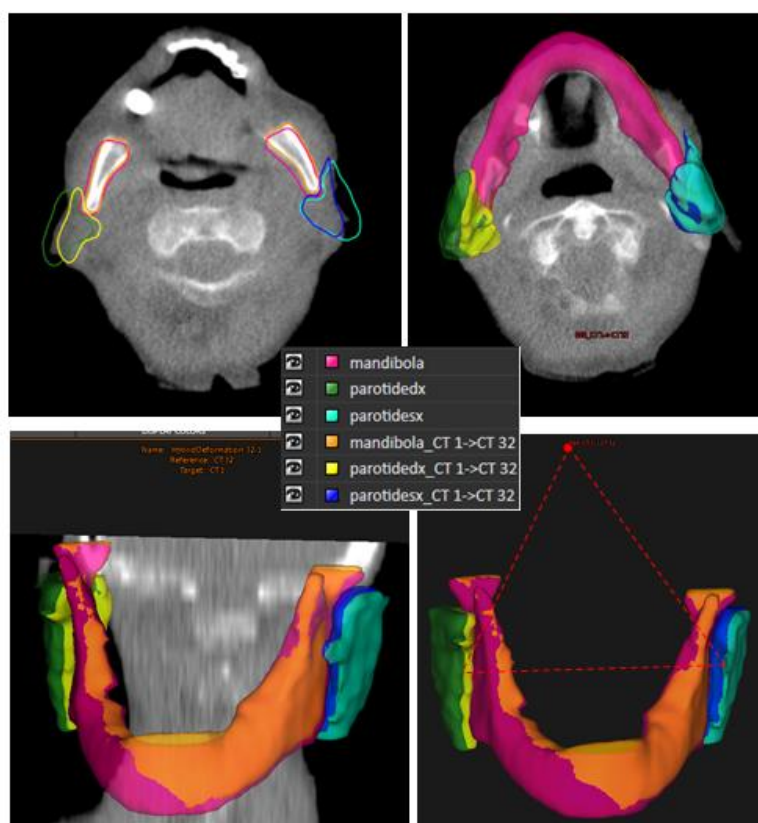


Figura 4.1 - Deformazione ibrida delle parotidi e della mandibola (Reference CT 32, Target CT 1) per un paziente H&N.

In Figura 4.1 a sinistra, l'immagine 2D delle ROI deformate e non deformate (colori rosa, verde e azzurro). A fianco, la stessa immagine ma 3D e in basso a sinistra: l'immagine 3D della deformazione da un altro punto di vista. In basso a destra: la stessa immagine dopo al creazione del POI (rosso) per il calcolo degli spostamenti delle parotidi.

Tramite lo sviluppo di uno script, si è voluto analizzare come varia la distanza tra queste da seduta a seduta. In particolare, lo script “*GeometryCheck.py*” permette di estrapolare:

- Le coordinate DICOM assolute (x, y, z) del centro di ogni parotide in cm.
- I moduli dei vettori $|\vec{d}_1|$ e $|\vec{d}_2|$ che congiungono il centro di ogni POI creato in ogni seduta con il centro di ogni parotide.
- L'angolo ϑ tra i vettori in gradi.
- La distanza $|\vec{d}_{12}|$ tra il centro di ogni parotide, in cm.
- L'area A del triangolo (POI, P_1 , P_2), in cm^2 .
- I versori \widehat{d}_1 e \widehat{d}_2 dei vettori \vec{d}_1 e \vec{d}_2 .

Questo script, avviato in ogni paziente H&N dopo il processo di deformazione e prima di estrapolare i dati elencati sopra, crea in modo automatico un POI nella kVCT nel punto (0, 0, 0) e lo mappa nelle altre CT, in modo tale che in ogni CT del paziente aperto ci sia un POI nello stesso punto della kVCT, ma avente coordinate differenti.

I POI, sono punti fissi creabili dall'utente e che servono come riferimento, nel calcolo delle grandezze utili ai fini del nostro studio.

Una modellizzazione del problema analizzato è visibile nella Figura 4.2 sottostante:

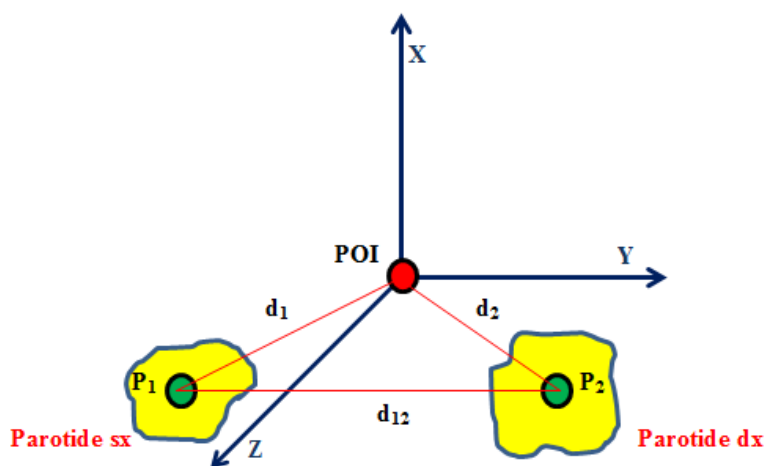


Figura 4.2 - Modello del problema delle parotidi.

È un triangolo, i cui vertici sono determinati dai punti indicati aventi le seguenti coordinate assolute:

- POI (x_0, y_0, z_0).
- P_1 (x_1, y_1, z_1).

- $P_2 (x_2, y_2, z_2)$.

La distanza tra il POI e P_1 viene calcolata come:

$$|\vec{d}_1|(POI, P_1) = \sqrt{(x_1-x_0)^2 + (y_1-y_0)^2 + (z_1-z_0)^2} \quad (26)$$

Lo stesso ragionamento vale per la distanza tra il POI e P_2 :

$$|\vec{d}_2|(POI, P_2) = \sqrt{(x_2-x_0)^2 + (y_2-y_0)^2 + (z_2-z_0)^2} \quad (27)$$

e per quella tra P_1 e P_2 :

$$|\vec{d}_{12}|(P_1, P_2) = \sqrt{(x_2-x_1)^2 + (y_2-y_1)^2 + (z_2-z_1)^2} \quad (28)$$

A questo punto si calcola l'angolo tra i vettori \vec{d}_1 e \vec{d}_2 , dalla definizione di prodotto scalare:

$$\cos(\widehat{d_1 d_2}) = \frac{\vec{d}_1 \cdot \vec{d}_2}{|\vec{d}_1| \cdot |\vec{d}_2|} = \frac{(x_{d1} \cdot x_{d2} + y_{d1} \cdot y_{d2} + z_{d1} \cdot z_{d2})}{|\vec{d}_1| \cdot |\vec{d}_2|} \quad (29)$$

$$\vartheta = \cos^{-1} \left[\frac{(x_{d1} \cdot x_{d2} + y_{d1} \cdot y_{d2} + z_{d1} \cdot z_{d2})}{|\vec{d}_1| \cdot |\vec{d}_2|} \right] \quad (30)$$

I versori dei vettori \vec{d}_1 e \vec{d}_2 saranno:

$$\widehat{d}_1 = \frac{(x_{d1}, y_{d1}, z_{d1})}{|\vec{d}_1|} \quad (31)$$

$$\widehat{d}_2 = \frac{(x_{d2}, y_{d2}, z_{d2})}{|\vec{d}_2|} \quad (32)$$

L'area del triangolo compreso tra il POI e i punti P_1 e P_2 si calcola con la seguente formula:

$$A = \frac{1}{2} |\vec{d}_1| \cdot |\vec{d}_2| \cdot \sin \vartheta \quad (33)$$

Lo script calcola e salva queste quantità (separate da caratteri di tabulazione) in un file .txt che può essere successivamente importato in Excel per l'elaborazione.

All'interno del foglio di lavoro di Excel vengono calcolate altre quantità elencate di seguito:

- $\Delta\vartheta = \vartheta_n - \vartheta_1$: differenza in gradi tra l'angolo fra i vettori \vec{d}_1 e \vec{d}_2 nella seduta n-esima del paziente e l'angolo della seduta 1.
- $\Delta A = A_n - A_1$: differenza in cm^2 tra l'area del triangolo con vertici P_1, P_1, POI della seduta n-esima e l'area del triangolo della seduta 1.
- $\Delta d_{12} = d_{12_n} - d_{12_1}$: differenza in cm tra la distanza fra le parotidi $|\vec{d}_{12}|$ della seduta n-esima e la distanza fra le parotidi della seduta 1.
- $\Delta \text{semilato}_1 = \text{semilato}_{1_n} - \text{semilato}_1$: differenza in cm tra il semilato 1 del triangolo nella seduta n-esima e il semilato 1 del triangolo nella prima seduta. Lo stesso calcolo è stato eseguito anche per il semilato 2 ($\Delta \text{semilato}_2$). I semilati del triangolo (Vedi Figura 4.3), sono calcolati con le formule seguenti, ricorrendo al teorema del coseno o di Carnot:

$$\text{Semilato}_1 = d_1 \cdot \cos \alpha = d_1 \cdot \frac{d_1^2 + d_{12}^2 - d_2^2}{2d_1 d_{12}} = \frac{d_1^2 + d_{12}^2 - d_2^2}{2d_{12}} \quad (34)$$

$$\text{Semilato}_2 = d_2 \cdot \cos \beta = d_2 \cdot \frac{d_2^2 + d_{12}^2 - d_1^2}{2d_2 d_{12}} = \frac{d_2^2 + d_{12}^2 - d_1^2}{2d_{12}} \quad (35)$$

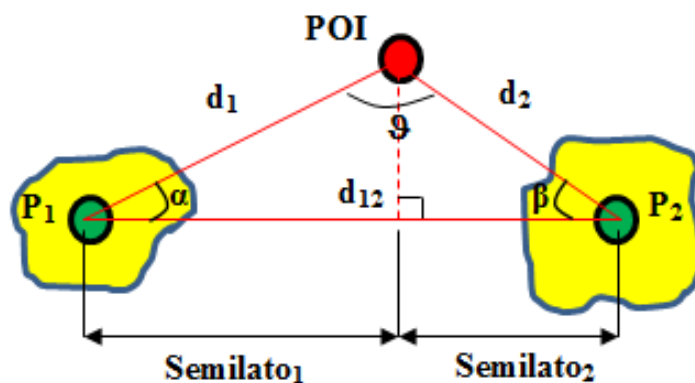


Figura 4.3 - Schema geometrico delle parotidi.

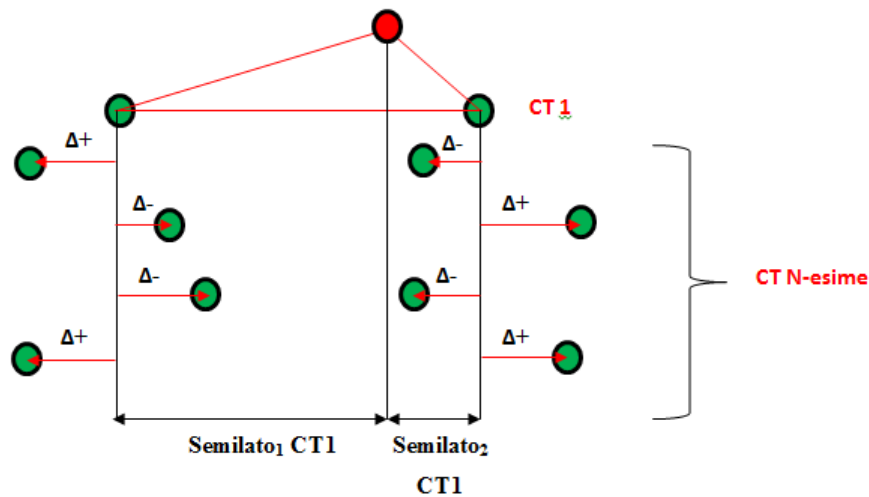


Figura 4.4 - Casi possibili di spostamento delle parotidi

Prima di proseguire con i risultati, sono fondamentali alcune considerazioni:

Δd_{12} ci può fornire indicazioni solo sullo spostamento del sistema “Parotide dx” e “Parotide sx”; in particolare, un valore negativo significa che le parotidi nella seduta n-esima hanno diminuito la loro distanza rispetto alla seduta 1 e viceversa.

Per questo motivo, si sono calcolati i semilati e, in particolare, la loro differenza da seduta a seduta rispetto alla prima. Se il $\Delta_{semilati}$ è positivo o negativo, si può valutare qual è stato lo spostamento effettivo di ogni parotide, come si può vedere nella Figura 4.4, dove sono riassunte tutte le possibili casistiche.

4.2 Analisi dei risultati

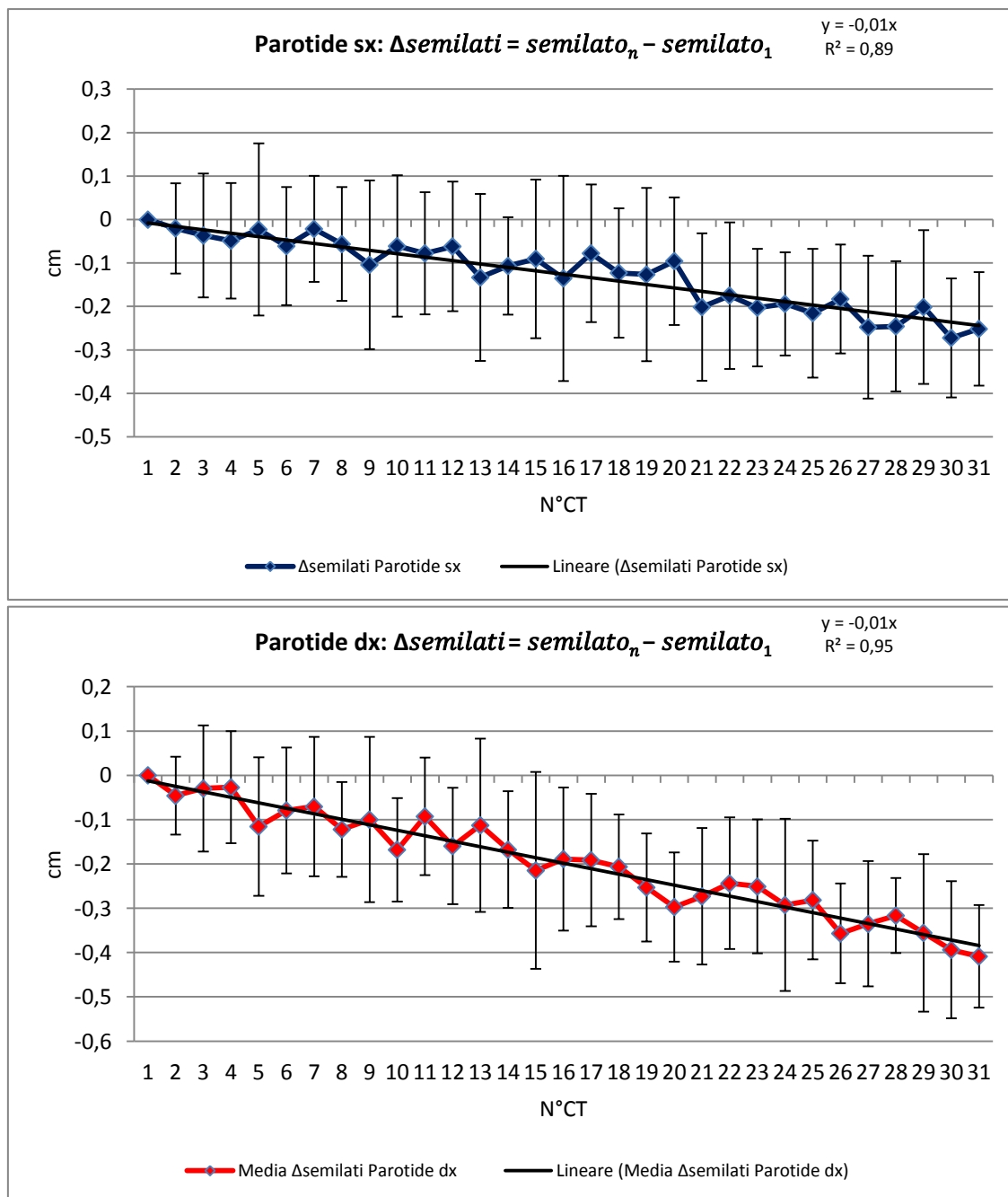


Figura 4.5 - In alto: media della differenza tra il semilato della CT N-esima e il semilato della CT 1 per la parotide sx. In basso: media della differenza tra il semilato della CT N-esima e il semilato della CT 1 per la parotide dx

Come accennato, dall'andamento medio della differenza tra i semilati di ogni parotide è possibile quantificare nei pazienti H&N qual è lo spostamento effettivo medio di ogni parotide rispetto alla seduta iniziale, che viene presa come riferimento.

In particolare, si nota il decremento all'incirca lineare (Parotide sinistra: $R^2 = 0.89$, Parotide destra: $R^2 = 0.95$) della variabile $\Delta semilati$: la differenza tra i semilati risulta essere sempre negativa sia per la parotide sinistra che su quella destra; cioè, il semilato alla seduta n-esima è in media sempre minore del semilato della prima seduta.

Tutto ciò risulta essere in accordo con quanto realmente accade nei pazienti H&N, in cui si riscontra un decremento del volume facciale rispetto alla prima seduta e, di conseguenza, un restringimento della distanza tra le parotidi destra e sinistra.

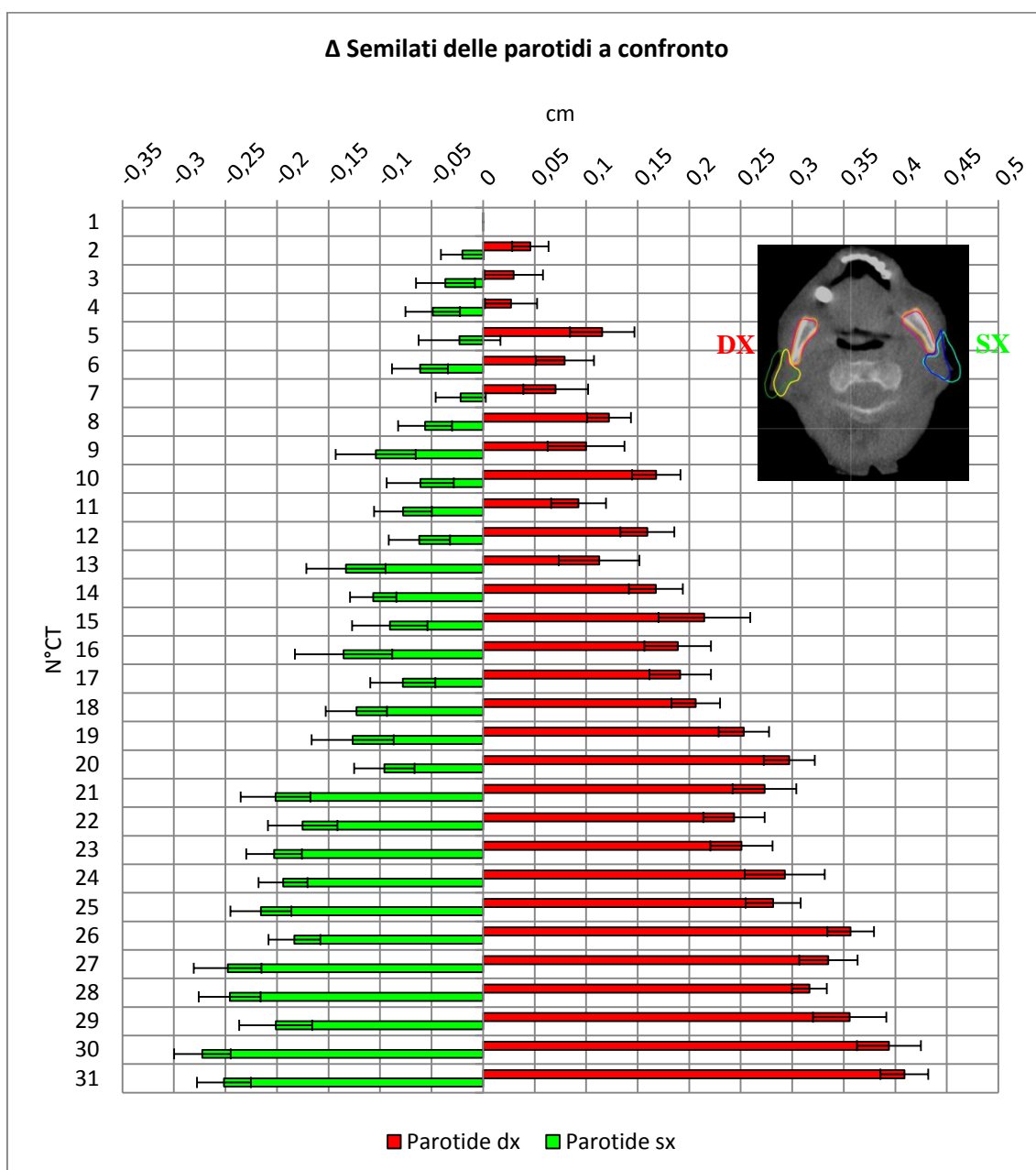


Figura 4.6 - Andamento medio della differenza tra i semilati della seduta n-esima e della seduta 1 per parotide sinistra e destra.

Nella Figura 4.6, sono stati graficati i dati relativi alla differenza tra i semilati per ogni parotide (parotide sinistra: colore verde, parotide destra: colore rosso), sotto forma di istogramma. Il fine è quello di rendere possibile un confronto più immediato ed intuitivo delle variazioni delle parotidi.

Prendendo come riferimento lo spostamento di 0 cm, relativo alla prima seduta, la parotide destra, tende a spostarsi verso destra mentre la parotide sinistra si restringe spostandosi verso sinistra della quantità indicata (es. la parotide destra, nella seconda seduta subisce un restringimento medio di circa 0,05 cm rispetto alla prima seduta).

Lo spostamento non è simmetrico, cioè la parotide destra presenta una variazione maggiore rispetto alla parotide sinistra. Le differenze, come si può notare risultano aumentare con l'aumentare del numero di sedute e l'analisi di un solo paziente può dare indicazioni al medico sulla risposta positiva o negativa di questi al trattamento eseguito, oppure sulla necessità di ripianificazione.

Un'altra indicazione, seppur meno significativa, della variazione delle parotidi è visibile in Figura 4.7. Il grafico riassume l'andamento medio della differenza tra la distanza delle parotidi in ogni seduta, rispetto alla seduta iniziale.

Anche in questo caso, la variabile Δd_{12} decresce linearmente con il numero di sedute ($R^2 = 0,97$), rispecchiando ciò che accade realmente nel paziente: cioè, un restringimento delle parotidi col proseguire del trattamento.

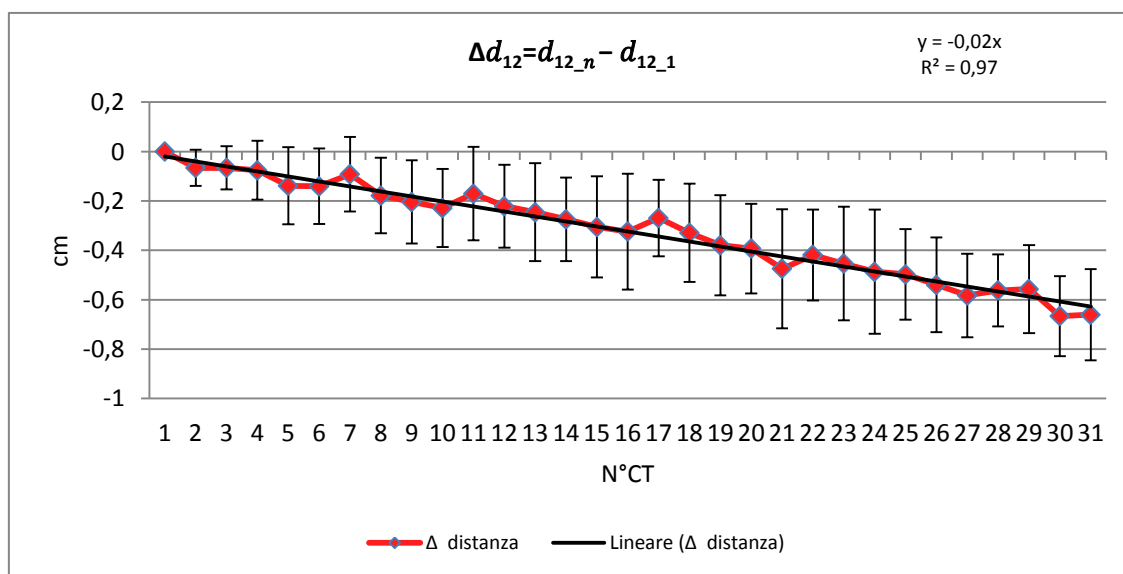


Figura 4.7 - Andamento medio nelle sedute della differenza della distanza tra le parotidi nella CT N-esima e nella CT 1.

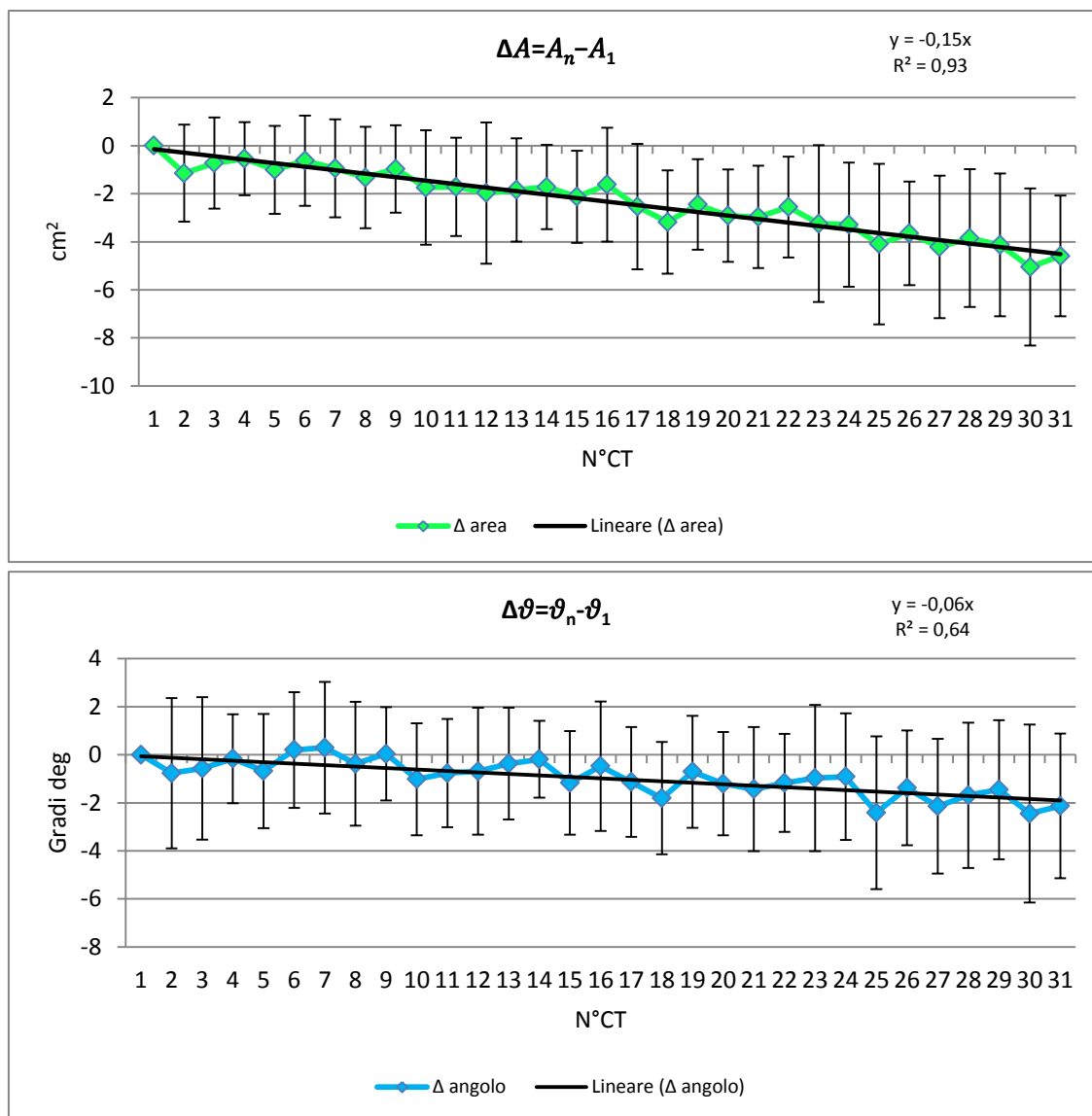


Figura 4.8 - In alto: andamento medio nelle sedute della differenza dell'area del triangolo nella CT N-esima e nella CT 1. In basso: andamento medio nelle sedute della differenza tra l'angolo dei vettori d_1 e d_2 nella CT N-esima e nella CT 1.

Anche le differenze in termini di area ΔA e angolo $\Delta \theta$, calcolate come descritto precedentemente (Vedi Figura 4.8), confermano i risultati già descritti riguardo al restringimento delle parotidi durante il corso del trattamento.

5 Valutazione Bayesiana nella pratica clinica delle deformazioni d'organo

5.1 Probabilità: concetti chiave

Teorema della probabilità totale:

Dati due eventi A e B , se questi non sono a due a due incompatibili, si deve tenere conto delle loro intersezioni. In particolare, la probabilità di unione di due eventi A e B (ovvero la probabilità che si verifichi almeno uno di essi), in generale, è pari alla somma delle singole probabilità $p(A)$ e $p(B)$ diminuita della probabilità della loro intersezione:

$$p(A \cup B) = p(A) + p(B) - p(A \cap B) \quad (36)$$

Con più di due eventi, alla somma delle probabilità di ciascuno si deve sottrarre la somma della probabilità delle loro intersezioni due a due, poi aggiungere la somma della probabilità delle loro intersezioni tre a tre e così via.

Teorema della probabilità composta:

$$p(A, B) \text{ o } p(A \cap B) = p(A|B)p(B) \quad (37)$$

$$p(A, B) = p(B|A)p(A) \quad (38)$$

$p(A, B)$ è equivalente a $p(B, A)$ e viene detta probabilità congiunta o probabilità composta: la probabilità che due eventi si verifichino contemporaneamente è pari alla probabilità di uno dei due eventi, moltiplicato con la probabilità dell'altro evento, condizionato al verificarsi del primo.

$p(B|A)$ è invece chiamata probabilità condizionale (o condizionata) ed è la probabilità che l'evento A si verifichi, sapendo che B si è già verificato. In altre parole, si tratta della probabilità che A sia condizionato da B , ovvero la probabilità di avere A , dato B .

L'analogo vale per $p(A/B)$. La probabilità congiunta di un insieme di eventi può essere espressa come catena di probabilità condizionali:

$$p(A, B, C, D) = p(A|B, C, D)p(B|C, D)p(C|D)p(D) \quad (39)$$

Dalle formule (37) e (38) deriva direttamente il teorema di Bayes:

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)} = \frac{p(A, B)p(A)}{p(B)} \quad (40)$$

Dove:

- $p(A)$ è la probabilità a priori di A .
- $p(A/B)$ è la probabilità a posteriori di A , dato B .
- $p(B/A)$ è detta verosimiglianza o 'rapporto tra i dati osservabili' ed è ciò su cui si fonda l'inferenza classica, o frequentista.
- $p(B)$ è detta 'probabilità marginale' o 'fattore di normalizzazione' ed è la probabilità di osservare B senza informazione pregressa.

Le stesse considerazioni valgono anche nel caso in cui si voglia calcolare $p(B/A)$:

$$p(B|A) = \frac{p(A|B)p(B)}{p(A)} = \frac{p(A, B)p(B)}{p(A)} \quad (41)$$

Il significato del teorema di Bayes è legato all'apprendimento dall'esperienza: spesso, infatti, nei ragionamenti probabilistici, capita che si debba valutare una probabilità, avendo già delle informazioni su quanto è già accaduto in precedenza. Dati due eventi A e B , se questi sono in qualche modo correlati, è ragionevole pensare che il sapere che uno dei due è già avvenuto possa migliorare la conoscenza della probabilità dell'altro.

Indipendenza marginale:

Due variabili A e B sono marginalmente indipendenti se:

$$p(A|B) = p(A) \quad (42)$$

$$p(B|A) = p(B) \quad (43)$$

che è sintetizzabile con la formula:

$$p(A, B) = p(A)p(B) \quad (44)$$

In altre parole la conoscenza del verificarsi di uno dei due non fornisce alcuna informazione sul verificarsi dell'altro. Il concetto di indipendenza non è da confondere con quello di incompatibilità: in effetti, se due eventi sono incompatibili, il verificarsi di uno dei due esclude il verificarsi dell'altro; quindi a rigore di termini essi non sono indipendenti (es. A = esce testa, B = esce croce).

Indipendenza condizionale:

Diciamo che l'evento A è condizionalmente indipendente da un evento B , dato l'evento C ($A \perp B/C$), se:

$$p(A|B, C) = p(A|C) \quad (45)$$

$$P(B|A, C) = P(B|C) \quad (46)$$

$$P(A, B|C) = P(A|C)P(B|C) \quad (47)$$

In pratica, A è condizionalmente indipendente da B , dato C , se la conoscenza di B non porta a nessuna ulteriore variazione della probabilità di A rispetto a quella apportata dall'avverarsi di C .

Per quanto riguarda A , se conosciamo C possiamo ignorare B .

Indipendenza marginale e condizionale sono due concetti molto differenti e non sono legati tra loro.

5.2 Le reti Bayesiane

Le reti Bayesiane (BN) sono modelli grafici probabilistici per la rappresentazione e l'analisi di modelli che coinvolgono incertezza; in particolare, per rappresentare la dipendenza fra variabili aleatorie e darne una specifica concisa della distribuzione di probabilità congiunta.

Basandosi sulla regola di Bayes, le reti suddette esprimono relazioni di dipendenza condizionale (archi) tra le variabili in gioco (nodi) usando grafi aciclici orientati o "Directed Acyclic Graph" (DAG).

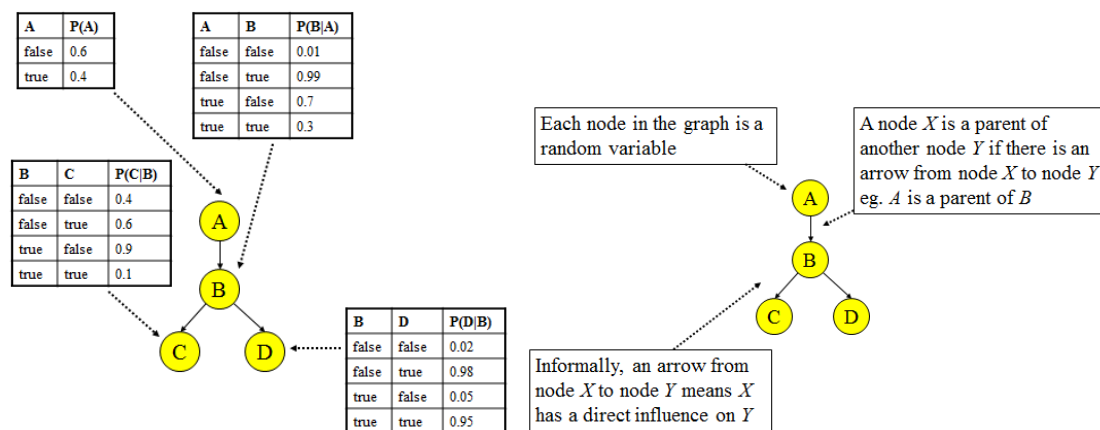


Figura 5.1 - Struttura di una rete Bayesiana con tabelle di probabilità per ogni variabile.

La rete bayesiana è costituita da un certo numero di elementi:

1. I nodi del grafo costituiscono le variabili casuali; discrete come nel nostro caso o continue.
2. Archi orientati, che collegano coppie di nodi. Se c'è un arco dal nodo X al nodo Y, X è genitore di Y (ed ha influenza diretta su Y).
3. Ogni nodo ha una tabella delle probabilità condizionate o "conditional probability table" (CPT) che quantifica gli effetti che i "genitori" hanno sul nodo. Se un nodo ha molti genitori o se i genitori di un nodo hanno numerosi stati possibili, la tabella di probabilità condizionata associata può essere molto grande. La dimensione di una CPT, infatti, varia con relazione 2^k , dove k è il numero dei genitori.
4. Il grafo non ha cicli diretti: non si può andare da un nodo a se stesso seguendo gli archi.

Per una rete Bayesiana, l'ipotesi base afferma che ogni variabile è condizionatamente indipendente dai suoi non discendenti, dove un discendente di un nodo è definito o come un figlio del nodo oppure un discendente di uno dei suoi figli, dati i suoi genitori. Questa condizione, detta proprietà locale di Markov, porta alla specificazione di un'unica distribuzione di probabilità congiunta, che risulta fattorizzabile in accordo con il grafo, permettendo una rappresentazione compatta ed efficiente.

In generale, la probabilità congiunta su tutte le variabili X_1, \dots, X_n della rete Bayesiana, la si può calcolare con la seguente formula:

$$p(X_1 = x_1, \dots, X_n = x_n) = \prod_{i=1}^n p(X_i = x_i | Parents(X_i)) \quad (48)$$

Dove $Parents(X_i)$ indica i valori dei genitori del nodo X_i rispetto al grafo considerato.

La fattorizzazione viene utilizzata quando si intende:

1. Apprendere le probabilità condizionali, una volta nota la struttura della rete.
2. Apprendere sia la struttura della rete che le probabilità condizionali.

Un nodo che non ha genitori diretti (cioè non ha frecce che puntano verso di lui) contiene una tabella di probabilità marginali tale che:

- Se il nodo è discreto, contiene una distribuzione di probabilità sugli stati della variabile che rappresenta;
- Se il nodo è continuo contiene una funzione gaussiana di densità (definita da media e varianza) della variabile casuale che rappresenta.

Se un nodo ha genitori (cioè una o più frecce che puntano verso di esso), allora il nodo contiene una tabella di probabilità condizionate:

- Se il nodo è discreto, la funzione di probabilità condizionata contiene la probabilità condizionata del nodo data una configurazione dei suoi nodi genitori.
- Se il nodo è continuo, la funzione di probabilità condizionata contiene media e varianza di ogni configurazione degli stati dei suoi nodi genitori.

Le reti Bayesiane possono essere utilizzate in ogni settore in cui sia necessario modellare la realtà in situazioni di incertezza, cioè in cui siano coinvolte delle probabilità (es. supporto alle decisioni, sistemi diagnostici, data mining, bioinformatica, ricerche intelligenti (Google), prevenzione del rischio, Computer Vision ecc.).

Uno degli obiettivi della costruzione delle reti Bayesiane è, data l'osservazione corrente sullo stato di alcune tra le variabili del dominio, rispondere a quesiti sulla distribuzione di probabilità di alcuni valori di variabili di interesse: questo processo è detto "inferenza probabilistica" o "*Belief Updating*".

Una rete Bayesiana, completamente specificata, contiene quindi l'informazione necessaria per rispondere a tutti i quesiti probabilistici circa le variabili d'interesse.

Il meccanismo che permette di trarre conclusioni in una rete Bayesiana è detto propagazione dell'evidenza, dove con evidenza si intende l'informazione corrente di cui si dispone.

La propagazione consiste nell'aggiornare le distribuzioni di probabilità delle variabili, in accordo con la nuova evidenza disponibile [Castillo et al., 1997, Cowell et al., 1999].

Le reti Bayesiane semplificano il processo d'inferenza probabilistica, aggiornando le probabilità del modello attraverso la regola di Bayes e il calcolo della probabilità a posteriori.

L'aggiornamento delle probabilità, basato semplicemente sulla regola di Bayes, è però trattabile solo a patto che la rete sia piccola e ogni nodo rappresenti variabili che possono assumere solo pochi valori.

Per risolvere questo problema, a seconda del tipo di struttura assunta dalla rete, sono stati sviluppati in letteratura algoritmi che permettono di eseguire inferenza esatta o approssimata. Nel nostro caso, data la semplicità della rete, l'inferenza è stata calcolata manualmente attraverso la regola di Bayes, per ogni patologia interessata dal nostro studio.

5.3 Applicazione pratica

I concetti delle reti Bayesiane, sono stati applicati in questa tesi alle co-registrazioni deformabili dei pazienti analizzati.

La rete Bayesiana è costituita semplicemente da due nodi, che costituiscono le variabili del nostro modello. Queste variabili sono:

1. Le valutazioni dei medici alle deformazioni: dopo la raccolta dei dati relativi ai pazienti analizzati, si sono coinvolti i medici i quali, sulla base della loro esperienza, hanno votato la qualità delle deformazioni eseguite dal software per ogni seduta, per ogni ROI di ogni paziente, limitatamente al caso di SBRT polmonare. Per le altre patologie, si è invece deciso, per necessità di tempo, di far votare al medico ogni ROI ogni 5 sedute di ogni paziente. I voti, espressi da 1 a 10, sono stati conferiti indipendentemente dalla qualità di ciascun contornamento iniziale in modo da svincolarsi da questo parametro e indipendentemente da seduta a seduta.
2. I valori dei volumi di ogni ROI votata, sono stati normalizzati sulla prima seduta del paziente.

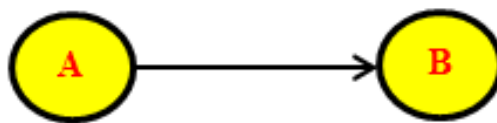


Figura 5.2 - Struttura rete Bayesiana.

In Figura 5.2, è visibile la struttura semplice della rete Bayesiana creata: dove A è la variabile corrispondente ai voti espressi dai medici, mentre B corrisponde alla percentuale di variazione dei volumi delle ROI considerate, rispetto alla seconda seduta. Mentre le percentuali relative ai volumi sono state estrapolate automaticamente, i voti dei medici hanno richiesto un lungo lavoro di raccolta, dipendente sia dalla disponibilità di tempo dei medici, sia dal tempo necessario per visualizzare i pazienti già elaborati in RayStation.

La Tabella 11, riassume i tempi necessari per il completamento di un solo paziente in base alla patologia esaminata, considerando in media 6 CT da votare per 7 ROI cioè 42 voti totali per paziente.

	H&N	ADK prostatico	SBRT polmonare
Tempo (min)	60	60	17

Tabella 11 - Tempi di votazione per un paziente di ogni patologia.

Si può notare come i tempi varino molto e diventino elevati per H&N e ADK prostatico. Questo ha permesso una raccolta di voti completa di tutti i pazienti analizzati per SBRT polmonare, ma una raccolta solo parziale nel caso di H&N e ADK prostatico.

Data la grande variabilità dei voti, conseguenza della soggettività e dell'esperienza di ogni medico votante, si renderà utile in un futuro la raccolta di una maggiore quantità di voti, ai fini di aumentare la statistica.

Nella rete Bayesiana sviluppata, la stima delle percentuali per ogni stato della variabile A (voti) è stata calcolata mediante conteggi. Dai conteggi, stabilendo una certa soglia, si possono ricavare le percentuali di voti per ogni stato della variabile A per la patologia d'interesse.

Nella Tabella 12, sono riassunti i conteggi e le relative percentuali per SBRT polmonare.

P(A)	Polmone dx		Polmone sx		Cuore		Midollo		Esofago		Trachea	
	Cont.	%	Cont.	%	Cont.	%	Cont.	%	Cont.	%	Cont.	%
A ≥ 6	153	96	132	94	79	82	146	91	124	86	55	98
A < 6	7	4	8	6	17	18	14	9	20	14	1	2
TOT.	160	100	140	100	96	100	160	100	144	100	56	100

Tabella 12 - Calcolo delle percentuali per i voti di alcune ROI.

Si sono quindi calcolati $P(A,B)$, $P(B)$, $P(A/B)$, $P(B/A)$ per ogni ROI: un esempio per la ROI "Polmone dx" è riportata di seguito.

$P(A,B)$ rappresenta la probabilità congiunta che A e B si verifichino contemporaneamente: per esempio, la probabilità che il medico abbia dato un voto maggiore o uguale a 6 e allo stesso tempo il volume abbia una variazione compresa del $\pm 5\%$ rispetto allo stesso volume nella seconda seduta.

$P(A,B)$ è stata calcolata mediante conteggi.

P(A,B) - Polmone dx		
	A ≥ 6	A < 6
95% ≤ B ≤ 105%	77%	3%
95% > B > 105%	19%	1%

Tabella 13 - Calcolo delle percentuali per $P(A,B)$ per la ROI Polmone dx.

Dalla Tabella 13, marginalizzando, si può estrarre la probabilità $P(B)$, cioè la probabilità sia di avere un volume compreso tra il 95% e il 105%, sia di avere un volume fuori da questo range.

P(B) - Polmone dx	
95% ≤ B ≤ 105%	80%
95% > B > 105%	20%

Tabella 14 - Calcolo delle percentuali per $P(B)$ per la ROI Polmone dx.

Dove $P(95 \leq B \leq 105) = 77\% + 3\% = 80\%$ e $P(95 > B > 105) = 19\% + 1\% = 20\%$.

Avendo queste probabilità a disposizione, si può calcolare quindi $P(A/B) = P(A,B)/P(B)$, che è la probabilità condizionata dato un volume B di avere un voto A .

P(A B) - Polmone dx		
	A ≥ 6	A < 6
95% ≤ B ≤ 105%	96%	4%
95% > B > 105%	94%	6%

Tabella 15 - Calcolo della probabilità congiunta P(A/B) per ROI Polmone dx.

Infine si ha $P(B/A)$, cioè la probabilità condizionata dato un voto del medico A di avere il volume B.

$P(B/A)$ può essere calcolata in due modi:

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)} \quad (49)$$

$$P(B|A) = \frac{P(A, B)}{P(A)} \quad (50)$$

P(B A) - Polmone dx		
	A ≥ 6	A < 6
95% ≤ B ≤ 105%	80%	71%
95% > B > 105%	20%	29%

Tabella 16 - Calcolo della probabilità congiunta P(B/A) per ROI Polmone dx.

Le tabelle riportate sopra sono state calcolate per ogni patologia, per ogni organo e per ogni seduta. Nei paragrafi successivi si riportano le tabelle riassuntive delle probabilità per ogni patologia.

Ad ogni variabile della rete Bayesiana, per ogni patologia considerata, sono associati due stati determinati da valori di soglia o *cut off* differenti. Questi valori limite, dovrebbero variare in base alla patologia, alle sedute e agli organi interessati (seriali o paralleli). In particolare, prendendo come esempio la patologia SBRT polmonare, alcuni valori limite potrebbero essere bassi per alcune ROI (es. polmoni) e alti per altre (es. midollo); quindi, si dovrebbe cercare di adattarli, in modo da renderli più specifici, in base all'importanza degli organi coinvolti in ogni patologia.

Una prima indicazione sul grado d'importanza degli organi coinvolti si basa sulla distinzione in organi seriali e paralleli. I seriali risultano più importanti perché, se danneggiati, perdono tutta la funzionalità (es. midollo); quindi, si cerca evidentemente

di preservarli. Gli organi paralleli (es. polmoni), anche se danneggiati, non perdono totalmente la propria funzionalità e risultano quindi relativamente meno importanti dal punto di vista clinico.

Prendendo come esempio la patologia SBRT polmonare: essendo molto importante il midollo, si dovrebbe cercare di essere più restrittivi impostando un valore di soglia per i voti alto (es. 8), mentre per i grandi vasi o i polmoni un valore di soglia impostato a 6 potrebbe essere più che accettabile.

Per introdurre valori di *cut off* variabili, è fondamentale il giudizio del medico che in questa prima implementazione della rete, non è stato possibile avere per motivi di tempo. Si è deciso, quindi, di adottare in un primo momento gli stessi valori di soglia per ogni patologia.

Per la variabile A “voto” della BN sono stati definiti i seguenti valori:

1. Voto ≥ 6 .
2. Voto < 6 .

Per quanto riguarda, invece, la variabile B “volume” della BN, sono stati stabiliti i seguenti stati:

1. $95\% \leq \text{Volume} \leq 105\%$.
2. $95\% > \text{Volume} > 105\%$.

Il modello Bayesiano creato costituisce quindi una base da cui partire in futuro, per lo sviluppo di un modello più completo che possa comprendere altre variabili come, ad esempio, D_{99} , D_{98} , $D_{Average}$, D_{50} , D_2 , D_1 ecc. (Vedi Figura 5.3).

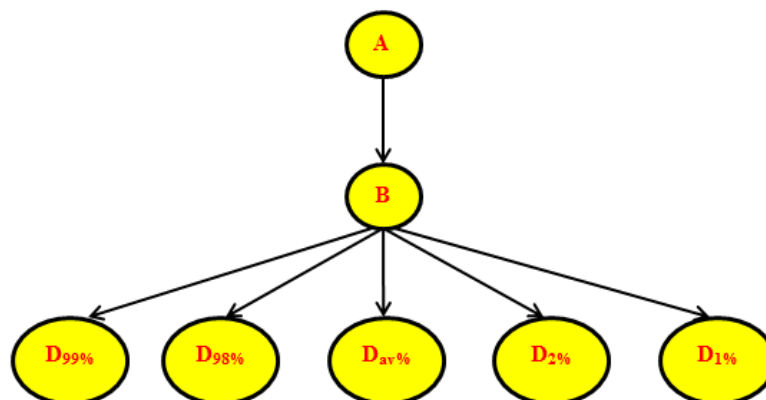


Figura 5.3 - Rete Bayesiana completa.

Il limite del modello sopracitato è il calcolo delle probabilità attraverso la regola di Bayes, che diventa sempre più oneroso se si aggiungono altre variabili al modello predittivo.

In particolare, se le variabili possiedono 2 stati, si hanno 2^k probabilità per ogni nodo, essendo k il numero di genitori per ogni nodo. In questo caso, è necessario ricorrere ad algoritmi o software più complessi, peraltro presenti in letteratura e in internet (es. Netica, Weka, MSBNx ecc.).

L'importanza della BN è data dalla predittività delle due variabili analizzate, voti e volumi, in termini di probabilità.

Anche se la probabilità nasconde un certo livello intrinseco d'incertezza, si può fornire all'utilizzatore del TPS, oltre che al medico, un'indicazione quantitativa (basata sui casi clinici analizzati) della qualità del processo di ART.

Uno dei vantaggi dell'impiego della BN è la possibilità di aggiornarla continuamente, in base a nuovi dati scaturiti da analisi su nuovi pazienti, conferendo alla rete sempre più potere predittivo aumentando la statistica, in modo da avere probabilità sempre più accurate.

L'utilità e la praticità di questo strumento, potrebbe aumentare con una sua successiva implementazione all'interno di RayStation e lo sviluppo di un processo di aggiornamento automatico della BN.

In base agli obiettivi prefissati, le probabilità più significative e utili da calcolare sono $P(B,A)$ e $P(A/B)$.

Nel caso venga avviato il processo di ART su un nuovo paziente:

- La $P(B,A)$ fornisce un'indicazione a priori, in base alla casistica già analizzata, della probabilità che il TPS esegua bene la deformazione della ROI considerata (in base ai criteri decisi nella BN) nella particolare seduta. In questo modo il MR o il fisico medico possono selezionare quali ROI o quali sedute controllare alla fine del processo di deformazione, risparmiando molto tempo.
- La $P(A/B)$, può essere utilizzata per avere indicazioni in che percentuale la deformazione della particolare ROI nella seduta considerata è giusta, leggendo dalla tabella *Dose Statistics* (tramite script o manualmente) quale è stata la variazione percentuale volumetrica della ROI normalizzata alla seconda seduta. Questo tipo di controllo, risulterebbe più veloce e potrebbe in un futuro essere sviluppato tramite script in modo automatico.

5.3.1 H&N

Nel grafico di Figura 5.4, si può notare l'andamento dei voti medi per ogni organo analizzato della patologia H&N, in gruppi di 5 sedute.

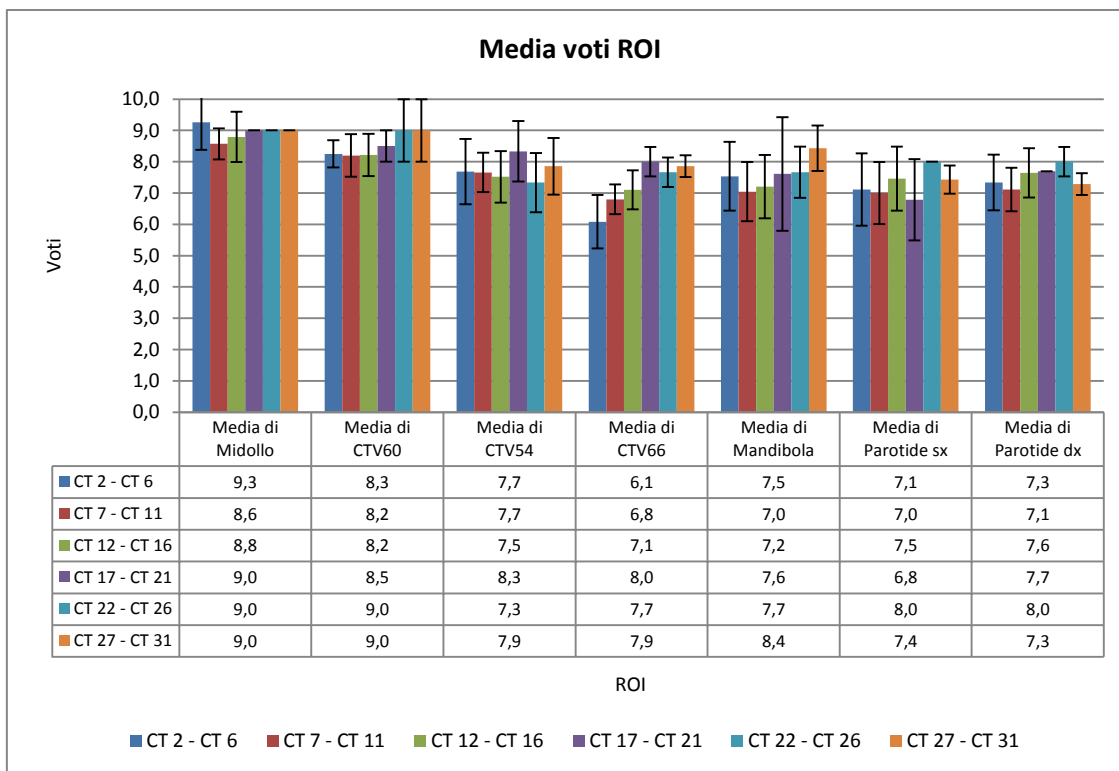


Figura 5.4 - Andamento dei voti medi per ogni ROI in base ai gruppi di sedute per H&N.

I voti risultano molto positivi per il midollo (organo seriale importante), elemento dovuto al fatto che si tratta di un organo che non ha particolari caratteristiche anatomiche che potrebbero rendere difficili una sua elaborazione.

I CTV in media ricevono sempre un voto maggiore di 7 e data la loro conformazione spesso irregolare, è possibile che il software possa commettere errori. Gli errori sono più frequenti nei CTV più piccoli (es. CTV66) rispetto ai CTV più grandi (es. CTV54), in quanto il software elabora meglio le strutture più grandi rispetto a quelle piccole.

I CTV possono variare anche molto da seduta a seduta ed è importante che il software segua queste variazioni.

Non si riscontrano nell'analisi compiuta, organi critici il cui voto in media non raggiunge la sufficienza.

Si riportano nella Tabella 17, le probabilità congiunte e condizionali per H&N per la ROI "Parotide dx" in gruppi di 5 sedute.

		A ≥ 6 95% ≤ B ≤ 105%	A < 6 95% ≤ B ≤ 105%	A ≥ 6 95% > B > 105%	A < 6 95% > B > 105%
Prob.	CT	Parotide dx	Parotide dx	Parotide dx	Parotide dx
P(A,B)	2 - 6	40%	7%	49%	4%
P(A,B)	7 - 11	27%	9%	62%	2%
P(A,B)	12 - 16	36%	0%	64%	0%
P(A,B)	17 - 21	43%	9%	39%	9%
P(A,B)	22 - 26	27%	0%	73%	0%
P(A B)	2 - 6	0%	0%	100%	0%
P(A B)	7 - 11	86%	14%	92%	8%
P(A B)	12 - 16	75%	25%	97%	3%
P(A B)	17 - 21	100%	0%	100%	0%
P(A B)	22 - 26	83%	17%	82%	18%
P(B A)	2 - 6	100%	0%	100%	0%
P(B A)	7 - 11	0%	0%	100%	0%
P(B A)	12 - 16	45%	60%	55%	40%
P(B A)	17 - 21	30%	80%	70%	20%
P(B A)	22 - 26	36%	0%	64%	0%

Tabella 17 - Riassunto delle probabilità congiunte P(A,B) e condizionali P(A|B), P(B|A) per i 4 stati possibili delle variabili A e B.

Le probabilità sono state infine graficate in Figura 5.5 e Figura 5.6 per la ROI "Parotide dx". Per le tabelle riassuntive di tutte le ROI e i grafici relativi si rimanda all'Appendice B.

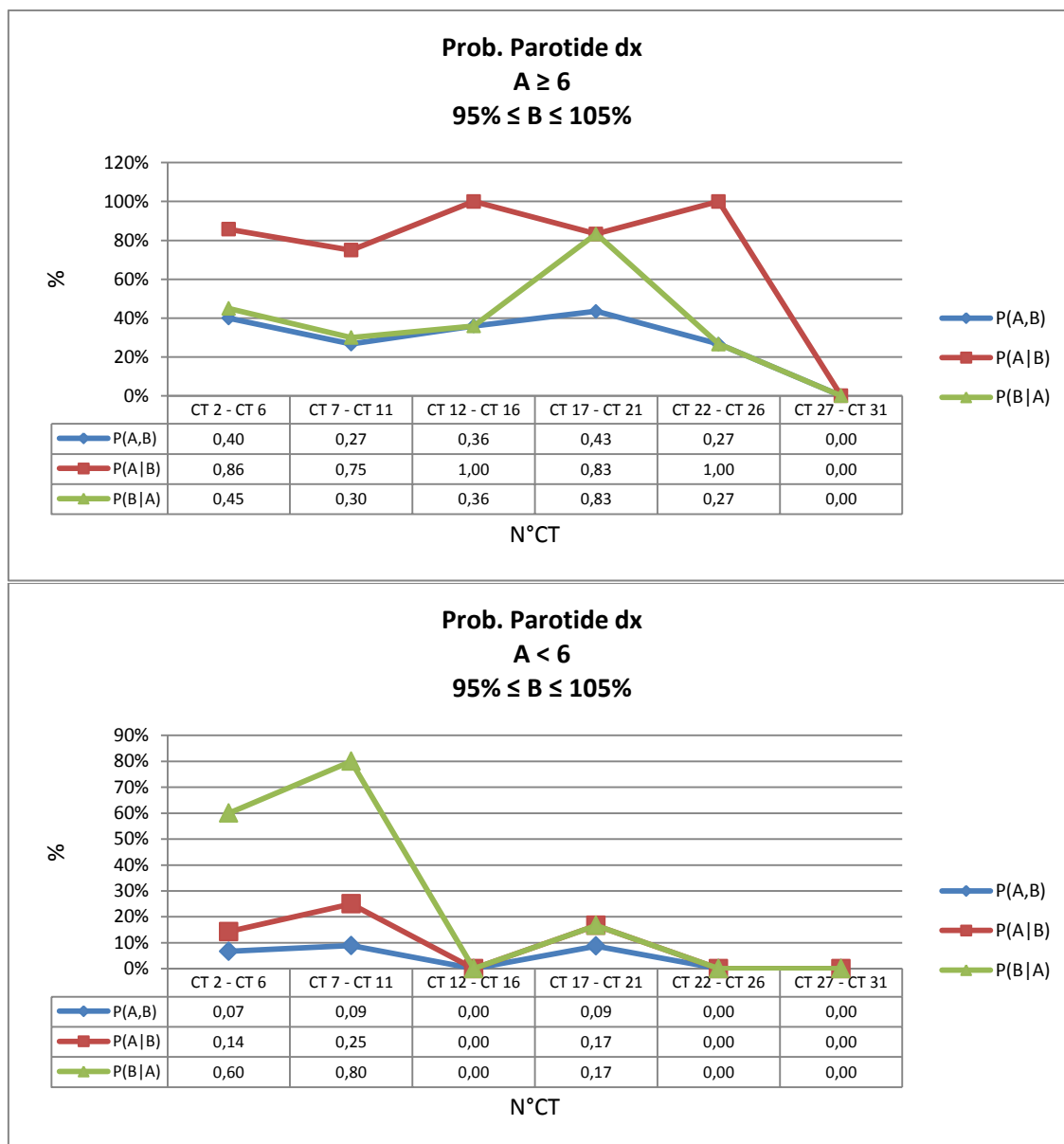


Figura 5.5 - In alto: parotide dx, andamento della probabilità per lo stato $A \geq 6$ e $95\% \leq B \leq 105\%$. In basso: parotide dx, andamento della probabilità per lo stato $A < 6$ e $95\% \leq B \leq 105\%$.

Osservando i grafici e/o le tabelle sopra si può sapere la $P(A,B)$, $P(A/B)$ e $P(B/A)$ per la parotide destra nei pazienti H&N e basandosi su queste si può decidere in quale seduta di un nuovo ipotetico paziente controllare i risultati delle deformazioni, a causa di una probabilità bassa.

Da un punto di vista clinico le probabilità più importanti sono quelle del primo grafico, in particolare $P(A,B)$ e $P(A/B)$. Dato che i valori di $P(A,B)$ sono al di sotto del 50%, ci si deve basare sulla $P(A/B)$ per avere un'indicazione utile al nostro scopo. Per questa ragione, per un paziente nuovo, il MR o il fisico analizzano i volumi della ROI "Parotide dx" nella seduta d'interesse: se la variazione in termini percentuali è compresa

tra il 95% e il 105%, si può sapere a priori, in base alla statistica, la probabilità che la deformazione sia giusta (voto maggiore o uguale a 6) o sbagliata (voto minore di 6) dal primo grafico di Figura 5.5. Se, invece, la variazione del volume è fuori da questo range, si può basare la valutazione sul primo grafico di Figura 5.6.

Paradossalmente sono più alte le probabilità nel secondo caso, rispetto al primo; questo può essere dovuto al fatto che le soglie di variazione del volume sono troppo restrittive per questo organo e necessitano di essere cambiate. Se la CT del paziente ricade nel gruppo contenente dalla CT 27 alla CT 31, sarà in ogni modo necessario un suo controllo, come si può notare dai valori di probabilità tutte nulle.



Figura 5.6 - In alto: parotide dx, andamento della probabilità per lo stato $A \geq 6$ e $95\% > B > 105\%$. In basso: parotide dx, andamento della probabilità per lo stato $A < 6$ e $95\% > B > 105\%$.

5.3.2 ADK prostatico

Nel grafico di Figura 5.7, si può notare l'andamento dei voti medi per ogni organo suddivisi in gruppi di 5 sedute.

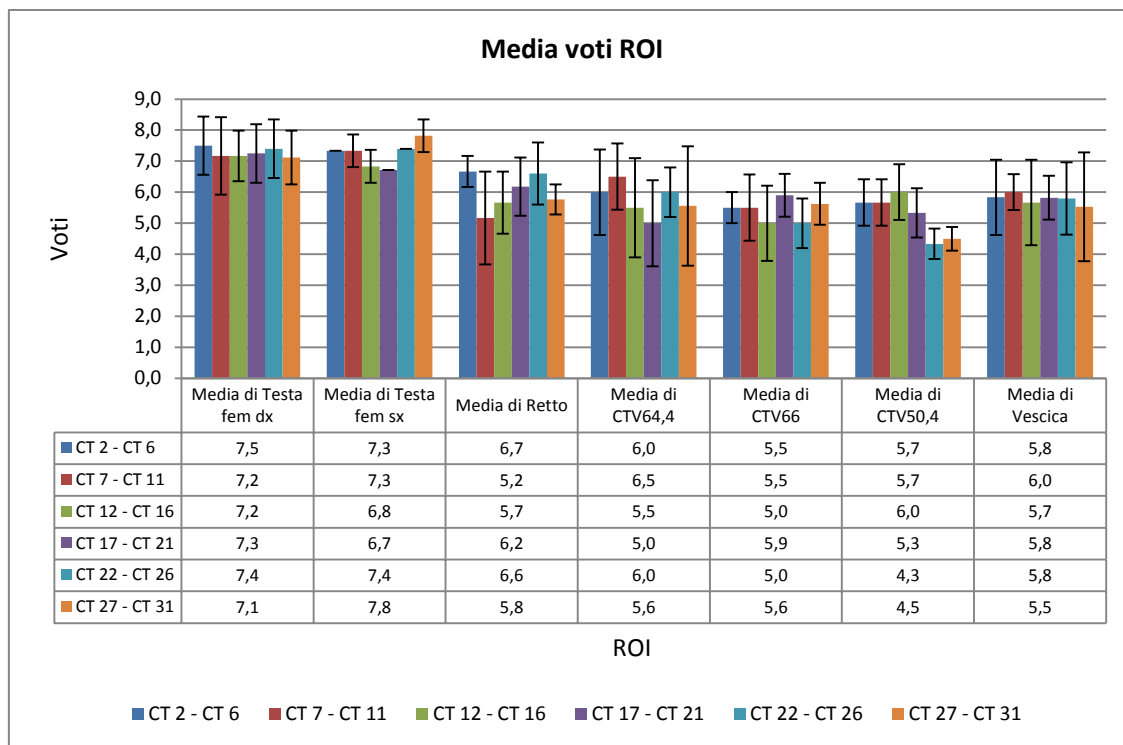


Figura 5.7 - Andamento dei voti medi per ogni ROI in base ai gruppi di sedute per ADK prostatico.

Le teste femorali mantengono in media una votazione alta in ogni gruppo di sedute, elemento dovuto al fatto che sono strutture semplici dalla forma pressochè circolare, non sovrapposte solitamente ad altre ROI per cui il software è in grado di elaborarle correttamente.

Il retto è uno degli organi critici: durante il corso del trattamento può variare in forma, sia per la vicinanza del CTV (che, riducendosi, libera spazio per il retto) sia per la presenza di feci in quantità variabili.

I CTV possono variare anch'essi molto durante il trattamento, se quest'ultimo risulta efficace. I voti sono in questo caso variabili e raggiungono in pochi casi la sufficienza.

Il CTV50.4 rispetto agli altri CTV di forma circolare, ha una struttura più complessa e, anche se maggiore come dimensioni rispetto ai CTV64,4 e CTV66, il software la deforma in modo peggiore.

Si riportano nella Tabella 18, le probabilità congiunte e condizionali per ADK prostatico per la ROI "CTV66", in gruppi di 5 sedute.

		A ≥ 6 95% ≤ B ≤ 105%	A < 6 95% ≤ B ≤ 105%	A ≥ 6 95% > B > 105%	A < 6 95% > B > 105%
Prob.	CT	CTV66	CTV66	CTV66	CTV66
P(A,B)	2 - 6	50%	50%	0%	0%
P(A,B)	7 - 11	40%	40%	10%	10%
P(A,B)	12 - 16	40%	40%	10%	10%
P(A,B)	17 - 21	50%	20%	0%	30%
P(A,B)	22 - 26	40%	50%	10%	0%
P(A B)	2 - 6	25%	25%	38%	13%
P(A B)	7 - 11	50%	50%	0%	0%
P(A B)	12 - 16	50%	50%	50%	50%
P(A B)	17 - 21	50%	50%	50%	50%
P(A B)	22 - 26	71%	29%	0%	100%
P(B A)	2 - 6	44%	56%	100%	0%
P(B A)	7 - 11	50%	50%	75%	25%
P(B A)	12 - 16	100%	100%	0%	0%
P(B A)	17 - 21	80%	80%	20%	20%
P(B A)	22 - 26	80%	80%	20%	20%

Tabella 18 - Riassunto delle probabilità congiunte $P(A,B)$ e condizionali $P(A|B)$, $P(B|A)$ per i 4 stati possibili delle variabili A e B.

Le probabilità, sono state infine graficate in Figura 5.8 e Figura 5.9 per la ROI "CTV66". Per le tabelle riassuntive di tutte le ROI e i grafici relativi si rimanda all'Appendice B.



Figura 5.8 - In alto: CTV66, andamento della probabilità per lo stato $A \geq 6$ e $95\% \leq B \leq 105\%$. In basso: CTV66, andamento della probabilità per lo stato $A < 6$ e $95\% \leq B \leq 105\%$.

Soffermando l'analisi su un particolare range di sedute, per esempio CT 22 - CT 26:

- Si supponga che, nella CT 23 di un nuovo paziente ADK prostatico, il volume del CTV66 ha una variazione compresa nel range indicato; leggendo allora il primo grafico di Figura 5.8, si ha un 40% di probabilità che la deformazione sia fatta bene e quindi, dal secondo grafico, un 60% di probabilità che la deformazione sia sbagliata. È opportuno perciò controllare il risultato.
- Se, invece, la variazione di volume è fuori dal range del $\pm 5\%$, il grafico superiore di Figura 5.9, fornisce la probabilità che il voto sia positivo. In questo caso, si ha il

100% di probabilità che il voto sia maggiore di 6, per cui non è necessario ricontrollare il risultato della deformazione eseguita dal software.

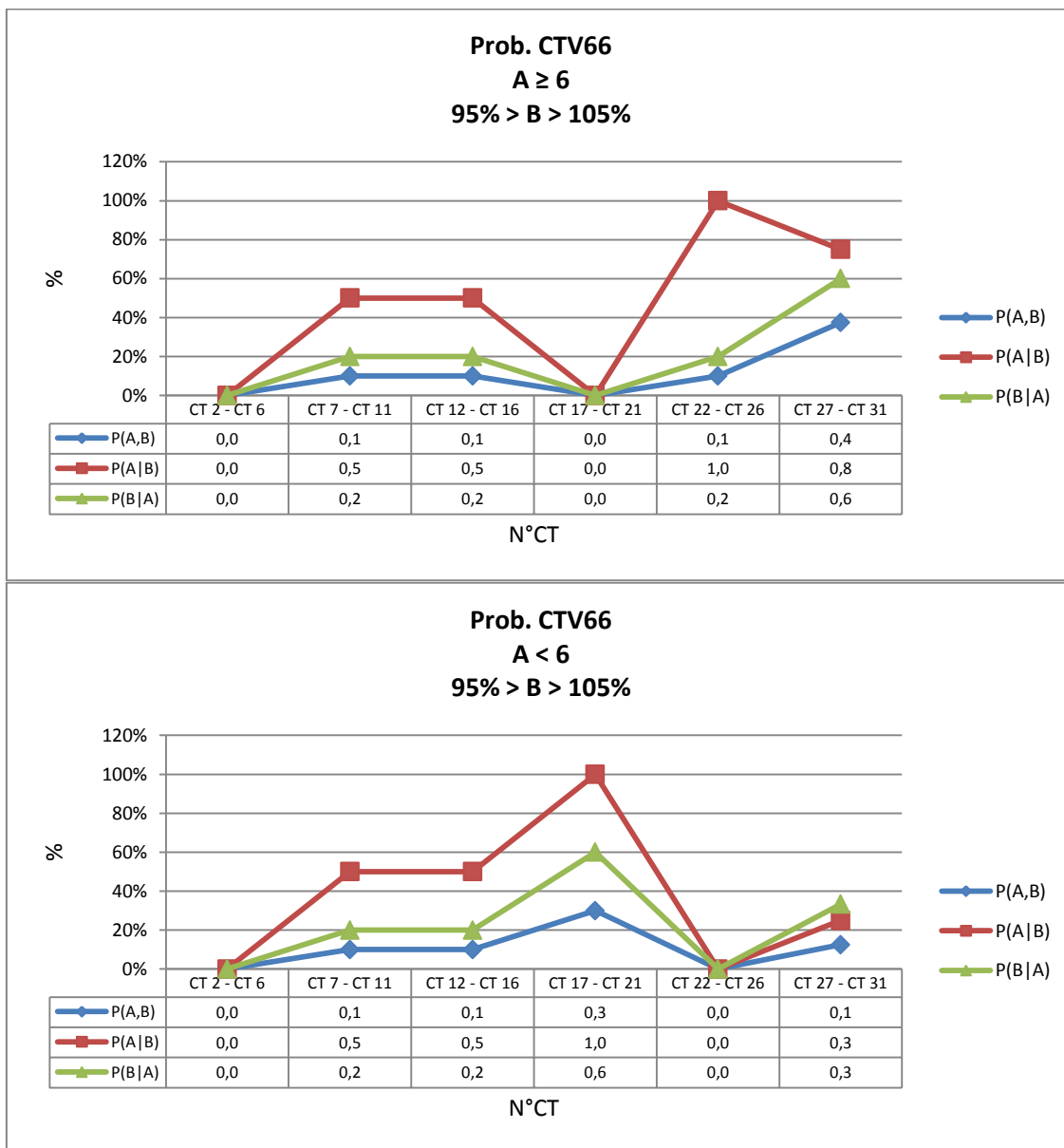


Figura 5.9 - In alto: CTV66, andamento della probabilità per lo stato $A \geq 6$ e $95\% > B > 105\%$. In basso: CTV66, andamento della probabilità per lo stato $A < 6$ e $95\% > B > 105\%$.

5.3.3 SBRT polmonare

Nel grafico di Figura 5.10, viene riportato l'andamento dei voti medi per ogni organo, suddivisi in base alle sedute.

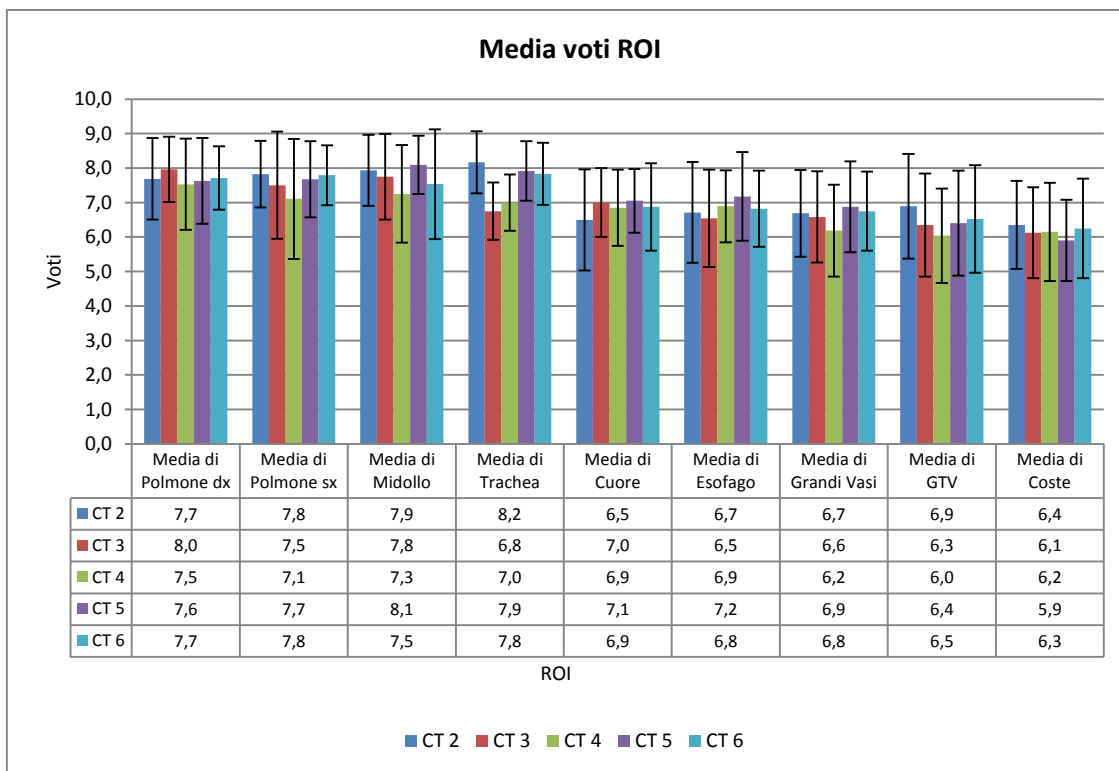


Figura 5.10 - Andamento dei voti medi per ogni ROI in base alla seduta per SBRT polmonare.

Risulta evidente dove si verificano eventuali criticità del software e carenze nella qualità delle deformazioni eseguite da RayStation.

Nel dettaglio, si riscontra che gli organi più critici sono le coste, seguite dal GTV e dai grandi vasi. Mentre gli organi che risultano essere deformati meglio sono i polmoni, il midollo e la trachea. Il software, infatti, presenta più difficoltà nel deformare le ROI aventi un volume piccolo rispetto a quelle aventi un volume grande; differenza che è stata notata durante la messa a punto del processo di deformazione. Queste criticità potrebbero dipendere anche dal contornamento iniziale non ottimale dell'organo.

Il fine di quest'analisi è comunque quello di fornire al MR un'indicazione, seppure approssimata, di quando si rende necessario il suo intervento per un eventuale ricontornamento dell'organo, oppure l'intervento del fisico medico per la modifica del metodo di deformazione (es. focus on ROI, controlling ROI, resolution ecc.).

Si riportano nella Tabella 19, le probabilità congiunte e condizionali per SBRT polmonare per la ROI “Polmone dx” di ogni seduta.

		A ≥ 6 95% ≤ B ≤ 105%	A < 6 95% ≤ B ≤ 105%	A ≥ 6 95% > B > 105%	A < 6 95% > B > 105%
Prob.	Sedute	Polmone dx	Polmone dx	Polmone dx	Polmone dx
P(A,B)	CT2	94%	6%	0%	0%
P(A,B)	CT3	86%	0%	14%	0%
P(A,B)	CT4	59%	3%	31%	6%
P(A,B)	CT5	81%	6%	13%	0%
P(A,B)	CT6	57%	0%	43%	0%
P(A B)	CT2	94%	6%	0%	0%
P(A B)	CT3	100%	0%	100%	0%
P(A B)	CT4	95%	5%	83%	17%
P(A B)	CT5	93%	7%	100%	0%
P(A B)	CT6	100%	0%	100%	0%
P(B A)	CT2	100%	100%	0%	0%
P(B A)	CT3	86%	0%	14%	0%
P(B A)	CT4	66%	33%	34%	67%
P(B A)	CT5	87%	100%	13%	0%
P(B A)	CT6	57%	0%	43%	0%

Tabella 19 - Riassunto delle probabilità congiunte $P(A,B)$ e condizionali $P(A|B)$, $P(B|A)$ per i 4 stati possibili delle variabili A e B.

Le probabilità, sono state infine graficate in Figura 5.11 e Figura 5.12 per la ROI “Polmone dx”, in modo da avere un riscontro visivo ed immediato dei dati.

Per le tabelle riassuntive di tutte le ROI e i relativi grafici si rimanda all'Appendice B.

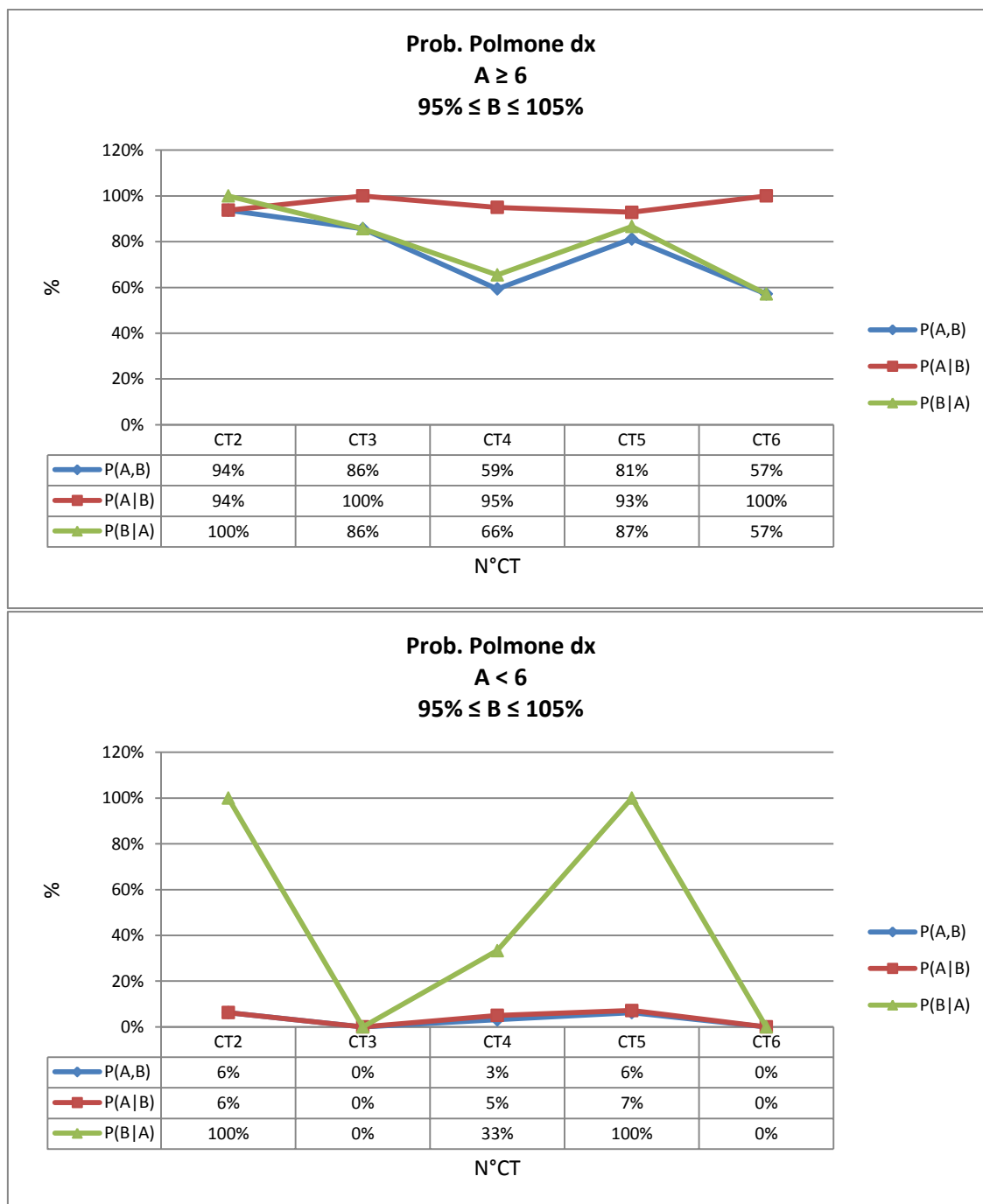


Figura 5.11 - In alto: polmone dx, andamento della probabilità per lo stato $A \geq 6$ e $95\% \leq B \leq 105\%$. In basso: polmone dx, andamento della probabilità per lo stato $A < 6$ e $95\% \leq B \leq 105\%$.

Analizzando il grafico superiore di Figura 5.11, le probabilità $P(A,B)$ e $P(A|B)$ risultano essere sempre superiori al 57%, per cui nel trattamento di un nuovo paziente SBRT polmonare si può affermare con buona probabilità che le deformazioni della ROI “polmone dx” vengano eseguite correttamente dal software in ogni seduta e che non sia necessario il controllo dell’utente nel caso che la variazione di volume della ROI, rispetto alla prima seduta, sia compresa tra 95% e 105%.

Se, invece, il volume risulta fuori dal range sopracitato si possono analizzare i grafici di Figura 5.12 e, in particolare, dal grafico superiore, esaminando la $P(A/B)$ risulta evidente che eventuali errori del software si potrebbero verificare con maggior probabilità nella seduta 2.

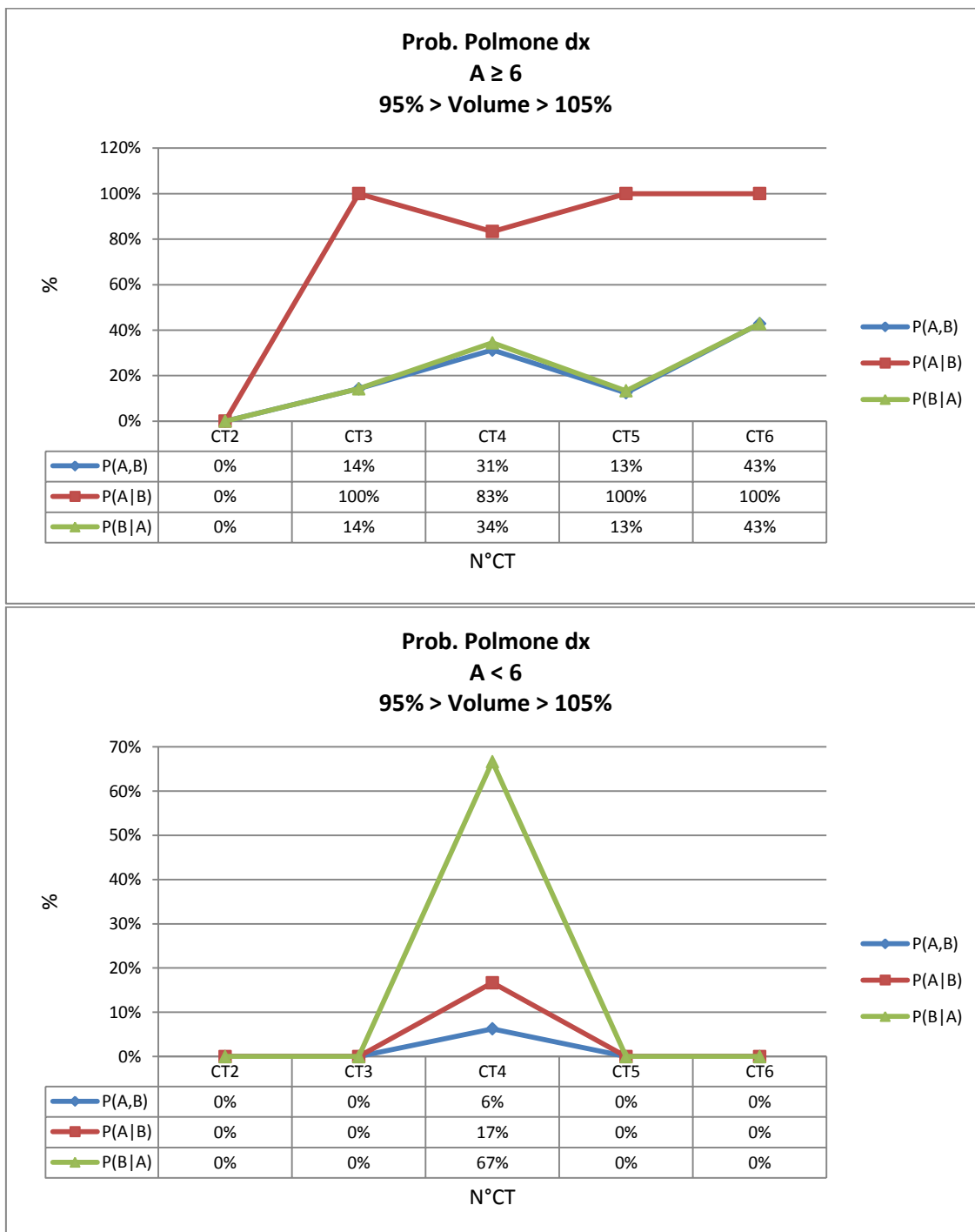


Figura 5.12 - In alto: polmone dx, andamento della probabilità per lo stato $A \geq 6$ e $95\% > B > 105\%$. In basso: polmone dx, andamento della probabilità per lo stato $A < 6$ e $95\% > B > 105\%$.

6 Conclusioni

La ricerca ha richiesto lo sviluppo di 3500 righe di codice in linguaggio IronPython, per un totale di 13 script, che consentono l'analisi automatizzata dei pazienti e delle deformazioni ad essi associate. Sono stati elaborati 51 pazienti (1340 immagini MVCT) con un risparmio di tempo medio pari al 164% per ogni patologia, rispetto allo stesso processo, svolto in manuale. Questa valutazione consente di proporre l'applicazione della tecnica di *Adaptive RT* nella pratica clinica. In particolare, si stima la possibilità di post-elaborare 30 pazienti a notte per singola macchina di calcolo.

Gli script di controllo permettono la verifica dei risultati ottenuti (es. il controllo delle rototraslazioni anomale) e forniscono informazioni su eventuali errori del software nelle co-registrazione rigide ed elastiche. La stima di andamenti anomali dei pazienti sottoposti a terapia può essere usata come indicatore predittivo per eventuali valutazioni da parte dell'utente e per pianificare attività che altrimenti risulterebbero inapplicabili nella routine clinica, a causa dell'elevato costo in termini temporali (*time consumption*).

Il medico radioterapista e/o il fisico medico, mediante lo sviluppo delle reti Bayesiane, possono calcolare le probabilità con cui alcuni organi, durante il ciclo di terapia, possono presentare anomalie o variazioni morfologiche, tali da provocare errori del software o incongruenze terapeutiche rispetto a quanto pianificato. Il fine è quello di limitare l'intervento dell'operatore, riducendo di conseguenza i tempi di calcolo e controllo.

Dalle valutazioni biomeccaniche preliminari sulle parotidi per le patologie H&N, si è potuto quantificare il movimento/contrazione durante il ciclo di terapia in un futuro estendibile anche ad altri organi (es. retto, vescica, polmoni ecc.). I risultati ottenuti sono correntemente oggetto di approfondimento, tramite tecniche di simulazione che utilizzano software FEM e fantocci antropomorfi, creati mediante LEGO *Mindstorms*, attualmente in fase di sviluppo.

Lo stesso procedimento, sviluppato durante la ricerca è oggi applicato e studiato sulle dosi erogate, quantificando la differenza di terapia tra sedute (*interplay*) o nella stessa seduta (*intraplay*). L'obiettivo finale potrebbe essere lo sviluppo e l'identificazione reale delle variazioni anatomiche del paziente e la ripianificazione delle nuove morfologie assunte dagli organi interni e dai tumori durante il ciclo terapeutico.

Legenda

ROI = Region Of Interest

POI = Point Of Interest

TPS = Treatment Planning System

IGRT = Image Guided Radiation Therapy

IMRT = Image Modulated Radiation Therapy

ART = Adaptive Radiation Therapy

TC = Computed Tomography

PET = Positron Emission Tomography

RMN = Nuclear Magnetic Resonance

MLC = Multi Leaf Collimator

CRT = Conformal Radiation Therapy

IORT = Intra-Operative Radiation Therapy

LET = Linear Energy Transfer

NDD = Normalized depth dose

TPR = Tissue phantom ratio

ID = Integral Dose

LINAC = Linear Accelerator

HU = Hounsfield Units

SPECT = Single Photon Emission Computed Tomography

fMRI = Functional Magnetic Resonance Imaging

FEM = Finite Element Method

US = Ultra Sound

SIB = Simultaneous Integrated Boost

MR = Medico Radioterapista

R&V = Record&Verified

IT = Information Technology

VMAT = Volumetric Modulated Arc Therapy

RS = Stereotactic radiosurgery

ADK = Adenocarcinoma

SBRT = Stereotactic body radiation therapy

CBCT = Cone Beam Computed Tomography
MVCT = Mega Voltage Computed Tomography
kVCT = Kilo Voltage Computed Tomography
DICOM = Digital Imaging and Communications in Medicine
NEMA = National Electrical Manufacturer Association)
ACR = American College of Radiology
TCP/IP = Transmission Control Protocol/Internet Protocol
OSI = Open Systems Interconnection
ISO = International Organization for Standardization
RT = RadioTherapy
WPF = Windows Presentation Foundation
SO = Operative System
CC = Cross Correlation
DVH = Dose Volume Histogram
MI = Mutual Information
HI = Homogeneity Index
CPU = Central Processing Unit
RAM = Random Access Memory
ICRU = International Commission on Radiations Units and Measurements
GTV = Gross Tumor Volume
CTV = Clinical Target Volume
ITV = Internal Target Volume
IM = Internal Margin
SM = Set-up Margin
OARs = Organ at Risks
PRV = Planning Organ at Risk Volume Landmarks
DAG = Directed Acyclic Graph
CPT = Conditional Probability Table
BN = Bayesian Network

Bibliografia

1. [Tehrani 2013] Joubin Nasehi Tehrani, Ricky T O'Brien, Per Rugaard Poulsen and Paul Keall, *Real-time estimation of prostate tumor rotation and translation with a kV imaging system based on an iterative closest point algorithm*, Phys. Med. Biol. 58, 8517–8533.
2. [Chao 2005] Anne Chao, Robin L. Chazdon, Robert K. Colwell and Tsung-Jen Shen, *A new statistical approach for assessing similarity of species composition with incidence and abundance data*, Ecology Letters 8, 148–159.
3. [Dalirsefat 2009] Seyed Benyamin Dalirsefat, Andréia da Silva Meyer and Seyed Ziyaeddin Mirhoseini, *Comparison of similarity coefficients used for cluster analysis with amplified fragment length polymorphism markers in the silkworm, Bombyx mori*, Journal of Insect Science 9(71).
4. [Turan 2005] Mustafa Deha Turan and Selçuk Çömlekçi, *Development of biological voxel-based computational models from Dicom files for 3-D electromagnetic simulations*, Ecology Letters 8, 148–159.
5. [J. McDonald 2009] Robert J. McDonald, Leigh A. Gray, Harry J. Cloft et al., *The Effect of Operator Variability and Experience in Vertebroplasty Outcomes*, Radiology 253(2), 478-485.
6. [A Vinson 2012] Joe A Vinson, Bryan R Burnham and Mysore V Nagendran, *Randomized, double-blind, placebo-controlled, linear dose, crossover study to evaluate the efficacy and safety of a green coffee bean extract in overweight subjects*, Diabetes, Metabolic Syndrome and Obesity: Targets and Therapy 5, 21–27.
7. [Vieira 2011] Sonia Vieira and José Eduardo Corrente, *Statistical methods for assessing agreement between double readings of clinical measurements*, J Appl Oral Sci. 19(5), 488-92.
8. [Webber 2013] Colleen Webber, Michael D. Brundage, David Robert Siemens et al., *Quality of care indicators and their related outcomes: A population-based study in prostate cancer patients treated with radiotherapy*, Radiotherapy and Oncology 107, 358–365.
9. [Mulliez 2013] Thomas Mulliez, Bruno Speleers, Indira Madani et al., *Whole breast radiotherapy in prone and supine position: is there a place for multi-beam IMRT?*, Radiation Oncology 8, 151.
10. [Feuvret 2006] Loïc Feuvret, Georges Noël et al., *Conformity Index: A Review*, Int. J. Radiation Oncology Biol. Phys. 64(2), 333–342.
11. [Chen 2013] Yie Chen, Jie Qu, Jack Yang et al., *Planning and Dosimetric Comparisons of IMRT Lung Cancers with Three Advanced Optimization Algorithms*, International Journal of Medical Physics, Clinical Engineering and Radiation Oncology 2, 52-60.
12. Health Technology Assessment Orientamenti 2, *Innovative radiation treatment in cancer: IGRT/IMRT*, DOSSIER 199-2010, ISSN 1591-223X.
13. [Eklund 2013] Anders Eklund, Paul Dufort et al., *Medical image processing on the GPU – Past, present and future*, Medical Image Analysis 17, 1073–1094.
14. [Sznitman 2013] Josué Sznitman, *Respiratory microflows in the pulmonary acinus*, Journal of Biomechanics 46, 284–298.

-
15. [Broggio 2013] D. Broggio, A. Moignier, K. Ben Brahim et al., *Comparison of organs' shapes with geometric and Zernike 3D moments*, Computer methods and programs in biomedicine III, 740–754.
 16. [C.Y. Chan 2013] Timothy C.Y. Chan and Velibor V. Mišić, *Adaptive and robust radiation therapy optimization for lung cancer*, European Journal of Operational Research 231, 745–756.
 17. [De Ruyscher 2013] Dirk De Ruyscher, José Belderbos, Bart Reymen et al., *State of the Art Radiation Therapy for Lung Cancer 2012: A Glimpse of the Future*, Clinical Lung Cancer 14(2), 89–95.
 18. [Foroudi 2013] Farshad Foroudi, Daniel Pham, Mathias Bressel et al., *Intrafraction Bladder Motion in Radiation Therapy Estimated From Pretreatment and Posttreatment Volumetric Imaging*, Int J Radiation Oncol Biol Phys 86(1), 77–82.
 19. [De Los Santos 2013] Jennifer De Los Santos, Richard Pople, Nzhde Agazaryan et al., *Image Guided Radiation Therapy (IGRT) Technologies for Radiation Therapy Localization and Delivery*, Int J Radiation Oncol Biol Phys 87(1), 33–45.
 20. [Fu 2013] Weihua Fu, Yong Yang, Ning J. Yue et al., *Dosimetric influences of rotational setup errors on head and neck carcinoma intensity-modulated radiation therapy treatments*, Medical Dosimetry 38, 125–132.
 21. [Champ 2012] Colin E Champ, Joshua Siglin, Mark V Mishra et al., *Evaluating changes in radiation treatment volumes from post-operative to same-day planning MRI in High-grade gliomas*, Radiation Oncology 7:220.
 22. [Sotiras 2013] Aristeidis Sotiras, Christos Davatzikos, Senior Member and Nikos Paragios, *Deformable Medical Image Registration: A Survey*, IEEE transactions on medical imaging 32(7), 1153–1190.
 23. [Mi 2013] Hongmei Mi, Caroline Petitjean, Su Ruan et al., *Predicting lung tumor evolution during radiotherapy from PET images using a patient specific model*, IEEE 10th International Symposium on Biomedical Imaging: From Nano to Macro, 1404–1407.
 24. [Castadot 2008] Pierre Castadot, John Aldo Lee, Adriane Parraga et al., *Comparison of 12 deformable registration strategies in adaptive radiation therapy for the treatment of head and neck tumors*, Radiotherapy and Oncology 89, 1–12.
 25. [Cheung 2009] Joey Cheung, Jean-Francois Aubry, Sue S. Yom et al., *Dose recalculation and the dose-guided radiation therapy (DGRT) process using Megavoltage Cone-Beam CT*, Int. J. Radiation Oncology Biol. Phys. 74(2), 583–592.
 26. [Lee 2008] Choonik Lee, Katja M. Langen, Weiguo Lu et al., *Evaluation of geometric changes of parotid glands during head and neck cancer radiotherapy using daily MVCT and automatic deformable registration*, Radiotherapy and Oncology 89, 81–88.
 27. [Kataria 2012] Tejinder Kataria, Kuldeep Sharma, Vikraman Subramani et al., *Homogeneity Index: An objective tool for assessment of conformal radiation treatments*, J Med Phys. 37(4), 207–213.
 28. [W. Pluim 2003] Josien P.W. Pluim, J.B. Antoine Maintz and Max A. Viergever et al., *Mutual-Information-Based Registration of Medical Images: A Survey*, IEEE transactions on medical imaging 22(8), 986–1004.

-
29. [H. Balagamwala 2012] Ehsan H. Balagamwala, John H. Suh, Gene H. Barnett et al., *The Importance of the Conformality, Heterogeneity, and Gradient Indices in Evaluating Gamma Knife Radiosurgery Treatment Plans for Intracranial Meningiomas*, Int J Radiation Oncol Biol Phys 83(5), 1406-1413.
 30. [D'Arrigo 2011] Graziella D'Arrigo, Fabio Provenzano, Claudia Torino et al., *I test diagnostici e l'analisi della curva ROC*, G Ital Nefrol 28 (6), 642-647.
 31. [Hun Oh 2009] Jung Hun Oh and Issam El Naqa, *Bayesian network learning for detecting reliable interactions of dose-volume related parameters in radiation pneumonitis*, International Conference on Machine Learning and Applications, 484-488.
 32. [J. Brown 2001] P.J. Brown, M.G. Kenward, E.E. Bassett, *Bayesian discrimination with longitudinal data*, Biostatistics 2(4), 417-432.
 33. [D. Johnson 2005] Timothy D. Johnson, Jeremy M.G. Taylor et al., *A Bayesian mixture model relating dose to critical organs and functional complication in 3D conformal radiation therapy*, Biostatistics 6(4), 615-632.
 34. [H. Zwinderman 2008] Aeilko H. Zwinderman, Afina S. Glas, Patrick M. Bossuyt et al., *A Statistical models for quantifying diagnostic accuracy with multiple lesions per patient*, Biostatistics 9(3), 513-522.
 35. [Rodriguez 2009] Abel Rodriguez, David B. Dunson and Jack Taylor, *A Bayesian hierarchically weighted finite mixture models for samples of distributions*, Biostatistics 10(1), 155-171.
 36. [P. Smith 2009] Wade P. Smith, Jason Doctor, Jürgen Meyer et al., *A decision aid for intensity-modulated radiationtherapy plan selection in prostate cancer based on a prognostic Bayesian network and a Markov model*, Artificial Intelligence in Medicine 46, 119-130.
 37. John Paul Mueller, *Professional IronPython, Design and Develop IronPython Techniques*, Wiley Publishing, Inc., 2010
 38. [Rohlfing 2005] Torsten Rohlfing, Robert Brandt, Randolf Menzel et al., *Quo Vadis, Atlas-Based Segmentation?*, The Handbook of Biomedical Image Analysis, in: Topics in Biomedical Engineering Internation Book Series, 435-486.
 39. José Silvio Falla, Tesi: *Sviluppo di algoritmi e interfaccia grafica per applicazioni in ambito medico a supporto dello studio perfusionale*, Corso di laurea in ing. Elettronica, Università degli Studi di Bologna - Sede di Cesena A.A. 2011-2012.
 40. Salvatore La Bua , Calogero Crapanzanierto Amato, Tesi: *Decodifica di files DICOM e restauro di immagini di Risonanza Magnetica*, Corso di Laurea in Ing. Informatica per i Sistemi Intelligenti, Università degli Studi di Palermo A.A. 2005-2006.
 41. Antonio Candito, Tesi: *Integrazione informatica dei sistemi di medicina nucleare nel sistema informativo ospedaliero*, Corso di Laurea in Ingegneria Biomedica, Università degli Studi di Bologna A.A. 2011-2012.
 42. Francesco Lucchi, Tesi: *Utilizzo dello standard IHE per la gestione dei flussi informativi in medicina nucleare*, Corso di Laurea in Ingegneria Biomedica, Università degli Studi di Bologna - Sede di Cesena A.A. 2011-2012.
 43. [Xu 2007] Peng Xu and DeZhong Yao, *A study on medical image registration by mutual information with pyramid data structure*, Computers in Biology and Medicine 37, 320 - 327

-
44. [W. Pluim 2000] Josien P.W. Pluim, J.B. Antoine Maintz, and Max A. Viergever, *Interpolation Artefacts in Mutual Information-Based Image Registration*, Computer Vision and Image Understanding 77, 211–232
 45. [Crop 2012] F. Crop, A. Bernard, and N. Reynaert, *Improving dose calculations on tomotherapy MVCT images*, Journal of applied clinical medical physics 13(6), 241-253.
 46. [V. Wyawahare 2009] Medha V. Wyawahare, Pradeep M. Patil, and Hemant K. Abhyankar, *Image Registration Techniques: An overview*, International Journal of Signal Processing, Image Processing and Pattern Recognition 2(3), 11-28.
 47. [Hun Oh 2009] Jung Hun Oh and Issam El Naqa, *Bayesian network learning for detecting reliable interactions of dose-volume related parameters in radiation pneumonitis*, International Conference on Machine Learning and Applications, 484-488.
 48. [Dalirsefat 2009] Seyed Benyamin Dalirsefat, Andréia da Silva Meyer and Seyed Ziyaeddin Mirhoseini, *Comparison of similarity coefficients used for cluster analysis with amplified fragment length polymorphism markers in the silkworm, Bombyx mori*, Journal of Insect Science 9(71), 1-8.
 49. [P. Smith 2009] Wade P. Smith, Jason Doctor et al., *A decision aid for intensity-modulated radiationtherapy plan selection in prostate cancer based on a prognostic Bayesian network and a Markov model*, Artificial Intelligence in Medicine 46, 119-130.
 50. Debora Slanzi, Tesi: *Reti Bayesiane: Approcci per la Selezione del Modello*, Dottorato di Ricerca in Statistica, Università degli Studi di Padova A.A. 2005-2006.
 51. [Meyer JL 2011] John L Meyer: *IMRT, IGRT, SBRT – Advances in the Treatment Planning and Delivery of Radiotherapy*, ed. 2, Front Radiat Ther Oncol. Basel, Karger, 43, 165-180.
 52. [D’Arienzo 2012] Marco D’Arienzo, Stefano G. Masciullo, Vitaliana de Sanctis et al.: *Integral Dose and Radiation-Induced Secondary Malignancies: Comparison between Stereotactic Body Radiation Therapy and Three-Dimensional Conformal Radiotherapy*, Int J Environ Res Public Health 9(11), 4223-4240.
 53. Mario Esposito, Tesi di dottorato: *Il problema della co-registrazione di immagini biomedicali e sue applicazioni nell’ambito radiologico e diagnostico*, Dottorato di Ricerca in Scienze Computazionali ed Informatiche, Università degli Studi di Napoli Federico II - CICLO XII.
 54. [Vecchi 2014] Vecchi C, Guidi G, Maffei N et al., *Deformable registration using python scripting for clinical automation*, ESTRO 33, (in press).
 55. [Maffei 2014] Maffei N, Guidi G, Vecchi C et al., *Warping methods for Tomotherapy and IGRT: challenge and predictive analysis in clinical practice*, ESTRO 33, (in press).

Sitografia

Geometria:

1. <http://electroncastle.com/wp/?p=416>
2. http://en.wikipedia.org/wiki/Rotation_matrix
3. http://www.dima.unige.it/~cavalier/ScAmb/GEOMda42a48_Vettori.PDF
4. <http://calvino.polito.it/~salamon/P/G/alga11.pdf>

Programmazione:

5. <http://www.redbaron85.com/guide-di-programmazione/49/>
6. <http://nicolaiarocci.com/impara-python-in-10-minuti/>
7. <http://www.html.it/pag/15610/linterprete-interattivo/>

DICOM:

8. <http://it.wikipedia.org/wiki/DICOM>
9. <http://www.federica.unina.it/smfn/metodologie-analisi-immagini/dicom/>
10. <http://www.rm-online.it/forums/topic/10825-f0-immagine-in-rm-e-dicom/>
11. <http://digilander.libero.it/openworks/dicom.html>
12. <http://medical.nema.org/standard.html>
13. http://www.unipd.it/nucmed/TF/TSRM/immagini_integrate/TSRM_11_DICOM.pdf
14. <http://www.trophos.org/it/partner-progetto/regolatorie/dicom>
15. <https://github.com/cvdlab-bio/web3d/blob/master/docs/dicom.md>
16. <http://www.sgsmp.ch/dicom/neumann1.pdf>

Software:

17. <http://www.slicer.org/slicerWiki/index.php/Documentation/4.3>
18. http://www.namic.org/Wiki/images/6/66/3DVisualizationDICOM_RadiologyApplications_SoniaPujol_RSNA2012.pdf
19. http://www.na-mic.org/Wiki/images/0/0f/Slicer4_ProgrammingTutorial_SPujol-SPieper.pdf
20. ftp://public.dhe.ibm.com/software/analytics/spss/documentation/statistics/22.0/en/client/Manuals/IBM_SPSS_Statistics_Brief_Guide.pdf
21. ftp://public.dhe.ibm.com/software/analytics/spss/documentation/statistics/22.0/en/client/Manuals/IBM_SPSS_Statistics_Core_System_User_Guide.pdf
22. http://www.slicer.org/slicerWiki/images/a/ae/Slicer4minute-tutorial_SoniaPujol-mj.pdf

Statistica:

23. http://en.wikipedia.org/wiki/Markov_chain
24. http://www.pp.rhul.ac.uk/~cowan/stat_course.html
25. <http://www.pr-owl.org/basics/bn.php>
26. <http://www.ccad.uiowa.edu/MIMX/projects/IA-FEMesh>
27. http://en.wikipedia.org/wiki/Root_mean_square

-
28. http://areeweb.polito.it/didattica/gcia/lucidi/Esercizi_Approfondimenti/Reti_credenza_Cheminod/Reti_credenza_Cheminod.pdf
 29. <http://vision.unipv.it/IA/10-RetiBayesiane.pdf>
 30. <http://www.cs.mcgill.ca/~dprecup/courses/Prob/Lectures/prob-lecture02.pdf>
 31. <http://www.autonlab.org/tutorials/bayesinf05.pdf>
 32. http://www.csse.monash.edu.au/bai/book/BAI_Chapter2.pdf
 33. <http://www.fit.vutbr.cz/study/courses/VPD/public/0809VPD-Vanek.pdf>
 34. <http://www.soa.org/library/newsletters/forecasting-futurism/2013/july/ffn-2013-iss7-heaton.aspx>
 35. http://www.agenarisk.com/resources/Introduction_To_Bayesian_Networks.pdf
 36. <http://doc.utwente.nl/36632/1/0000006a.pdf>
 37. <http://www.disco.unimib.it/upload/MAD23aprile.pdf>
 38. http://aim.labmedinfo.org/bellazzi/mpm/dati/Reti_Bayesiane.pdf
 39. http://www.cse.buffalo.edu/courses/cse725/peter/Murphy_2003.pdf
 40. <http://www.csee.umbc.edu/courses/671/fall09/notes/bbn.ppt>
 41. <http://www.dii.unisi.it/~mocenni/lezione7.ppt>
 42. http://www.ics.uci.edu/~smyth/courses/cs271/topic11_bayesian_networks.ppt
 43. <http://www.ecse.rpi.edu/Homepages/qji/TutFinbd.ppt>
 44. <http://dimacs.rutgers.edu/Workshops/Surveillance/slides/wong.ppt>

Analisi immagini:

45. http://en.wikipedia.org/wiki/Image_registration
46. http://en.wikipedia.org/wiki/Medical_image_computing
47. http://depts.washington.edu/bicg/documents/MII_registration10.pdf
48. <http://www.fil.ion.ucl.ac.uk/spm/doc/books/hbf2/pdfs/Ch2.pdf>
49. http://www.cse.msu.edu/~cse902/S03/mut_info.ppt
50. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.39.4417&rep=rep1&type=pdf>
51. http://appuntamentoiafisica.to.infn.it/2012/1/doc/Riccardo_Ragona.pdf

Radioterapia:

52. <http://en.wikipedia.org/wiki/Tomotherapy>
53. http://en.wikipedia.org/wiki/Radiation_therapy
54. http://www.radioterapiaitalia.it/cont__56.phtml
55. [http://www.avl.nl/media/517544/Adaptive%20radiotherapy%20\(ART\).pdf](http://www.avl.nl/media/517544/Adaptive%20radiotherapy%20(ART).pdf)
56. <http://medicaldosimetry.org/meetings/2013regionIII/dasimrt.pdf>
57. <http://pablocirrone.wikispaces.com/file/view/MasterAdroterapia2009Lezione4.pdf>
58. http://www.salutedomani.com/il_weblog_di_antonio/2009/10/radioterapia-guidata-dalle-immagini-igrt-incontro-a-torino.html
59. <http://aapm.org/meetings/2011SS/documents/MackieUncertainty.pdf>
60. <http://www.aapm.org/meetings/amos2/pdf/49-14512-46489-282.pdf>

Appendice A

Gli script di seguiti riportati, sono stati implementati nell'ambito del progetto italiano:
BANDO 2010 RICERCA FINALIZZATA E GIOVANI RICERCATORI del Ministero
della Salute.

MoH2010 (GR-2010-2318757) Dose warping methods for IGRT and ADAPTIVERT:
DOSE ACCUMULATION BASED ON ORGAN MOTION AND ANATOMICAL
VARIATIONS OF THE PATIENTS DURING RADIATION THERAPY
TREATMENTS.

Ogni script pertanto è parte integrante del progetto e verrà utilizzato per l'attività clinica
giornaliera in radioterapia dell'Azienda Ospedaliera Universitaria del Policlinico di
Modena. Si riporta di seguito l'intestazione di ogni script sviluppato, che nella presente
appendice verrà omessa per ridondanza.

```
# RayStation version: 3.99.0.7 Not for clinical use #
#####
# Azienda Ospedaliera Universitaria di Modena - Policlinico #
# Servizio Fisica Sanitaria #
# Dott. Gabriele Guidi #
# E-mail: guidi.gabriele@policlinico.mo.it #
# Research Project: MoH2010 (GR-2010-2318757) Dose warping methods for #
# IGRT and ADAPTIVERT #
#####
# Universita' degli Studi di Bologna #
# Script made by: Dott. Claudio Vecchi #
# E-mail: claudio.vecchi@studio.unibo.it #
#####
```

1. ImportPatient_WithoutRTPLAN.py

```
#Importa tutte le serie del paziente in base al filtro impostato
(es.PatientID), tranne RTPLAN#

from connect import *

import connect

#Accede all'oggetto PatientDB tramite il metodo get_current
db = connect.get_current('PatientDB')

p_id = "068779"

#Crea una variabile "series" con la lista degli esami in base al filtro
PatientID della query
series = db.QuerySeriesFromPath(Path = r"Y:\DICOM", Filter =
{'PatientID': '%s' %(p_id)})

#Crea una variabile "end_exam" con il numero totale degli esami del
paziente
end_exam = len(series)

#Crea una lista vuota di nome "lista"
lista = []

#Ciclo che inserisce i SeriesNumber degli esami nella variabile lista
#Inserisce in variabili separate i SeriesNumber dei file
RTPLAN,RTDOSE,RTSTRUCT
for i in range(0,end_exam):
    select_parola = series[i]
    modality = select_parola['Modality']
    Series_Desc = select_parola['SeriesDescription']
    if modality == "RTPLAN":
        series_number = select_parola['SeriesNumber']
        RTPLAN = series_number
        continue
    else:
        if modality == "RTDOSE":
            series_number = select_parola['SeriesNumber']
            RTDOSE = series_number
            continue
        else:
            if modality == "RTSTRUCT":
                series_number = select_parola['SeriesNumber']
                RTSTRUCT = series_number
                continue
            else:
                if Series_Desc == "kVCT Image Set":
                    series_number = select_parola['SeriesNumber']
                    KVCT = series_number
                    continue
                else:
                    series_number = select_parola['SeriesNumber']
                    lista[len(lista):len(lista)] = [series_number]

lista.sort() #Ordina "lista"
endlista = len(lista) #Inserisce nella variabile "endlista" il numero
totale degli esami (senza RTPLAN, RTSTRUCT, RTDOSE)
lista.insert(0, KVCT) #Inserisce il valore di KVCT nella posizione 0
di "lista" (prima posizione)
lista.insert(1, RTSTRUCT) #Inserisce il valore di RTSTRUCT nella
posizione 1 di "lista" (seconda posizione)
lista.insert(endlista+1, RTDOSE) #Inserisce il valore di RTDOSE
alla fine di "lista"

#Importa il primo esame del paziente
warnings = db.ImportPatientFromPath(Path = r"Y:\DICOM", Patient
= {'Id': '%s' %(p_id)}, SeriesFilter = {'SeriesNumber': '%d' %
lista[0]})

#Accede all'oggetto Patient tramite il metodo get_current
patient = get_current("Patient")

#Salva paziente
patient.Save()

#Importa i restanti esami
for i in range(1,len(lista)):
    warnings = patient.ImportDicomDataFromPath(Path =
r"Y:\DICOM", SeriesFilter = {'SeriesNumber': '%d' % lista[i]})

#Salva paziente
patient.Save()
```

2. RigidRegistration.py

```
#Esegue la co-registrazione rigida tra gli esami del paziente#

from connect import *

#Accede all'oggetto Patient tramite il metodo get_current
patient = get_current("Patient")

#Esegue un ciclo di controllo in cui trova la ROI "body" e la
rinomina in "Body"
for r in patient.PatientModel.RegionsOfInterest:
    if r.Name == "body":
        r.Name = "Body"
        break
    else: continue

#Esegue un ciclo di controllo in cui trova la ROI "BODY" e la
rinomina in "Body"
for r in patient.PatientModel.RegionsOfInterest:
    if r.Name == "BODY":
        r.Name = "Body"
        break
    else: continue

#Setta la ROI "Body" come esterna
patient.PatientModel.RegionsOfInterest['Body'].SetAsExternal()

#Crea variabile "ExamNames" e "ExamNames2" con la lista dei nomi
degli esami importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]
ExamNames2 = [r.Name for r in patient.Examinations]

#Crea variabile "ExamNum" con numero elementi della lista
"ExamNames" (numero totale esami)
ExamNum = len(ExamNames)

#Ordina elementi della lista ExamNames per nome
ExamNames.sort()

#Salva il primo elemento della lista ExamNames (CT 1) ed eliminalo
FirstExam = ExamNames[0]

#Ciclo che esegue la co-registrazione rigida tra gli esami
for i in range(2,ExamNum+1):
    RefExam = ExamNames2[i-1] #Inserisce in "RefExam" il nome
del Reference Exam

patient.ComputeRigidImageRegistration(FloatingExaminationName
="%s" %(FirstExam), ReferenceExaminationName="%s" %
(RefExam),
    UseOnlyTranslations=False, HighWeightOnBones=True,
    InitializeImages=True, FocusRoisNames=[])
patient.Save() #Salva il paziente
```

2.1. RigidRegistration_NoRotations.py

#Esegue la co-registrazione rigida tra gli esami del paziente, senza rotazioni di immagine#

```
from connect import *

#Accede all'oggetto Patient tramite il metodo get_current
patient = get_current("Patient")

#Esegue un ciclo di controllo in cui trova la ROI "body" e la
rinomina in "Body"
for r in patient.PatientModel.RegionsOfInterest:
    if r.Name == "body":
        r.Name = "Body"
        break
    else: continue

#Esegue un ciclo di controllo in cui trova la ROI "BODY" e la
rinomina in "Body"
for r in patient.PatientModel.RegionsOfInterest:
    if r.Name == "BODY":
        r.Name = "Body"
        break
    else: continue

#Setta la ROI "Body" come esterna
patient.PatientModel.RegionsOfInterest['Body'].SetAsExternal()

#Crea variabile "ExamNames" e "ExamNames2" con la lista dei nomi
degli esami importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]
ExamNames2 = [r.Name for r in patient.Examinations]

#Crea variabile "ExamNum" con numero elementi della lista
"ExamNames" (numero totale esami)
ExamNum = len(ExamNames)

#Ordina elementi della lista ExamNames per nome
ExamNames.sort()

#Salva il primo elemento della lista ExamNames (CT 1) ed eliminalo
FirstExam = ExamNames[0]

#Ciclo che esegue la co-registrazione rigida tra gli esami
for i in range(2,ExamNum+1):

    RefExam = ExamNames2[i-1] #Inserisce in "RefExam" il nome
del Reference Exam

patient.ComputeRigidImageRegistration(FloatingExaminationName
="%s" %(FirstExam), ReferenceExaminationName="%s" %
(RefExam),
    UseOnlyTranslations=True, HighWeightOnBones=True,
InitializeImages=True, FocusRoisNames=[])
patient.Save() #Salva il paziente
```

2.2. RigidRegistration_NoRotations_NoFocusOnBones.py

#Esegue la co-registrazione rigida tra gli esami del paziente, senza focus sulle ossa#

```
from connect import *

#Accede all'oggetto Patient tramite il metodo get_current
patient = get_current("Patient")

#Esegue un ciclo di controllo in cui trova la ROI "body" e la
rinomina in "Body"
for r in patient.PatientModel.RegionsOfInterest:
    if r.Name == "body":
        r.Name = "Body"
        break
    else: continue

#Esegue un ciclo di controllo in cui trova la ROI "BODY" e la
rinomina in "Body"
for r in patient.PatientModel.RegionsOfInterest:
    if r.Name == "BODY":
        r.Name = "Body"
        break
    else: continue

#Setta la ROI "Body" come esterna
patient.PatientModel.RegionsOfInterest['Body'].SetAsExternal()

#Crea variabile "ExamNames" e "ExamNames2" con la lista dei nomi
degli esami importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]
ExamNames2 = [r.Name for r in patient.Examinations]

#Crea variabile "ExamNum" con numero elementi della lista
"ExamNames" (numero totale esami)
ExamNum = len(ExamNames)

#Ordina elementi della lista ExamNames per nome
ExamNames.sort()

#Salva il primo elemento della lista ExamNames (CT 1) ed eliminalo
FirstExam = ExamNames[0]

#Ciclo che esegue la co-registrazione rigida tra gli esami
for i in range(2,ExamNum+1):

    RefExam = ExamNames2[i-1] #Inserisce in "RefExam" il nome
del Reference Exam

patient.ComputeRigidImageRegistration(FloatingExaminationName
="%s" %(FirstExam), ReferenceExaminationName="%s" %
(RefExam),
    UseOnlyTranslations=True, HighWeightOnBones=False,
InitializeImages=True, FocusRoisNames=[])
patient.Save() #Salva il paziente
```

3. CopyROI.py

```
#Esegue il Copy ROI dal primo esame a tutti gli altri#

from connect import *

#Accede all'oggetto Patient tramite il metodo get_current
patient = get_current("Patient")

#Crea variabile roi_names di tipo lista con i nomi delle ROI del
paziente
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]

#Crea variabile roi_numbers di tipo lista con i numeri delle ROI del
paziente
roi_numbers = [r.RoiNumber for r in
patient.PatientModel.RegionsOfInterest]

#Crea variabile ExamNames di tipo lista con i nomi degli esami
importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]

#Crea variabile ExamNum con numero elementi della lista
ExamNames (numero totale esami)
ExamNum = len(ExamNames)

#Ordina elementi della lista ExamNames per nome
ExamNames.sort()

#Salva il primo elemento della lista ExamNames (CT 1) in
"FirstExam" e rimuovilo
FirstExam = ExamNames[0]
del ExamNames[0:1]

#Copia le ROI dal primo esame a tutti gli altri
Copy_roi =
patient.PatientModel.CopyRoiGeometries(SourceExamination =
patient.Examinations["%s" % (FirstExam)],
TargetExaminationNames = ExamNames, RoiNames = roi_names)

#Salva paziente
patient.Save()
```

4. HybridDeformation_CC.py

```
#Esegue la deformazione ibrida con algoritmo CC(cross-correlation)
tra gli esami del paziente#

from connect import *

import sys

#Accede all'oggetto Patient tramite il metodo get_current
patient = get_current("Patient")

#Crea variabile "ExamNames" e "ExamNames2" con la lista dei
nomi degli esami importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]
ExamNames2 = [r.Name for r in patient.Examinations]

#Crea variabile "ExamNum" con numero elementi della lista
ExamNames (numero totale esami)
ExamNum = len(ExamNames)

#Ordina elementi della lista ExamNames per nome
ExamNames.sort()

#Salva il primo elemento della lista ExamNames (CT 1)
FirstExam = ExamNames[0]

#Crea variabile "roi_names" con la lista dei nomi delle ROI del
paziente
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]

#Crea variabile "roi_numbers" con la lista dei numeri delle ROI del
paziente
roi_numbers = [r.RoiNumber for r in
patient.PatientModel.RegionsOfInterest]

#Crea variabile "BodyObj" con l'oggetto Body
BodyObj = patient.PatientModel.RegionsOfInterest['Body']

#Crea variabile "BodyNumber" con il numero della ROI Body
BodyNumber = BodyObj.RoiNumber

#Rimuovi il nome della ROI Body dalla lista "roi_names"
roi_names.remove('Body')

#Rimuovi il numero della ROI Body dalla lista "roi_numbers"
roi_numbers.remove(BodyNumber)

try:
    #Prova ad eseguire il ciclo, se un'eccezione
    si verifica esegui except
    for i in range(2,ExamNum+1): #Ciclo per deformare tutte le
    ROI in tutti gli esami del paziente

        RefExam = ExamNames2[i-1]

        #Crea un dizionario "diz" con i
        valori della griglia di deformazione
        diz =
        patient.GetDeformationGrid(ReferenceExaminationName = "%s" %
        (RefExam), TargetExaminationsNames = ["%s" % (FirstExam)],
        ControllingROIsNames = ['Body'], ControllingPOIsNames
        = [], FocusROIsNames = roi_names, CoverDoseGrids = True,
        VoxelSize={ 'x': 0.25, 'y': 0.25, 'z': 0.25 })

        #Esegue la deformazione a livello avanzato con
        algoritmo di cross-correlation "CC"

        patient.ComputeHybridDeformableImageRegistrationAdvanced(Reg
        istrationGroupName="HybridDef %d-" % (i),
        ReferenceExaminationName="%s" % (RefExam),
        TargetExaminationsNames=["%s" % (FirstExam)],
        ControllingRoiNumbers=[BodyNumber],
        ControllingPoiNumbers=[],
        FocusROINumbers=roi_numbers,
        DeformationGridCorner=diz['Corner'],
        DeformationGridSize=diz['GridSize'],
        DeformationGridVoxelSize={ 'x': 0.25, 'y': 0.25, 'z': 0.25 },

        GaussianSmoothingSigma=[[0.3333333333333333,0.3333333333333333
        33,0.3333333333333333]],
        ResolutionLevels=[[{ 'x': 1, 'y': 1, 'z': 1 }, { 'x': 0.5, 'y': 0.5,
        'z': 0.5 }, { 'x': 0.25, 'y': 0.25, 'z': 0.25 }]],
        GridSmoothingWeight=[[200,200,200]],
        ShapeConstrainingWeight=[1], LandmarkConstrainingWeight=[10],
        ControllingRoiWeight=[10],
```

```
MaxNumberOfIterationsAtResolutionLevel=[[1000,1000,1000]],  
ImageSimilarityMeasures=["CC","CC","CC"])
```

```
patient.Save() #Salva paziente
```

```
except:
```

```
print "Cannot do the cycle" #Stampa la stringa in caso di errore  
sys.exit() #Esci dallo script
```

4.1. HybridDeformation_CCmixedMI.py

```
#Esegue la deformazione ibrida con algoritmo misto CC(cross-  
correlation) e MI(mutual information) tra gli esami del paziente#
```

```
from connect import *
```

```
import sys
```

```
#Accede all'oggetto Patient tramite il metodo get_current  
patient = get_current("Patient")
```

```
#Crea variabile "ExamNames" e "ExamNames2" con la lista dei  
nomi degli esami importati (es. CT 1, CT2..)  
ExamNames = [r.Name for r in patient.Examinations]  
ExamNames2 = [r.Name for r in patient.Examinations]
```

```
#Crea variabile "ExamNum" con numero elementi della lista  
ExamNames (numero totale esami)  
ExamNum = len(ExamNames)
```

```
#Ordina elementi della lista ExamNames per nome  
ExamNames.sort()
```

```
#Salva il primo elemento della lista ExamNames (CT 1) ed eliminato  
FirstExam = ExamNames[0]
```

```
#Crea variabile "roi_names" con la lista dei nomi delle ROI del  
paziente  
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]
```

```
#Crea variabile "roi_numbers" con la lista dei numeri delle ROI del  
paziente  
roi_numbers = [r.RoiNumber for r in  
patient.PatientModel.RegionsOfInterest]
```

```
#Crea variabile "BodyObj" con l'oggetto Body  
BodyObj = patient.PatientModel.RegionsOfInterest['Body']
```

```
#Crea variabile "BodyNumber" con il numero della ROI Body  
BodyNumber = BodyObj.RoiNumber
```

```
#Rimuovi il nome della ROI Body dalla lista "roi_names"  
roi_names.remove('Body')
```

```
#Rimuovi il numero della ROI Body dalla lista "roi_numbers"  
roi_numbers.remove(BodyNumber)
```

```
try: #Prova ad eseguire il ciclo, se un'eccezione  
si verifica esegui except  
for i in range(2,ExamNum+1): #Ciclo per deformare tutte le  
ROI in tutti gli esami del paziente
```

```
RefExam = ExamNames2[i-1]
```

```
#Crea un dizionario "diz" con i  
valori della griglia di deformazione
```

```
diz =  
patient.GetDeformationGrid(ReferenceExaminationName = "%s" %  
(RefExam), TargetExaminationsNames = ["%s" % (FirstExam)],  
ControllingROIsNames = ['Body'], ControllingPOIsNames  
= [], FocusROIsNames = roi_names, CoverDoseGrids = True,  
VoxelSize={ 'x': 0.25, 'y': 0.25, 'z': 0.25 })
```

```
#Esegue la deformazione a livello avanzato con  
algoritmo di cross-correlation "CC"
```

```
patient.ComputeHybridDeformableImageRegistrationAdvanced(Reg  
istrationGroupName="HybridDef %d-" % (i),  
ReferenceExaminationName="%s" % (RefExam),  
TargetExaminationsNames=["%s" % (FirstExam)],  
ControllingRoiNumbers=[BodyNumber],  
ControllingPoiNumbers=[],  
FocusROINumbers=roi_numbers,  
DeformationGridCorner=diz['Corner'],  
DeformationGridSize=diz['GridSize'],  
DeformationGridVoxelSize={ 'x': 0.25, 'y': 0.25, 'z': 0.25 },
```

```
GaussianSmoothingSigma=[[0.3333333333333333,0.3333333333333333  
33,0.3333333333333333]],
```

```
ResolutionLevels=[[{ 'x': 1, 'y': 1, 'z': 1 }, { 'x': 0.5, 'y': 0.5,  
'z': 0.5 }, { 'x': 0.25, 'y': 0.25, 'z': 0.25 }]],
```

```
GridSmoothingWeight=[[200,200,200]],  
ShapeConstrainingWeight=[1], LandmarkConstrainingWeight=[10],  
ControllingRoiWeight=[10],
```

```

MaxNumberOfIterationsAtResolutionLevel=[[1000,1000,1000]],
ImageSimilarityMeasures=["CC_MIXED_MI","CC_MIXED_MI","
CC_MIXED_MI"])

```

```

patient.Save() #Salva paziente

```

```

except:
    print "Cannot do the cycle" #Stampa la stringa in caso di errore
    sys.exit() #Esci dallo script

```

4.2. HybridDeformation_ML.py

```

#Esegue la deformazione ibrida con algoritmo MI(mutual
information) tra gli esami del paziente#

```

```

from connect import *

```

```

import sys

```

```

#Accede all'oggetto Patient tramite il metodo get_current
patient = get_current("Patient")

```

```

#Crea variabile "ExamNames" e "ExamNames2" con la lista dei
nomi degli esami importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]
ExamNames2 = [r.Name for r in patient.Examinations]

```

```

#Crea variabile "ExamNum" con numero elementi della lista
ExamNames (numero totale esami)
ExamNum = len(ExamNames)

```

```

#Ordina elementi della lista ExamNames per nome
ExamNames.sort()

```

```

#Salva il primo elemento della lista ExamNames (CT 1) ed eliminato
FirstExam = ExamNames[0]

```

```

#Crea variabile "roi_names" con la lista dei nomi delle ROI del
paziente
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]

```

```

#Crea variabile "roi_numbers" con la lista dei numeri delle ROI del
paziente
roi_numbers = [r.RoiNumber for r in
patient.PatientModel.RegionsOfInterest]

```

```

#Crea variabile "BodyObj" con l'oggetto Body
BodyObj = patient.PatientModel.RegionsOfInterest['Body']

```

```

#Crea variabile "BodyNumber" con il numero della ROI Body
BodyNumber = BodyObj.RoiNumber

```

```

#Rimuovi il nome della ROI Body dalla lista "roi_names"
roi_names.remove('Body')

```

```

#Rimuovi il numero della ROI Body dalla lista "roi_numbers"
roi_numbers.remove(BodyNumber)

```

```

try: #Prova ad eseguire il ciclo, se un'eccezione
si verifica esegui except
    for i in range(2,ExamNum+1): #Ciclo per deformare tutte le
ROI in tutti gli esami del paziente

```

```

        RefExam = ExamNames2[i-1]

```

```

        #Crea un dizionario "diz" con i
valori della griglia di deformazione
        diz =

```

```

patient.GetDeformationGrid(ReferenceExaminationName = "%s" %
(RefExam), TargetExaminationsNames = ["%s" % (FirstExam)],
ControllingROIsNames = ['Body'], ControllingPOIsNames
= [], FocusROIsNames = roi_names, CoverDoseGrids = True,
VoxelSize={ 'x': 0.25, 'y': 0.25, 'z': 0.25 })

```

```

        #Esegue la deformazione a livello avanzato con
algoritmo di cross-correlation "CC"

```

```

patient.ComputeHybridDeformableImageRegistrationAdvanced(Reg
istrationGroupName="HybridDef %d-" % (i),
ReferenceExaminationName="%s" % (RefExam),
TargetExaminationsNames=["%s" % (FirstExam)],
ControllingRoiNumbers=[BodyNumber],
ControllingPoiNumbers=[],
FocusROINumbers=roi_numbers,
DeformationGridCorner=diz['Corner'],
DeformationGridSize=diz['GridSize'],
DeformationGridVoxelSize={ 'x': 0.25, 'y': 0.25, 'z': 0.25 },

```

```

GaussianSmoothingSigma=[[0.3333333333333333,0.333333333333
33,0.3333333333333333]],

```

```

ResolutionLevels=[[{ 'x': 1, 'y': 1, 'z': 1 }, { 'x': 0.5, 'y': 0.5,
'z': 0.5 }, { 'x': 0.25, 'y': 0.25, 'z': 0.25 }]],

```

```

GridSmoothingWeight=[[200,200,200]],
ShapeConstrainingWeight=[1], LandmarkConstrainingWeight=[10],
ControllingRoiWeight=[10],

```

```
MaxNumberOfIterationsAtResolutionLevel=[[1000,1000,1000]],
ImageSimilarityMeasures=[["MI","MI","MI"]])
```

```
patient.Save() #Salva paziente
```

```
except:
```

```
print "Cannot do the cycle" #Stampa la stringa in caso di errore
sys.exit() #Esci dallo script
```

5. MapRoiReverse.py

```
#Esegue il Map ROI inverso tra le CT della deformazione ibrida
considerata#
```

```
from connect import *
```

```
#Accede all'oggetto Patient tramite il metodo get_current
patient = get_current("Patient")
```

```
#Crea variabile "roi_names" con la lista dei nomi delle ROI del
paziente
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]
```

```
#Rimuovi il nome della ROI "Body" dalla lista "roi_names"
roi_names.remove("Body")
```

```
#Crea variabile "struct_reg" con la lista dei nomi delle registrazioni
fatte
struct_reg = [r.Name for r in
patient.PatientModel.StructureRegistrationGroups]
```

```
#Crea variabile "ExamNames" e "ExamNames2" con la lista dei
nomi degli esami importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]
ExamNames2 = [r.Name for r in patient.Examinations]
```

```
#Crea variabile "ExamNum" con numero elementi della lista
"ExamNames" (numero totale esami)
ExamNum = len(ExamNames)
```

```
#Ordina elementi della lista ExamNames per nome
ExamNames.sort()
```

```
#Salva il primo elemento della lista ExamNames (CT 1) ed eliminalo
FirstExam = ExamNames[0]
```

```
#Ciclo che esegue il Reverse MapROI su tutti gli esami per le ROI
presenti in "roi_names"
for i in range(2,ExamNum+1):
```

```
RefExam = ExamNames2[i-1] #Inserisce in "RefExam" il
nome del Reference Exam
```

```
Struct_group = struct_reg[i-2] #Inserisce in "Struct_group" il
Nome di Structure Registration Group
```

```
patient.MapRoiGeometriesDeformably(StructureRegistrationGroup
Names = ["%s" %(Struct_group)], ReferenceExaminationNames =
["%s" %(RefExam)],
```

```
TargetExaminationNames = ["%s" %(FirstExam)],
ReverseMapping = True, AbortWhenBadDisplacementField = True,
RoiGeometryNames = roi_names, CreateNewRois = True)
```

```
#Salva paziente
patient.Save()
```

5.1. MapRoiReverse_NoAbort.py

```
#Esegue il Map ROI inverso tra le CT della deformazione ibrida
considerata,#
#interrompendo l'operazione in caso di errore del campo di
spostamento#

from connect import *

#Accede all'oggetto Patient tramite il metodo get_current
patient = get_current("Patient")

#Crea variabile "roi_names" con la lista dei nomi delle ROI del
paziente
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]

#Rimuovi il nome della ROI "Body" dalla lista "roi_names"
roi_names.remove('Body')

#Crea variabile "struct_reg" con la lista dei nomi delle registrazioni
fatte
struct_reg = [r.Name for r in patient.PatientModel.StructureRegistrationGroups]

#Crea variabile "ExamNames" e "ExamNames2" con la lista dei
nomi degli esami importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]
ExamNames2 = [r.Name for r in patient.Examinations]

#Crea variabile "ExamNum" con numero elementi della lista
"ExamNames" (numero totale esami)
ExamNum = len(ExamNames)

#Ordina elementi della lista ExamNames per nome
ExamNames.sort()

#Salva il primo elemento della lista ExamNames (CT 1) ed eliminalo
FirstExam = ExamNames[0]

#Ciclo che esegue il Reverse MapROI su tutti gli esami per le ROI
presenti in "roi_names"
for i in range(2,ExamNum+1):

    RefExam = ExamNames2[i-1] #Inserisce in "RefExam" il
nome del Reference Exam
    Struct_group = struct_reg[i-2] #Inserisce in "Struct_group" il
Nome di Structure Registration Group

patient.MapRoiGeometriesDeformably(StructureRegistrationGroup
Names = ["%s" %(Struct_group)], ReferenceExaminationNames =
["%s" %(RefExam)],
TargetExaminationNames = ["%s" %(FirstExam)],
ReverseMapping = True, AbortWhenBadDisplacementField = False,
RoiGeometryNames = roi_names, CreateNewRois = True)

#Salva paziente
patient.Save()
```

6. GetDoseStatistics_db.py

```
#Estrapola dosi e (volumi) calcolati della tabella Dose Statistics,
aggiungendo le colonne presenti nel nostro database per la statistica
e li scrive nel file DoseStatistics.txt#
#Dosi relative al secondo workspace layout#

from connect import *

patient = get_current("Patient")

#Inserisce in "nome_paziente" il nome del paziente
nome_paziente = patient.PatientName

#Crea file con nome del paziente
f = open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_paziente),'w')
f.close()

#Accede all'oggetto FractionEvaluations
frac_evaluation = patient.TreatmentDelivery.FractionEvaluations[0]

#Accede all'oggetto FractionDose
frac_dose = patient.TreatmentPlans['Empty plan'].BeamSets['Empty
plan'].FractionDose

#Crea variabile roi_names di tipo lista con i nomi delle ROI del
paziente e rimuove la ROI "Body" dalla lista
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]
roi_names.remove('Body')

#Inserisce nella variabile "lunghezza" il numero di elementi della
lista "roi_names"
lunghezza = len(roi_names)

#Crea variabile ExamNames di tipo lista con i nomi degli esami
importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]

#Crea variabile ExamNum con numero elementi della lista
ExamNames (numero totale esami)
ExamNum = len(ExamNames)

#Inserisce in "totalroi" il numero totale di ROI per esame
totalroi = lunghezza/(ExamNum)

#Imposta una variabile "offset" per selezionare le ROI giuste ad ogni
seduta
offset = 0

#Crea una lista in cui salvare i valori di volume della riga 4 della
seconda seduta
lista = []

for r in range(0,ExamNum-1):
#Ciclo su ogni esame
    offset = offset+1
#Aggiorna offset
    for i in range(0,totalroi):
#Ciclo su tutte le ROI dell'esame
        if r==0:
            #Crea una lista con i valori di dose della prima riga al
99%,98%,95%,50%,2%,1% del volume
            prima_riga = frac_dose.GetDoseAtRelativeVolumes(RoiName = roi_names[i],
RelativeVolumes = [0.99,0.98,0.95,0.50,0.02,0.01])
            n_ele = len(prima_riga)
#Ritorna il numero di elementi di prima_riga
            f = prima_riga
            open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_paziente),'r+') #Apre il file creato in
scrittura
            f.seek(0, 2)
#Si posiziona all'ultima lettera
            f.write("\n")
#Scrivo uno spazio
            f.write(roi_names[i])
#Scrivo il nome della ROI
            f.write(";")
            seduta = str(r+1)
            f.write(seduta)
#Scrivo il nome della seduta
            f.write(";")
            f.write("1 - ROI on kVCT")
#Scrivo il nome della deformation
```



```

f.write(";")
f.write("1;")
#Scrive bool
roi_vol_kvct =
patient.PatientModel.StructureSets['%s' % (ExamNames[0])].RoiGeom
metries['%s' % (roi_names[i])].GetRoiVolume()
V_rifKVCT = (roi_vol_kvct/roi_vol_kvct)*100
V_rifKVCT = str(V_rifKVCT)
roi_vol_kvct = str(roi_vol_kvct)
f.write(roi_vol_kvct)
f.write(";")
f.write(V_rifKVCT)
f.write(";;")
f.close()
#Chiude il file
for t in range(0,n_ele):
#Ciclo per scrivere valori di dose nel file di testo
Dose_value_1 = prima_riga[t]
#Mette in Dose_value il valore di dose
Dose_value_1 = Dose_value_1/100
#Converte da cGy a Gy
Dose_value_1 = str(Dose_value_1)
#Converte in stringa Dose_value
f =
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" % (n
ome_patient),r+) #Scrive nel file
f.seek(0, 2)
f.write(Dose_value_1)
f.write(";")
if t == 2 or t == 3: print("")
else:
Dose_value_percento =
(float(Dose_value_1)/float(Dose_value_1))*100
Dose_value_percento = str(Dose_value_percento)
f.write(Dose_value_percento)
f.write(";")
f.close()
if t == 2:
#Dopo il valore di dose al 98% del volume scrive la dose media
D_media_1 = frac_dose.GetDoseStatistic(RoiName =
roi_names[i], DoseType = "Average")
D_media_1 = D_media_1/100 #Converte da cGy a Gy
D_media_1 = str(D_media_1)
f =
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" % (n
ome_patient),r+)
f.seek(0, 2)
f.write(D_media_1)
f.write(";")
Dose_media_percento =
(float(D_media_1)/float(D_media_1))*100
Dose_media_percento = str(Dose_media_percento)
f.write(Dose_media_percento)
f.write(";")
f.close()
continue
else: continue
f.close()
#Crea una lista con i valori di dose della seconda riga al
99%,98%,95%,50%,2%,1% del volume
seconda_riga =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDose
AtRelativeVolumes(RoiName = roi_names[i], RelativeVolumes =
[0.99,0.98,0.95,0.50,0.02,0.01])
n_ele = len(seconda_riga)
f =
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" % (n
ome_patient),r+)
f.seek(0, 2)
f.write("\n")
f.write(roi_names[i])
f.write(";")
seduta = str(r+2)
f.write(seduta)
f.write(";")
f.write("2 - ROI on MVCT")
f.write(";")
f.write("2;")
roi_vol =
patient.PatientModel.StructureSets['%s' % (ExamNames[r+1])].RoiG
eometries['%s' % (roi_names[i])].GetRoiVolume()
V_rifKVCT = (roi_vol/float(roi_vol_kvct))*100
V_rifKVCT = str(V_rifKVCT)
roi_vol = str(roi_vol)
f.write(roi_vol)
f.write(";")
f.write(V_rifKVCT)
f.write(";;")
f.close()
for t in range(0,n_ele):
Dose_value = terza_riga[t]
Dose_value = Dose_value/100
Dose_value = str(Dose_value)
f =
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" % (n
ome_patient),r+)
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
if t == 2 or t == 3: print("")
else:
Dose_value_percento = (terza_riga[t]/prima_riga[t])*100
Dose_value_percento = str(Dose_value_percento)
f.write(V_rifKVCT)
f.write(";;")
f.close()
for t in range(0,n_ele):
Dose_value = seconda_riga[t]
Dose_value = Dose_value/100
Dose_value = str(Dose_value)
f =
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" % (n
ome_patient),r+)
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
if t == 2 or t == 3: print("")
else:
Dose_value_percento =
(seconda_riga[t]/prima_riga[t])*100
Dose_value_percento = str(Dose_value_percento)
f.write(Dose_value_percento)
f.write(";")
f.close()
if t == 2:
D_media =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDose
eStatistic(RoiName = roi_names[i], DoseType = "Average")
D_media = D_media/100
D_media = str(D_media)
f =
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" % (n
ome_patient),r+)
f.seek(0, 2)
f.write(D_media)
f.write(";")
Dose_media_percento =
(float(D_media)/float(D_media_1))*100
Dose_media_percento = str(Dose_media_percento)
f.write(Dose_media_percento)
f.write(";")
f.close()
continue
else: continue
f.close()
#Crea una lista con i valori di dose della terza riga al
99%,98%,95%,50%,2%,1% del volume
terza_riga = frac_dose.GetDoseAtRelativeVolumes(RoiName =
roi_names[i+(totalroi*offset)], RelativeVolumes =
[0.99,0.98,0.95,0.50,0.02,0.01])
n_ele = len(terza_riga)
f =
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" % (n
ome_patient),r+)
f.seek(0, 2)
f.write("\n")
f.write(roi_names[i])
f.write(";")
seduta = str(r+2)
f.write(seduta)
f.write("3 - Def ROI on KVCT")
f.write(";")
f.write("3;")
roi_vol =
patient.PatientModel.StructureSets['%s' % (ExamNames[0])].RoiGeom
etries['%s' % (roi_names[i+(totalroi*offset)])].GetRoiVolume()
V_rifKVCT = (roi_vol/float(roi_vol_kvct))*100
roi_vol = str(roi_vol)
V_rifKVCT = str(V_rifKVCT)
f.write(roi_vol)
f.write(";")
f.write(V_rifKVCT)
f.write(";;")
f.close()
for t in range(0,n_ele):
Dose_value = terza_riga[t]
Dose_value = Dose_value/100
Dose_value = str(Dose_value)
f =
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" % (n
ome_patient),r+)
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
if t == 2 or t == 3: print("")
else:
Dose_value_percento = (terza_riga[t]/prima_riga[t])*100
Dose_value_percento = str(Dose_value_percento)

```

```

        f.write(Dose_value_percento)
        f.write(";")
    f.close()
    if t == 2:
        D_media = frac_dose.GetDoseStatistic(RoiName =
roi_names[i+(totalroi*offset)], DoseType = "Average")
        D_media = D_media/100
        D_media = str(D_media)
        f
    open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),r+)
        f.seek(0, 2)
        f.write(D_media)
        f.write(";")
        Dose_media_percento
    =
(float(D_media)/float(D_media_1))*100
        Dose_media_percento = str(Dose_media_percento)
        f.write(Dose_media_percento)
        f.write(";")
        f.close()
        continue
    else: continue
        f.close()
        Dose_media_percento
    =
(float(D_media)/float(D_media_1))*100
        Dose_media_percento = str(Dose_media_percento)
        f.write(Dose_media_percento)
        f.write(";")
        f.close()
        continue
    else: continue
        f.close()
    #Crea una lista con i valori di dose della quarta riga al
99%,98%,95%,50%,2%,1% del volume
    quarta_riga
    =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDose
eAtRelativeVolumes(RoiName = roi_names[i+(totalroi*offset)],
RelativeVolumes = [0.99,0.98,0.95,0.50,0.02,0.01])
    n_ele = len(quarta_riga)
    f
    open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),r+)
        f.seek(0, 2)
        f.write("\n")
        f.write(roi_names[i])
        f.write(";")
        seduta = str(r+2)
        f.write(seduta)
        f.write(";")
        f.write("4 - Def ROI on MVCT")
        f.write(";")
        f.write("4;")
        roi_vol_mvct
    =
patient.PatientModel.StructureSets['%s' %(ExamNames[r+1])].RoiG
eometries['%s' %(roi_names[i+(totalroi*offset)])].GetRoiVolume()
        lista[len(lista):len(lista)] = [roi_vol_mvct]
        V_rifKVCT = (roi_vol_mvct/float(roi_vol_kvct))*100
        V_rifKVCT = str(V_rifKVCT)
        roi_vol_mvct = str(roi_vol_mvct)
        f.write(roi_vol_mvct)
        f.write(";")
        f.write(V_rifKVCT)
        f.write(";")
        V_rifMVCT = (float(roi_vol_mvct)/float(roi_vol_mvct))*100
        V_rifMVCT = str(V_rifMVCT)
        f.write(V_rifMVCT)
        f.write(";")
        f.close()
    for t in range(0,n_ele):
        Dose_value = quarta_riga[t]
        Dose_value = Dose_value/100
        Dose_value = str(Dose_value)
        f
    open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),r+)
        f.seek(0, 2)
        f.write(Dose_value)
        f.write(";")
        if t == 2 or t == 3: print ("")
        else:
            Dose_value_percento
    =
(seconda_riga[t]/prima_riga[t])*100
            Dose_value_percento = str(Dose_value_percento)
            f.write(Dose_value_percento)
            f.write(";")
            f.close()
            if t == 2:
                D_media
    =
                D_media
    =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDose
eStatistic(RoiName = roi_names[i], DoseType = "Average")
                D_media = D_media/100 #Converte da cGy a Gy
                D_media = str(D_media)
                f
    =
    open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),r+)
        f.seek(0, 2)
        f.write(D_media)
        f.write(";")

```

```

Dose_media_percento =
((float(D_media)*100)/D_media_1)*100
Dose_media_percento = str(Dose_media_percento)
f.write(Dose_media_percento)
f.write(";")
f.close()
continue
else: continue
f.close()

#Crea una lista con i valori di dose della terza riga al
99%,98%,95%,50%,2%,1% del volume
terza_riga = frac_dose.GetDoseAtRelativeVolumes(RoiName
= roi_names[i+(totalroi*offset)], RelativeVolumes
= [0.99,0.98,0.95,0.50,0.02,0.01])
n_ele = len(terza_riga)
f
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),'+')
f.seek(0, 2)
f.write("\n")
f.write(roi_names[i])
f.write(";")
seduta = str(r+2)
f.write(seduta)
f.write(";")
f.write("3 - Def ROI on kVCT")
f.write(";")
f.write("3;")
roi_vol
patient.PatientModel.StructureSets['%s' %(ExamNames[0])].RoiGeo
metries['%s' %(roi_names[i+(totalroi*offset)]]).GetRoiVolume()
V_rifKVCT = (roi_vol/roi_vol_kvct)*100
roi_vol = str(roi_vol)
V_rifKVCT = str(V_rifKVCT)
f.write(roi_vol)
f.write(";")
f.write(V_rifKVCT)
f.write(";;")
f.close()
for t in range(0,n_ele):
Dose_value = terza_riga[t]
Dose_value = Dose_value/100
Dose_value = str(Dose_value)
f
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),'+')
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
if t == 2 or t == 3: print ("")
else:
Dose_value_percento = (terza_riga[t]/prima_riga[t])*100
Dose_value_percento = str(Dose_value_percento)
f.write(Dose_value_percento)
f.write(";")
f.close()
if t == 2:
D_media = frac_dose.GetDoseStatistic(RoiName =
roi_names[i+(totalroi*offset)], DoseType = "Average")
D_media = D_media/100
D_media = str(D_media)
f
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),'+')
f.seek(0, 2)
f.write(D_media)
f.write(";")
Dose_media_percento
((float(D_media)*100)/D_media_1)*100
Dose_media_percento = str(Dose_media_percento)
f.write(Dose_media_percento)
f.write(";")
f.close()
continue
else: continue
f.close()

#Crea una lista con i valori di dose della quarta riga al
99%,98%,95%,50%,2%,1% del volume
quarta_riga
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDose
AtRelativeVolumes(RoiName = roi_names[i+(totalroi*offset)],
RelativeVolumes = [0.99,0.98,0.95,0.50,0.02,0.01])
n_ele = len(quarta_riga)
f
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),'+')
f.seek(0, 2)
f.write("n")
f.write(roi_names[i])
f.write(";")
seduta = str(r+2)
f.write(seduta)
f.write(";")
f.write("4 - Def ROI on MVCT")
f.write(";")
f.write("4;")
roi_vol
patient.PatientModel.StructureSets['%s' %(ExamNames[r+1])].RoiG
eometries['%s' %(roi_names[i+(totalroi*offset)]]).GetRoiVolume()
V_rifKVCT = (roi_vol/roi_vol_kvct)*100
V_rifKVCT = str(V_rifKVCT)
roi_vol = str(roi_vol)
f.write(roi_vol)
f.write(";")
f.write(V_rifKVCT)
f.write(";;")
V_rifMVCT = (float(roi_vol)/lista[i])*100
V_rifMVCT = str(V_rifMVCT)
f.write(V_rifMVCT)
f.write(";;")
f.close()
for t in range(0,n_ele):
Dose_value = quarta_riga[t]
Dose_value = Dose_value/100
Dose_value = str(Dose_value)
f
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),'+')
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
if t == 2 or t == 3: print ("")
else:
Dose_value_percento
(quarta_riga[t]/prima_riga[t])*100
Dose_value_percento = str(Dose_value_percento)
f.write(Dose_value_percento)
f.write(";")
f.close()
if t == 2:
D_media = frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDose
Statistic(RoiName = roi_names[i+(totalroi*offset)], DoseType =
"Average")
D_media = D_media/100
D_media = str(D_media)
f
open("c:\Users\utente\Desktop\Statistics\s_DoseStatistics.txt" %(n
ome_patient),'+')
f.seek(0, 2)
f.write(D_media)
f.write(";")
Dose_media_percento
((float(D_media)*100)/D_media_1)*100
Dose_media_percento = str(Dose_media_percento)
f.write(Dose_media_percento)
f.write(";")
f.close()
continue
else: continue
f.close()

```

6.1. GetDoseStatistics_solo_dosi.py

```
#Estrapola le dosi calcolate della tabella Dose Statistics e le scrive
nel file DoseStatistics.txt#
#Dosi relative al secondo workspace layout, prima di avviarlo
selezionare il secondo workspace,#
#caricare EmptyPlan e cliccare in Update ROI Voxel Volumes#

from connect import *

patient = get_current("Patient")

#Inserisce in "nome_patient" il nome del paziente
nome_patient = patient.PatientName

#Crea file con nome del paziente nel percorso indicato
f = open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patient), 'w')
f.close()

#Accede all'oggetto FractionEvaluations
frac_evaluation = patient.TreatmentDelivery.FractionEvaluations[0]

#Accede all'oggetto FractionDose
frac_dose = patient.TreatmentPlans['Empty plan'].BeamSets['Empty plan'].FractionDose

#Crea variabile "roi_names" di tipo lista con i nomi delle ROI del
paziente e rimuove la ROI "Body" dalla lista
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]
roi_names.remove('Body')

#Inserisce nella variabile "lunghezza" il numero di elementi della
lista "roi_names"
lunghezza = len(roi_names)

#Crea variabile ExamNames di tipo lista con i nomi degli esami
importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]

#Crea variabile ExamNum con numero elementi della lista
ExamNames (numero totale esami)
ExamNum = len(ExamNames)

#Inserisce in "totalroi" il numero totale di ROI per esame
totalroi = lunghezza/(ExamNum)

#Imposta una variabile "offset" per selezionare le ROI giuste ad ogni
seduta
offset = 0

for r in range(0,ExamNum-1):
#Ciclo su ogni esame
    offset = offset+1
#Aggiorna offset
    for i in range(0,totalroi):
#Ciclo su tutte le ROI dell'esame
        #Crea una lista con i valori di dose della prima riga al
        99%,98%,95%,50%,2%,1% del volume
        prima_riga = frac_dose.GetDoseAtRelativeVolumes(RoiName
= roi_names[i], RelativeVolumes = [0.99,0.98,0.95,0.50,0.02,0.01])
        n_ele = len(prima_riga) #Ritorna il numero di elementi di
        prima_riga
        f = open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patient), 'r+')
        #Scrivi nel file
        f.seek(0, 2)
#Si posiziona all'ultima lettera
        f.write("\n")
#Manda a capo
        f.write(roi_names[i])
#Scrivi il nome della ROI
        f.write(";")
        f.close()
#Chiude il file
        for t in range(0,n_ele):
#Ciclo per scrivere valori di dose nel file di testo
            Dose_value = prima_riga[t]
#Mette in "Dose_value" il valore di dose
            Dose_value = Dose_value/100
#Converte da cGy a Gy
            Dose_value = str(Dose_value)
#Converte in stringa "Dose_value"

        f = open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patient), 'r+')
        #Scrivi nel file
        f.seek(0, 2)
        f.write(Dose_value)
        f.write(";")
        f.close()
        continue
        else: continue
        f.close()

        #Crea una lista con i valori di dose della seconda riga al
        99%,98%,95%,50%,2%,1% del volume
        seconda_riga = frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDose
        eAtRelativeVolumes(RoiName = roi_names[i], RelativeVolumes =
        [0.99,0.98,0.95,0.50,0.02,0.01])
        n_ele = len(seconda_riga)
        f = open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patient), 'r+')
        f.seek(0, 2)
        f.write("\n")
        f.write(roi_names[i])
        f.write(";")
        f.close()
        for t in range(0,n_ele):
            Dose_value = seconda_riga[t]
            Dose_value = Dose_value/100
            Dose_value = str(Dose_value)
            f = open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patient), 'r+')
            f.seek(0, 2)
            f.write(Dose_value)
            f.write(";")
            f.close()
            continue
            else: continue
            f.close()

            #Crea una lista con i valori di dose della terza riga al
            99%,98%,95%,50%,2%,1% del volume
            terza_riga = frac_dose.GetDoseAtRelativeVolumes(RoiName =
            roi_names[i+(totalroi*offset)], RelativeVolumes =
            [0.99,0.98,0.95,0.50,0.02,0.01])
            n_ele = len(terza_riga)
            f = open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patient), 'r+')
            f.seek(0, 2)
            f.write("\n")
            f.write(roi_names[i+(totalroi*offset)])
            f.write(";")
            f.close()
            continue
            else: continue
            f.close()

            for t in range(0,n_ele):
                Dose_value = terza_riga[t]
                Dose_value = Dose_value/100
```

```

Dose_value = str(Dose_value)
f
open("c:\Users\utente\Desktop%s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
f.close()
if t == 2:
    D_media = frac_dose.GetDoseStatistic(RoiName =
roi_names[i+(totalroi*offset)], DoseType = "Average")
    D_media = D_media/100
    D_media = str(D_media)
f
open("c:\Users\utente\Desktop%s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write(D_media)
f.write(";")
f.close()
continue
else: continue
f.close()

#Crea una lista con i valori di dose della quarta riga al
99%,98%,95%,50%,2%,1% del volume
quarta_riga =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDose
AtRelativeVolumes(RoiName = roi_names[i+(totalroi*offset)],
RelativeVolumes = [0.99,0.98,0.95,0.50,0.02,0.01])
n_ele = len(quarta_riga)
f
open("c:\Users\utente\Desktop%s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write("\n")
f.write(roi_names[i+(totalroi*offset)])
f.write(";")
f.close()
for t in range(0,n_ele):
    Dose_value = quarta_riga[t]
    Dose_value = Dose_value/100
    Dose_value = str(Dose_value)
f
open("c:\Users\utente\Desktop%s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
f.close()
if t == 2:
    D_media =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDose
Statistic(RoiName = roi_names[i+(totalroi*offset)], DoseType =
"Average")
    D_media = D_media/100
    D_media = str(D_media)
f
open("c:\Users\utente\Desktop%s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write(D_media)
f.write(";")
f.close()
continue
else: continue
f.close()

```

6.2. GetDoseStatistics_dosi_volumi.py

```

#Estrapola le dosi e (volumi) calcolati nella tabella Dose Statistics e
li scrive nel file DoseStatistics.txt#
#Dosi e volumi relativi al secondo workspace layout, prima di
avviarlo selezionare il secondo workspace,#
#caricare EmptyPlan e cliccare in Update ROI Voxel Volumes#

from connect import *

patient = get_current("Patient")

#Inserisce in "nome_patient" il nome del paziente
nome_patient = patient.PatientName

#Crea file con nome del paziente
f
open("c:\Users\utente\Desktop%s_DoseStatistics.txt" %(nome_patie
nt),w)
f.close()

#Accede all'oggetto FractionEvaluations
frac_evaluation = patient.TreatmentDelivery.FractionEvaluations[0]

#Accede all'oggetto FractionDose
frac_dose = patient.TreatmentPlans['Empty plan'].BeamSets['Empty
plan'].FractionDose

#Crea variabile roi_names di tipo lista con i nomi delle ROI del
paziente e rimuove la ROI "Body" dalla lista
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]
roi_names.remove('Body')

#Inserisce nella variabile "lunghezza" il numero di elementi della
lista "roi_names"
lunghezza = len(roi_names)

#Crea variabile ExamNames di tipo lista con i nomi degli esami
importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]

#Crea variabile ExamNum con numero elementi della lista
ExamNames (numero totale esami)
ExamNum = len(ExamNames)

#Inserisce in "totalroi" il numero totale di ROI per esame
totalroi = lunghezza/(ExamNum)

#Imposta una variabile "offset" per selezionare le ROI giuste ad ogni
seduta
offset = 0

for r in range(0,ExamNum-1):
#Ciclo su ogni esame
    offset = offset+1
#Aggiorna offset
    for i in range(0,totalroi):
#Ciclo su tutte le ROI dell'esame
        #Crea una lista con i valori di dose della prima riga al
99%,98%,95%,50%,2%,1% del volume
        prima_riga = frac_dose.GetDoseAtRelativeVolumes(RoiName
= roi_names[i], RelativeVolumes = [0.99,0.98,0.95,0.50,0.02,0.01])
        n_ele = len(prima_riga)
#Ritorna il numero di elementi di prima_riga
        f
open("c:\Users\utente\Desktop%s_DoseStatistics.txt" %(nome_patie
nt),r+')
#Scrivo nel file
        f.seek(0, 2)
#Si posiziona all'ultima lettera
        f.write("\n")
#Manda a capo
        f.write(roi_names[i])
#Scrivo il nome della ROI
        f.write(";")
        roi_vol =
patient.PatientModel.StructureSets["%s" %(ExamNames[0])].RoiGeo
metries["%s" %(roi_names[i])].GetRoiVolume()
#Mette in
"roi_vol" il volume della roi
        roi_vol = str(roi_vol)
#Converte in stringa
        f.write(roi_vol)
#Scrivo il valore nel file
        f.write(";")
        f.close()
#Chiude il file

```

```

for t in range(0,n_ele):
#Ciclo per scrivere valori di dose nel file di testo
Dose_value = prima_riga[t]
#Mette in Dose_value il valore di dose
Dose_value = Dose_value/100
#Converte da cGy a Gy
Dose_value = str(Dose_value)
#Converte in stringa "Dose_value"
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+') #Scrive nel file
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
f.close()
if t == 2:
#Dopo il valore di dose al 98% del volume scrive la dose media
D_media = frac_dose.GetDoseStatistic(RoiName =
roi_names[i], DoseType = "Average")
D_media = D_media/100
#Converte da cGy a Gy
D_media = str(D_media)
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+') #Scrive nel file
f.seek(0, 2)
f.write(D_media)
f.write(";")
f.close()
continue
else: continue
f.close()

#Crea una lista con i valori di dose della seconda riga al
99%,98%,95%,50%,2%,1% del volume
seconda_riga =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDoseAtRelativeVolumes(RoiName = roi_names[i], RelativeVolumes = [0.99,0.98,0.95,0.50,0.02,0.01])
n_ele = len(seconda_riga)
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write("\n")
f.write(roi_names[i])
f.write(";")
roi_vol =
patient.PatientModel.StructureSets['%s' %(ExamNames[r+1])].RoiGeometries['%s' %(roi_names[i])].GetRoiVolume()
roi_vol = str(roi_vol)
f.write(roi_vol)
f.write(";")
f.close()
for t in range(0,n_ele):
Dose_value = seconda_riga[t]
Dose_value = Dose_value/100
Dose_value = str(Dose_value)
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
f.close()
if t == 2:
D_media =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDoseStatistic(RoiName = roi_names[i], DoseType = "Average")
D_media = D_media/100
D_media = str(D_media)
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write(D_media)
f.write(";")
f.close()
continue
else: continue
f.close()

#Crea una lista con i valori di dose della terza riga al
99%,98%,95%,50%,2%,1% del volume
terza_riga = frac_dose.GetDoseAtRelativeVolumes(RoiName =
roi_names[i+(totalroi*offset)], RelativeVolumes =
[0.99,0.98,0.95,0.50,0.02,0.01])
n_ele = len(terza_riga)
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write("\n")
f.write(roi_names[i+(totalroi*offset)])
f.write(";")
roi_vol =
patient.PatientModel.StructureSets['%s' %(ExamNames[0])].RoiGeometries['%s' %(roi_names[i+(totalroi*offset)])].GetRoiVolume()
roi_vol = str(roi_vol)
f.write(roi_vol)
f.write(";")
f.close()
for t in range(0,n_ele):
Dose_value = terza_riga[t]
Dose_value = Dose_value/100
Dose_value = str(Dose_value)
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
f.close()
if t == 2:
D_media = frac_dose.GetDoseStatistic(RoiName =
roi_names[i+(totalroi*offset)], DoseType = "Average")
D_media = D_media/100
D_media = str(D_media)
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write(D_media)
f.write(";")
f.close()
continue
else: continue
f.close()

#Crea una lista con i valori di dose della quarta riga al
99%,98%,95%,50%,2%,1% del volume
quarta_riga =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDoseAtRelativeVolumes(RoiName = roi_names[i+(totalroi*offset)], RelativeVolumes = [0.99,0.98,0.95,0.50,0.02,0.01])
n_ele = len(quarta_riga)
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write("\n")
f.write(roi_names[i+(totalroi*offset)])
f.write(";")
roi_vol =
patient.PatientModel.StructureSets['%s' %(ExamNames[r+1])].RoiGeometries['%s' %(roi_names[i+(totalroi*offset)])].GetRoiVolume()
roi_vol = str(roi_vol)
f.write(roi_vol)
f.write(";")
f.close()
for t in range(0,n_ele):
Dose_value = quarta_riga[t]
Dose_value = Dose_value/100
Dose_value = str(Dose_value)
f =
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+')
f.seek(0, 2)
f.write(Dose_value)
f.write(";")
f.close()
if t == 2:
D_media =
frac_evaluation.DoseOnExaminations[r].DoseEvaluations[0].GetDoseStatistic(RoiName = roi_names[i+(totalroi*offset)], DoseType =
"Average")
D_media = D_media/100
D_media = str(D_media)

```

```

f
open("c:\Users\utente\Desktop\s_DoseStatistics.txt" %(nome_patie
nt),r+')
    f.seek(0, 2)
    f.write(D_media)
    f.write(";")
    f.close()
    continue
else: continue
f.close()

```

7. ImportScriptInFolder.py

```

#Importa in successione gli script#

from connect import *

import connect

import os, sys

import datetime

try:
    os.chdir(r"Y:\Documents Disco Y\Vecchi\scripting") #Imposta il
    percorso indicato come predefinito per la lettura
    tstart = datetime.datetime.now() #Avvia timer
    #Inserire nomi file da importare dopo l'istruzione import separati
    da ,
    import RigidRegistration, RotationValue

    tend = datetime.datetime.now() #Ferma timer
    timediff = tend - tstart #Tempo di esecuzione
    timediff = str(timediff)[-4] #Tolgo le ultime 4 cifre
    convertendo in stringa

    #Inserisce in "nome_patient" il nome del paziente
    patient = get_current("Patient")
    nome_patient = patient.PatientName
    nome_patient2 = nome_patient.replace('^',' ')

    #Crea file con nome del paziente
    f
    open("c:\Users\utente\Desktop\Tempi\s_Tempi.txt" %(nome_patie
nt),w')
    f.close()
    f
    open("c:\Users\utente\Desktop\Tempi\s_Tempi.txt" %(nome_patie
nt),r+') #Apre il file creato in scrittura
    f.seek(0, 2)
    #Si posiziona all'ultima lettera
    f.write(nome_patient2)
    f.write(";")
    #Scrivo il nome della ROI
    f.write("Import CT - Rigid registration;")
    f.write(timediff)
    f.write(";")
    f.close()

except:
    print "File mancante o errore script"

    tend = datetime.datetime.now() #Ferma timer
    timediff = tend - tstart #Tempo di esecuzione
    timediff = str(timediff)[-4] #Tolgo le ultime 4 cifre
    convertendo in stringa

    #Inserisce in "nome_patient" il nome del paziente
    patient = get_current("Patient")
    nome_patient = patient.PatientName
    nome_patient2 = nome_patient.replace('^',' ')

    #Crea file con nome del paziente
    f
    open("c:\Users\utente\Desktop\Tempi\s_Tempi.txt" %(nome_patie
nt),w')
    f.close()
    f
    open("c:\Users\utente\Desktop\Tempi\s_Tempi.txt" %(nome_patie
nt),r+') #Apre il file creato in scrittura
    f.seek(0, 2)
    #Si posiziona all'ultima lettera
    f.write(nome_patient2)
    f.write(";")
    #Scrivo il nome della ROI
    f.write("Import CT - Rigid registration;")
    f.write(timediff)
    f.write(";")
    f.close()

    sys.exit()

```

8. RotationValue.py

```
#Calcola l'angolo di rotazione tra le immagini dalla matrice di
trasformazione#

from connect import *

from math import *

patient = get_current("Patient")

#Inserisce in "nome_patient" il nome del paziente
nome_patient = patient.PatientName

#Crea file con nome del paziente
f =
open("c:\Users\utente\Desktop\Check\Rigid_registration\s_Rotatio
nAngle.txt" %(nome_patient), 'w')
f.close()

#Crea variabile "ExamNames" con la lista dei nomi degli esami
importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]

#Salva il primo elemento della lista ExamNames (CT 1)
FirstExam = ExamNames[0]

for i in range(1, len(ExamNames)):
    T = patient.GetTransformForExaminations(FromExamination =
"%s" %(ExamNames[i]), ToExamination = FirstExam)
    trace = T[0] + T[5] + T[10]
    angle_rad = acos((trace-1) / 2)
    angle_deg = angle_rad * 180 / pi
    f =
open("c:\Users\utente\Desktop\Check\Rigid_registration\s_Rotatio
nAngle.txt" %(nome_patient), 'r+')
    f.seek(0, 2)
    f.write("\n")
    f.write("From:{0}-
To:{1};{2};".format(FirstExam, "%s" %(ExamNames[i]), angle_deg))
    if angle_deg > 5:
        f.write('Check;')
    else:
        f.write(';')
    f.close()
```

9. ROI_position.py

```
#Controlla che la ROI "Body" sia all'interno dei bordi DICOM#

from connect import *

patient = get_current("Patient")

#Inserisce in "nome_patient" il nome del paziente
nome_patient = patient.PatientName

#Crea file con nome del paziente
f =
open("c:\Users\utente\Desktop\Check\Rigid_registration\s_ROI_P
osition.txt" %(nome_patient), 'w')
f.close()

#Crea variabile "roi_names" con la lista dei nomi delle ROI del
paziente
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]
roi_names.remove('Body')

#Crea variabile ExamNames di tipo lista con i nomi degli esami
importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]

for i in range(0, len(ExamNames)):
    dicom_border =
patient.Examinations['%s' %(ExamNames[i])].Series[0].ImageStack.
Corner
    center_of_roi_body =
patient.PatientModel.StructureSets['%s' %(ExamNames[i])].RoiGeo
metries['Body'].GetCenterOfRoi()
    f =
open("c:\Users\utente\Desktop\Check\Rigid_registration\s_ROI_P
osition.txt" %(nome_patient), 'r+')
    f.seek(0, 2)
    f.write("\n")
    f.write('{num_esame};'.format(num_esame = i+1))
    if (abs(center_of_roi_body.x) > ((abs(dicom_border.x)/3)) or \
(abs(center_of_roi_body.y) > ((abs(dicom_border.y)/3)) or \
(abs(center_of_roi_body.z) > ((abs(dicom_border.z)*2))):
        f.write("Check ROI Position")
    else:
        f.write(";")
    f.close()
```


10. GeometryCheck.py

```

#Crea un POI in ogni seduta nel punto (0,0,0) e calcola, scrivendo in
un file di testo i seguenti valori:#
#Coordinate x,y,z del centro della ROI indicate es. Parotide Dx e
Parotide Sx#
#Modulo dei vettori che congiungono il centro delle ROI con il
centro di ogni POI#
#Angolo tra i vettori
#Area del triangolo creato con i vettori
#Versore dei vettori
#Distanza tra le ROI es. Parotide Dx e Parotide Sx

from connect import *
from math import *
patient = get_current("Patient")

#Inserisce in "nome_paziente" il nome del paziente
nome_paziente = patient.PatientName
nome_paziente2 = nome_paziente.replace('^','')

#Crea file con nome del paziente
f = open("c:\Users\utente\Desktop\Check\Geometry\%s_Geometry.txt"
        =
        %(nome_paziente),w)
f.close()

#Crea variabile "roi_names" con la lista dei nomi delle ROI del
paziente
roi_names = [r.Name for r in patient.PatientModel.RegionsOfInterest]

for roi in range(0,len(roi_names)):
    if roi_names[roi]=="parotidedx":
        parotide_dx_num = roi
    elif roi_names[roi]=="parotidesx":
        parotide_sx_num = roi
    else: continue

roi_names2 = [r.Name for r in
patient.PatientModel.RegionsOfInterest]
roi_names2.remove('Body')

#Crea variabile ExamNames di tipo lista con i nomi degli esami
importati (es. CT 1, CT2..)
ExamNames = [r.Name for r in patient.Examinations]
ExamNames2 = [r.Name for r in patient.Examinations]

#Salva il primo elemento della lista ExamNames (CT 1) ed eliminalo
FirstExam = ExamNames[0]

#Crea variabile "struct_reg" con la lista dei nomi delle registrazioni
fatte
struct_reg = [r.Name for r in
patient.PatientModel.StructureRegistrationGroups]

#Inserisce in "totalroi" il numero totale di ROI per esame
totalroi = len(roi_names2)/len(ExamNames)

#Imposta una variabile "offset" per selezionare le ROI giuste ad ogni
seduta
offset = -1

patient.PatientModel.CreatePoi(Examination
=
patient.Examinations["%s" % (FirstExam)], Point = { 'x': 0.0, 'y': 0.0,
'z': 0.0}, Volume = 0, Name = "000", Color = "Red", Type =
"Undefined")

for i in range(2,len(ExamNames)+1):
    RefExam = ExamNames2[i-1] #Inserisce in "RefExam" il
nome del Reference Exam
    Struct_group = struct_reg[i-2] #Inserisce in "Struct_group" il
Nome di Structure Registration Group

patient.MapPoiGeometriesDeformably(StructureRegistrationGroupN
ames = ["%s" % (Struct_group)], ReferenceExaminationNames =
["%s" % (RefExam)],
    TargetExaminationNames = ["%s" % (FirstExam)],
    ReverseMapping = True, AbortWhenBadDisplacementField = True,
    PoiGeometryNames = ["000"], CreateNewPois = True)

#Salva paziente
patient.Save()

poi_names = [r.Name for r in patient.PatientModel.PointsOfInterest]

```

```

for i in range(0,len(ExamNames)):
    offset = offset + 1
    if i==0:
        center_of_roi
        =
        patient.PatientModel.StructureSets["%s" % (ExamNames[i])].RoiGeo
metries["%s" % (roi_names[parotide_dx_num])].GetCenterOfRoi()
        poi_point = patient.PatientModel.StructureSets["%s" %
(ExamNames2[i])].PoiGeometries[i].Point
        modulo_vettore1 = sqrt(((center_of_roi.x
-
poi_point.x)**2)+(center_of_roi.y
-
poi_point.y)**2)+(center_of_roi.z - poi_point.z)**2))
        center_of_roi1 = center_of_roi
        versore_x = (center_of_roi1.x/abs(modulo_vettore1))
        versore_y = (center_of_roi1.y/abs(modulo_vettore1))
        versore_z = (center_of_roi1.z/abs(modulo_vettore1))
        versore = [versore_x,versore_y,versore_z]
        f
        =
        open("c:\Users\utente\Desktop\Check\Geometry\%s_Geometry.txt"
        =
        %(nome_paziente),r+)
        f.seek(0, 2)
        f.write("\n")

f.write('{ nome_paziente};{ num_esame};{ roi_name};{0};{1};{2};{3
};;{4};{5};{6};'.format(center_of_roi1.x,
        center_of_roi1.y,
        center_of_roi1.z, modulo_vettore1, versore_x, versore_y, versore_z,
        roi_name = roi_names[parotide_dx_num], num_esame = i+1,
        nome_paziente = nome_paziente2))
        f.close()
        center_of_roi
        =
        patient.PatientModel.StructureSets["%s" % (ExamNames[i])].RoiGeo
metries["%s" % (roi_names[parotide_sx_num])].GetCenterOfRoi()
        modulo_vettore2 = sqrt(((center_of_roi.x
-
poi_point.x)**2)+(center_of_roi.y
-
poi_point.y)**2)+(center_of_roi.z - poi_point.z)**2))
        center_of_roi2 = center_of_roi
        versore_x = (center_of_roi2.x/abs(modulo_vettore2))
        versore_y = (center_of_roi2.y/abs(modulo_vettore2))
        versore_z = (center_of_roi2.z/abs(modulo_vettore2))
        versore = [versore_x,versore_y,versore_z]
        distanza
        =
        sqrt(((center_of_roi1.x
-
center_of_roi2.x)**2)+(center_of_roi1.y
-
center_of_roi2.y)**2)+(center_of_roi1.z - center_of_roi2.z)**2))
        alfa_rad
        =
        acos(((center_of_roi1.x*center_of_roi2.x)+(center_of_roi1.y*center
_of_roi2.y)+(center_of_roi1.z*center_of_roi2.z))/(modulo_vettore1*
modulo_vettore2))
        alfa_deg = (alfa_rad * 180)/pi
        area
        =
        (0.5*(abs(modulo_vettore1*modulo_vettore2))*sin(alfa_rad))
        f
        =
        open("c:\Users\utente\Desktop\Check\Geometry\%s_Geometry.txt"
        =
        %(nome_paziente),r+)
        f.seek(0, 2)
        f.write("\n")

f.write('{ nome_paziente};{ num_esame};{ roi_name};{0};{1};{2};{3
};{4};{5};{6};{7};{8};{9}'.format(center_of_roi2.x,
        center_of_roi2.y, center_of_roi2.z, modulo_vettore2, alfa_deg, area,
        versore_x, versore_y, versore_z, distanza, roi_name =
        roi_names[parotide_sx_num], num_esame = i+1, nome_paziente =
        nome_paziente2))
        f.close()
    else:
        if i>=1:
            center_of_roi
            =
            patient.PatientModel.StructureSets["%s" % (ExamNames[i])].RoiGeo
metries["%s" % (roi_names[parotide_dx_num+(totalroi*offset))].Get
CenterOfRoi()
            poi_point = patient.PatientModel.StructureSets["%s" %
(ExamNames2[i])].PoiGeometries[i].Point
            modulo_vettore1 = sqrt(((center_of_roi.x
-
poi_point.x)**2)+(center_of_roi.y
-
poi_point.y)**2)+(center_of_roi.z - poi_point.z)**2))
            center_of_roi1 = center_of_roi
            versore_x = (center_of_roi1.x/abs(modulo_vettore1))
            versore_y = (center_of_roi1.y/abs(modulo_vettore1))
            versore_z = (center_of_roi1.z/abs(modulo_vettore1))
            versore = [versore_x,versore_y,versore_z]
            f
            =
            open("c:\Users\utente\Desktop\Check\Geometry\%s_Geometry.txt"
            =
            %(nome_paziente),r+)
            f.seek(0, 2)
            f.write("\n")

f.write('{ nome_paziente};{ num_esame};{ roi_name};{0};{1};{2};{3

```

```

};;{4};{5};{6};;'.format(center_of_roi1.x, center_of_roi1.y,
center_of_roi1.z, modulo_vettore1, versore_x, versore_y, versore_z,
roi_name = roi_names[parotide_dx_num], num_esame = i+1,
nome_paziente = nome_patient2))
f.close()
center_of_roi =
patient.PatientModel.StructureSets['%s' % (ExamNames[i])].RoiGeometries['%s' % (roi_names[parotide_dx_num+(totalroi*offset))].GetCenterOfRoi()
modulo_vettore2 = sqrt(((center_of_roi.x - poi_point.x)**2)+(center_of_roi.y - poi_point.y)**2)+(center_of_roi.z - poi_point.z)**2))
center_of_roi2 = center_of_roi
versore_x = (center_of_roi2.x/abs(modulo_vettore2))
versore_y = (center_of_roi2.y/abs(modulo_vettore2))
versore_z = (center_of_roi2.z/abs(modulo_vettore2))
versore = [versore_x,versore_y,versore_z]
distanza = sqrt(((center_of_roi1.x - center_of_roi2.x)**2)+(center_of_roi1.y - center_of_roi2.y)**2)+(center_of_roi1.z - center_of_roi2.z)**2))
alfa_rad =
acos(((center_of_roi1.x*center_of_roi2.x)+(center_of_roi1.y*center_of_roi2.y)+(center_of_roi1.z*center_of_roi2.z))/(modulo_vettore1*modulo_vettore2))
alfa_deg = (alfa_rad * 180)/pi
area =
(0.5*(abs(modulo_vettore1*modulo_vettore2))*sin(alfa_rad))
f =
open("c:\Users\utente\Desktop\Check\Geometry\%s_Geometry.txt" % (nome_patient),r+')
f.seek(0, 2)
f.write("\n")

f.write('{ nome_paziente};{ num_esame};{ roi_name};{0};{1};{2};{3};{4};{5};{6};{7};{8};{9}'.format(center_of_roi2.x, center_of_roi2.y, center_of_roi2.z, modulo_vettore2, alfa_deg, area, versore_x, versore_y, versore_z, distanza, roi_name = roi_names[parotide_dx_num], num_esame = i+1, nome_paziente = nome_patient2))
f.close()
else: continue

```

11. Delete_all_doses.py

```

#Elimina tutte le dosi deformate#

from connect import *

patient = get_current("Patient")

dose_exam = [r for r in patient.TreatmentDelivery.FractionEvaluations[0].DoseOnExaminations]

dose_num = len(dose_exam)

for i in range(0,dose_num-1):
    delete_dose = patient.TreatmentDelivery.FractionEvaluations[0].DoseOnExaminations[i].DoseEvaluations[0].DeleteEvaluationDose()

patient.Save()

```

12. New_console.py

```
#Apre una nuova console RayStation#

from connect import *

from os import system

#Setta la directory di installazione RayStation come predefinita
os.chdir(r"C:\Program Files\RaySearch Laboratories\RayStation
3.99")

#Apre il file RayStation.exe
system('start RayStation.exe')

GetPatientName_Id.py

#Stampa i nomi e gli ID dei pazienti contenuti nella cartella
DICOM#

from connect import *

import connect

#Accede all'oggetto PatientDB (database pazienti) tramite il metodo
get_current
db = connect.get_current('PatientDB')

#Crea una variabile "patients" con la lista dei pazienti
patients = db.QueryPatientsFromPath(Path = r"Y:\DICOM", Filter =
{})

#Crea una lista vuota di nome "lista"
lista = []

#Crea una lista vuota di nome "lista2"
lista2 = []

#Ciclo che inserisce in "lista" i nomi dei pazienti e in "lista2" gli Id
dei pazienti
for i in range(0,len(patients)):
    select_parola = patients[i]          #Seleziona primo paziente
    patient_name = select_parola['NamÈ'] #Seleziona il nome
del paziente
    patient_id = select_parola['Id']     #Seleziona l'Id del
paziente
    lista[len(lista):len(lista)] = [patient_name] #Aggiunge a "lista" il
nome del paziente
    lista2[len(lista2):len(lista2)] = [patient_id] #Aggiunge a "lista2"
l'Id del paziente

print lista          #Stampa i nomi dei pazienti
print lista2        #Stampa gli Id dei pazienti
```

Appendice B

In questa appendice sono contenuti i grafici relativi alla probabilità $P(A,B)$, $P(A/B)$, $P(B/A)$ di ogni stato:

1. $A \geq 6$ e $95\% \leq B \leq 105\%$.
2. $A < 6$ e $95\% \leq B \leq 105\%$.
3. $A \geq 6$ e $95\% > B > 105\%$.
4. $A < 6$ e $95\% > B > 105\%$.

scaturite con la BN per ogni ROI analizzata della patologia H&N, ADK prostatico e SBRT polmonare.

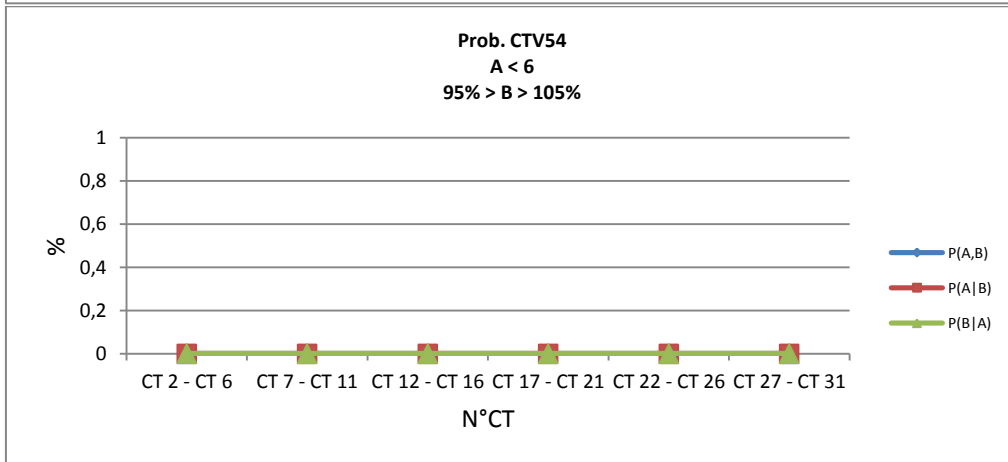
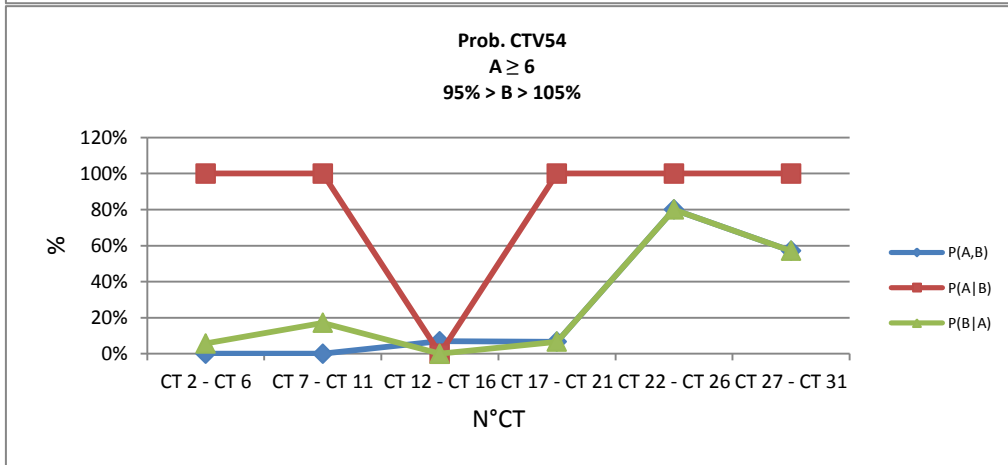
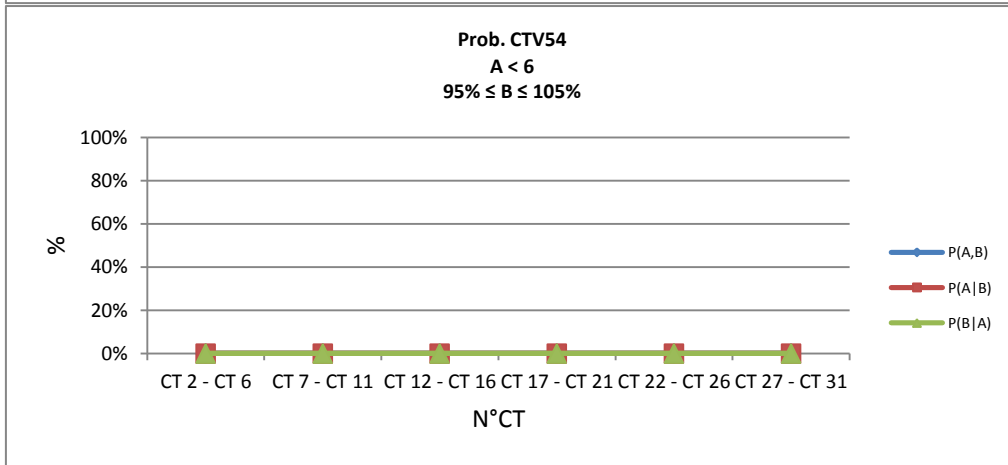
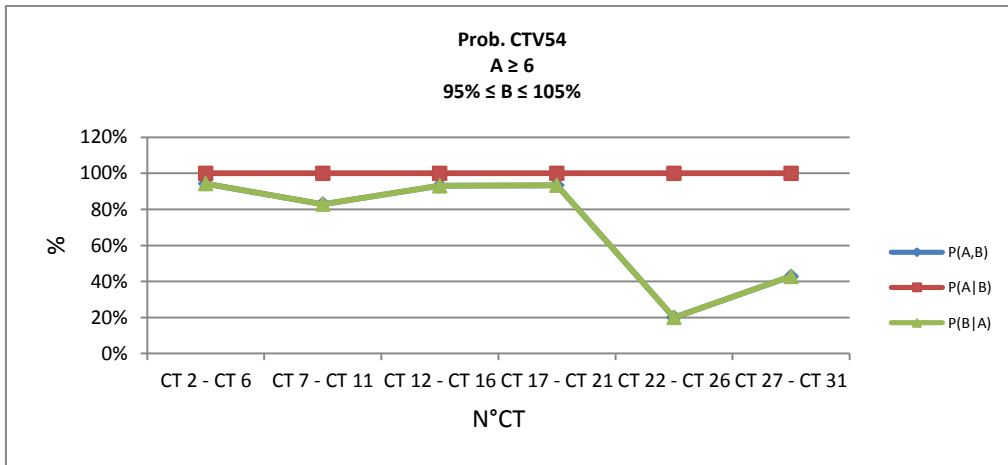
B.1 Head & Neck

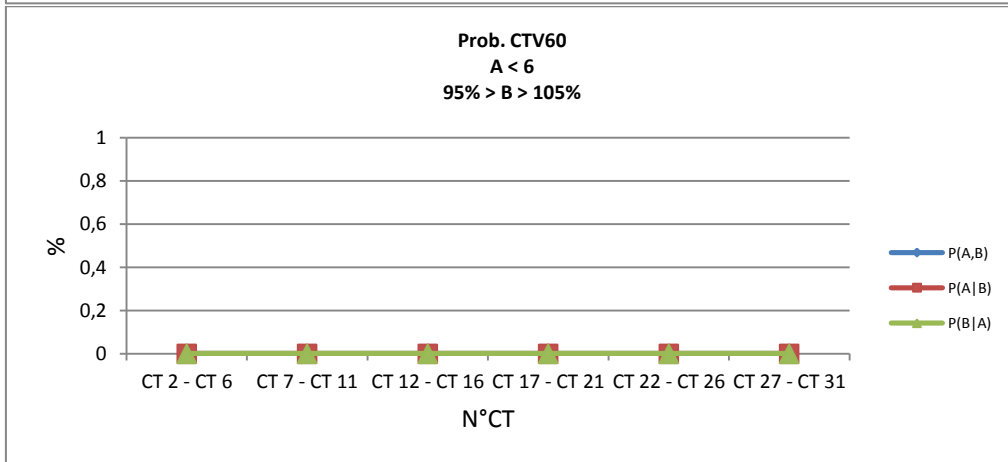
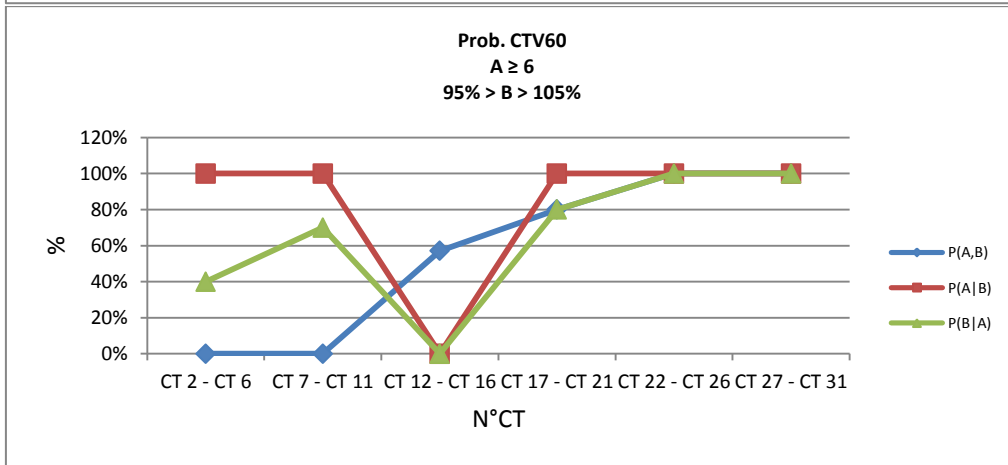
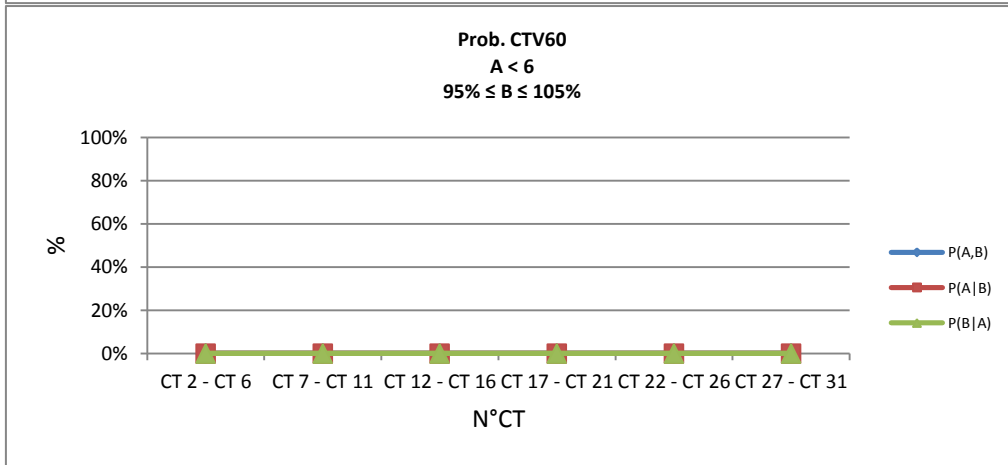
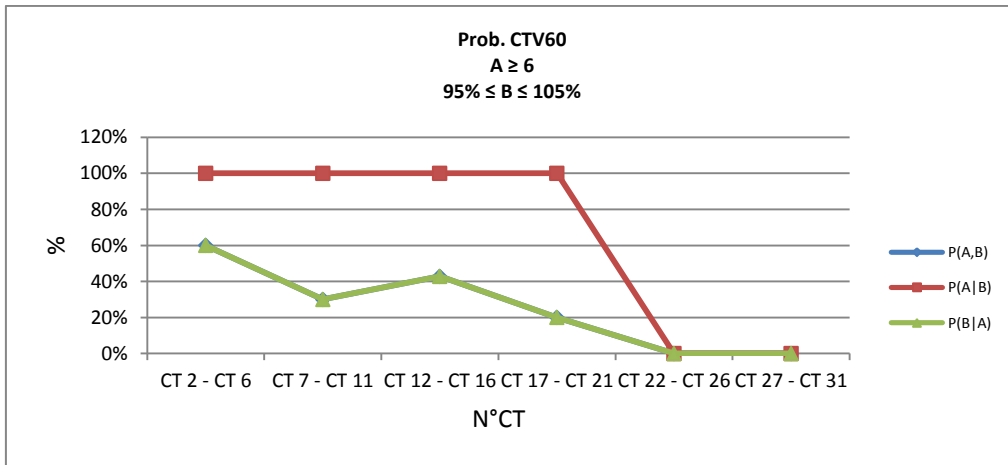
	Prob.	CT	CTV54	CTV60	CTV66	Mand.	Midollo	Par. dx	Par. sx
A ≥ 6 95% ≤ B ≤ 105%	P(A,B)	2 - 6	94%	60%	48%	71%	77%	40%	47%
	P(A,B)	7 - 11	83%	30%	84%	82%	26%	27%	29%
	P(A,B)	12 - 16	93%	43%	89%	54%	41%	36%	26%
	P(A,B)	17 - 21	93%	20%	93%	61%	47%	43%	26%
	P(A,B)	22 - 26	20%	0%	73%	80%	53%	27%	27%
	P(A,B)	27 - 31	43%	0%	43%	43%	43%	0%	0%
	P(A B)	2 - 6	100%	100%	57%	100%	100%	86%	81%
	P(A B)	7 - 11	100%	100%	91%	100%	100%	75%	72%
	P(A B)	12 - 16	100%	100%	94%	100%	100%	100%	100%
	P(A B)	17 - 21	100%	100%	100%	100%	100%	83%	60%
	P(A B)	22 - 26	100%	0%	100%	100%	100%	100%	100%
	P(A B)	27 - 31	100%	0%	100%	100%	100%	0%	0%
	P(B A)	2 - 6	94%	60%	75%	71%	77%	45%	53%
	P(B A)	7 - 11	83%	30%	91%	82%	26%	30%	33%
	P(B A)	12 - 16	93%	43%	94%	54%	41%	36%	26%
	P(B A)	17 - 21	93%	20%	93%	61%	47%	83%	40%
	P(B A)	22 - 26	20%	0%	73%	80%	53%	27%	27%
	P(B A)	27 - 31	43%	0%	43%	43%	43%	0%	0%
A < 6 95% ≤ B ≤ 105%	P(A,B)	2 - 6	0%	0%	36%	0%	0%	7%	11%
	P(A,B)	7 - 11	0%	0%	8%	0%	0%	9%	11%
	P(A,B)	12 - 16	0%	0%	5%	0%	0%	0%	0%
	P(A,B)	17 - 21	0%	0%	0%	0%	0%	9%	17%
	P(A,B)	22 - 26	0%	0%	0%	0%	0%	0%	0%
	P(A,B)	27 - 31	0%	0%	0%	0%	0%	0%	0%
	P(A B)	2 - 6	0%	0%	43%	0%	0%	14%	19%
	P(A B)	7 - 11	0%	0%	9%	0%	0%	25%	28%
	P(A B)	12 - 16	0%	0%	6%	0%	0%	0%	0%
	P(A B)	17 - 21	0%	0%	0%	0%	0%	17%	40%
	P(A B)	22 - 26	0%	0%	0%	0%	0%	0%	0%
	P(A B)	27 - 31	0%	0%	0%	0%	0%	0%	0%
	P(B A)	2 - 6	0%	0%	100%	0%	0%	60%	100%
	P(B A)	7 - 11	0%	0%	100%	0%	0%	80%	100%
	P(B A)	12 - 16	0%	0%	100%	0%	0%	0%	0%
	P(B A)	17 - 21	0%	0%	0%	0%	0%	17%	50%
	P(B A)	22 - 26	0%	0%	0%	0%	0%	0%	0%
	P(B A)	27 - 31	0%	0%	0%	0%	0%	0%	0%

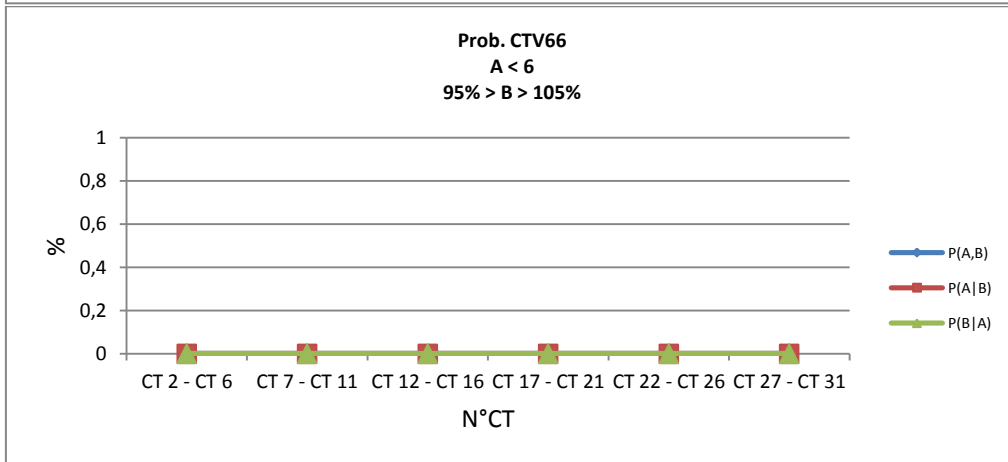
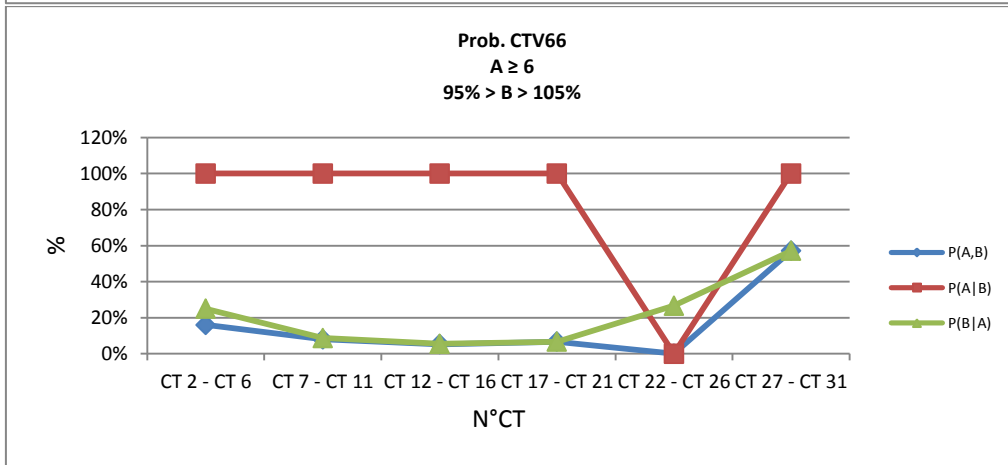
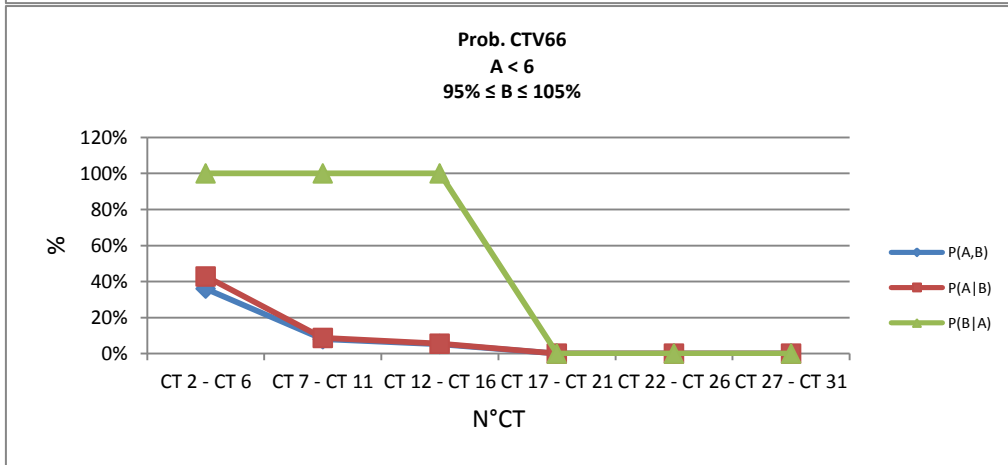
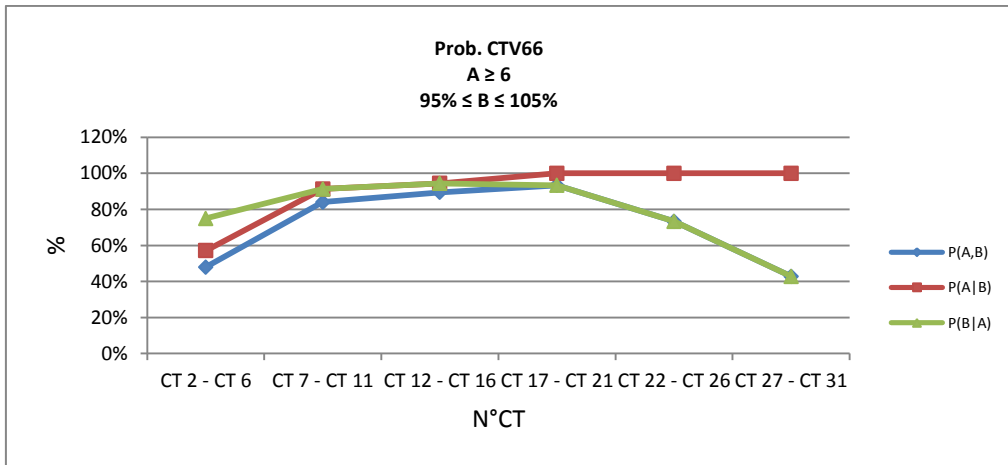
Tabella 20 - Riassunto delle probabilità congiunte $P(A,B)$ e condizionali $P(A|B)$, $P(B|A)$ per gli stati $A \geq 6$ e $95\% \leq B \leq 105\%$, $A < 6$ e $95\% \leq B \leq 105\%$.

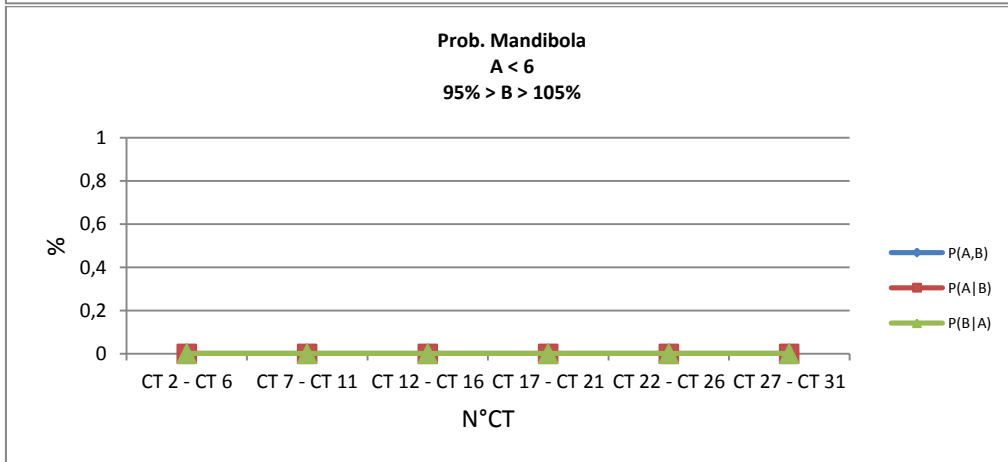
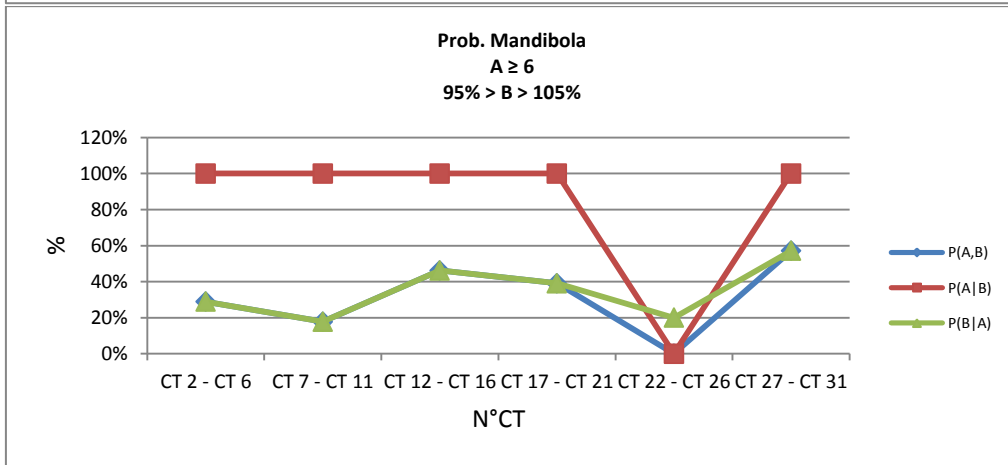
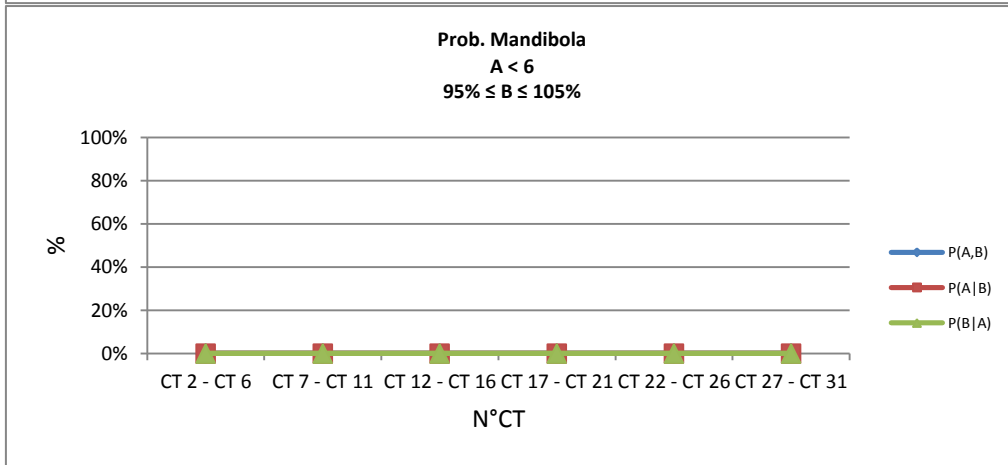
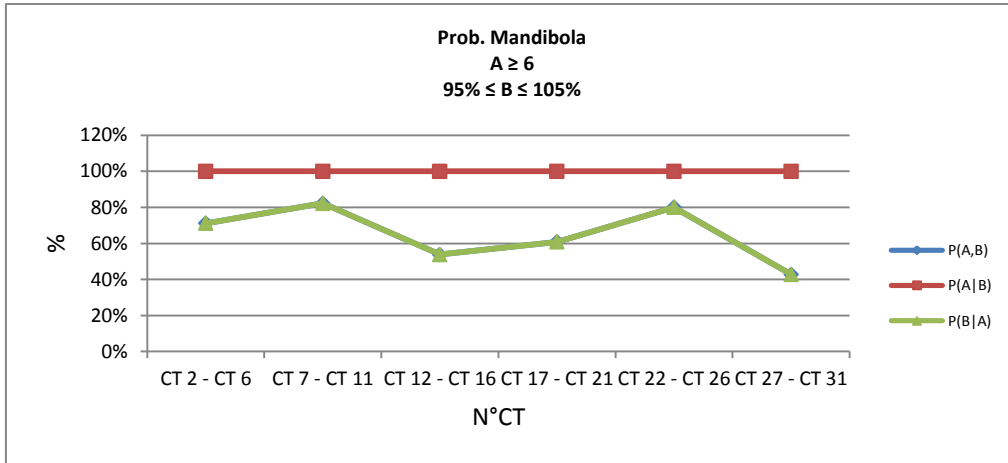
	Prob.	CT	CTV54	CTV60	CTV66	Mand.	Midollo	Par. dx	Par. sx
A ≥ 6 95% > B > 105%	P(A,B)	2 - 6	0%	0%	16%	29%	23%	49%	42%
	P(A,B)	7 - 11	0%	0%	8%	18%	74%	62%	60%
	P(A,B)	12 - 16	7%	57%	5%	46%	59%	64%	74%
	P(A,B)	17 - 21	7%	80%	7%	39%	53%	39%	39%
	P(A,B)	22 - 26	80%	100%	0%	0%	47%	73%	73%
	P(A,B)	27 - 31	57%	100%	57%	57%	57%	100%	100%
	P(A B)	2 - 6	100%	100%	100%	100%	100%	92%	100%
	P(A B)	7 - 11	100%	100%	100%	100%	100%	97%	100%
	P(A B)	12 - 16	0%	0%	100%	100%	100%	100%	100%
	P(A B)	17 - 21	100%	100%	100%	100%	100%	82%	69%
	P(A B)	22 - 26	100%	100%	0%	0%	100%	100%	100%
	P(A B)	27 - 31	100%	100%	100%	100%	100%	100%	100%
	P(B A)	2 - 6	6%	40%	25%	29%	23%	55%	48%
	P(B A)	7 - 11	17%	70%	9%	18%	74%	70%	68%
	P(B A)	12 - 16	0%	0%	6%	46%	59%	64%	74%
	P(B A)	17 - 21	7%	80%	7%	39%	53%	82%	60%
	P(B A)	22 - 26	80%	100%	27%	20%	47%	73%	73%
	P(B A)	27 - 31	57%	100%	57%	57%	57%	100%	100%
A < 6 95% > B > 105%	P(A,B)	2 - 6	0%	0%	0%	0%	0%	4%	0%
	P(A,B)	7 - 11	0%	0%	0%	0%	0%	2%	0%
	P(A,B)	12 - 16	0%	0%	0%	0%	0%	0%	0%
	P(A,B)	17 - 21	0%	0%	0%	0%	0%	9%	17%
	P(A,B)	22 - 26	0%	0%	0%	0%	0%	0%	0%
	P(A,B)	27 - 31	0%	0%	0%	0%	0%	0%	0%
	P(A B)	2 - 6	0%	0%	0%	0%	0%	8%	0%
	P(A B)	7 - 11	0%	0%	0%	0%	0%	3%	0%
	P(A B)	12 - 16	0%	0%	0%	0%	0%	0%	0%
	P(A B)	17 - 21	0%	0%	0%	0%	0%	18%	31%
	P(A B)	22 - 26	0%	0%	0%	0%	0%	0%	0%
	P(A B)	27 - 31	0%	0%	0%	0%	0%	0%	0%
	P(B A)	2 - 6	0%	0%	0%	0%	0%	40%	0%
	P(B A)	7 - 11	0%	0%	0%	0%	0%	20%	0%
	P(B A)	12 - 16	0%	0%	0%	0%	0%	0%	0%
	P(B A)	17 - 21	0%	0%	0%	0%	0%	18%	50%
	P(B A)	22 - 26	0%	0%	0%	0%	0%	0%	0%
	P(B A)	27 - 31	0%	0%	0%	0%	0%	0%	0%

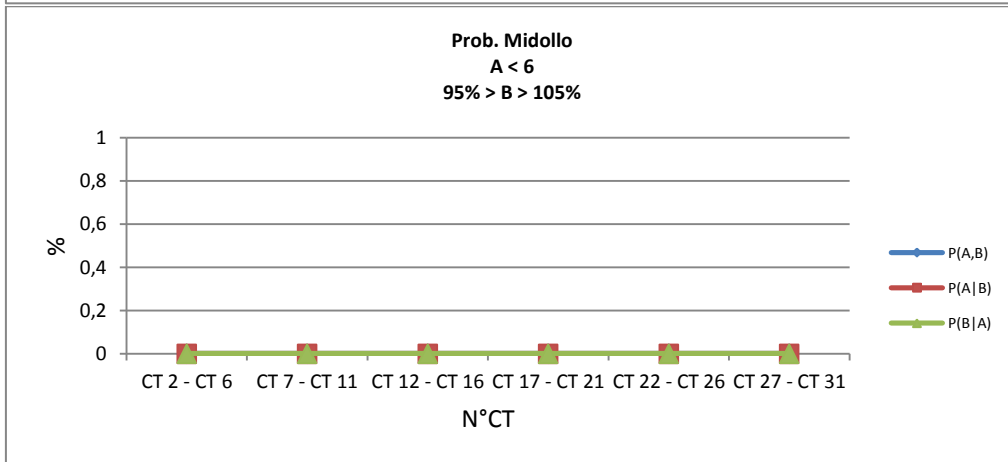
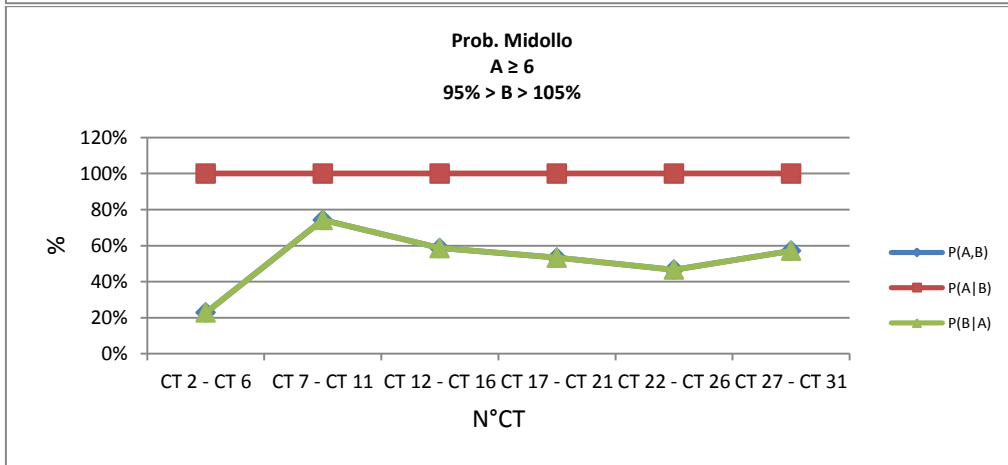
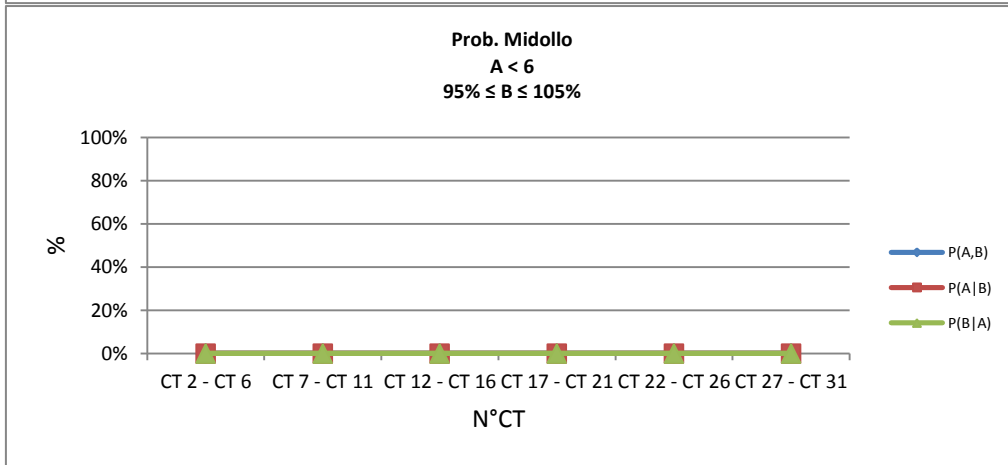
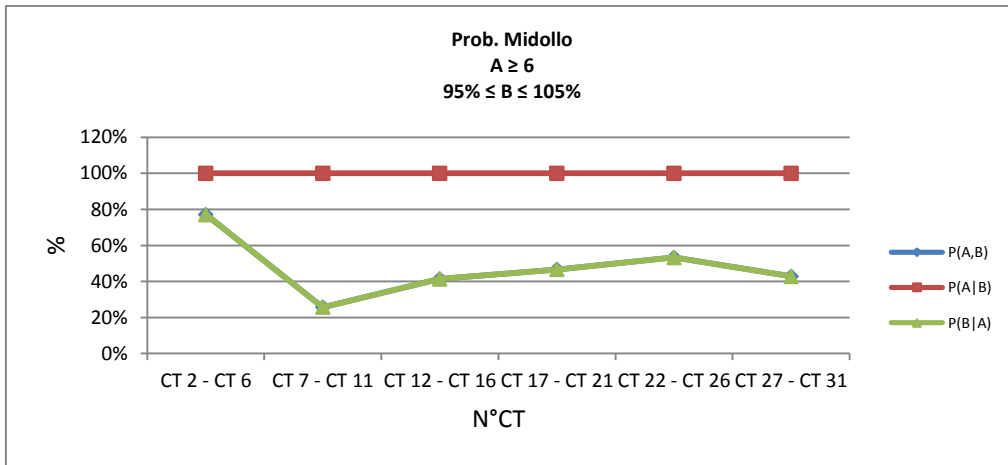
Tabella 21 - Riassunto delle probabilità congiunte $P(A,B)$ e condizionali $P(A|B)$, $P(B|A)$ per gli stati $A \geq 6$ e $95\% > B > 105\%$, $A < 6$ e $95\% > B > 105\%$.

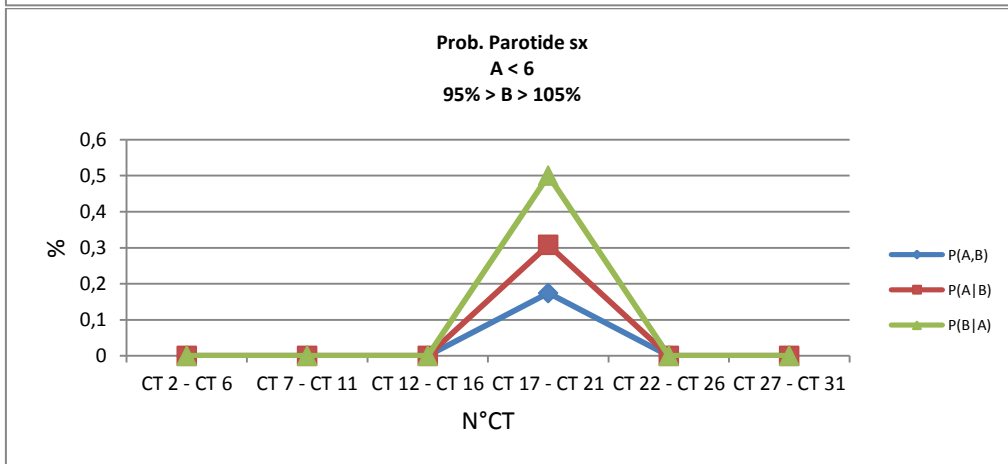
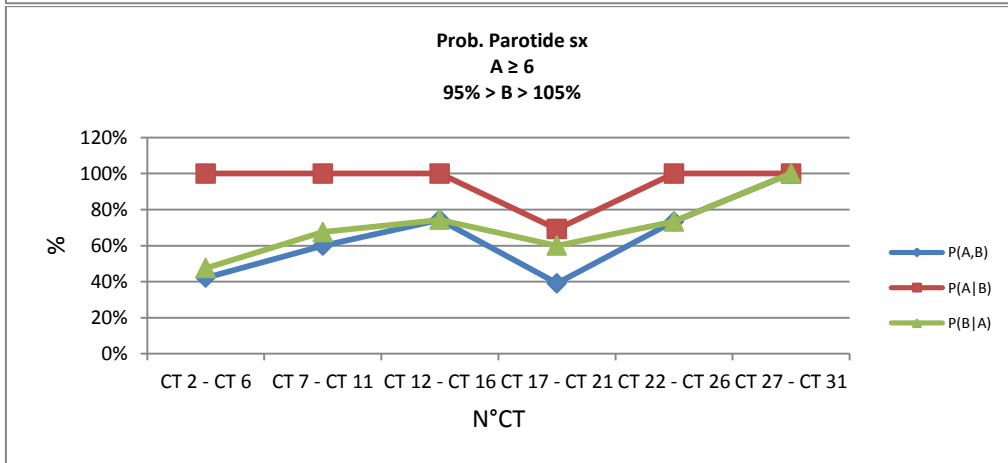
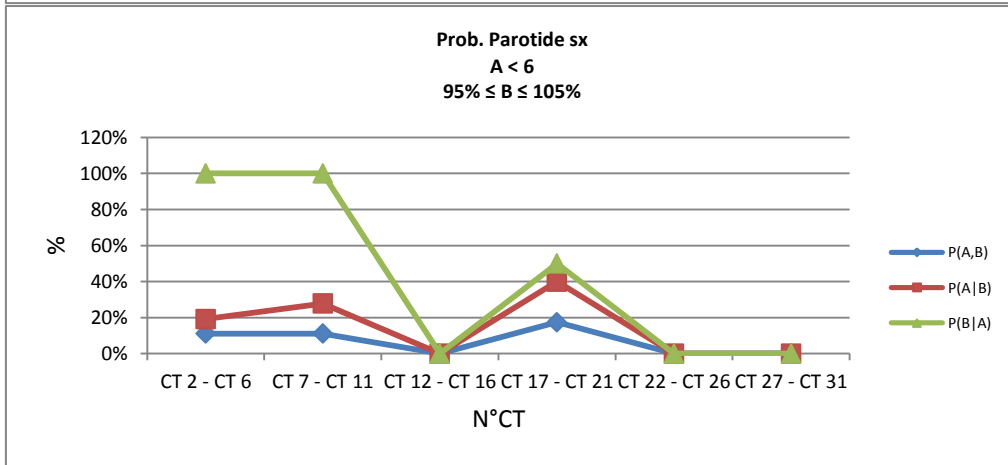
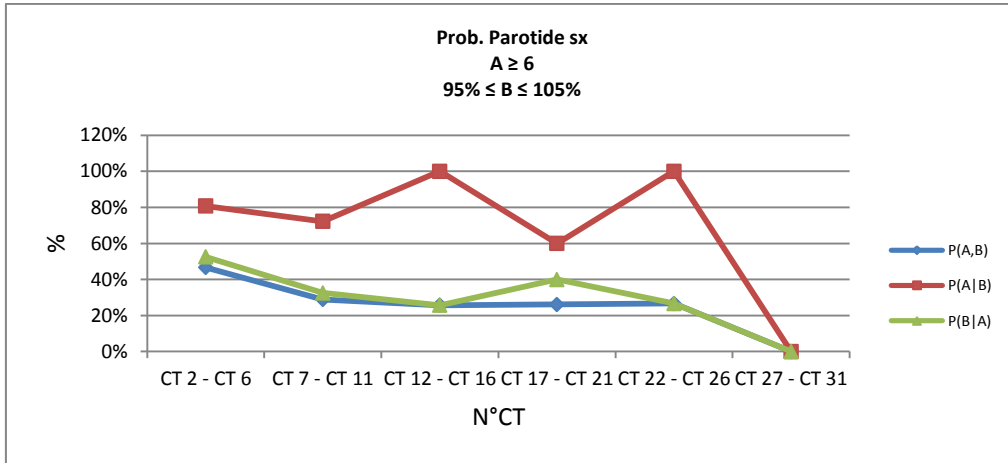












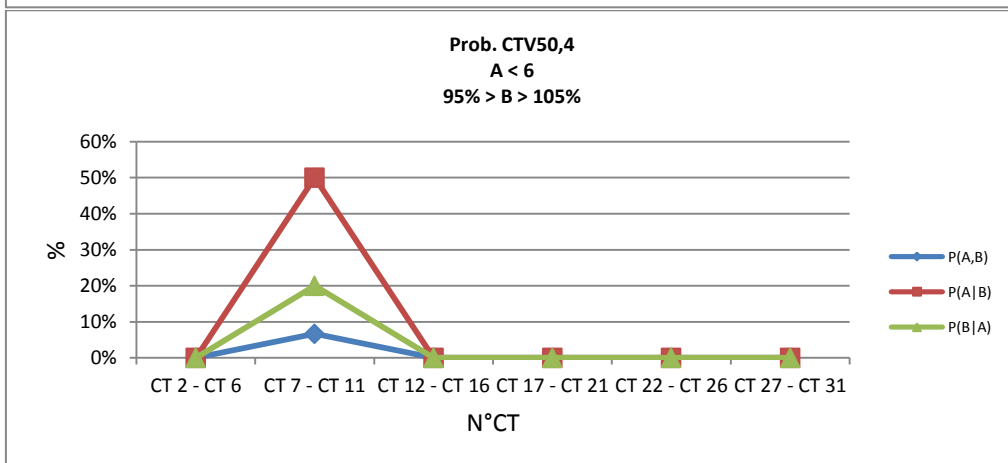
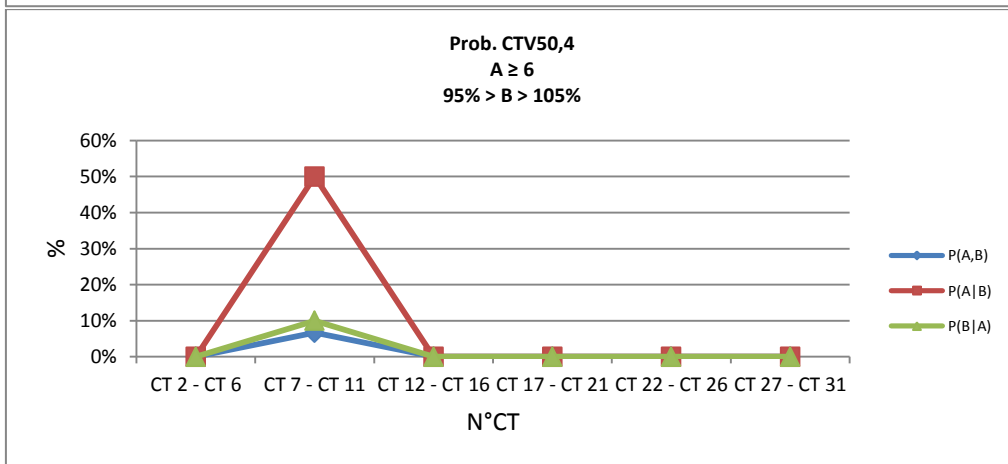
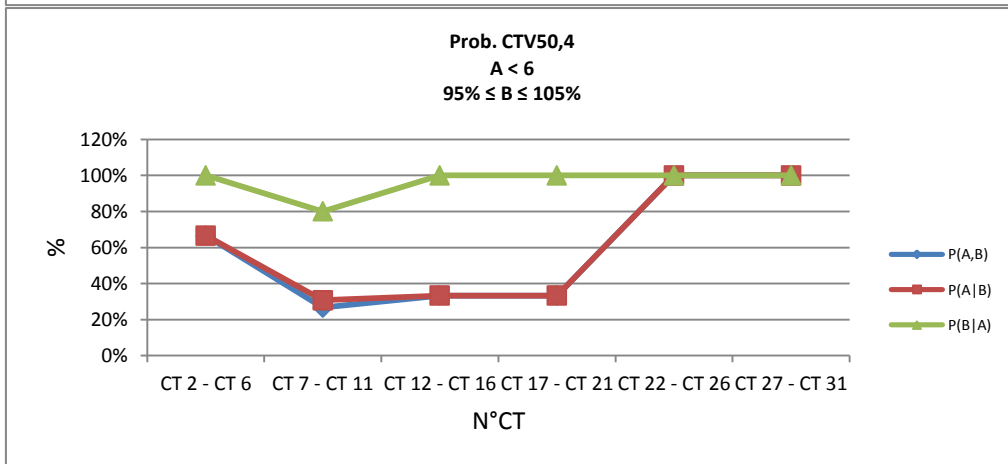
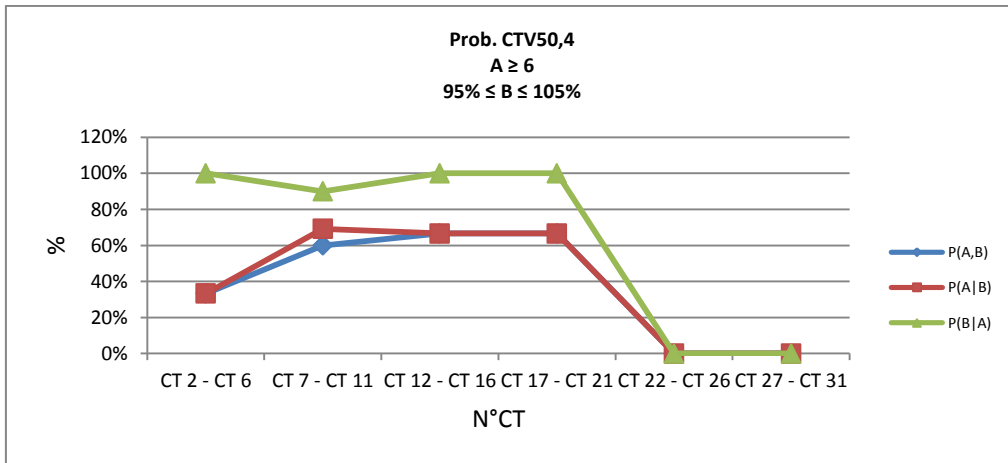
B.2 ADK prostatico

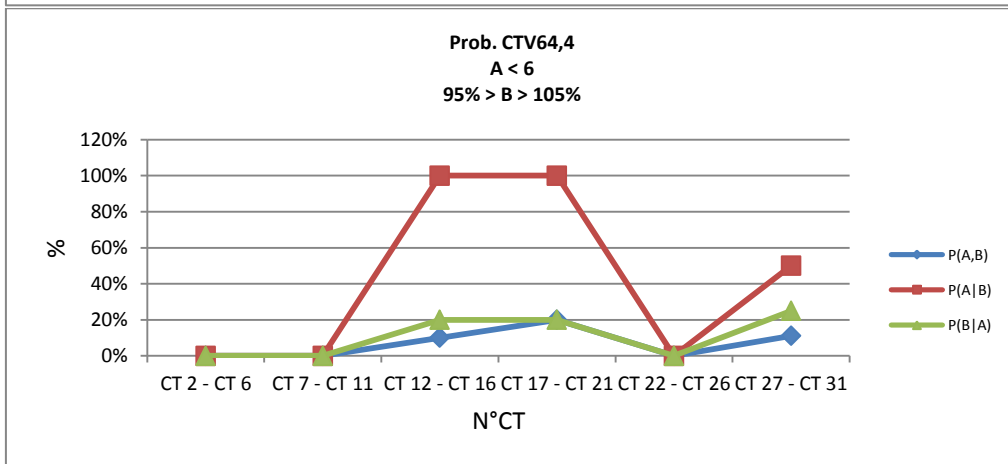
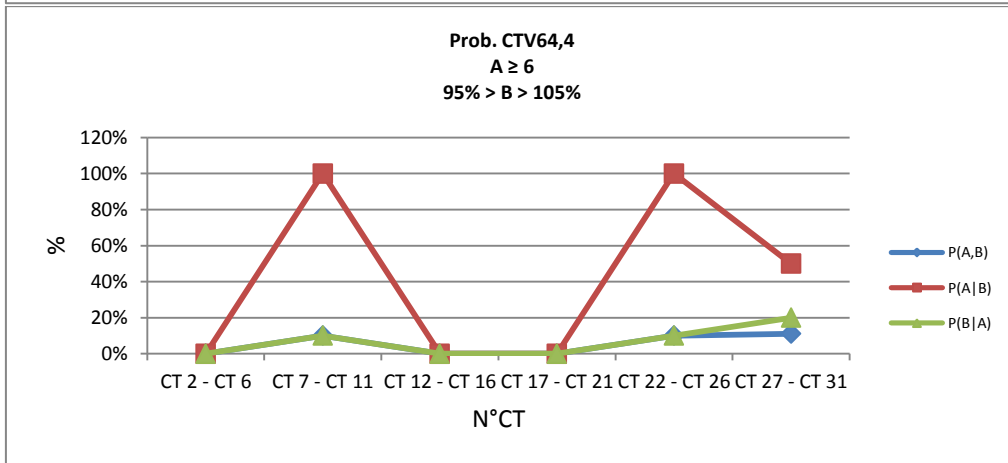
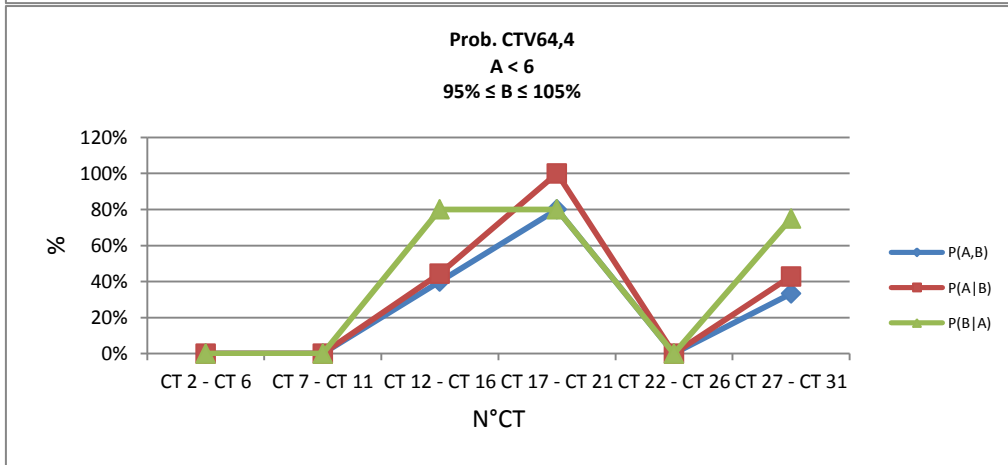
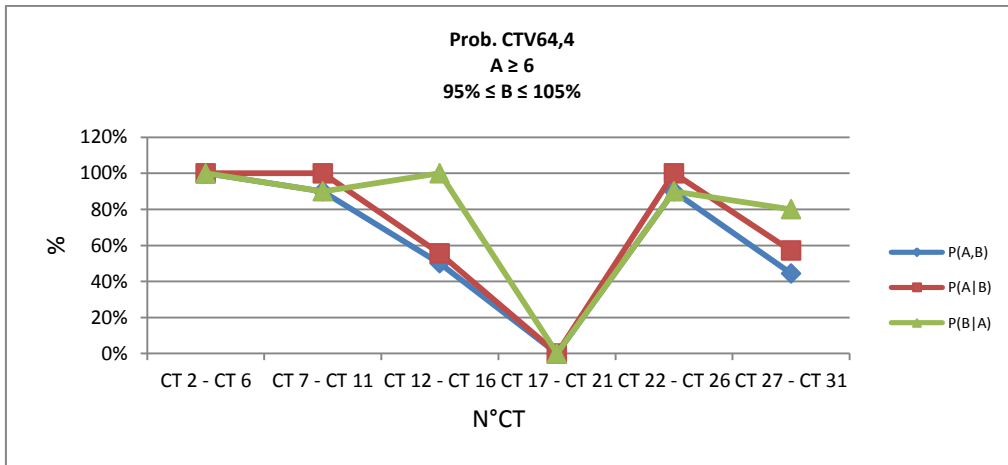
	Prob.	CT	CTV50,4	CTV64,4	CTV66	Retto	T. fem. dx	T. fem. sx	Vescica	
A ≥ 6 95% ≤ B ≤ 105%	P(A,B)	2 - 6	33%	100%	50%	50%	87%	90%	20%	
	P(A,B)	7 - 11	60%	90%	40%	17%	70%	90%	13%	
	P(A,B)	12 - 16	67%	50%	40%	13%	77%	97%	7%	
	P(A,B)	17 - 21	67%	0%	50%	25%	86%	93%	7%	
	P(A,B)	22 - 26	0%	90%	40%	36%	84%	80%	8%	
	P(A,B)	27 - 31	0%	44%	25%	24%	94%	100%	0%	
	P(A B)	2 - 6	33%	100%	50%	94%	100%	100%	46%	
	P(A B)	7 - 11	69%	100%	50%	45%	81%	100%	57%	
	P(A B)	12 - 16	67%	56%	50%	80%	92%	100%	100%	
	P(A B)	17 - 21	67%	0%	71%	64%	100%	100%	40%	
	P(A B)	22 - 26	0%	100%	44%	100%	100%	100%	29%	
	P(A B)	27 - 31	0%	57%	50%	80%	100%	100%	0%	
	P(B A)	2 - 6	100%	100%	100%	60%	87%	90%	40%	
	P(B A)	7 - 11	90%	90%	80%	33%	84%	90%	16%	
	P(B A)	12 - 16	100%	100%	80%	27%	92%	97%	10%	
	P(B A)	17 - 21	100%	0%	100%	39%	86%	93%	11%	
	P(B A)	22 - 26	0%	90%	80%	36%	84%	80%	13%	
	P(B A)	27 - 31	0%	80%	40%	33%	94%	100%	0%	
	A < 6 95% ≤ B ≤ 105%	P(A,B)	2 - 6	67%	0%	50%	3%	0%	0%	23%
		P(A,B)	7 - 11	27%	0%	40%	20%	17%	0%	10%
P(A,B)		12 - 16	33%	40%	40%	3%	7%	0%	0%	
P(A,B)		17 - 21	33%	80%	20%	14%	0%	0%	11%	
P(A,B)		22 - 26	100%	0%	50%	0%	0%	0%	20%	
P(A,B)		27 - 31	100%	33%	25%	6%	0%	0%	12%	
P(A B)		2 - 6	67%	0%	50%	6%	0%	0%	54%	
P(A B)		7 - 11	31%	0%	50%	55%	19%	0%	43%	
P(A B)		12 - 16	33%	44%	50%	20%	8%	0%	0%	
P(A B)		17 - 21	33%	100%	29%	36%	0%	0%	60%	
P(A B)		22 - 26	100%	0%	56%	0%	0%	0%	71%	
P(A B)		27 - 31	100%	43%	50%	20%	0%	0%	100%	
P(B A)		2 - 6	100%	0%	100%	20%	0%	0%	47%	
P(B A)		7 - 11	80%	0%	80%	40%	50%	0%	60%	
P(B A)	12 - 16	100%	80%	80%	7%	20%	0%	0%		
P(B A)	17 - 21	100%	80%	40%	40%	0%	0%	30%		
P(B A)	22 - 26	100%	0%	100%	0%	0%	0%	50%		
P(B A)	27 - 31	100%	75%	67%	20%	0%	0%	17%		

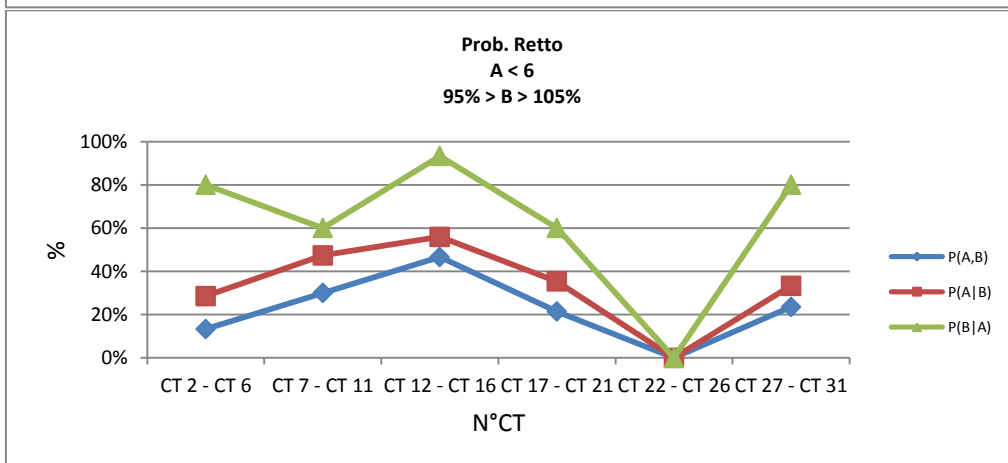
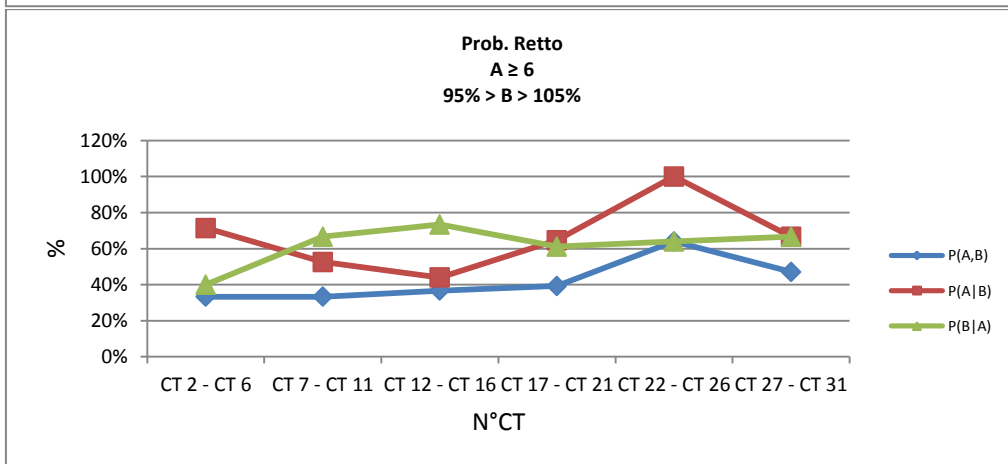
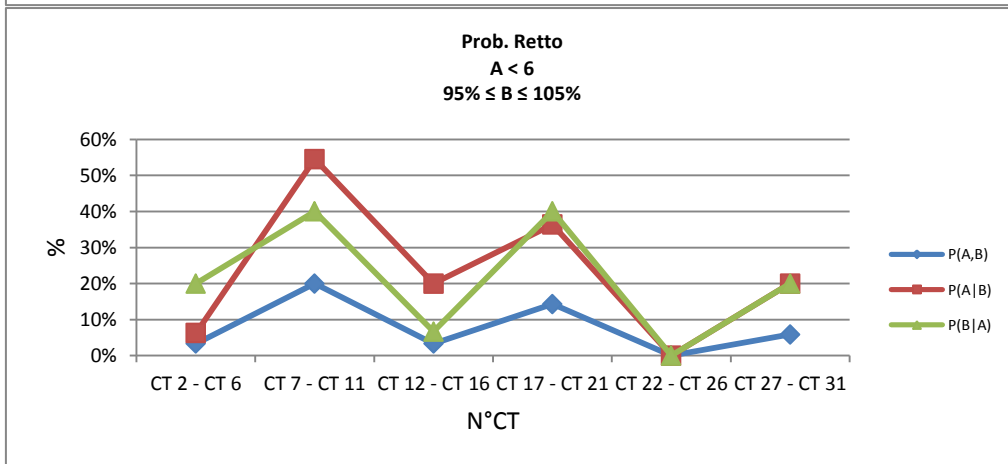
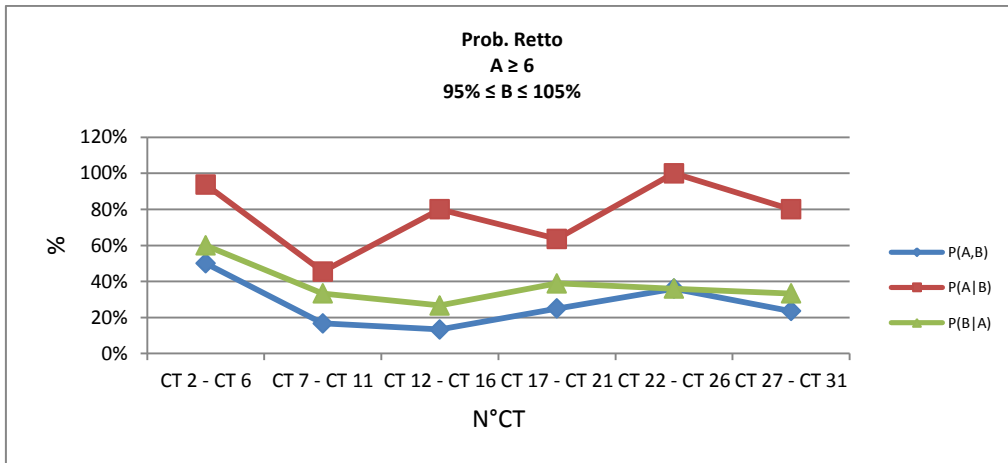
Tabella 22 - Riassunto delle probabilità congiunte $P(A,B)$ e condizionali $P(A|B)$, $P(B|A)$ per gli stati $A \geq 6$ e $95\% \leq B \leq 105\%$, $A < 6$ e $95\% \leq B \leq 105\%$.

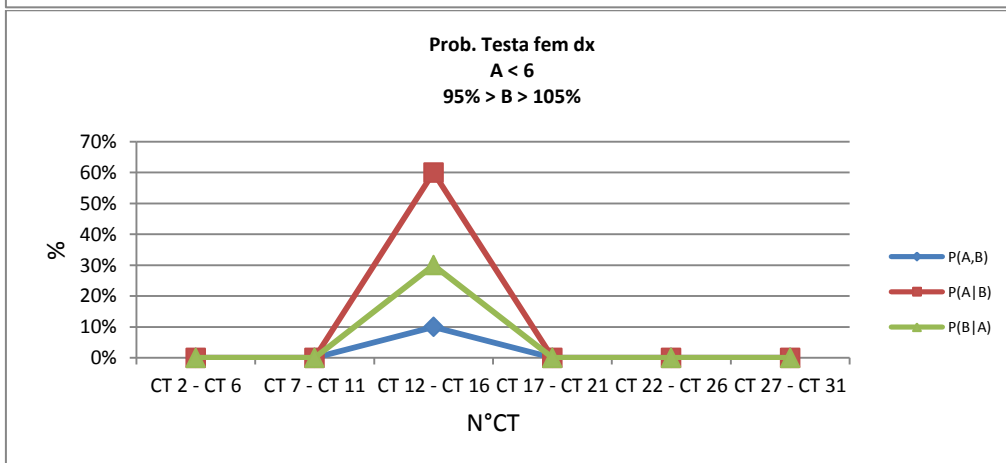
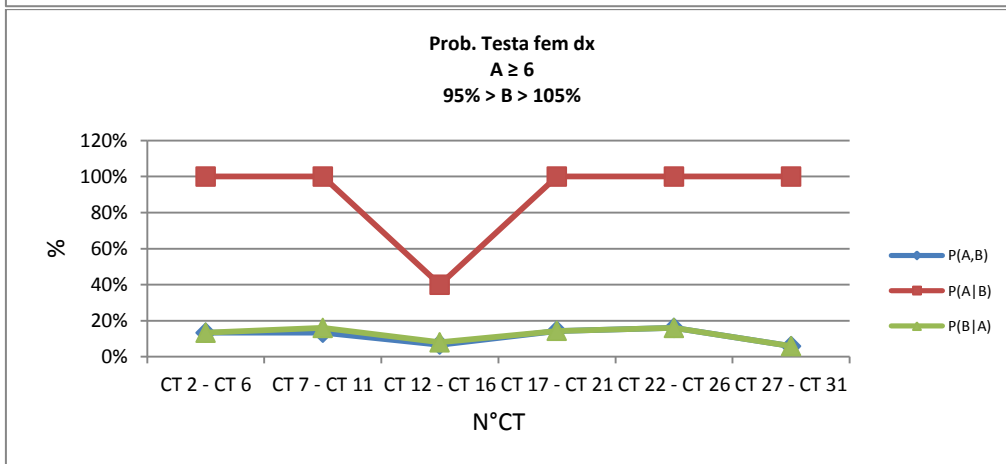
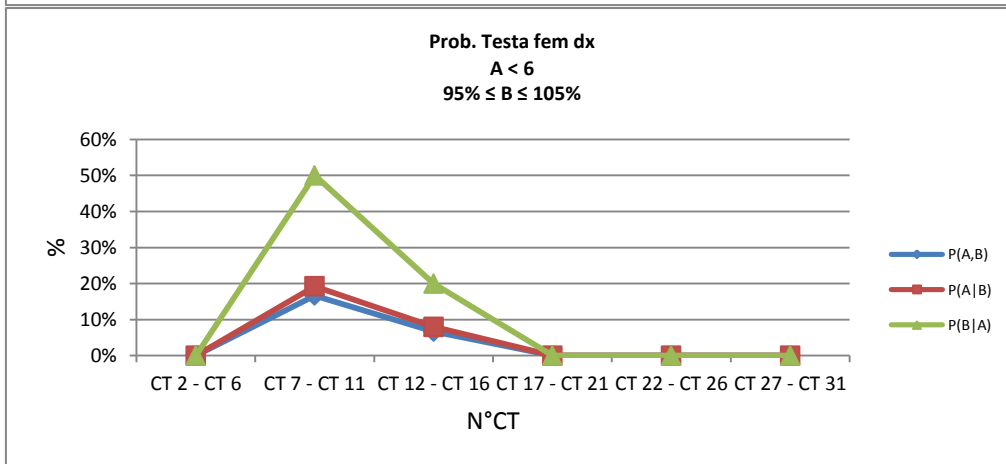
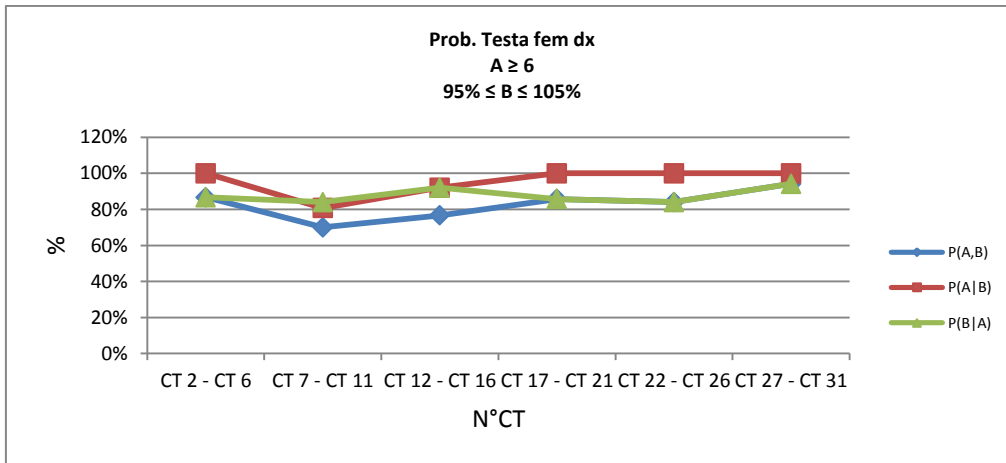
	Prob.	CT	CTV50,4	CTV64,4	CTV66	Retto	T. fem. dx	T. fem. sx	Vescica
A ≥ 6 95% > B > 105%	P(A,B)	2 - 6	0%	0%	0%	33%	13%	10%	30%
	P(A,B)	7 - 11	7%	10%	10%	33%	13%	10%	70%
	P(A,B)	12 - 16	0%	0%	10%	37%	7%	3%	60%
	P(A,B)	17 - 21	0%	0%	0%	39%	14%	7%	57%
	P(A,B)	22 - 26	0%	10%	10%	64%	16%	20%	52%
	P(A,B)	27 - 31	0%	11%	38%	47%	6%	0%	29%
	P(A B)	2 - 6	0%	0%	0%	71%	100%	100%	53%
	P(A B)	7 - 11	50%	100%	50%	53%	100%	100%	91%
	P(A B)	12 - 16	0%	0%	50%	44%	40%	100%	64%
	P(A B)	17 - 21	0%	0%	0%	65%	100%	100%	70%
	P(A B)	22 - 26	0%	100%	100%	100%	100%	100%	72%
	P(A B)	27 - 31	0%	50%	75%	67%	100%	0%	33%
	P(B A)	2 - 6	0%	0%	0%	40%	13%	10%	60%
	P(B A)	7 - 11	10%	10%	20%	67%	16%	10%	84%
	P(B A)	12 - 16	0%	0%	20%	73%	8%	3%	90%
	P(B A)	17 - 21	0%	0%	0%	61%	14%	7%	89%
	P(B A)	22 - 26	0%	10%	20%	64%	16%	20%	87%
	P(B A)	27 - 31	0%	20%	60%	67%	6%	0%	100%
A < 6 95% > B > 105%	P(A,B)	2 - 6	0%	0%	0%	13%	0%	0%	27%
	P(A,B)	7 - 11	7%	0%	10%	30%	0%	0%	7%
	P(A,B)	12 - 16	0%	10%	10%	47%	10%	0%	33%
	P(A,B)	17 - 21	0%	20%	30%	21%	0%	0%	25%
	P(A,B)	22 - 26	0%	0%	0%	0%	0%	0%	20%
	P(A,B)	27 - 31	0%	11%	13%	24%	0%	0%	59%
	P(A B)	2 - 6	0%	0%	0%	29%	0%	0%	47%
	P(A B)	7 - 11	50%	0%	50%	47%	0%	0%	9%
	P(A B)	12 - 16	0%	100%	50%	56%	60%	0%	36%
	P(A B)	17 - 21	0%	100%	100%	35%	0%	0%	30%
	P(A B)	22 - 26	0%	0%	0%	0%	0%	0%	28%
	P(A B)	27 - 31	0%	50%	25%	33%	0%	0%	67%
	P(B A)	2 - 6	0%	0%	0%	80%	0%	0%	53%
	P(B A)	7 - 11	20%	0%	20%	60%	0%	0%	40%
	P(B A)	12 - 16	0%	20%	20%	93%	30%	0%	100%
	P(B A)	17 - 21	0%	20%	60%	60%	0%	0%	70%
	P(B A)	22 - 26	0%	0%	0%	0%	0%	0%	50%
	P(B A)	27 - 31	0%	25%	33%	80%	0%	0%	83%

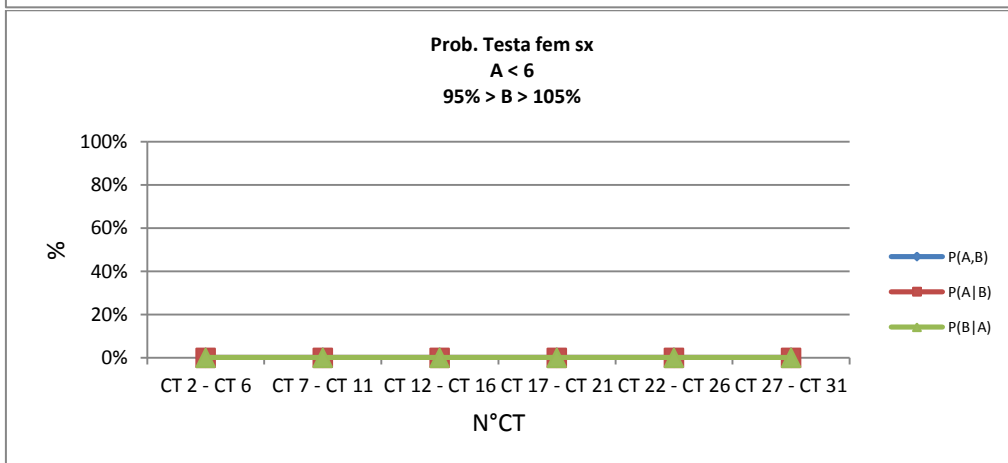
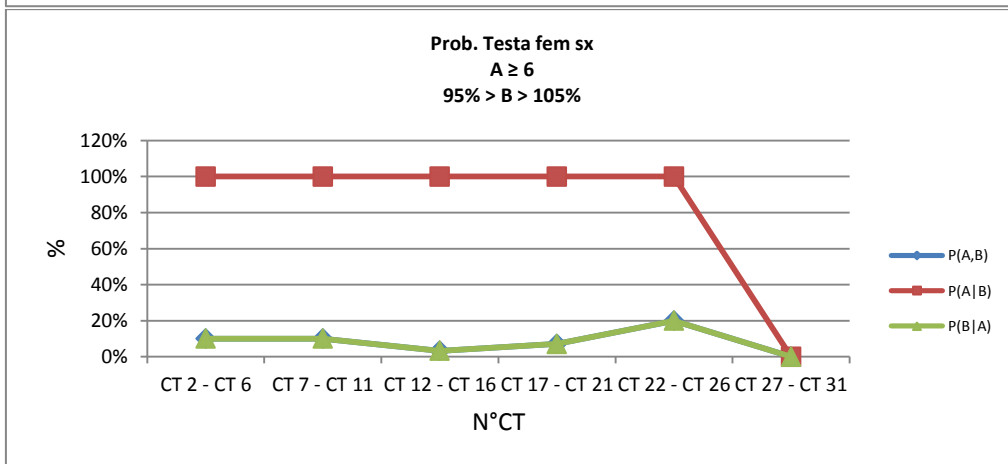
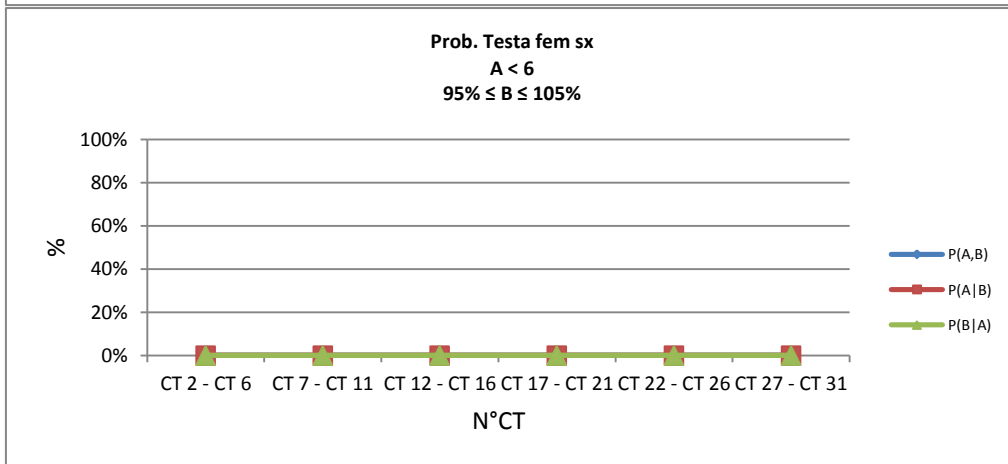
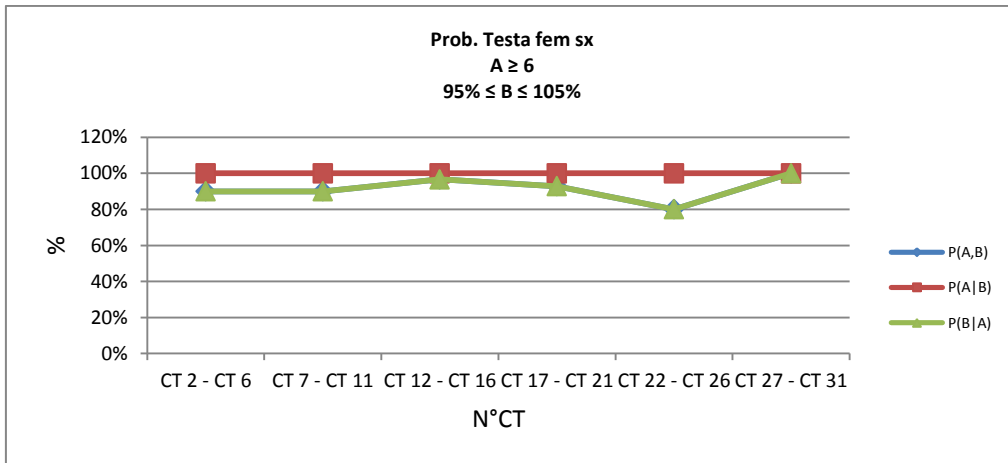
Tabella 23 - Riassunto delle probabilità congiunte $P(A,B)$ e condizionali $P(A|B)$, $P(B|A)$ per gli stati $A \geq 6$ e $95\% > B > 105\%$, $A < 6$ e $95\% > B > 105\%$.

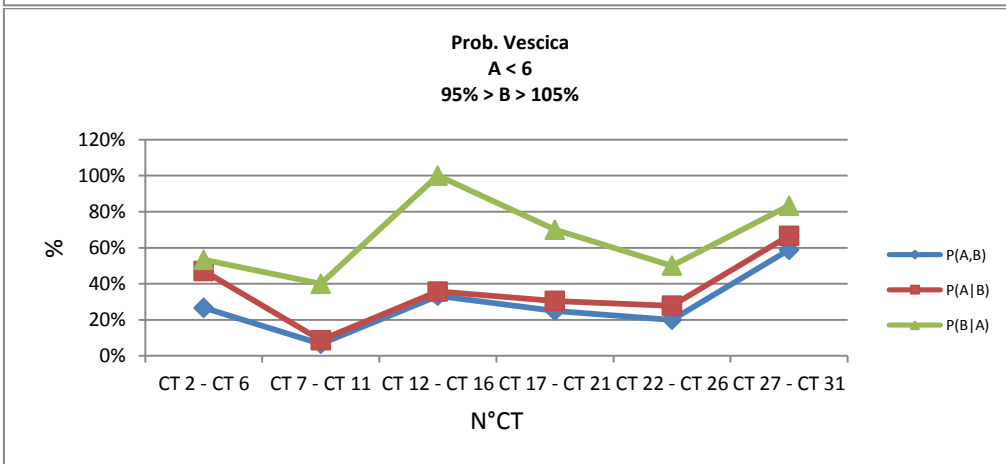
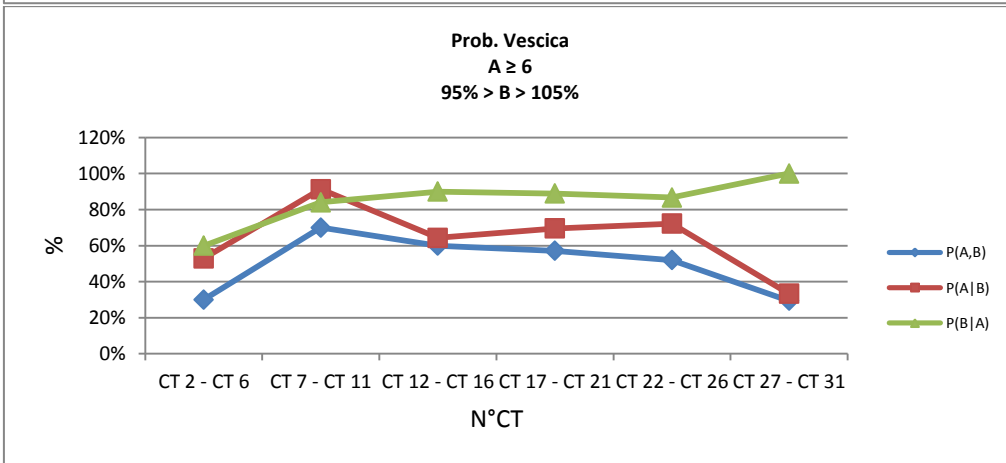
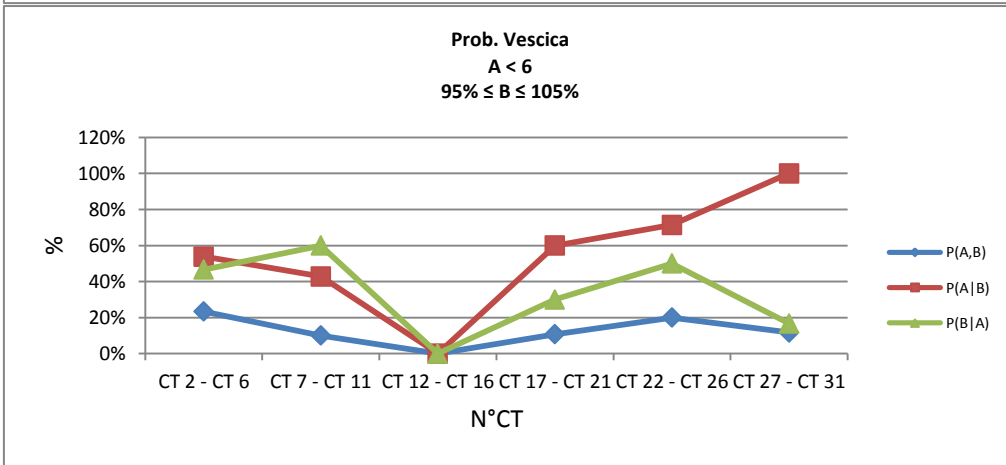
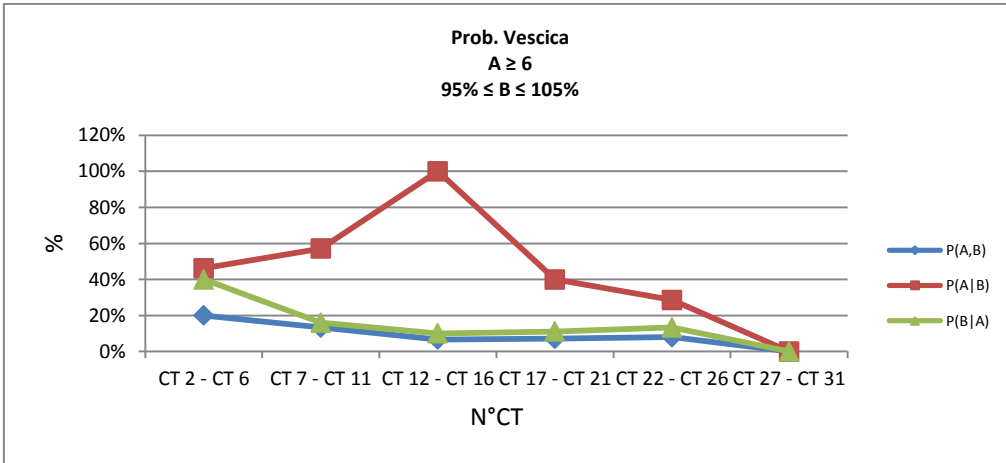












B.3 SBRT polmonare

	Prob.	CT	Polm dx	Polm sx	Cuore	Midolo	Esofago	Trachea	GTV	Coste	Grandi vasi	
A ≥ 6 95% ≤ B ≤ 105%	P(A,B)	CT2	94%	100%	70%	97%	86%	100%	82%	65%	75%	
	P(A,B)	CT3	86%	42%	60%	68%	71%	38%	0%	19%	17%	
	P(A,B)	CT4	59%	64%	50%	69%	75%	33%	8%	35%	13%	
	P(A,B)	CT5	81%	71%	90%	75%	68%	33%	32%	20%	56%	
	P(A,B)	CT6	57%	67%	50%	57%	57%	33%	29%	10%	38%	
	P(A B)	CT2	94%	100%	70%	97%	86%	100%	82%	65%	75%	
	P(A B)	CT3	100%	83%	100%	95%	85%	75%	0%	75%	50%	
	P(A B)	CT4	95%	90%	83%	92%	88%	100%	50%	88%	50%	
	P(A B)	CT5	93%	100%	90%	100%	95%	100%	89%	50%	75%	
	P(A B)	CT6	100%	100%	100%	100%	80%	100%	75%	50%	75%	
	P(B A)	CT2	100%	100%	100%	100%	100%	100%	100%	100%	100%	
	P(B A)	CT3	86%	45%	67%	73%	89%	43%	0%	30%	25%	
	P(B A)	CT4	66%	78%	63%	81%	84%	33%	13%	64%	22%	
	P(B A)	CT5	87%	74%	100%	75%	76%	33%	44%	36%	69%	
	P(B A)	CT6	57%	67%	62%	70%	67%	33%	43%	18%	46%	
	A < 6 95% ≤ B ≤ 105%	P(A,B)	CT2	6%	0%	30%	3%	14%	0%	18%	35%	25%
		P(A,B)	CT3	0%	8%	0%	4%	13%	13%	0%	6%	17%
		P(A,B)	CT4	3%	7%	10%	6%	11%	0%	8%	5%	13%
P(A,B)		CT5	6%	0%	10%	0%	4%	0%	4%	20%	19%	
P(A,B)		CT6	0%	0%	0%	0%	14%	0%	10%	10%	13%	
P(A B)		CT2	6%	0%	30%	3%	14%	0%	18%	35%	25%	
P(A B)		CT3	0%	17%	0%	5%	15%	25%	25%	25%	50%	
P(A B)		CT4	5%	10%	17%	8%	13%	0%	50%	13%	50%	
P(A B)		CT5	7%	0%	10%	0%	5%	0%	11%	50%	25%	
P(A B)		CT6	0%	0%	0%	0%	20%	0%	25%	50%	25%	
P(B A)		CT2	100%	0%	100%	100%	100%	0%	100%	100%	100%	
P(B A)		CT3	0%	100%	0%	50%	60%	100%	0%	17%	50%	
P(B A)		CT4	33%	40%	50%	40%	100%	0%	20%	11%	29%	
P(B A)		CT5	100%	0%	100%	0%	33%	0%	14%	44%	100%	
P(B A)	CT6	0%	0%	0%	0%	100%	0%	29%	22%	67%		

Tabella 24 - Riassunto delle probabilità congiunte $P(A,B)$ e condizionali $P(A|B)$, $P(B|A)$ per gli stati $A \geq 6$ e $95\% \leq B \leq 105\%$, $A < 6$ e $95\% \leq B \leq 105\%$.

	Prob.	CT	Polmone dx	Polmone sx	Cuore	Midollo	Esofago	Trachea	GTV	Coste	Grandi vasi	
A ≥ 6 95% > B > 105%	P(A,B)	CT2	0%	0%	0%	0%	0%	0%	0%	0%	0%	
	P(A,B)	CT3	14%	50%	30%	25%	8%	50%	65%	44%	50%	
	P(A,B)	CT4	31%	18%	30%	16%	14%	67%	52%	20%	44%	
	P(A,B)	CT5	13%	25%	0%	25%	21%	67%	40%	35%	25%	
	P(A,B)	CT6	43%	33%	31%	25%	29%	67%	38%	45%	44%	
	P(A B)	CT2	0%	0%	0%	0%	0%	0%	0%	0%	0%	
	P(A B)	CT3	100%	100%	75%	88%	50%	100%	65%	58%	75%	
	P(A B)	CT4	83%	63%	75%	63%	100%	100%	62%	33%	58%	
	P(A B)	CT5	100%	88%	0%	100%	75%	100%	63%	58%	100%	
	P(A B)	CT6	100%	100%	63%	58%	100%	100%	62%	56%	88%	
	P(B A)	CT2	0%	0%	0%	0%	0%	0%	0%	0%	0%	
	P(B A)	CT3	14%	55%	33%	27%	11%	57%	100%	70%	75%	
	P(B A)	CT4	34%	22%	38%	19%	16%	67%	87%	36%	78%	
	P(B A)	CT5	13%	26%	0%	25%	24%	67%	56%	64%	31%	
	P(B A)	CT6	43%	33%	38%	30%	33%	67%	57%	82%	54%	
	A < 6 95% > B > 105%	P(A,B)	CT2	0%	0%	0%	0%	0%	0%	0%	0%	0%
		P(A,B)	CT3	0%	0%	10%	4%	8%	0%	35%	31%	17%
		P(A,B)	CT4	6%	11%	10%	9%	0%	0%	32%	40%	31%
P(A,B)		CT5	0%	4%	0%	0%	7%	0%	24%	25%	0%	
P(A,B)		CT6	0%	0%	19%	18%	0%	0%	24%	35%	6%	
P(A B)		CT2	0%	0%	0%	0%	0%	0%	0%	0%	0%	
P(A B)		CT3	0%	0%	25%	13%	50%	0%	35%	42%	25%	
P(A B)		CT4	17%	38%	25%	38%	0%	0%	38%	67%	42%	
P(A B)		CT5	0%	13%	0%	0%	25%	0%	38%	42%	0%	
P(A B)		CT6	0%	0%	38%	42%	0%	0%	38%	44%	13%	
P(B A)		CT2	0%	0%	0%	0%	0%	0%	0%	0%	0%	
P(B A)		CT3	0%	0%	100%	50%	40%	0%	100%	83%	50%	
P(B A)		CT4	67%	60%	50%	60%	0%	0%	80%	89%	71%	
P(B A)		CT5	0%	100%	0%	0%	67%	0%	86%	56%	0%	
P(B A)	CT6	0%	0%	100%	100%	0%	0%	71%	78%	33%		

Tabella 25 - Riassunto delle probabilità congiunte $P(A,B)$ e condizionali $P(A|B)$, $P(B|A)$ per gli stati $A \geq 6$ e $95\% > B > 105\%$, $A < 6$ e $95\% > B > 105\%$.

