



SAPIENZA
UNIVERSITÀ DI ROMA

Identificazione ed Analisi di Biomarcatori dai Dati di Espressione Genica e Proteica

Facoltà di Ingegneria dell'Informazione, Informatica e Statistica
Dipartimento di Informatica
Corso di laurea in Informatica

Valerio Mesi
Matricola 1936543

Relatore
Maurizio Mancini

Correlatore
Enrico Tronci

A.A. 2022 - 2023

Identificazione ed Analisi di Biomarcatori dai Dati di Espressione Genica e Proteica
Relazione di Tirocinio. Sapienza – Università di Roma

© Valerio Mesi. Tutti i diritti riservati

E-mail dell'autore: mesiti.1936543@studenti.uniroma1.it

Ai miei nonni

Sommario

Abstract	vi
Capitolo 1: Introduzione	2
1.1 Contesto e Motivazione	2
1.2 Scopo e Obiettivi della Tesi	3
Capitolo 2: Revisione e Background	4
2.1 Importanza dei Biomarcatori Correlati alle Malattie	4
2.2 Fonti di Dati Genomici e GDC	5
2.3 Metodologie di Identificazione dei Biomarcatori	5
2.4 Scoperte e Sfide nella Ricerca dei Biomarcatori	6
2.5 Panoramica del database e strumenti utilizzati	7
Capitolo 3: Metodologia ed Implementazione	11
3.1 Creazione della Base di Dati	12
3.1.1 Progettazione Concettuale	12
3.1.2 Ristrutturazione dello Schema Concettuale	23
3.1.3 Modello Relazionale	31
3.1.4 Specifica del Database in SQL	34
3.2 Script di downloading e gestione dati	39
3.3 Script di creazione di alberi decisionali	43
3.3.1 Cos'è un albero decisionale?	43
3.3.2 Codice ed implementazione	45
Capitolo 4: Risultati	49
4.1 Dati Caricati	49
4.2 Interrogazioni e Analisi di Esempio	51
4.3 Biomarcatori e Alberi Decisionali	53
4.4 Limitazioni e Sviluppi Futuri	55
4.4.1 Limitazioni del Database	55
4.4.2 Possibili Sviluppi Futuri	56
Capitolo 5: Conclusioni	57
5.1 Sintesi dei Risultati	57
5.2 Impatto della Ricerca	57
5.3 Sfide Superate	58
5.4 Chiusura	58
Bibliografia	59
Ringraziamenti	61

Abstract

La medicina di precisione sta rivoluzionando l'approccio alla diagnosi, alla prognosi e al trattamento delle malattie, consentendo una personalizzazione più accurata delle terapie in base alle caratteristiche individuali dei pazienti. Questa tesi si propone di **identificare ed analizzare biomarcatori** correlati alle malattie utilizzando dati di **espressione genica e proteica** provenienti dal **Genomic Data Commons (GDC)**, una fonte ricca di informazioni genomiche.

La revisione della letteratura ha sottolineato l'importanza dei biomarcatori come **indicatori di malattia** e la necessità di identificarli con precisione per un trattamento mirato. Nel contesto della ricerca sui biomarcatori, i dati di espressione genica e proteica forniscono un'enorme quantità di informazioni sul **profilo genico** dei pazienti.

In questa tesi, vengono descritti i metodi e le tecniche utilizzate per **raccogliere, preparare e analizzare** i dati di espressione genica dal **GDC**. I criteri di selezione dei dati, la normalizzazione e le analisi statistico-bioinformatiche vengono discussi in dettaglio. I risultati delle analisi rivelano una serie di biomarcatori potenziali correlati a malattie specifiche, dimostrando l'utilità dei dati di espressione genica e proteica per identificare **segnali biologici rilevanti**.

La discussione si concentra sull'interpretazione dei biomarcatori identificati e sulle loro implicazioni per la **medicina di precisione**. Sono affrontate le limitazioni dello studio, tra cui la necessità di **ulteriori validazioni sperimentali**, e sono delineate le direzioni future della ricerca nel campo dei biomarcatori correlati alle malattie.

In conclusione, questa tesi fornisce una panoramica approfondita sull'identificazione di biomarcatori utilizzando dati di espressione genica dal GDC e dimostra il loro potenziale contributo alla **comprensione e al trattamento delle malattie**. Questo lavoro costituisce una panoramica di una medicina personalizzata più efficace e basata sull'evidenza

Capitolo 1: Introduzione

Nel corso delle ultime due decadi, l'avvento delle tecnologie di **sequenziamento ad alto rendimento** ha rivoluzionato la ricerca in campo biomedico e ha aperto nuove prospettive per la comprensione delle malattie a livello molecolare. La medicina di precisione, un approccio basato **sull'individuo alla diagnosi** e al trattamento delle malattie, è emersa come uno dei risultati più promettenti di questa rivoluzione scientifica. Un pilastro fondamentale della medicina di precisione è l'identificazione e l'utilizzo di **biomarcatori**, indicatori biologici che consentono una diagnosi più accurata, una prognosi più affidabile e una terapia mirata.

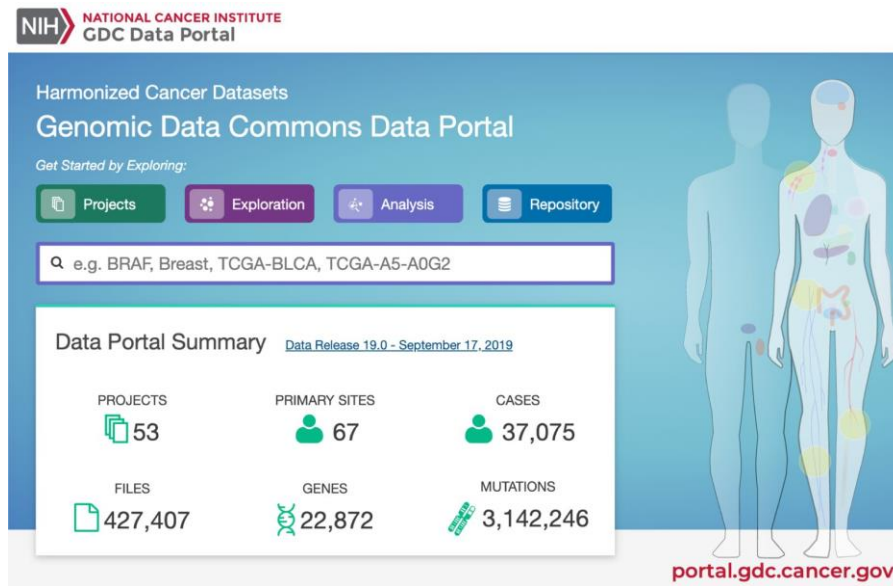
I **dati di espressione genica e proteica** rappresentano una risorsa straordinaria per l'identificazione dei biomarcatori, poiché riflettono direttamente l'attività genica all'interno delle cellule e dei tessuti. Questi dati offrono una finestra senza precedenti sulla **complessità dei profili genici dei pazienti**, consentendo la ricerca di modelli distintivi associati a specifiche condizioni patologiche.

1.1 Contesto e Motivazione

La comprensione delle malattie a livello molecolare è fondamentale per lo sviluppo di nuovi approcci terapeutici e diagnostici. I dati di espressione genica, che riflettono l'attività dei geni all'interno delle **cellule e dei tessuti**, forniscono un'opportunità senza precedenti per acquisire una visione dettagliata delle basi biologiche delle malattie.

Il **Genomic Data Commons (GDC)** è diventato un'importante fonte di dati genomici, fornendo accesso a una vasta raccolta di dati di espressione genica provenienti da una varietà di progetti di ricerca e fonti. Questa piattaforma offre l'opportunità di sfruttare appieno il potenziale dei dati di espressione genica e proteica per l'identificazione dei biomarcatori correlati alle malattie.

Come si presenta il portale dati



1.2 Scopo e Obiettivi della Tesi

Questa tesi si pone l'obiettivo di **esplorare** il ricco terreno dei dati di espressione genica e proteica disponibili nel GDC al fine di identificare biomarcatori associati a malattie specifiche. Attraverso una combinazione di metodologie bioinformatiche e analisi statistiche, questo studio si propone di individuare segnali biologici distintivi e rilevanti per la diagnosi e il trattamento delle malattie.

Nel corso delle prossime pagine, esamineremo in dettaglio i metodi e le analisi utilizzate per raggiungere questo obiettivo, presenteremo i risultati delle nostre indagini e discuteremo delle implicazioni delle scoperte per la medicina di precisione e la ricerca sulle malattie. Questo lavoro rappresenta un contributo alla comprensione dei biomarcatori correlati alle malattie e al **potenziale impatto positivo** che possono avere sulla salute umana.

Capitolo 2: Revisione e Background

La ricerca dei biomarcatori correlati alle malattie rappresenta un campo di crescente interesse nella biomedicina, con implicazioni significative per la **diagnosi precoce, la prognosi accurata e la terapia mirata**. Questo capitolo offre una revisione dei dati e delle informazioni sulla ricerca dei biomarcatori, concentrandosi sulle **metodologie utilizzate, le scoperte chiave e le sfide in questo ambito**. In particolare, esploreremo come i dati di espressione genica e proteica provenienti dal Genomic Data Commons (GDC) abbiano contribuito a questa ricerca.

2.1 Importanza dei Biomarcatori Correlati alle Malattie

L'identificazione dei biomarcatori correlati alle malattie svolge un ruolo cruciale in una serie di contesti clinici e di ricerca:

- **Diagnosi;** I biomarcatori possono essere utilizzati per diagnosticare precocemente malattie specifiche, consentendo interventi terapeutici tempestivi.
- **Prognosi;** I biomarcatori possono fornire informazioni sulla progressione e l'out come delle malattie, aiutando a guidare le decisioni terapeutiche.
- **Terapia Personalizzata;** La presenza di biomarcatori specifici può indicare quale terapia è più probabile essere efficace per un paziente individuale, contribuendo alla medicina di precisione.
- **Ricerca sulle Malattie;** L'identificazione di biomarcatori fornisce nuove intuizioni sulla biologia delle malattie e può aprire nuove strade per lo sviluppo di terapie.

2.2 Fonti di Dati Genomici e GDC

Il principale scopo di questa analisi è creare un database **robusto e ben strutturato** in grado di archiviare, gestire ed eseguire analisi sui dati genomici provenienti dalla piattaforma GDC (Genomic Data Commons). GDC rappresenta una fonte inestimabile di informazioni genetiche, inclusi dati di **espressione genica, sequenziamento genomico, copy number variation (CNV) e molto altro**. La nostra analisi si concentra su come raccogliere, organizzare e sfruttare questi dati in modo efficiente per scopi di ricerca scientifica.

GDC fornisce un vasto archivio di dati genomici da progetti di ricerca di tutto il mondo. Tuttavia, per sfruttare appieno queste risorse, è essenziale avere un sistema che **semplifichi l'accesso e la ricerca dei dati rilevanti**. Il nostro database è stato creato per colmare questa esigenza, consentendo agli utenti di **accedere rapidamente** ai dati di loro interesse.

La diversità dei dati presenti su GDC richiede un sistema di gestione che consenta l'integrazione di dati di diversi **tipi e formati**. Il nostro database offre un'infrastruttura che facilita **l'analisi dei dati e la creazione di connessioni** tra le diverse informazioni genetiche. Ciò rende possibile l'analisi avanzata e l'identificazione di modelli o correlazioni.

2.3 Metodologie di Identificazione dei Biomarcatori

Nella ricerca dei biomarcatori correlati alle malattie, vengono utilizzate diverse metodologie e approcci:

- **Analisi di Differenza di Espressione:** Questa tecnica confronta i profili di espressione genica tra gruppi di pazienti con e senza una determinata malattia per identificare geni differenzialmente espressi.
- **Clustering e Classificazione:** L'analisi di clustering può rivelare sottotipi di malattie basati su pattern di espressione genica simili. Algoritmi di classificazione possono essere addestrati per predire la presenza di una malattia basandosi sui dati di espressione genica.
- **Analisi di Reti Geniche:** Questo approccio identifica reti di geni coinvolti in processi biologici chiave correlati alla malattia.

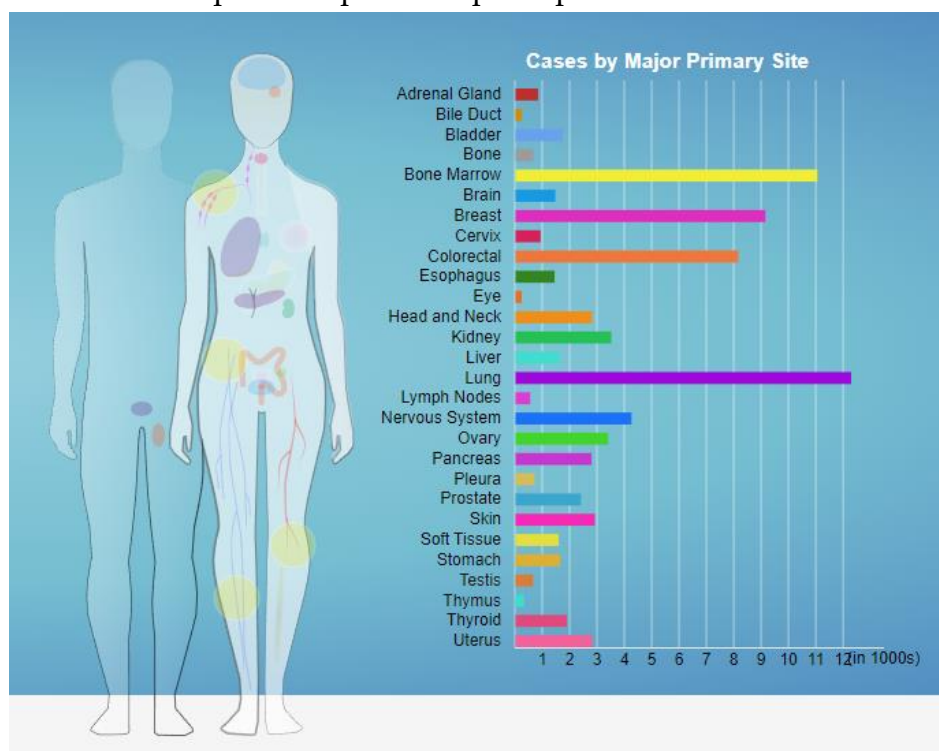
2.4 Scoperte e Sfide nella Ricerca dei Biomarcatori

La letteratura esistente ha evidenziato numerose scoperte significative nella ricerca dei biomarcatori correlati alle malattie. Tuttavia, sono presenti alcune sfide:

- **Validazione:** La validazione sperimentale dei biomarcatori è spesso necessaria per confermare le scoperte e garantire l'affidabilità.
- **Variabilità Biologica:** La biologia umana è complessa, e la variabilità biologica può influenzare l'identificazione dei biomarcatori.
- **Dimensionalità Elevata:** I dati di espressione genica possono avere dimensioni elevate, il che richiede metodi sofisticati per l'analisi.

In questo contesto, questa tesi si propone di utilizzare i dati di espressione genica e proteica dal GDC per identificare biomarcatori correlati a malattie specifiche, quali i 68 siti più comuni dove si può contrarre un tumore, contribuendo così ad un aiuto per future analisi in merito.

Casi per sede primaria principale della malattia

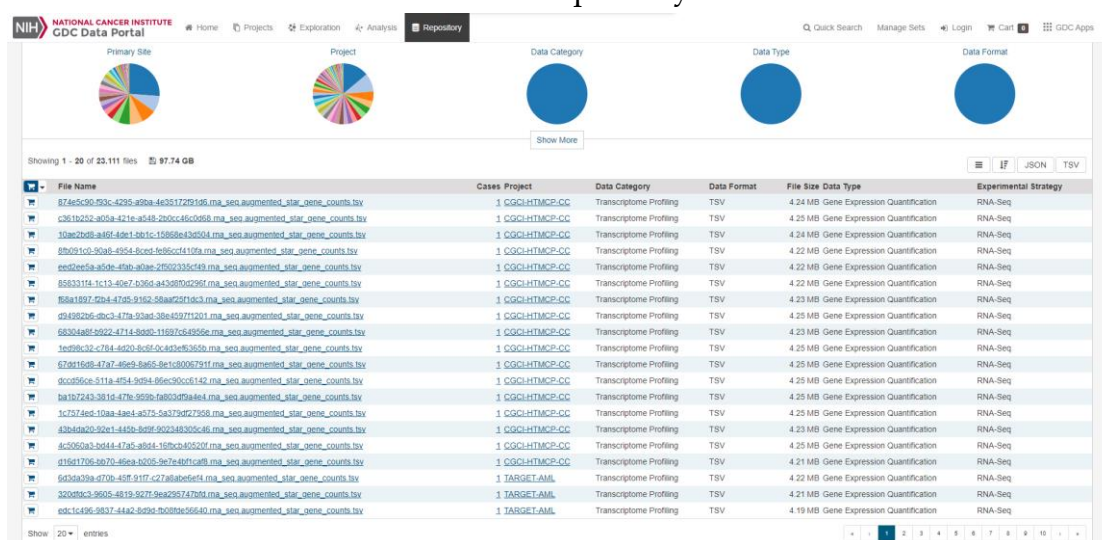


2.5 Panoramica del database e strumenti utilizzati

I primi passi per creare un database solido sulla base del GDC sono comprendere in primis quali dati contiene e quali sono utili al fine della ricerca. Facciamo una **panoramica** del tutto:

Tutta la ricerca ed a sua volta la base di dati ruotano attorno ad un componente principale, delle **analisi** archiviate in file .tsv detti **Gene Expression File**. Un Gene Expression File (File di Espressione Genica) è un insieme strutturato di dati che rappresenta le quantità di espressione genica di specifici geni in **campioni biologici**. Questi file sono utilizzati per studiare il livello di attività genica in vari contesti, tra cui studi di malattie, sviluppo, risposte cellulari e altro.

Panoramica del repository dei dati



Struttura del File:

- **Identificatore del File:** Ogni Gene Expression File ha un identificatore univoco che consente di riferirsi in modo univoco a un insieme specifico di dati di espressione genica.
- **Informazioni del Campione:** Il file contiene informazioni sui campioni biologici dai quali sono stati estratti i dati di espressione genica. Queste informazioni possono includere dettagli sul tipo di campione, sulla fonte biologica, sulla malattia associata e su altri metadati rilevanti.

Informazioni interne al file

File Properties

Name	874ec90c-f93c-4290-a5ba-4c35172f91d5.ma_seq.augmented_star_gene_counts.tsv
Access	open
UUID	56450c13-a0bb-48c3-9e84-c90905ed597b
Data Format	TSV
Size	4.24 MB
MD5 Checksum	c054d9abc17c21920a6f0ad53f3234
Archive	--
Project	CGCL-HTMCP-QC

Showing 1 - 1 of 1 associated cases/biospecimen

Associated Cases/Biospecimen

Q

Entity ID

eg. YCBA-13*, *13*, *09

Entity ID	Entity Type	Sample Type	Case UUID	Annotations
HTMCP-03-06-02266-01A-01R-9005	aliquot	Primary Tumor	37941ba3-0167-4b7b-b609-29621ee854d8	0

Show

10

entries

«

«

1

»

»

Analysis

Analysis ID	f0d36eac-f0b3-49ec-b900-9dbd276b6499
Workflow Type	STAR - Counts
Workflow Completion Date	2021-12-22
Source Files	1

Reference Genome

Genome Build	GRCh38.p0
Genome Name	GRCh38.d1.vd1

- **Tabella dei Dati di Espressione:** Questa è la parte centrale del file e consiste in una tabella in cui le righe rappresentano i geni specifici e le colonne contengono le quantità di espressione (ad esempio, TPM - Transcripts Per Million, FPKM - Fragments Per Kilobase per Million) associate a ciascun gene nei campioni.
- **Metadati Aggiuntivi:** Il file può contenere metadati aggiuntivi, come le annotazioni dei geni, informazioni sull'analisi, i tipi di misurazioni di espressione e altro.

Un esempio di file gene_expression_quantification.tsv

# gene-model: GENCODE v36								
gene_id	gene_name	gene_type	unstranded	stranded_first	stranded_second	tpm_unstranded	fpkm_unstranded	fpkm_uq_unstranded
ENSG00000000003.15	TSPAN6	protein_coding	6082	2	6080	55.4834	16.1430	15.8165
ENSG00000000005.6	TNMD	protein_coding	5	0	5	0.1402	0.0408	0.0400
ENSG000000000419.13	DPM1	protein_coding	1543	21	1522	52.8991	15.3911	15.0798
ENSG000000000457.14	SCYL3	protein_coding	2698	1583	2449	16.2281	4.7193	4.6238
ENSG000000000460.17	C1orf112	protein_coding	2244	619	2994	15.5539	4.5254	4.4339
ENSG000000000938.13	FGR	protein_coding	783	2	781	9.5803	2.7874	2.7310
ENSG000000000971.16	CFH	protein_coding	13922	5	13919	72.1919	21.0044	20.5795
ENSG00000001036.14	FUCA2	protein_coding	3723	37	6368	54.5916	15.8836	15.5622
ENSG00000001084.13	GLCLC	protein_coding	6633	1	7821	31.8488	9.2665	9.0790
ENSG00000001167.14	NFYA	protein_coding	4624	5	5184	50.2075	14.6080	14.3125
ENSG00000001460.18	STPG1	protein_coding	1661	41	1658	8.0757	2.3496	2.3021
ENSG00000001461.17	NIPAL3	protein_coding	7571	8	7599	33.3426	9.7011	9.5049
ENSG00000001497.18	LAS1L	protein_coding	4857	6	4885	16.0081	4.6576	4.5634
ENSG00000001561.7	ENPP4	protein_coding	2090	1	2089	18.6227	5.4183	5.3087
ENSG00000001617.12	SEMA3F	protein_coding	22324	9	22347	191.4142	55.6925	54.5659
ENSG00000001626.16	CFTR	protein_coding	509	19	500	2.1185	0.6164	0.6039
ENSG00000001629.10	ANKIB1	protein_coding	4510	7	4520	25.2569	7.3486	7.1999
ENSG00000001630.17	CYP51A1	protein_coding	48	1	49	0.5628	0.1638	0.1604
ENSG00000001631.16	KRIT1	protein_coding	544	15	544	3.6284	1.0557	1.0343
ENSG00000002016.18	RAD52	protein_coding	1290	160	1130	11.8543	3.4491	3.3793
ENSG00000002330.14	BAD	protein_coding	598	48	2145	14.4878	4.2153	4.1300
ENSG00000002549.12	LAP3	protein_coding	3268	1	3267	35.7277	10.3951	10.1848
ENSG00000002586.20	CD99	protein_coding	16559	16	16547	141.0477	41.0382	40.2080
ENSG00000002587.10	HS3ST1	protein_coding	270	7	263	1.5027	0.4372	0.4284
ENSG00000002726.21	AOC1	protein_coding	7641	65	7576	83.5140	24.2986	23.8071
ENSG00000002745.13	WNT16	protein_coding	33	11	22	0.4184	0.1217	0.1193
ENSG00000002746.15	HECW1	protein_coding	109	5	113	0.3237	0.0942	0.0923
ENSG00000002822.15	MAD1L1	protein_coding	37	0	37	0.2118	0.0616	0.0604
ENSG00000002834.18	LASP1	protein_coding	23157	15	27179	135.7080	39.4846	38.6859
ENSG00000002919.15	SNX11	protein_coding	2646	43	2686	26.4919	7.7079	7.5520
ENSG00000002933.9	TNEM176A	protein_coding	3513	72	3486	41.8083	12.1642	11.9182
ENSG00000003056.8	MEPR	protein_coding	6722	163	8155	77.5888	22.5747	22.1180
ENSG00000003066.14	KLHL13	protein_coding	2085	0	2085	12.3553	3.5948	3.5221
ENSG00000003137.8	CYP26B1	protein_coding	621	0	621	5.1240	1.4908	1.4607
ENSG00000003147.19	ICAI1	protein_coding	1980	48	2167	16.7791	4.8819	4.7832
ENSG00000003249.15	DBND01	protein_coding	1046	14	1045	9.7004	2.8224	2.7653
ENSG00000003393.15	ALS2	protein_coding	1983	0	1984	7.8613	2.2873	2.2410
ENSG00000003400.15	CASP10	protein_coding	1721	2	1719	9.3939	2.7332	2.6779
ENSG00000003402.20	CFLAR	protein_coding	8008	16	8797	14.7413	4.2890	4.2023
ENSG00000003436.16	TFPI	protein_coding	479	9	484	1.6921	0.4923	0.4824
ENSG00000003509.16	NDUFAF7	protein_coding	1430	639	1445	16.2074	4.7156	4.6202
ENSG00000003756.17	RBMS	protein_coding	8443	13	10960	33.7328	9.8146	9.6161
ENSG00000003987.14	MTMR7	protein_coding	289	536	178	1.7262	0.5022	0.4921
ENSG00000003989.18	SLC7A2	protein_coding	5181	1	5180	27.1207	7.8908	7.7312
ENSG00000004059.11	ARF5	protein_coding	7666	6	7856	173.8183	50.5729	49.5498
ENSG00000004139.14	SARM1	protein_coding	1085	4048	3425	4.0430	1.1763	1.1525
ENSG00000004142.12	POLDIP2	protein_coding	9324	553	9347	144.5042	42.0439	41.1934
ENSG00000004399.13	PLXND1	protein_coding	10787	3	10784	46.7056	13.5891	13.3142

Importanza:

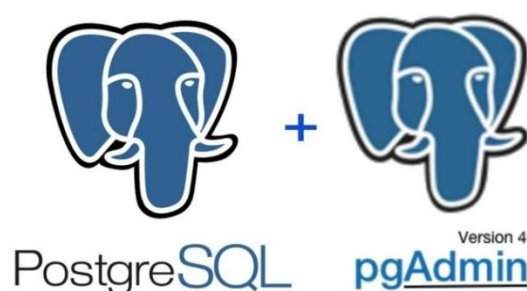
- **Analisi della Regolazione Genica:** I Gene Expression Files sono essenziali per comprendere come i geni sono regolati in risposta a diversi stimoli, condizioni o malattie. Consentono di identificare i geni che sono sovraespressi o sottoregolati in un determinato contesto biologico.
- **Identificazione di Biomarcatori:** Sono ampiamente utilizzati nella ricerca di biomarcatori, che possono essere indicatori diagnostici o prognostici utili in campo medico.
- **Studi di Malattie:** Sono fondamentali per studi di malattie, consentendo di identificare i processi biologici coinvolti in condizioni patologiche e potenziali bersagli terapeutici.
- **Personalizzazione della Terapia:** Possono essere utilizzati per personalizzare l'approccio terapeutico in base al profilo di espressione genica di un paziente.

In questo caso verranno usati nel contesto di identificazione di Biomarcatori costruendo dai dati raccolti degli **alberi decisionali**.

Per la realizzazione di quanto detto, è stato essenziale avere una buona conoscenza dei linguaggi **Python** e **SQL**. Inoltre, è stato fondamentale avere una metodica conoscenza delle procedure di creazione di un database, ossia seguire le tecniche di progettazione **concettuale, logica e fisica**.



Per la creazione della base di dati ho utilizzato **PostgreSQL** con l'interfaccia utente **pgAdmin**.



È stata necessaria la conoscenza teorica appresa nei corsi di Basi di Dati per la realizzazione del database locale, per la creazione e gestione delle varie **tabelle** e soprattutto per la realizzazione di **query** con cui acquisire i dati necessari per le varie **sperimentazioni**.

Vedremo successivamente tutto il processo di creazione del database con focus su ogni aspetto.

Ed è stata necessaria, inoltre, la conoscenza pratica appresa nel corso di Fondamenti di Programmazione per la realizzazione dello **script di download**, per la familiarità con i **moduli** di Python e per la creazione delle analisi sui dati mediante **alberi decisionali**.

Grazie a Python, ho creato degli script ben strutturati per:

- **Download dei dati;**

Avendo implementato vari script per il download di grandissime quantità di dati; ogni file di analisi ha un peso di 4.24 MB, il database è stato pensato per contenere 100 GB di dati.

- **Gestione dei dati;**

Talvolta i dati scaricati dall'API presentavano informazioni non pertinenti al fine del progetto o disposte in modo confusionario.

È stata essenziale la conoscenza dei moduli JSON e Request per richiedere ed organizzare i dati dalla repository online con precisione ed accuratezza, inoltre una buona dimestichezza nella lettura e gestione del file si è rivelata utile per riordinare le informazioni al loro interno e facilitare l'inserimento nella base di dati.

- **Interazione con il database locale;**

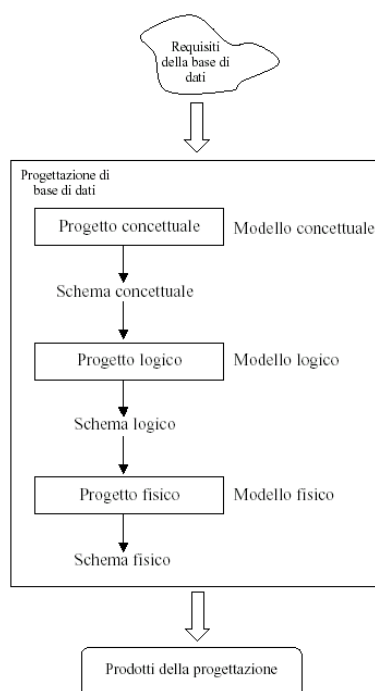
Per gestire tutti gli inserimenti e le successive analisi è stato necessario imparare a conoscere il modulo psycopg2 che mi ha permesso di utilizzare un cursore per realizzare query di inserimento e interrogazione al database.

Capitolo 3: Metodologia ed Implementazione

Nel contesto di questo lavoro di ricerca questo capitolo rappresenta una fase fondamentale dell'intero processo. In questa sezione, vengono descritte le metodologie e le procedure utilizzate per **creare** la base di dati, **gestire** i dati scaricati dalla piattaforma GDC (Genomic Data Commons), e **sviluppare** gli alberi decisionali per l'analisi di espressione genica.

L'obiettivo di questo capitolo è fornire una panoramica dettagliata delle fasi chiave del processo di creazione del database, **dalla raccolta dei dati alla loro elaborazione**. Inoltre, saranno descritte le **strategie adottate** per l'estrazione e la gestione delle informazioni dai file di espressione genica e la costruzione di modelli di alberi decisionali. La metodologia adottata rappresenta **una guida completa e riproducibile** per la creazione di un database genomico e per l'analisi dei dati.

Fasi della progettazione di una base di dati



Voglio quindi fornire un'ampia visione delle metodologie utilizzate nel processo di ricerca, in modo da rendere **chiara e replicabile** l'intera procedura. La metodologia descritta sarà il fondamento su cui si baseranno tutte le **conclusioni e le discussioni** presentate nei capitoli successivi, contribuendo a garantire **l'integrità e la validità** delle analisi condotte.

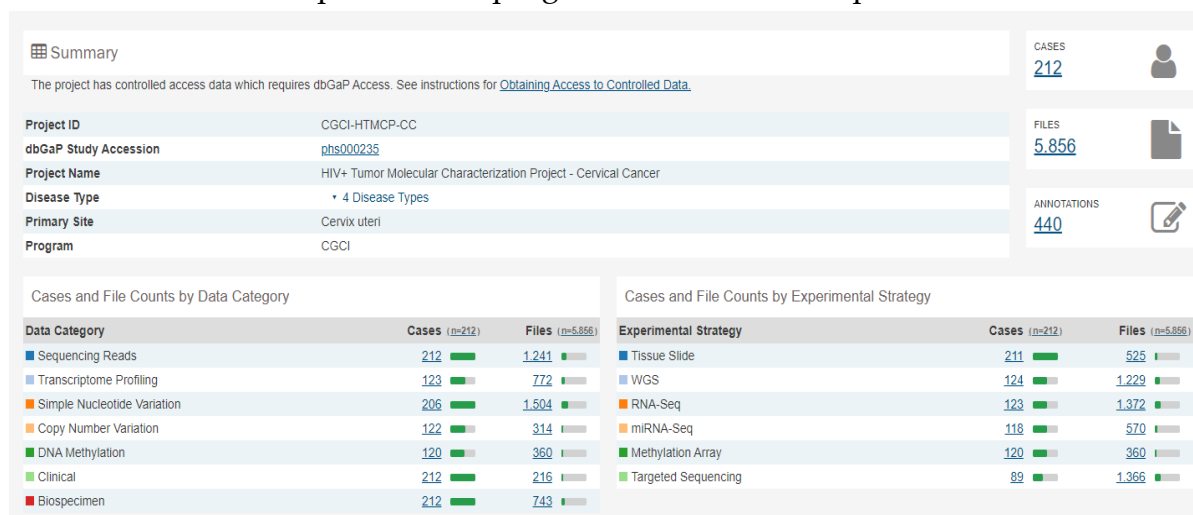
3.1 Creazione della Base di Dati

Questa sezione si concentra sulla creazione della base di dati che servirà come fondamento per tutte le successive analisi. Verranno descritti i passaggi per l'organizzazione e l'archiviazione dei dati provenienti dalla piattaforma GDC, inclusi i dettagli sulla **struttura del database, la progettazione delle tabelle e la definizione dei campi**. La relazione della progettazione è stata realizzata seguendo quanto imparato dai corsi di Basi di Dati e quindi seguendo tutti i procedimenti teorici richiesti.

3.1.1 Progettazione Concettuale

Nel presente capitolo, esploreremo il processo di progettazione concettuale di una base di dati destinata a ospitare dati biologici, in particolare i dati di espressione genica raccolti dal Genomic Data Commons (GDC). Con un focus particolare sugli **Expression Files**, analizzeremo come i dati sono strutturati e come possono essere modellati concettualmente in una base di dati.

Come si presenta un progetto all'interno della piattaforma



Nel contesto della creazione di una base di dati basata sul GDC, la progettazione concettuale è il **fondamento** su cui verranno costruite le fasi successive, inclusa la progettazione **logica e fisica**. Quindi, procediamo con attenzione e precisione in questo capitolo, poiché una progettazione concettuale accurata è essenziale per il successo del progetto nel suo complesso.

Richiesta dei Requisiti

Si vuole realizzare una base di dati per analizzare la mole di dati derivanti dai maggiori siti di biomedicina riguardanti la “gene expression” e la “protein expression”.

Tutto il database è diviso in **progetti** (Project) che hanno un codice univoco ed un nome, i progetti contengono tutti le **analisi** (Analysis) sotto forma di file e i **casi** (Case) che ne fanno parte

Di un caso sappiamo l'id, il **tipo di malattia** ed il **sito primario** dove risiede la malattia. Da questo paziente (che può avere **un'anagrafica**) scaturiscono più **campioni biologici** (Biospecimen). Questi prelievi primari, chiamati **sample**, possono essere trattati e resi **porzioni** (Portion), quest'ultime possono diventare **analiti** (Analyte) che infine con determinati trattamenti diventano **aliquote** (Aliquot)

Per i campioni (Sample) identificati da un id, vogliamo sapere **il tipo del campione** e di che **tumore** si tratta. Per i derivati di ciascuno avremo i loro identificatori e per gli analiti e le aliquote anche la **concentrazione** nel campione.

Rappresentiamo quindi l'analisi sotto forma di file, dove è contenuto un determinato insieme di dati di espressione presi da **uno o più** biospecie e per ognuno di essi da chi proviene. Le analisi hanno: un codice, un nome, la **categoria dei dati**, il peso del file, la data di creazione del file, la data di ultimo aggiornamento del file, **il tipo di dati ed una strategia sperimentale**

Per le due espressioni (geni e proteine), contenute in più file e identificate da un codice univoco, si vuole memorizzare il nome e le misurazioni che ne derivano.

Per scelta progettuale inoltre:

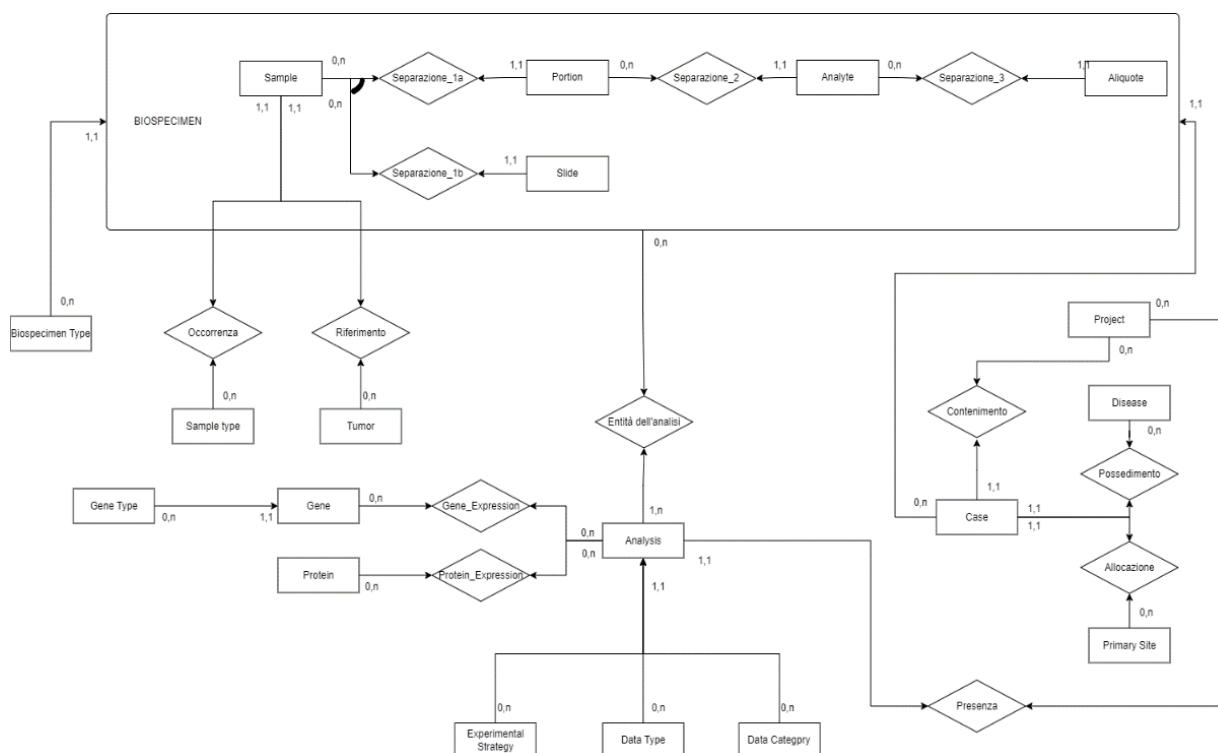
- Decidiamo di creare un'entità **Tipo del campione/tumore/malattia** così da differenziarle con un codice univoco
- Decidiamo di creare tre entità: Experimental Strategy, Data Type e Data Category per prevedere **futuri ampliamenti** di database

Modello Concettuale

Un modello concettuale è una rappresentazione astratta dei dati, delle entità, delle relazioni e delle regole che governano un sistema o un dominio specifico, indipendentemente dai dettagli di implementazione tecnica. Si concentra **sull'organizzazione logica** dei dati piuttosto che sui dettagli di come vengono memorizzati o elaborati fisicamente.

Il modello concettuale è uno strumento fondamentale nella progettazione dei database, poiché aiuta a definire in modo chiaro e comprensibile la struttura dei dati e le relazioni tra le diverse entità, consentendo la creazione di una base **solida** per la progettazione del database fisico. Inoltre, un modello concettuale fornisce una base comune per la **comunicazione tra i membri del gruppo di sviluppo**, gli stakeholder e gli utenti finali, contribuendo a garantire una comprensione condivisa e una progettazione accurata del sistema informativo.

Il modello concettuale proposto rappresenta una struttura dati per la gestione e l'analisi di informazioni biomediche relative alla "gene expression" e alla "protein expression". Questo modello è **organizzato in modo gerarchico**, con una chiara **struttura a livelli**, che parte dai progetti biomedici come entità di alto livello e scende fino ai **dettagli** delle analisi e delle espressioni genetiche e proteiche.



Glossario dei Termini

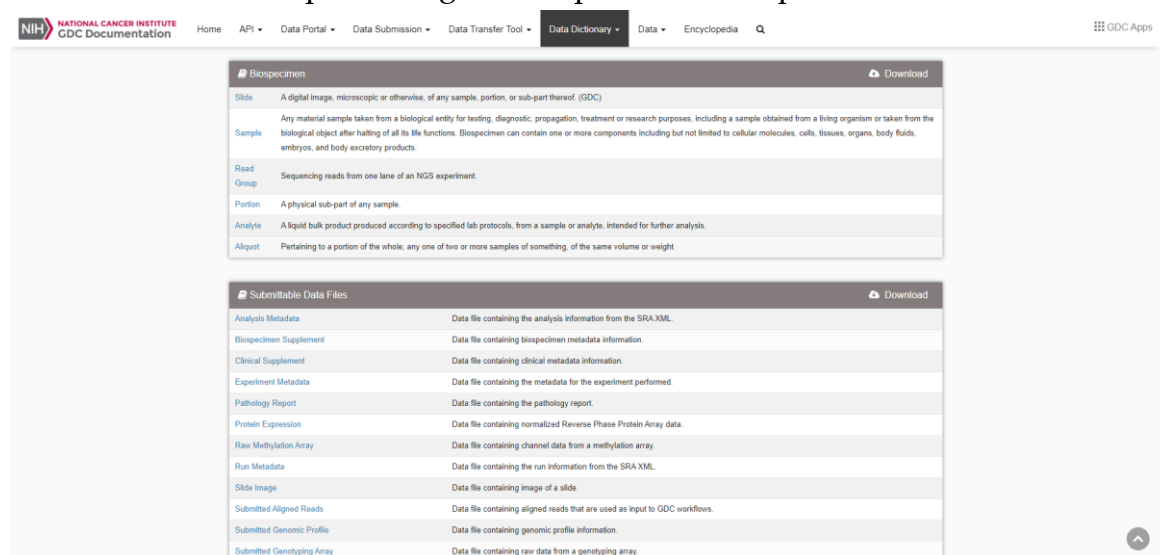
Il glossario dei termini è un elenco di definizioni chiare e concise dei principali termini e concetti utilizzati nel contesto del database biomedico.

Questo glossario è stato creato per offrire una **risorsa di riferimento** rapida e utile per chiunque stia lavorando con il database o stia cercando di comprendere meglio le informazioni biomediche.

Fornisce spiegazioni **chiare dei termini**, consentendo agli utenti di familiarizzare con il linguaggio specifico utilizzato nel database e di interpretare correttamente i dati e le informazioni contenuti all'interno del sistema.

Le definizioni nel glossario sono state create in modo da essere facilmente comprensibili anche da parte di coloro che potrebbero **non avere una conoscenza approfondita del campo della biomedicina**, contribuendo così a rendere accessibili e utili le informazioni contenute nel database a un pubblico più ampio.

Una parte del glossario presente sulla piattaforma




Biospecimen		Download
Slide	A digital image, microscopic or otherwise, of any sample, portion, or sub-part thereof. (GDC)	
Sample	Any material sample taken from a biological entity for testing, diagnostic, propagation, treatment or research purposes, including a sample obtained from a living organism or taken from the biological object after halting of all its life functions. Biospecimen can contain one or more components including but not limited to cellular molecules, cells, tissues, organs, body fluids, embryos, and body excretory products.	
Read	Sequencing reads from one lane of an NGS experiment.	
Group		
Portion	A physical sub-part of any sample.	
Analyte	A liquid bulk product produced according to specified lab protocols, from a sample or analyte, intended for further analysis.	
Aliquot	Pertaining to a portion of the whole; any one of two or more samples of something, of the same volume or weight.	
Submittable Data Files		Download
Analysis Metadata	Data file containing the analysis information from the SRA XML.	
Biospecimen Supplement	Data file containing biospecimen metadata information.	
Clinical Supplement	Data file containing clinical metadata information.	
Experiment Metadata	Data file containing the metadata for the experiment performed.	
Pathology Report	Data file containing the pathology report.	
Protein Expression	Data file containing normalized Reverse Phase Protein Array data.	
Raw Methylation Array	Data file containing channel data from a methylation array.	
Run Metadata	Data file containing the run information from the SRA XML.	
Slide Image	Data file containing image of a slide.	
Submitted Aligned Reads	Data file containing aligned reads that are used as input to GDC workflows.	
Submitted Genomic Profile	Data file containing genomic profile information.	
Submitted Genotyping Array	Data file containing raw data from a genotyping array.	

TERMINE	DESCRIZIONE	SINONIMI	COLLEGAMENTI
PROJECT	Questa entità rappresenta un progetto biomedico con un codice univoco e un nome. Può contenere una serie di analisi, casi, e altre informazioni specifiche del progetto.	Progetto	Case, Analysis
CASE	La raccolta di tutti i dati relativi ad un soggetto specifico nel contesto di un progetto specifico.	Paziente, Caso	Project, Biospecimen, Disease, Primary Site
PRIMARY SITE	Contiene i dati sul sito primario di analisi del caso preso in questione	Sito, Locazione	Case
DISEASE	Contiene i dati sulla malattia del caso preso in questione	Malattia	Case
BIOSPECIMEN	Qualsiasi campione materiale prelevato da un'entità biologica a fini di analisi, diagnosi, propagazione, trattamento o ricerca, compreso un campione ottenuto da un organismo vivente o prelevato dall'oggetto biologico dopo l'arresto di tutte le sue funzioni vitali. Il campione biologico può contenere uno o più componenti inclusi, ma non limitati a, molecole cellulari, cellule, tessuti, organi, fluidi corporei, embrioni e prodotti escretori corporei.	Campione, Porzione, Analita, Aliquota, Entità	Case, File, Biospecimen Type
ANALYSIS	Rappresenta i file che contengono dati di espressione genica o proteica. Contiene informazione come il nome del file, il suo submitter, la dimensione del file e altro. È collegato a campioni biologici o altri elementi	Insieme di espressioni, File	Project, Biospecimen, Gene, Protein
GENE	Questa entità rappresenta i dati di un gene specifico. Contiene informazioni sul gene, come il suo ID e il nome.	Espressione	File
PROTEIN	Questa entità rappresenta i dati di una proteina specifica. Contiene informazioni sul gene, come il suo ID e il nome.	Espressione	File
BIOSPECIMEN TYPE	Tipologia di un Campione Biologico: Sample, Portion, Analyte o Aliquot	Tipo di Campione	Biospecimen

Dizionario dei Dati (Entità)

Il Dizionario dei Dati, nell'ambito del database biomedico, è un documento o una risorsa che elenca e **descrive in dettaglio** le diverse entità, tabelle o categorie di dati presenti nel sistema.

Esempio di descrizione di un'entità nella piattaforma

 Data Dictionary Viewer





[Dictionary Viewer](#) > Sample [Print](#)

Sample [Download Template](#) [TSV](#)

Summary

Type	sample
Category	biospecimen
Description	Any material sample taken from a biological entity for testing, diagnostic, propagation, treatment or research purposes, including a sample obtained from a living organism or taken from the biological object after halting of all its life functions. Biospecimen can contain one or more components including but not limited to cellular molecules, cells, tissues, organs, body fluids, embryos, and body excretory products.
Unique Keys	<ul style="list-style-type: none">idproject_id, submitter_id

Links

Links to Entity	Link Name	Relationship	Required?
 Case	cases	Samples Derived From Case	Yes
 Diagnosis	diagnoses	Samples Related To Diagnosis	No
 Sample	parent_samples	Child Samples Derived From Sample	No
 Tissue Source Site	tissue_source_sites	Samples Processed At Tissue Source Site	No

Questo dizionario fornisce informazioni chiare e complete su ciascuna entità, compresi i **campi o attributi associati**, i **tipi di dati**, le **relazioni con altre entità** e una spiegazione del significato e dell'utilizzo di ciascun attributo.

ENTITÀ	DESCRIZIONE	ATTRIBUTI	IDENTIFICATORE
PROJECT	Un progetto biomedico con un codice univoco e un nome. Può contenere una serie di file, casi, e altre informazioni specifiche del progetto.	project_id, name	project_id
CASE	La raccolta di tutti i dati relativi ad un soggetto specifico nel contesto di un progetto specifico.	case_id, ethnicity, gender, race, vital_status	case_id
BIOSPECIMEN	Qualsiasi campione materiale prelevato da un'entità biologica a fini di analisi, diagnosi, propagazione, trattamento o ricerca, compreso un campione ottenuto da un organismo vivente o prelevato dall'oggetto biologico dopo l'arresto delle funzioni vitali.	id	id
DISEASE	Contiene i dati sulla malattia del caso preso in questione	id, type	id
PRIMARY SITE	Contiene i dati sul sito preso in analisi del caso in questione	id, site	id
SAMPLE IS-A BIOSPECIMEN	Rappresenta un campione biologico specifico, identificato da un ID univoco. Contiene informazioni sul tipo di campione e il tipo di tumore, se applicabile. È collegato a un caso o paziente.	(Ereditati da Biospecimen)	biospecimen.id
SAMPLE TYPE	Tipologia di un campione	type_id, type	type_id
TUMOR	Contiene i dati sul tumore del campione preso in questione	tumor_code_id, code, descriptor	tumor_code_id
BIOSPECIMEN TYPE	Tipologia di un Campione Biologico: Sample, Slide, Portion, Analyte o Aliquot	id, type	id

PORTION IS-A BIOSPECIMEN	Una sottoparte fisica di qualsiasi campione. Contiene informazioni sulla porzione del campione, ma può essere collegato ad analiti più piccoli.	(Ereditati da Biospecimen)	biospecimen.id
ANALYTE IS-A BIOSPECIMEN	Un prodotto sfuso liquido prodotto secondo protocolli di laboratorio specificati, da una porzione, destinato a ulteriori analisi.	(Ereditati da Biospecimen), concentration	biospecimen.id
ALIQUEOT IS-A BIOSPECIMEN	Una piccola porzione di un analita con una concentrazione specifica. Contiene informazioni sulla concentrazione dell'aliquota.	(Ereditati da Biospecimen), concentration	biospecimen.id
ANALYSIS	File che contengono dati di espressione genica o proteica. Contiene informazione come il nome del file, il suo submitter, la dimensione del file e altro. È collegato a campioni biologici o altri elementi	id, filename, file_size, created_date, updated_date	id
EXPERIMENTAL STRATEGY	Contiene i dati sulla strategia sperimentale usata nell'analisi	strategy_id, strategy	strategy_id
DATA TYPE	Contiene i dati sul tipo di dati nel file dell'analisi	type_id, type	type_id
DATA CATEGORY	Contiene i dati sulla categoria di dati nel file dell'analisi	category_id, category	category_id
GENE	Dati di un gene specifico. Contiene informazioni sul gene, come il suo ID e il nome.	gene_id, gene_name	gene_id
GENE TYPE	Tipologia di un gene	type_id, type	type_id
PROTEIN	Dati di una proteina specifica. Contiene informazioni sul gene, come il suo ID e il nome.	agid, lab_id, catalog_number, set_id, peptide_target	agid

Dizionario dei Dati (Relazioni)

RELAZIONI	DESCRIZIONE	COMPONENTI	ATTRIBUTI
CONTENIMENTO	Associa un progetto ai casi che contiene	Project, Case	
PRESENZA	Associa un progetto ai file che presenta	Project, File	
POSSEDIMENTO	Associa un caso alla malattia che possiede	Case, Disease	
ALLOCAZIONE	Associa un caso al sito dove si alloca la malattia	Case, Primary Site	
SCATURIRE	Associa dei casi alle biospecie che hanno scaturito	Case, Biospecimen	
OCCORRENZA	Associa un campione biologico al tipo di occorrenza che è	Sample/Analyte/ Aliquot, Type	
RIFERIMENTO	Associa un tumore al campione che fa riferimento	Sample, Tumor	
SEPARAZIONE_1	Associa un campione alle porzioni separate da esso	Sample, Portion	
SEPARAZIONE_2	Associa una porzione agli analiti separati da esso	Portion, Analyte	
SEPARAZIONE_3	Associa un analita alle aliquot separate da esso	Analyte, Aliquot	
ENTITÀ DELL'ANALISI	Associa più campioni biologici ai file dove è contenuto	Biospecimen, File	
GENE EXPRESSION	File di dati contenente informazioni sull'espressione genica.	File, Gene	unstranded, stranded_first, stranded_second tpm, fpkm, fpkm_uq
PROTEIN EXPRESSION	File di dati contenente dati normalizzati dell'array di proteine a fase inversa.	File, Protein	peptide_target, expression

Tavola dei Volumi

La Tabella dei Volumi, in un contesto di database, è una risorsa che fornisce una panoramica delle **dimensioni dei dati** memorizzati all'interno del sistema. Essa elenca le diverse entità o tabelle presenti nel database insieme alle rispettive dimensioni, solitamente espresse in termini di **quantità di record o righe** e dimensione totale in byte o altre unità di misura appropriate.

Una panoramica del volume dei dati nella piattaforma

The screenshot displays the 'Cases' tab of a data platform interface. It features a sidebar with filters and a main content area with search and filter options.

Filters (Left Sidebar):

- Search Files:** Search bar with placeholder text: e.g. 142682.bam, 4f6e2e7a-b...
- Data Category:**
 - ☐ simple nucleotide variation (365.968)
 - ☐ copy number variation (165.903)
 - ☐ sequencing reads (149.745)
 - ☐ structural variation (86.411)
 - ☐ transcriptome profiling (81.096)
 - 6 More...
- Data Type:**
 - ☐ Annotated Somatic Mutation (152.422)
 - ☐ Aligned Reads (149.745)
 - ☐ Raw Simple Somatic Mutation (94.338)
 - ☐ Transcript Fusion (89.543)
 - ☐ Masked Annotated Somatic Mutation (44.755)
 - 25 More...
- Experimental Strategy:**
 - ☐ WXS (265.553)
 - ☐ RNA-Seq (201.394)
 - ☐ Genotyping Array (147.734)
 - ☐ Targeted Sequencing (140.457)
 - ☐ WGS (54.086)
 - 7 More...

Main Content Area (Cases Tab):

- Search Cases:** Search bar with placeholder text: e.g. TCGA-A5-A0G2, 432fe4a9-2...; Upload Case Set button.
- Case ID:** Search bar with placeholder text: eg. TCGA-DD*, *DD*, TCGA-DD-AAVP; Go! button.
- Primary Site:**
 - ☐ bronchus and lung (12.345)
 - ☐ hematopoietic and reticuloendothelial ... (11.376)
 - ☐ breast (9.155)
 - ☐ colon (6.951)
 - ☐ spinal cord, cranial nerves, and other p... (3.703)
 - 63 More...
- Program:**
 - ☐ GENIE (44.756)
 - ☐ FM (18.004)
 - ☐ TCGA (11.428)
 - ☐ TARGET (6.543)
 - ☐ CPTAC (1.577)
 - 18 More...
- Project:**
 - ☐ FM-AD (18.004)
 - ☐ GENIE-MSK (16.824)
 - ☐ GENIE-DFCI (14.232)
 - ☐ GENIE-MDA (3.857)

CONCETTO	TIPO	VOLUME	SPIEGAZIONE DEI VOLUMI
PROJECT	E	100	Scelta basata sulle analisi fatte nella piattaforma
CASE	E	20.000	Scelta basata sulle analisi fatte nella piattaforma
DISEASE	E	200	Scelta basata sulle analisi fatte nella piattaforma
PRIMARY SITE	E	68	Scelta basata sulle analisi fatte nella piattaforma
BIOSPECIMEN	E	600.000	Ogni caso produce circa 30 Biospecie
SAMPLE IS-A BIOSPECIMEN	E	60.000	10% delle Biospecie
SAMPLE TYPE	E	20	Numero dei campionamenti più comuni
TUMOR	E	33	Scelta basata sulle analisi fatte nella piattaforma
PORTION IS-A BIOSPECIMEN	E	90.000	15% delle Biospecie
ANALYTE IS-A BIOSPECIMEN	E	150.000	25% delle Biospecie
ALQUOT IS-A BIOSPECIMEN	E	300.000	50% delle Biospecie
BIOSPECIMEN TYPE	E	4	Sample, Portion, Analyte and Aliquot
ANALYSIS	E	35.000	Scelta basata sulle analisi fatte nella piattaforma (circa 100GB)
EXPERIMENTAL STRATEGY	E	12	Scelta basata sulle analisi fatte nella piattaforma
DATA TYPE	E	30	Scelta basata sulle analisi fatte nella piattaforma
DATA CATEGORY	E	11	Scelta basata sulle analisi fatte nella piattaforma
GENE	E	60.660	Geni presenti in questo tipo di analisi
GENE TYPE	E	50	Numero dei tipi di geni più comuni
PROTEIN	E	500	Proteine presenti in questo tipo di analisi
ENTITÀ DELL'ANALISI	R	35.000	Difficile trovare un'entità che sia presente in più file quindi prevediamo un record per ognuna
GENE EXPRESSION	R	1.500.000.000	60.000 geni presenti * 25.000 file di analisi
PROTEIN EXPRESSION	R	5.000.000	500 proteine presenti * 10.000 file di analisi

Dizionario dei Vincoli Esterni

1. (Biospecimen)
Per aggiungere un record alle tabelle sottostanti campione deve essere presente un record a cui si riferiscono a cascata
2. (Biospecimen)
Se un id è presente già in una tabella sottostante a Biospecimen non può essere in un'altra
3. (Analysis)
La data di creazione di un file non può essere successiva alla data dell'ultima modifica

3.1.2 Ristrutturazione dello Schema Concettuale

Modifiche schema ER

Per la ristrutturazione del diagramma ER ho effettuato le seguenti modifiche:

Entità

- Creazione "Entità del File": dalla risoluzione della relazione n-n (Entità del File)
- Creazione "Gene Expression File" dalla risoluzione della relazione n-n (Gene Expression)
- Creazione "Protein Expression File" dalla risoluzione della relazione n-n (Protein Expression)

Relazioni

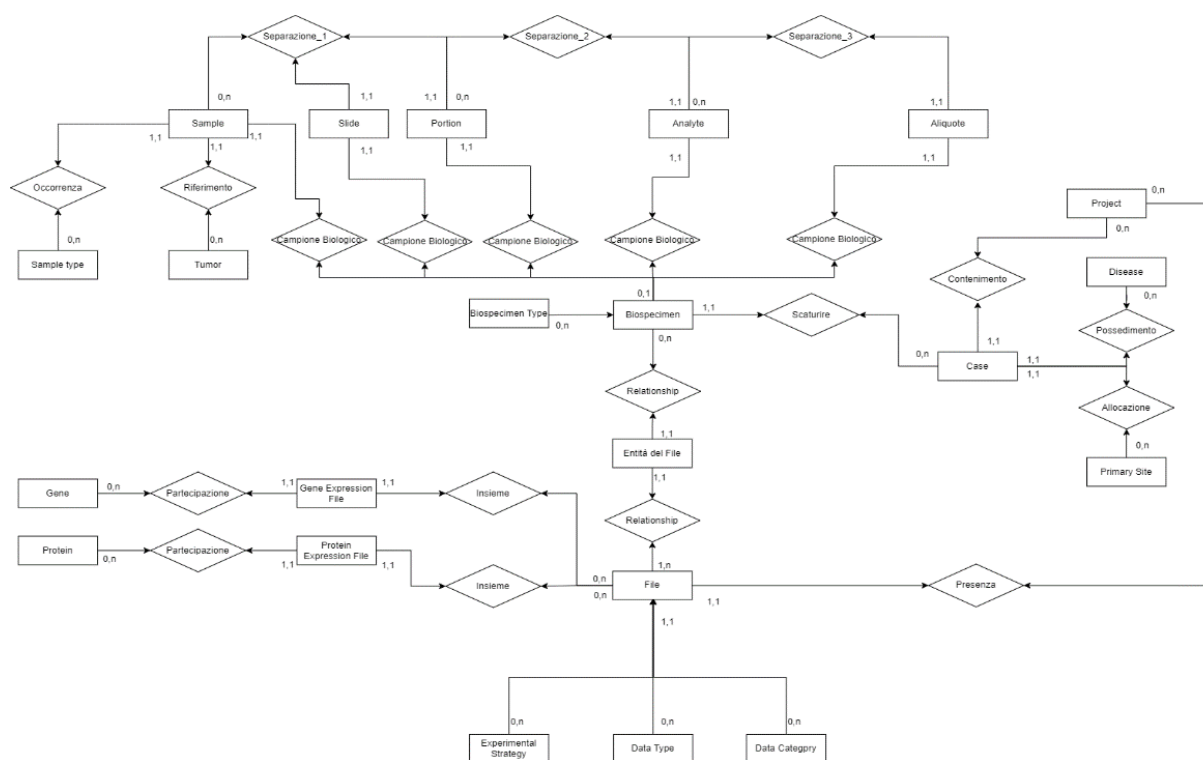
- Aggiunta "Campione Biologico": applicato il metodo 3 (Sostituzione) dalla risoluzione della generalizzazione di Biospecimen
- Aggiunta "Insieme": tra File e Gene/Protein Expression File

Diagramma ER Ristrutturato

Un Diagramma ER (Entity-Relationship Diagram) Ristrutturato è una rappresentazione visuale della struttura di un database dopo aver applicato determinate **modifiche o miglioramenti** per ottimizzarne la progettazione. L'obiettivo principale di questa ristrutturazione è semplificare il modello dei dati, **eliminare ridondanze, migliorare le prestazioni e garantire una migliore gestibilità** del database.

Il Diagramma ER Ristrutturato offre una visione chiara delle modifiche apportate al modello dei dati e come queste influenzino la struttura complessiva del database. Questa rappresentazione aiuta gli sviluppatori, gli amministratori di database e altri professionisti IT a comprendere e valutare le modifiche apportate per garantire che il database sia **efficiente, scalabile e in grado di soddisfare i requisiti** dell'applicazione.

In breve, un Diagramma ER Ristrutturato è uno strumento **essenziale** per ottimizzare la progettazione di un database al fine di migliorare le **prestazioni, l'affidabilità e la gestibilità** complessiva del sistema.



Dizionario dei Dati (Entità)

ENTITÀ	DESCRIZIONE	ATTRIBUTI	IDENTIFICATORE	VINCOLI ESTERNI
PROJECT	Un progetto biomedico con un codice univoco e un nome. Può contenere una serie di file, casi, e altre informazioni specifiche del progetto.	project_id, name	project_id	
CASE	La raccolta di tutti i dati relativi ad un soggetto specifico nel contesto di un progetto specifico.	case_id, ethnicity, gender, race, vital_status	case_id	
BIOSPECIMEN	Qualsiasi campione materiale prelevato da un'entità biologica a fini di analisi, diagnosi, propagazione, trattamento o ricerca.	id	id	Se un id è presente già in una tabella sottostante a Biospecimen non può essere in un'altra
DISEASE	Dati sulla malattia del caso collegato	id, type	id	
PRIMARY SITE	Dati sul sito preso in analisi del caso in questione	id, site	id	

SAMPLE IS-A BIOSPECIMEN	Un campione biologico specifico, identificato da un ID univoco. Contiene informazioni sul tipo di campione e il tipo di tumore, se applicabile.	(Ereditati da Biospecimen)	biospecimen.id	Per aggiungere un record alle tabelle sottostanti campione deve essere presente un record a cui si riferiscono a cascata
SAMPLE TYPE	Tipologia di un campione	type_id, type	id	
TUMOR	Contiene i dati sul tumore del campione preso in questione	tumor_code_id, code, descriptor	id	
PORTION IS-A BIOSPECIMEN	Una sottoparte fisica di qualsiasi campione. Contiene informazioni sulla porzione del campione, ma può essere collegato ad analiti più piccoli.	(Ereditati da Biospecimen)	biospecimen.id	Per aggiungere un record alle tabelle sottostanti porzione deve essere presente un record a cui si riferiscono a cascata
ANALYTE IS-A BIOSPECIMEN	Un prodotto sfuso liquido prodotto secondo protocolli di laboratorio specificati, da una porzione, destinato a ulteriori analisi.	(Ereditati da Biospecimen) concentration	biospecimen.id	Per aggiungere un record alle tabelle sottostanti analita deve essere presente un record a cui si riferiscono a cascata

ALIQOT IS-A BIOSPECIMEN	Una piccola porzione di un analita con una concentrazione specifica. Contiene informazioni sulla concentrazione dell'aliquota.	(Ereditati da Biospecimen) concentration	biospecimen.id	
BIOSPECIMEN TYPE	Tipologia di un Campione Biologico: Sample, Slide, Portion, Analyte o Aliquot	id, type	id	
ANALYSIS	File che contengono dati di espressione genica o proteica. Contiene informazione come il nome del file, il suo submitter, la dimensione del file e altro.	id, filename, file_size, created_datetime, updated_datetime	id	La data di creazione di un file non può essere successiva alla data dell'ultima modifica
EXPERIMENTAL STRATEGY	Dati sulla strategia sperimentale usata nell'analisi	strategy_id, strategy	id	
DATA TYPE	Dati sul tipo di dati nel file dell'analisi	type_id, type	id	
DATA CATEGORY	Dati sulla categoria di dati nel file dell'analisi	category_id, category	id	

GENE	Dati di un gene specifico. Contiene informazioni sul gene, come il suo ID e il nome.	gene_id, gene_name	gene_id
GENE TYPE	Tipologia di un gene	type_id, type	id
PROTEIN	Dati di una proteina specifica. Contiene informazioni sul gene, come il suo ID e il nome.	agid, lab_id, catalog_num, set_id, peptide_tget	agid
GENE EXPRESSION FILE	File di dati contenente informazioni sull'espressione genica.	unstranded, stranded_firt stranded_sec, tpm, fpkm, fpkm_uq	FK
PROTEIN EXPRESSION FILE	File di dati contenente dati normalizzati dell'array di proteine a fase inversa.	expression	FK

Dizionario dei Dati (Relazioni)

RELAZIONI	DESCRIZIONE	COMPONENTI
CONTENIMENTO	Associa un progetto ai casi che contiene	Project, Case
PRESENZA	Associa un progetto ai file che presenta	Project, File
POSSEDIMENTO	Associa un caso alla malattia che possiede	Case, Disease
ALLOCAZIONE	Associa un caso al sito dove si alloca la malattia	Case, Primary Site
SCATURIRE	Associa dei casi alle biospecie che hanno scaturito	Case, Biospecimen
OCCORRENZA	Associa un campione biologico al tipo di occorrenza che è	Sample/Analyte/Aliquot, Type
RIFERIMENTO	Associa un tumore al campione che fa riferimento	Sample, Tumor
SEPARAZIONE_1	Associa un campione alle porzioni separate da esso	Sample, Portion
SEPARAZIONE_2	Associa una porzione agli analiti separati da esso	Portion, Analyte
SEPARAZIONE_3	Associa un analita alle aliquot separate da esso	Analyte, Aliquot
ENTITÀ DELL'ANALISI	Associa più campioni biologici ai file dove è contenuto	Biospecimen, File
GENE EXPRESSION	Associa più geni ai file dove sono contenute le misurazioni	File, Gene
PROTEIN EXPRESSION	Associa più proteine ai file dove sono contenute le misurazioni	File, Protein
INSIEME	Associa un File alle concentrazioni in esso	File, Gene/Protein Expression File
PARTECIPAZIONE	Associa un Gene/Proteina ai file dov'è contenuto	Gene/Protein, Gene/Protein Expression File

Tavola dei Volumi (2 Versione)

CONCETTO	TIPO	VOLUME
PROJECT	E	100
CASE	E	20.000
BIOSPECIMEN	E	600.000
DISEASE	E	200
PRIMARY SITE	E	68
SAMPLE IS-A BIOSPECIMEN	E	60.000
SAMPLE TYPE	E	20
TUMOR	E	33
PORTION IS-A BIOSPECIMEN	E	90.000
ANALYTE IS-A BIOSPECIMEN	E	150.000
ALIUQUOT IS-A BIOSPECIMEN	E	300.000
BIOSPECIMEN TYPE	E	4
ANALYSIS	E	35.000
EXPERIMENTAL STRATEGY	E	12
DATA TYPE	E	30
DATA CATEGORY	E	11
GENE	E	60.660
GENE TYPE	E	50
PROTEIN	E	500
GENE EXPRESSION FILE	E	1.500.000.000
PROTEIN EXPRESSION FILE	E	5.000.000
CONTENIMENTO	R	20.000
PRESENZA	R	N.A.
POSSEDIMENTO	R	N.A.
ALLOCAZIONE	R	N.A.
OCCORRENZA	R	N.A.
SCATURIRE	R	30.000
RIFERIMENTO	R	N.A.
ENTITÀ DELL'ANALISI	R	N.A.
SEPARAZIONE_1	R	90.000
SEPARAZIONE_2	R	150.000
SEPARAZIONE_3	R	300.000
GENE EXPRESSION	R	N.A.
PROTEIN EXPRESSION	R	N.A.
INSIEME	R	N.A.
PARTECIPAZIONE	R	N.A.
CONTENIMENTO	R	N.A.

3.1.3 Modello Relazionale

Il modello relazionale è un modello di dati che rappresenta le informazioni all'interno di un database in **forma tabellare**, utilizzando tabelle (relazioni) composte da righe e colonne. È uno dei modelli di dati più utilizzati nei database relazionali, che sono ampiamente diffusi in applicazioni di gestione dati, sistemi aziendali e applicazioni web.

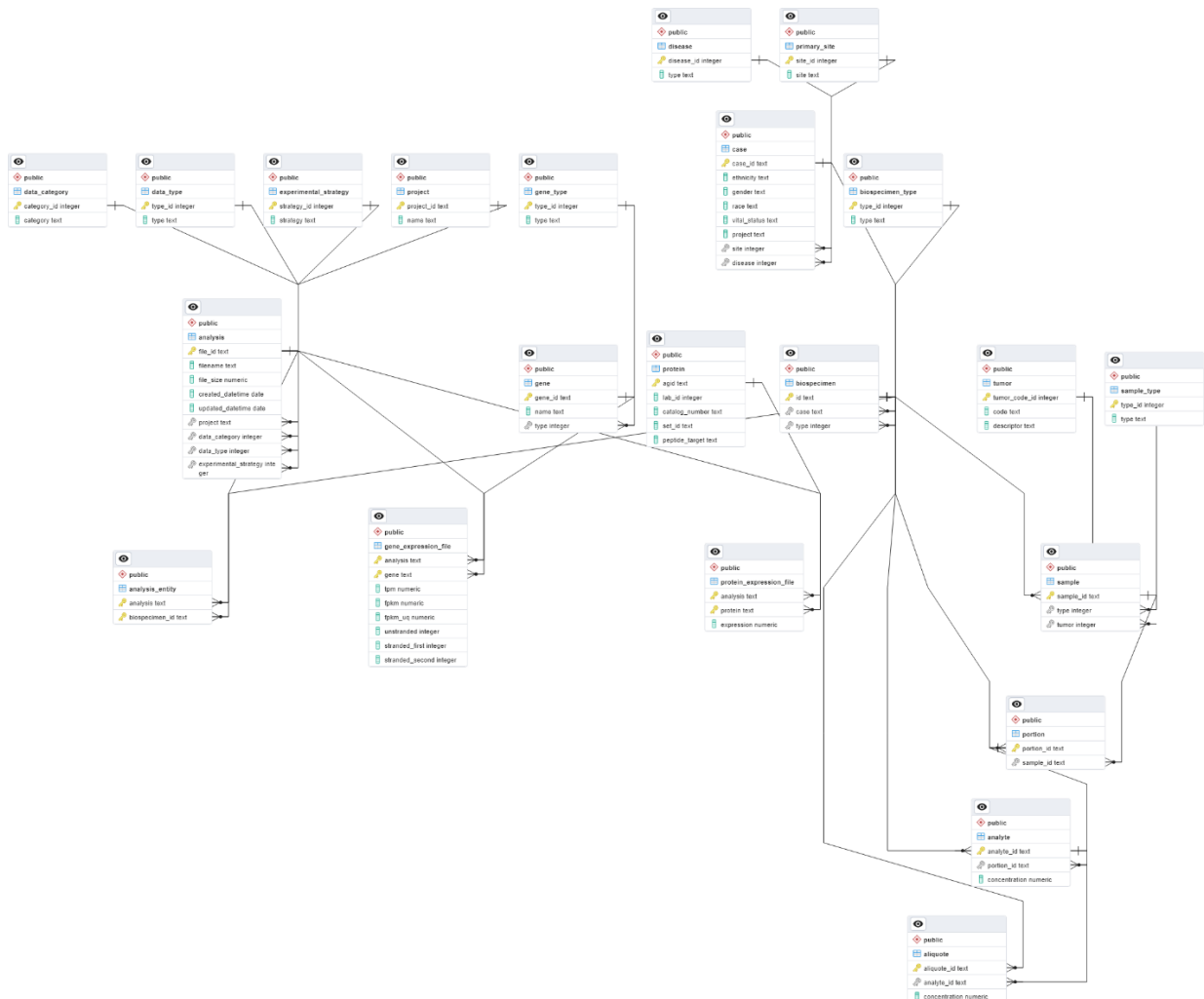
Schema Logico

Questo schema dati è stato progettato per garantire l'integrità e la coerenza dei dati, consentendo agli utenti di registrare e analizzare informazioni provenienti da studi biomedici. È stato strutturato per fornire **flessibilità e scalabilità** in modo che ulteriori informazioni **possano essere aggiunte in futuro** senza dover ridisegnare completamente il database. Il modello relazionale è uno strumento fondamentale per la gestione efficiente dei dati biomedici e l'analisi delle espressioni geniche e proteiche per scopi di ricerca e studio.

project (<u>project_id</u> , name)
case (<u>case_id</u> , ethnicity, gender, race, vital_status, project, site, disease) Foreign Key project references project (project_id), site references primary_site (id), disease references disease (id)
disease (<u>disease_id</u> , type)
primary_site (<u>site_id</u> , site)
biospecimen (<u>id</u> , case, type) Foreign Key case references case (case_id) Foreign Key type references biospecimen_type (id)
biospecimen_type (<u>type_id</u> , type)
sample (<u>sample_id</u> , type, tumor) Foreign Key sample_id references biospecimen (id), type references sample_type (id), tumor references tumor (tumor_code)
portion (<u>portion_id</u> , sample_id) Foreign Key portion_id references biospecimen (id), sample_id references sample (sample_id)
analyte (<u>analyte_id</u> , portion_id, concentration) Foreign Key analyte_id references biospecimen (id), portion_id references portion (portion_id)
aliquot (<u>aliquot_id</u> , <u>analyte_id</u> , concentration) Foreign Key aliquot_id references biospecimen (id), analyte_id references analyte (analyte_id)
sample_type (<u>type_id</u> , type)
tumor (<u>tumor_code_id</u> , code, descriptor)
analysis_entity (<u>file</u> , biospecimen_id) Foreign Key biospecimen_id references biospecimen (id), file references file (file_id)
gene_type (<u>type_id</u> , type)
analysis (<u>file_id</u> , filename, file_size, created_datetime, updated_datetime, project, data_type, data_category, experimental_strategy) Foreign Key project references project (project_id), data_type references data_type (type_id), data_category references data_category (category_id), experimental_strategy references experimental_strategy (strategy_id)
data_type (<u>type_id</u> , type)
data_category (<u>category_id</u> , category)
experimental_strategy (<u>strategy_id</u> , strategy)
gene_expression_file (<u>gene</u> , <u>analysis</u> , unstranded, stranded_first, stranded_second tpm, fpkm, fpkm_uq) Foreign Key gene references gene (gene_id), analysis references analysis (file_id)
protein_expression_file (<u>protein</u> , <u>analysis</u> , expression) Foreign Key analysis references analysis (file_id), protein references protein (protein_id)
gene (<u>gene_id</u> , name, type) Foreign Key type references gene_type (type_id)
protein (<u>agid</u> , lab_id, catalog_number, set_id, peptide_target)

Schema Relazionale

Questa è una panoramica dello schema realizzato su PostgreSQL



Ristrutturazione dello Schema Relazionale

Dato che il numero degli accessi per le operazioni è ridotto al minimo e il carico generale dell'applicazione non è eccessivo, non è necessaria una ristrutturazione dello schema relazionale.

3.1.4 Specifica del Database in SQL

Creazione Database

```
CREATE DATABASE IF NOT EXIST "GDC";  
USE "GDC";
```

```
CREATE TABLE public.aliquote (  
    aliquote_id text PRIMARY KEY,  
    analyte_id text NOT NULL,  
    concentration numeric,  
    FOREIGN KEY (analyte_id) REFERENCES public.analyte (analyte_id)  
    FOREIGN KEY (aliquote_id) REFERENCES public.biospecimen ("id")  
);
```

```
CREATE TABLE public.analysis (  
    file_id text PRIMARY KEY,  
    filename text,  
    file_size numeric,  
    created_datetime date,  
    updated_datetime date,  
    project text NOT NULL,  
    data_category integer,  
    data_type integer,  
    experimental_strategy integer,  
    CONSTRAINT "CK_Data" CHECK ((updated_datetime >= created_datetime)),  
    FOREIGN KEY (project) REFERENCES public.project (project_id),  
    FOREIGN KEY (data_category) REFERENCES public.data_category (category_id),  
    FOREIGN KEY (data_type) REFERENCES public.data_type (type_id),  
    FOREIGN KEY (experimental_strategy) REFERENCES  
public.experimental_strategy (strategy_id),  
);
```

```
CREATE TABLE public.analysis_entity (  
    analysis text PRIMARY KEY,  
    biospecimen_id text PRIMARY KEY,  
    FOREIGN KEY (biospecimen_id) REFERENCES public.biospecimen (id),  
    FOREIGN KEY ("File") REFERENCES public.file (file_id)  
);
```

```
CREATE TABLE public.analyte (  
  analyte_id text PRIMARY KEY,  
  portion_id text NOT NULL,  
  concentration numeric,  
  FOREIGN KEY (analyte_id) REFERENCES public.biospecimen (id),  
  FOREIGN KEY (portion_id) REFERENCES public.portion (portion_id)  
);
```

```
CREATE TABLE public.biospecimen (  
  "id" text PRIMARY KEY,  
  "case" text NOT NULL,  
  "type" integer NOT NULL,  
  FOREIGN KEY ("case") REFERENCES public.case (case_id),  
  FOREIGN KEY ("type") REFERENCES public.biospecimen_type (type_id)  
);
```

```
CREATE TABLE public.biospecimen_type (  
  type_id integer AUTO_INCREMENT PRIMARY KEY,  
  type text  
);
```

```
CREATE TABLE public."data_category" (  
  "category_id" integer AUTO_INCREMENT PRIMARY KEY,  
  "category" text  
);
```

```
CREATE TABLE public."data_type" (  
  "type_id" integer AUTO_INCREMENT PRIMARY KEY,  
  "type" text  
);
```

```
CREATE TABLE public.disease (  
  disease_id integer AUTO_INCREMENT PRIMARY KEY,  
  type text  
);
```

```

CREATE TABLE public.case (
  case_id text PRIMARY KEY,
  ethnicity text,
  race text,
  gender text,
  vital_status text,
  project text NOT NULL,
  site integer,
  disease integer,
  CONSTRAINT "CK_Gender" CHECK ((gender = ANY (ARRAY['Male'::text,
'Female'::text, 'Unknown'::text, 'Not reported'::text]])),
  CONSTRAINT "CK_Vital_Status" CHECK ((vital_status = ANY
(ARRAY['Alive'::text, 'Dead'::text]])),
  FOREIGN KEY (disease) REFERENCES public.disease (disease_id),
  FOREIGN KEY (project) REFERENCES public.project (project_id),
  FOREIGN KEY (site) REFERENCES public.primary_site (site_id)
);

```

```

CREATE TABLE public.experimental_strategy (
  strategy_id integer AUTO_INCREMENT PRIMARY KEY,
  strategy text
);

```

```

CREATE TABLE public.gene (
  gene_id text PRIMARY KEY,
  name text,
  type integer,
  FOREIGN KEY (type) REFERENCES public.gene_type (type_id)
);

```

```

CREATE TABLE public."gene_type" (
  "type_id" integer AUTO_INCREMENT PRIMARY KEY,
  "type" text
);

```

```
CREATE TABLE public.gene_expression_file (  
  analysis text PRIMARY KEY,  
  gene text PRIMARY KEY,  
  tpm numeric,  
  fpkm numeric,  
  fpkm_uq numeric  
  unstranded integer,  
  stranded_first integer,  
  stranded_second integer,  
  FOREIGN KEY (analysis) REFERENCES public.analysis (file_id),  
  FOREIGN KEY (gene) REFERENCES public.gene (gene_id)  
);
```

```
CREATE TABLE public.portion (  
  portion_id text PRIMARY KEY,  
  sample_id text NOT NULL,  
  FOREIGN KEY (portion_id) REFERENCES public.biospecimen (id),  
  FOREIGN KEY (sample_id) REFERENCES public.sample (sample_id)  
);
```

```
CREATE TABLE public.primary_site (  
  site_id integer AUTO_INCREMENT PRIMARY KEY,  
  site text  
);
```

```
CREATE TABLE public.project (  
  project_id text PRIMARY KEY,  
  name text  
);
```

```
CREATE TABLE public.protein (  
  agid text PRIMARY KEY  
  lab_id integer,  
  catalog_number text,  
  set_id text,  
  peptide_target text  
);
```

```

CREATE TABLE public.protein_expression_file (
  analysis text PRIMARY KEY,
  protein text PRIMARY KEY,
  expression numeric,
  FOREIGN KEY (analysis) REFERENCES public.analysis (file_id),
  FOREIGN KEY (protein) REFERENCES public.protein (agid)
);

CREATE TABLE public.sample (
  sample_id text PRIMARY KEY,
  type integer,
  tumor integer,
  FOREIGN KEY (sample_id) REFERENCES public.biospecimen (id),
  FOREIGN KEY (tumor) REFERENCES public.tumor (tumor_code_id),
  FOREIGN KEY (type) REFERENCES public.sample_type (type_id)
);

CREATE TABLE public.sample_type (
  type_id integer PRIMARY KEY,
  type text
);

CREATE TABLE public.tumor (
  tumor_code_id integer PRIMARY KEY,
  code text,
  descriptor text
);

```

3.2 Script di downloading e gestione dati

Qui si esplorerà il processo di scaricamento dei dati genomici dalla piattaforma GDC, illustrando gli script e le API utilizzate per acquisire i dati di espressione genica e i metadati associati. Sarà anche descritto come vengono gestiti i dati scaricati per renderli pronti per l'analisi.

La costante su cui si basa l'intero progetto è rappresentata dai dati, indispensabili per la creazione del database ma soprattutto per le sperimentazioni.

È stata quindi implementata una procedura complessa e dettagliata per scaricare e gestire i dati relativi all'espressione genica dalla piattaforma GDC (Genomic Data Commons). Questa sezione è fondamentale per acquisire i dati necessari all'analisi e inserirli nel database descritto. Ecco una descrizione delle **componenti chiave** di questo script:

1. **Connessione al Database:** Il codice inizia con l'impostazione dei parametri per la connessione al database PostgreSQL. Questi parametri includono l'indirizzo del server del database, il nome del database, il nome utente, la password e la porta di connessione.
2. **Funzione di Download ed Elaborazione dei Dati:** La funzione `'download_and_process_expression_data'` è il **cuore** di questo script. Questa funzione **gestisce il download** dei dati da GDC e li inserisce nel database PostgreSQL. Ecco come funziona:
 - a. Si stabilisce una connessione al database e si crea un cursore per eseguire query SQL. La transazione inizia in modo che le operazioni di inserimento nel database possano essere eseguite come **un'unica transazione**.

```
# Crea una connessione al database PostgreSQL
connection = psycopg2.connect(**db_params)

# Crea un cursore per eseguire query SQL
cursor = connection.cursor()

# Inizia la transazione
connection.autocommit = False
```

- b. Viene specificato l'URL dell'API GDC per scaricare i file relativi all'analisi dell'espressione genica.

I possibili **endpoint** della piattaforma

API Endpoints

Communicating with the GDC API involves making calls to API endpoints. Each GDC API endpoint represents specific API functionality, as summarized in the following table:

Endpoint	Type	Description
status	Status	Get the API status and version information
projects	Search & Retrieval	Search all data generated by a project
cases	Search & Retrieval	Find all files related to a specific case, or sample donor.
files	Search & Retrieval	Find all files with specific characteristics such as file_name, md5sum, data_format and others.
annotations	Search & Retrieval	Search annotations added to data after curation
data	Download	Used to download GDC data
manifest	Download	Generates manifests for use with GDC Data Transfer Tool
slicing	BAM Slicing	Allows remote slicing of BAM format objects
submission	Submission	Returns the available resources at the top level above programs i.e., registered programs

The HTTP URL that corresponds to the latest version of a GDC API endpoint is `https://api.gdc.cancer.gov/<endpoint>`, where `<endpoint>` is the name of the endpoint.

The HTTP URL of an endpoint corresponding to a specific major version of the GDC API is `https://api.gdc.cancer.gov/<version>/<endpoint>`, where `<endpoint>` is the name of the endpoint and `<version>` is the GDC API version.

For example, the address of the latest version of the `status` endpoint is `https://api.gdc.cancer.gov/status`, whereas the address of the `status` endpoint corresponding to version 0 of GDC API is `https://api.gdc.cancer.gov/v0/status`.

- c. I dati vengono scaricati utilizzando una richiesta HTTP a GDC. Vengono applicati diversi **filtri** per selezionare i file desiderati in base al tipo di dati, alla malattia, all'accesso aperto e al formato del file.
- d. I dati scaricati vengono quindi elaborati. I dettagli sui **progetti, casi e file** vengono estratti e confrontati con il database locale per **evitare duplicati**.

Parte di codice che confronta se un progetto è già nel DB

```
# Verifica se il progetto è già presente nel database
cursor.execute(cerca_progetto, (project_id,))
result = cursor.fetchone()
if result[0] == 0:
    # Se il progetto non è presente, esegui la funzione project() per
    inserirlo
    project(project_id, cursor)
    connection.commit()
```

- e. Viene inizializzata una **serie di query SQL** che verranno utilizzate per inserire i dati nel database. Le informazioni sui **progetti, casi e campioni** vengono inserite prima.

- f. Per ciascun file scaricato, i dati di espressione genica vengono elaborati e quindi inseriti nel database. In base al tipo di dati (genico o proteico), **i dati vengono gestiti in modo appropriato.**

Un esempio di inserimento di Gene_Expression_File

```
if type_id == 1:
    for data_row in expression_data:
        # Inserimento dei dati di espressione genica nel database
        gene_id = data_row["gene_id"]
        stranded_first = data_row["stranded_first"]
        stranded_second = data_row["stranded_second"]

        if stranded_first != 0 and stranded_second != 0:
            cursor.execute(inserisci_espressione_genica, (file_id, gene_id,
data_row["tpm_unstranded"], data_row["fpkm_unstranded"],
data_row["fpkm_uq_unstranded"], data_row["unstranded"], stranded_first,
stranded_second))
            connection.commit()
```

- g. Alla fine della procedura, la transazione viene confermata per **rendere permanenti** tutte le modifiche nel database.
3. **Funzioni Ausiliarie:** Nel codice sono incluse diverse funzioni ausiliarie per la gestione di **progetti, casi, campioni e dati di espressione**. Queste funzioni consentono di inserire dati di supporto nel database.

Funzione ausiliaria per l'inserimento di un nuovo progetto

```
# Funzione per inserire un nuovo progetto nel database
def project(id, cursor):
    project_url = "https://api.gdc.cancer.gov/projects/" + id
    inserisci_progetto = "INSERT INTO public.project VALUES (%s, %s) ON CONFLICT
(project_id) DO NOTHING;"

    params = {
        #Puoi aggiungere altri campi che danno più info relative al progetto
        "fields": "name",
        "format": "JSON",
        "pretty": "true"
    }

    response = requests.get(project_url, params=params)

    if response.status_code == 200:
        data = json.loads(response.content.decode("utf-8"))["data"]

        cursor.execute(inserisci_progetto, (id, data["name"]))
        print("Progetto inserito nel database")
    else:
        print(f"Errore durante il download del progetto: {response.status_code}")
        return []
```


4. **Gestione degli Errori:** Il codice gestisce gli errori sia relativi al database che alle richieste HTTP. Ogni volta che si verifica un errore, vengono stampate le relative **informazioni di errore per il debug**.

Gestione degli errori con il blocco try except

```
except psycopg2.Error as db_error:
    # Gestione degli errori del database
    connection.rollback()
    print(f"Errore nel database: {db_error}")

# Gestione degli errori di richiesta HTTP
except requests.RequestException as request_error:
    print(f"Errore nella richiesta HTTP: {request_error}")

except Exception as error:
    # Gestione generica degli errori
    connection.rollback()
    print(f"Errore sconosciuto: {error}")
```

Questo script rappresenta un passo fondamentale nel processo di creazione di un database per l'analisi dei dati genomici. Esso **scarica i dati necessari** dalla piattaforma GDC, **li elabora e li inserisce** in un database locale, preparandoli per analisi successive.

3.3 Script di creazione di alberi decisionali

Questa sezione affronterà il cuore dell'analisi, fornendo dettagli sullo sviluppo di modelli di alberi decisionali utilizzando tecniche di machine learning. Verranno descritti gli algoritmi impiegati e come vengono addestrati i modelli. Inoltre, si analizzeranno i risultati ottenuti e come essi possano essere interpretati per ottenere informazioni biologiche significative.

3.3.1 Cos'è un albero decisionale?

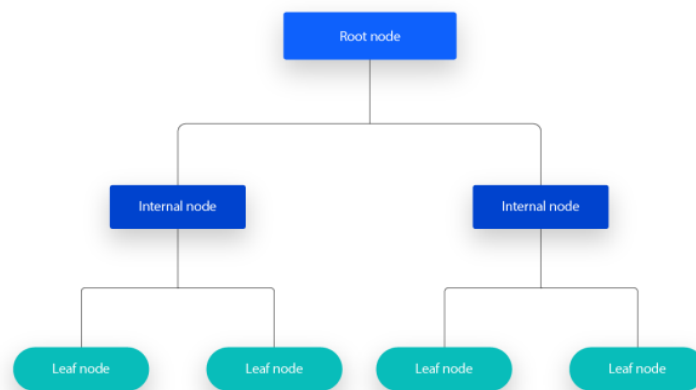
Facciamo innanzitutto chiarezza su cos'è un albero decisionale:

Un albero decisionale è un **modello di apprendimento automatico** utilizzato nell'ambito dell'analisi dei dati e della classificazione. Questo modello rappresenta un'analisi dei dati sotto forma di una struttura ad albero, in cui ogni **nodo rappresenta una decisione** o un **test** su un attributo dei dati, ciascun **ramo rappresenta l'outcome di una decisione** e ogni **foglia rappresenta una classe** o una **previsione**. Gli alberi decisionali sono ampiamente utilizzati per problemi di classificazione, ma possono essere applicati anche alla regressione.

Un albero decisionale è costituito da tre tipi di nodi:

1. **Nodo Radice:** Questo è il punto di partenza dell'albero e rappresenta l'intero set di dati. In genere, il nodo radice rappresenta la variabile più influente o il test iniziale che dividerà i dati in sottoinsiemi.
2. **Nodi Interni:** Questi nodi rappresentano i test intermedi su attributi specifici dei dati. Ad esempio, potrebbero rappresentare domande come "È il valore dell'attributo X maggiore di Y?". I nodi interni hanno rami uscenti che portano ai nodi successivi.
3. **Nodi Foglia:** Questi nodi rappresentano le classi o le previsioni finali. Ogni foglia rappresenta una categoria o un valore previsto in base ai test effettuati nei nodi interni lungo il percorso dal nodo radice alla foglia.

Un esempio basico di albero decisionale



L'albero decisionale costruisce una **serie di domande** (test) basate sugli attributi dei dati, cercando di suddividere il set di dati in modo che gli esempi di classi simili siano **raggruppati insieme**. Questo processo continua ricorsivamente fino a quando viene soddisfatta una condizione di stop, ad esempio la profondità massima dell'albero o il numero minimo di esempi in una foglia.

L'albero decisionale è utilizzato per prendere decisioni o effettuare **previsioni per nuovi dati**. Basta seguire il percorso dall'alto verso il basso, rispondendo alle domande nei nodi interni, fino a raggiungere una foglia che rappresenta la classe o la previsione finale.

Gli alberi decisionali sono apprezzati per la loro **semplicità e interpretabilità**. Tuttavia, possono diventare complessi con dati ricchi di attributi. Alcuni dei vantaggi e delle limitazioni degli alberi decisionali includono:

- **Vantaggi:**
 - Semplici da interpretare e spiegare.
 - Possono gestire dati categorici e numerici.
 - Sono adatti per la selezione delle caratteristiche (feature selection).
 - Non richiedono molta preparazione dei dati.
- **Limitazioni:**
 - Possono essere suscettibili all'over fitting (adattamento eccessivo) con alberi troppo profondi.
 - Potrebbero non gestire bene problemi con relazioni complesse tra le variabili.
 - L'accuratezza potrebbe non essere competitiva rispetto ad altri modelli più avanzati.

3.3.2 Codice ed implementazione

In questo progetto è stato implementato un codice che utilizza il modulo scikit-learn per creare un albero decisionale a partire dai dati di espressione genica presenti nel database locale (precedentemente scaricati dalla piattaforma GDC). Ecco una descrizione delle componenti chiave di questo script:

1. **Connessione al Database:** Il codice inizia stabilendo una connessione al database PostgreSQL. Vengono forniti i parametri necessari per connettersi al database "GDC".
2. **Query per Estrazione dei Dati:** Viene definita una query SQL complessa per estrarre i dati di addestramento dal database. La query unisce dati da diverse tabelle, inclusi dati di espressione genica, informazioni sul campione e il tipo di tessuto associato.

Nella query di esempio qui sotto ricerchiamo tessuti sani o tumorali contenenti il gene X per fare analisi su di essi.

```
SELECT tpm, fpkm, fpkm_uq, unstranded, stranded_first, stranded_second,  
sample_type.type as tissue_label  
FROM gene_expression_file  
JOIN analysis_entity ON gene_expression_file.analysis = analysis_entity.analysis  
JOIN aliquote a ON analysis_entity.biospecimen_id = a.aliquote_id  
JOIN Analyte ay ON a.Analyte_Id = ay.analyte_id  
JOIN portion ON ay.portion_id = portion.Portion_Id  
JOIN sample s ON portion.sample_id = s.sample_id  
JOIN sample_type ON s.type = sample_type.type_id  
WHERE s.Type in (1, 11) and gene.name = "X";
```

3. **Esecuzione della Query ed Estrazione dei Dati:** La query viene eseguita e i dati estratti vengono memorizzati in un DataFrame di Pandas. Questi dati costituiranno le feature e il target per l'addestramento del modello.

```
# Esecuzione della query e ottenimento dei dati  
cursor.execute(query2)  
data = cursor.fetchall()  
column_names = [desc[0] for desc in cursor.description]  
df = pd.DataFrame(data, columns=column_names)
```

4. **Preparazione dei Dati:** I dati vengono suddivisi in due parti: le feature (X) e il target (y). Nella fase successiva, i dati vengono ulteriormente suddivisi in set di addestramento e test utilizzando la funzione `'train_test_split'`. Questa suddivisione è essenziale per valutare le prestazioni del modello.

```
# Separazione dei dati in features (X) e target (y)
X = df.drop('tissue_label', axis=1)
y = df['tissue_label']

# Dividi i dati in set di addestramento e test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
```

5. **Creazione e Addestramento del Modello:** Viene creato un modello di albero decisionale utilizzando il `'DecisionTreeClassifier'` di scikit-learn. Il modello viene addestrato utilizzando i dati di addestramento (X_train e y_train). Nella creazione dell'albero decisionale, può essere anche impostata una profondità massima, il che limita la complessità dell'albero.
6. **Valutazione del Modello:** Il modello addestrato viene utilizzato per fare previsioni sui dati di test (X_test) e le previsioni vengono confrontate con i valori effettivi (y_test). L'accuratezza del modello viene calcolata utilizzando la funzione `'accuracy_score'` di scikit-learn. Viene anche generato un report di classificazione utilizzando la funzione `'classification_report'`, che fornisce informazioni dettagliate sulla precisione, il richiamo e l'F1-score per ciascuna classe.

```
# Dividi i dati in set di addestramento e test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Crea e addestra il modello di albero decisionale
model = DecisionTreeClassifier(random_state = 2, criterion = "entropy",
min_samples_split = 100, min_samples_leaf = 60)
model.fit(X_train, y_train)

# Valuta il modello
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)
```

7. **Visualizzazione dell'Albero Decisionale:** L'albero decisionale viene visualizzato come grafico utilizzando il modulo `pydotplus`. L'immagine risultante dell'albero viene salvata come "decision_tree.png" e successivamente aperta e mostrata utilizzando la libreria `PIL`.

```
# Visualizza l'albero decisionale come grafico
dot_data = StringIO()
tree.export_graphviz(model, out_file=dot_data, feature_names=list(X.columns))
graph = pydotplus.graph_from_dot_data(dot_data.getvalue())
graph.write_png("decision_tree.png")

# Visualizza l'immagine dell'albero
img = Image.open("decision_tree.png")
img.show()
```

8. **Chiusura della Connessione al Database:** La connessione al database viene chiusa in modo appropriato.

Questo script è fondamentale per la creazione e la valutazione di un modello di albero decisionale basato sui dati di espressione genica del database. I risultati dell'addestramento del modello e l'immagine dell'albero decisionale vengono utilizzati per comprendere meglio le relazioni nei dati e, eventualmente, per prendere decisioni basate su tali relazioni.

Capitolo 4: Risultati

Nel capitolo precedente, abbiamo esaminato dettagliatamente la progettazione concettuale e logica del nostro database per la gestione dei dati di espressione genica e proteica in ambito biomedico. Successivamente abbiamo realizzato un esempio di albero decisionale che simula un'analisi diretta sul database. Ora, entriamo nella fase di **presentazione dei risultati ottenuti** attraverso l'implementazione di questo modello di database.

Questo capitolo offre un **quadro completo dei risultati ottenuti** attraverso l'implementazione del nostro database, fornendo una panoramica delle analisi condotte e delle scoperte significative nel campo della biomedicina. I risultati presentati qui rappresentano un contributo importante alla nostra comprensione della **regolazione genica e dell'espressione proteica**.

4.1 Dati Caricati

Nel nostro database, abbiamo caricato una vasta quantità di dati di espressione genica e proteica provenienti da fonti autorevoli nel campo della biomedicina. Questi dati rappresentano una risorsa preziosa per una vasta gamma di analisi e ricerche nel campo della genetica, dell'oncologia e di altre discipline correlate.

I dati importati includono informazioni dettagliate, come:

- **Tipo di Dati:** Abbiamo acquisito dati di espressione genica, tra cui i valori di TPM (Transcripts Per Million), FPKM (Fragments Per Kilobase per Million), e FPKM-UQ (Upper Quartile), nonché dati di espressione proteica.
- **Origine dei Campioni:** Ogni campione biologico è associato a dettagli sul paziente, come l'etnia, il genere ed il sito primario del tumore.
- **Dettagli delle Analisi:** Ogni analisi è caratterizzata da un codice univoco, un nome, un identificativo del caso preso in analisi ed informazioni quali la categoria di dati, le dimensioni del file e la strategia sperimentale utilizzata.
- **Annotazioni di Geni e Proteine:** I dati di espressione genica e proteica sono associati a informazioni di annotazione dettagliate sui geni e le proteine coinvolte.
- **Concentrazioni:** Per i dati di espressione proteica, memorizziamo anche le concentrazioni misurate nelle diverse analisi.

I dati di espressione genica e proteica costituiscono il fulcro del nostro database, fornendo una base solida per una vasta gamma di analisi e ricerche nel campo della biomedicina. La disponibilità di dati di alta qualità e ben annotati è fondamentale per l'identificazione di biomarcatori, lo studio delle malattie e il progresso della medicina personalizzata. Nel prosieguo di questo capitolo, esploreremo come questi dati possano essere utilizzati per condurre analisi significative e fare scoperte importanti nel campo della ricerca biomedica.

Tabelle gene_expression e protein_expression correttamente popolate

Data Output Messages Notifications										Data Output Messages Notifications									
	analysis [PK] text	gene [PK] text	tpm numeric	fpm numeric	fpm_uq numeric	unstranded integer	stranded_first integer	stranded_second integer		analysis [PK] text	protein [PK] text	expression numeric							
1	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000000003.15	126.6434	37.3131	44.0706	10673	5365	5308	1	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000001	0.24475							
2	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000000005.6	0.1094	0.0322	0.0381	3	1	2	2	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000002	0.224							
3	08971a88-cb64-4030-93e5-08d77d2ed654	ENS00000000000419.13	89.2742	26.303	31.0665	2002	979	1023	3	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000003	1.0151							
4	08971a88-cb64-4030-93e5-08d77d2ed654	ENS00000000000457.14	4.7309	1.3939	1.6463	605	476	513	4	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000004	0.23645							
5	08971a88-cb64-4030-93e5-08d77d2ed654	ENS00000000000460.17	3.0202	0.8899	1.051	335	420	349	5	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000005	0.44812							
6	08971a88-cb64-4030-93e5-08d77d2ed654	ENS00000000000938.13	18.8588	5.5564	6.5627	1185	580	605	6	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000006	-0.071739							
7	08971a88-cb64-4030-93e5-08d77d2ed654	ENS00000000000971.16	92.8618	27.36	32.3149	13768	6946	6822	7	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000007	0.1208							
8	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001036.14	76.1573	22.4383	26.5019	3993	3092	3055	8	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000008	0.20843							
9	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001084.13	16.744	4.9333	5.8267	2681	1558	1556	9	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000009	-0.16021							
10	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001167.14	24.2211	7.1363	8.4287	1715	890	1032	10	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000010	-0.054093							
11	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001460.18	4.4141	1.3005	1.5361	698	383	365	11	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000011	0.38368							
12	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001461.17	11.3478	3.3434	3.9489	1981	1025	1007	12	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000012	-0.0060764							
13	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001497.18	10.5288	3.1021	3.6639	2456	1215	1248	13	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000013	-0.13943							
14	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001561.7	26.8304	7.9051	9.3367	2315	1113	1202	14	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000014	0.11111							
15	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001617.12	20.6883	6.0954	7.1993	1855	930	928	15	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000015	0.77041							
16	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001626.16	6.2312	1.8359	2.1684	1151	588	571	16	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000016	-0.34215							
17	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001629.10	31.6718	9.3315	11.0215	4348	2252	2119	17	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000017	-0.3255							
18	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001630.17	3.8434	1.1324	1.3375	252	142	117	18	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000018	-0.30442							
19	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000001631.16	1.8999	0.5598	0.6612	219	121	120	19	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000019	0.059141							
20	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002016.18	2.1993	0.648	0.7653	184	92	92	20	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000020	-0.45601							
21	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002079.14	0.0256	0.0075	0.0089	3	1	2	21	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000021	-0.5513							
22	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002330.14	30.567	9.006	10.637	970	1831	1963	22	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000022	0.018463							
23	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002549.12	115.3963	33.9994	40.1567	8115	4071	4044	23	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000023	-0.36305472893027							
24	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002586.20	300.0713	88.4104	104.4216	27084	12438	14648	24	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000024	1.1311							
25	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002587.10	37.5857	11.0739	13.0794	5192	2660	2536	25	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000025	-0.0654427065849525							
26	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002726.21	73.7118	21.7178	25.6509	5185	2619	2566	26	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000026	-0.11425							
27	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002745.13	0.1979	0.0583	0.0689	12	4	8	27	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000027	-0.186494341122066							
28	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002746.15	1.3016	0.3835	0.453	337	191	180	28	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000028	1.7292							
29	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002822.15	0.1563	0.0461	0.0544	21	11	10	29	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000029	-0.49224							
30	08971a88-cb64-4030-93e5-08d77d2ed654	ENS0000000002834.18	165.6771	48.8137	57.6539	21735	12892	13078	30	0179fd16-fbdf-4a80-9ctc-d018effa5dc6	AGID000030	-0.599833409820084							

Tabella case correttamente popolata

Data Output Messages Notifications									
	case_id [PK] text	ethnicity text	gender text	race text	vital_status text	project text	site integer	disease integer	
1	TCGA-05-4244	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
2	TCGA-05-4249	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
3	TCGA-05-4250	not reported	female	not reported	Dead	TCGA-LUAD	1	1	
4	TCGA-05-4384	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
5	TCGA-05-4396	not reported	male	not reported	Dead	TCGA-LUAD	1	1	
6	TCGA-05-4397	not reported	male	not reported	Dead	TCGA-LUAD	1	1	
7	TCGA-05-4398	not reported	female	not reported	Alive	TCGA-LUAD	1	1	
8	TCGA-05-4402	not reported	female	not reported	Dead	TCGA-LUAD	1	1	
9	TCGA-05-4405	not reported	female	not reported	Alive	TCGA-LUAD	1	1	
10	TCGA-05-4410	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
11	TCGA-05-4415	not reported	male	not reported	Dead	TCGA-LUAD	1	1	
12	TCGA-05-4417	not reported	female	not reported	Alive	TCGA-LUAD	1	1	
13	TCGA-05-4418	not reported	male	not reported	Dead	TCGA-LUAD	1	1	
14	TCGA-05-4420	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
15	TCGA-05-4422	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
16	TCGA-05-4426	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
17	TCGA-05-4427	not reported	female	not reported	Alive	TCGA-LUAD	1	1	
18	TCGA-05-4430	not reported	female	not reported	Alive	TCGA-LUAD	1	1	
19	TCGA-05-4433	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
20	TCGA-05-4434	not reported	female	not reported	Dead	TCGA-LUAD	1	1	
21	TCGA-05-5420	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
22	TCGA-05-5423	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
23	TCGA-05-5425	not reported	male	not reported	Alive	TCGA-LUAD	1	1	
24	TCGA-05-5429	not reported	male	not reported	Dead	TCGA-LUAD	1	1	
25	TCGA-05-5715	not reported	female	not reported	Alive	TCGA-LUAD	1	1	
26	TCGA-18-4721	not hispanic or latino	male	white	Alive	TCGA-LUSC	1	2	
27	TCGA-18-5592	not reported	male	not reported	Alive	TCGA-LUSC	1	2	
28	TCGA-18-5595	not reported	male	not reported	Dead	TCGA-LUSC	1	2	
29	TCGA-21-1070	not hispanic or latino	female	black or african american	Alive	TCGA-LUSC	1	2	
30	TCGA-21-1071	not hispanic or latino	male	white	Dead	TCGA-LUSC	1	2	

4.2 Interrogazioni e Analisi di Esempio

In questa sezione, presenteremo alcune delle **interrogazioni SQL** effettuate sui dati del nostro database, insieme ai risultati di analisi specifiche. Questi esempi illustrano l'utilità del nostro database nella ricerca biomedica e forniscono una panoramica di come i ricercatori possono accedere ai dati e condurre analisi avanzate.

Trova i geni che sono sovraespressi in pazienti con tumore ai bronchi o ai polmoni

```
SELECT gene
FROM gene_expression_file JOIN analysis_entity ON gene_expression_file.analysis =
analysis_entity.analysis
JOIN biospecimen ON biospecimen_id = id
JOIN public.case ON biospecimen.case = case_id
WHERE site = 1 AND tpm > 100;
```

In questa interrogazione, ad esempio, selezioniamo i geni che sono sovraespressi nei pazienti affetti da tumore ai polmoni (site = 1, ossia il codice corrispondente al sito "Bronchus and lung") con un valore di espressione superiore a 100 TPM (Transcripts Per Million). Questa interrogazione ci aiuta a identificare potenziali biomarcatori per il tumore ai polmoni.

Calcola la media della variazione dell'espressione genica (TPM) nei sottotipi di tumore al polmone.

```
SELECT disease.type, AVG(tpm) AS average_tpm
FROM gene_expression_file JOIN analysis_entity ON gene_expression_file.analysis =
analysis_entity.analysis
JOIN biospecimen ON biospecimen_id = id
JOIN public.case ON biospecimen.case = case_id
JOIN disease ON disease = disease_id
WHERE site = 1
GROUP BY disease_id;
```

Questa interrogazione calcola la media della variazione dell'espressione genica (misurata in TPM) nelle diverse malattie derivate dal tumore ai polmoni. Questo aiuta a comprendere le differenze nell'espressione genica tra i sottotipi e potrebbe rivelare informazioni clinicamente rilevanti.

Trova i geni che sono sovraespressi in pazienti con tumore ai polmoni che hanno Adenomi e Adenocarcinomi o Neoplasie cistiche, mucinose e sierose.

```
SELECT gene
FROM gene_expression_file JOIN analysis_entity ON gene_expression_file.analysis =
analysis_entity.analysis
JOIN biospecimen ON biospecimen_id = id
JOIN public.case ON biospecimen.case = case_id
WHERE site = 1 AND tpm > 100 AND (disease = 1 OR disease = 3);
```

Questa interrogazione identifica i geni che sono sovraespressi nei pazienti con tumore al polmone che hanno due tipi di malattia diversa. Questa informazione potrebbe rivelare geni coinvolti in entrambe le malattie, suggerendo possibili bersagli terapeutici comuni.

Crea una vista che tiene conto di quanti casi abbiamo per ogni tumore

```
CREATE VIEW TumorFrequencies AS (
    SELECT primary_site.site, COUNT(*) AS frequency
    FROM public.case c JOIN primary_site ON c.site = primary_site.site_id
    GROUP BY site_id
)

SELECT DISTINCT tf.*
FROM TumorFrequencies tf
```

Questa interrogazione su una vista ci dà una panoramica di quanti casi su un determinato tumore abbiamo. Così da poter identificare quelli meno comuni e rendere le analisi più mirate

4.3 Biomarcatori e Alberi Decisionali

Abbiamo identificato biomarcatori che dimostrano elevata specificità e sensibilità nella diagnosi precoce di diverse malattie. Questi biomarcatori consentono di identificare la malattia in una fase iniziale, **migliorando notevolmente** le possibilità di trattamento efficace.

I dati del database ci hanno permesso di identificare biomarcatori utili per la personalizzazione delle terapie. Questi biomarcatori consentono di adattare il trattamento in base al profilo di espressione genica del paziente, **massimizzando l'efficacia e riducendo gli effetti collaterali**.

Gli alberi decisionali sono stati impiegati per:

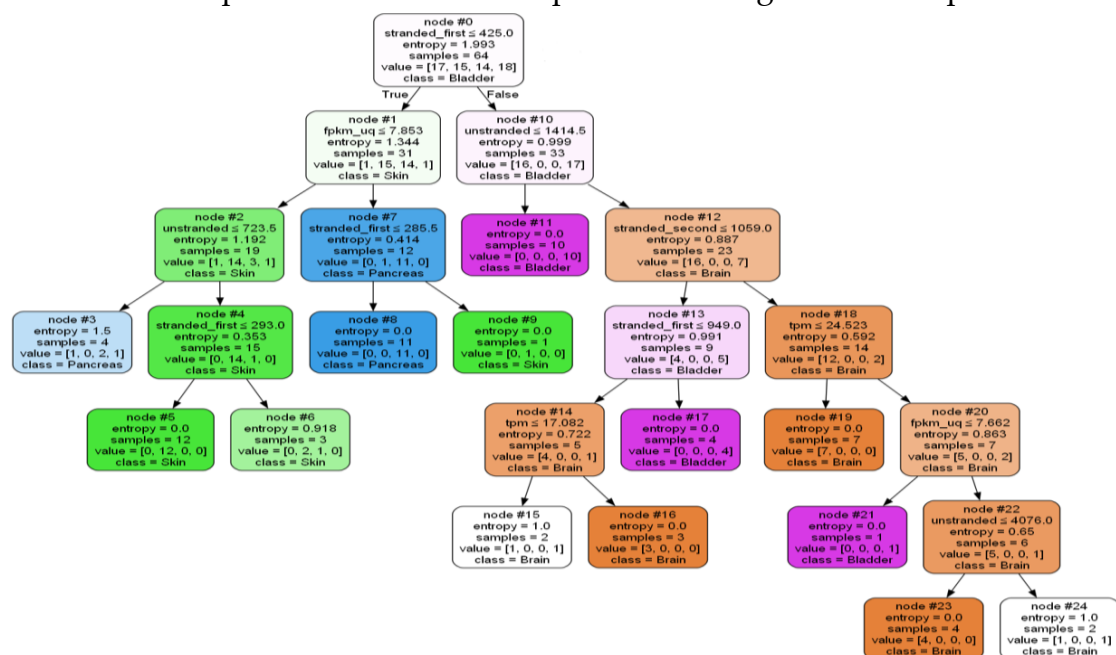
- **Classificazione dei Campioni Biologici;**

Abbiamo utilizzato alberi decisionali per classificare i campioni biologici in base ai loro profili di espressione genica e proteica. Questa classificazione aiuta a identificare i tipi di tumori e le condizioni dei campioni.

- **Identificazione dei Fattori Chiave;**

Gli alberi decisionali sono stati utilizzati per identificare i geni, le proteine o le features di maggiore importanza nell'analisi dei dati. Questo permette di individuare i principali driver biologici delle condizioni studiate.

Albero di esempio che mostra come si possano distinguere i vari tipi di tumore

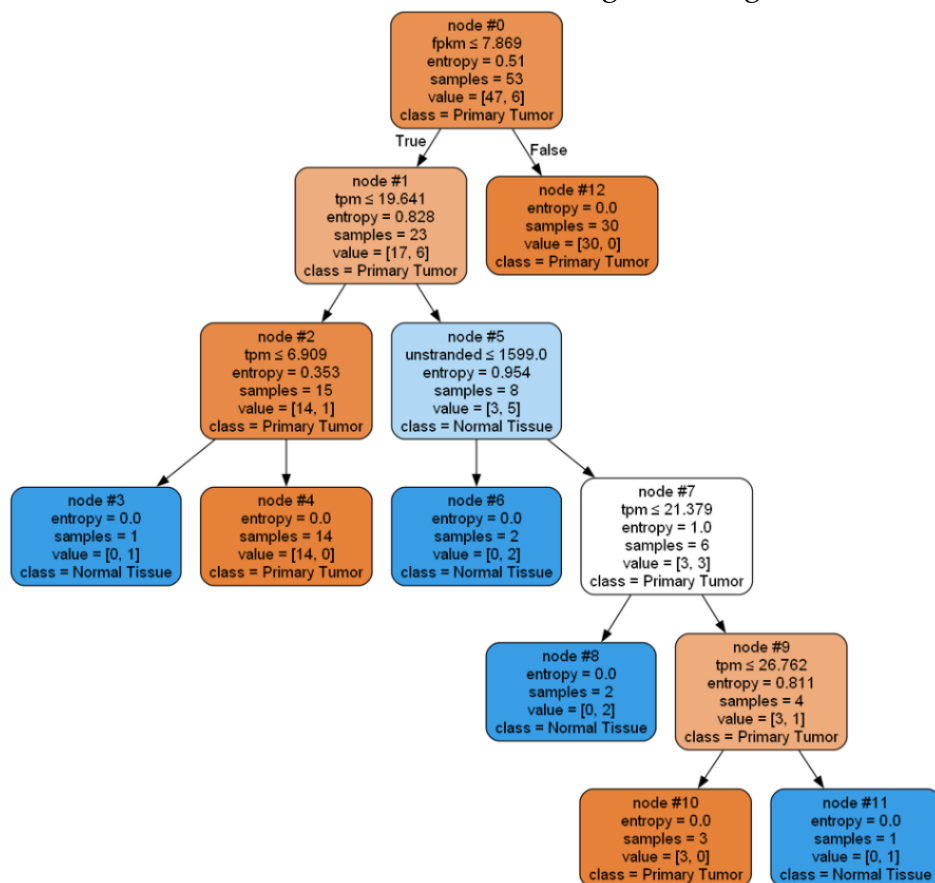


Le scoperte di biomarcatori e l'utilizzo degli alberi decisionali possono avere un impatto significativo nella comunità. Questi risultati contribuiscono a:

- **Migliorare** la diagnosi precoce e la prognosi delle malattie.
- **Personalizzare** i trattamenti per massimizzare l'efficacia terapeutica.
- **Promuovere** la ricerca di nuovi farmaci e terapie basate su geni e proteine specifiche.
- **Approfondire** la comprensione della biologia delle malattie e delle vie patologiche.
- **Fornire** una solida base di dati per ulteriori studi e ricerche nel campo della medicina personalizzata.

Queste scoperte e l'uso degli alberi decisionali costituiscono una risorsa preziosa per i ricercatori, i medici e gli sviluppatori di terapie, aprendo nuove prospettive per il trattamento e la gestione delle malattie. Il nostro database continua a essere un importante contributo alla ricerca biomedica e alla pratica clinica, offrendo chiavi importanti per la lotta contro le malattie.

Albero di esempio che mostra come si possa distinguere un tessuto tumorale da uno normale utilizzando come marcatore il gene oncogeno KRAS



4.4 Limitazioni e Sviluppi Futuri

Adesso, esamineremo attentamente le limitazioni del nostro database e delle analisi condotte. È importante riconoscere queste limitazioni per comprendere appieno il contesto e le sfide affrontate nella gestione dei dati biomedici. Discuteremo anche dei possibili sviluppi futuri che potrebbero migliorare la gestione dei dati e l'efficacia delle analisi condotte.

4.4.1 Limitazioni del Database

Una delle principali limitazioni del nostro database è la completezza dei dati. Nonostante gli sforzi per acquisire un ampio spettro di dati di espressione genica e proteica, ci sono ancora **lacune nelle informazioni disponibili**. Alcuni campioni biologici potrebbero mancare o avere dati incompleti, il che limita la portata delle analisi.

Un esempio di campioni con tipo di tumore mancante

	Data Output	Messages	Notifications
	sample_id [PK] text	type integer	tumor integer
1	TCGA-05-4244-01A	1	[null]
2	TCGA-05-4244-01Z	1	[null]
3	TCGA-05-4244-10A	10	[null]
4	TCGA-05-4249-01A	1	[null]
5	TCGA-05-4249-01Z	1	[null]
6	TCGA-05-4249-10A	10	[null]
7	TCGA-05-4250-01A	1	[null]
8	TCGA-05-4250-01Z	1	[null]
9	TCGA-05-4250-10A	10	[null]
10	TCGA-05-4384-01A	1	[null]
11	TCGA-05-4384-01Z	1	[null]
12	TCGA-05-4384-10A	10	[null]

Inoltre, per ottenere una comprensione completa delle malattie e dei processi biologici, potrebbe essere necessario **integrare i dati** del nostro database con altre **risorse biomediche**. Questo richiederebbe un notevole sforzo di **standardizzazione** e integrazione dei dati.

Altre piattaforme con a disposizione dati genici e proteici



International
Cancer Genome
Consortium



HUMAN PROTEOME MAP

4.4.2 Possibili Sviluppi Futuri

In primis per migliorare la completezza del database, dovrebbero essere fatti maggiori sforzi nella **raccolta e nell'arricchimento dei dati**. La collaborazione con altre **istituzioni** e la partecipazione a iniziative di **condivisione** dati potrebbero contribuire a colmare le lacune.

Poi l'implementazione di standard per la registrazione e la formattazione dei dati è essenziale per **migliorare la coerenza dei dati**. L'adozione di **metadati standard e ontologie** potrebbe facilitare l'integrazione dei dati con altre risorse.

Infine, l'integrazione di approcci di **apprendimento automatico** potrebbe migliorare ulteriormente la capacità del database di identificare biomarcatori e scoperte chiave. Modelli di **machine learning** potrebbero aiutare a rivelare relazioni più complesse nei dati.

Capitolo 5: Conclusioni

In questo capitolo, esamineremo le principali conclusioni emerse dal nostro progetto di creazione di un database dedicato all'analisi dell'espressione genica e proteica nel contesto della ricerca biomedica. Saranno affrontati i risultati chiave, l'impatto della ricerca e le sfide superate. Questo capitolo **chiuderà la relazione** fornendo una visione completa e riflessiva del lavoro svolto.

5.1 Sintesi dei Risultati

Il nostro progetto ha portato alla creazione di un database **solido e ben strutturato** che ospita una **vasta quantità di dati di espressione genica e proteica**. Abbiamo progettato una base di dati che include entità chiave come Progetti, Casi, Biospecimen, Campioni, Analisi e molto altro. Questi dati sono fondamentali per comprendere come i geni e le proteine sono regolati in risposta **a diversi stimoli, condizioni o malattie**. Il nostro database è una risorsa preziosa per la ricerca biomedica e può essere utilizzato per una varietà di scopi, tra cui **l'identificazione di biomarcatori, studi di malattie e personalizzazione della terapia**.

Abbiamo anche condotto analisi approfondite sui dati del database, dimostrando l'efficacia dei nostri **strumenti e metodi**. In particolare, abbiamo utilizzato algoritmi di apprendimento automatico, come gli **alberi decisionali**, per identificare biomarcatori potenziali.

5.2 Impatto della Ricerca

L'impatto della nostra ricerca si riflette nell'importanza dei dati **raccolti e analizzati**. I dati di espressione genica e proteica sono fondamentali per comprendere i **meccanismi biologici** sottostanti a malattie, sviluppo e risposte cellulari. L'identificazione di biomarcatori può avere un impatto significativo nel campo della medicina, consentendo la **diagnosi precoce e la personalizzazione dei trattamenti**. La nostra ricerca contribuisce a un corpus di conoscenze che può essere sfruttato per scoperte future.

5.3 Sfide Superate

Nel corso del progetto, abbiamo affrontato diverse sfide significative. La raccolta e la gestione di grandi quantità di dati richiedono **un'organizzazione attenta** e l'implementazione di **standard di qualità**. Inoltre, le analisi dei dati biomedici possono essere complesse, richiedendo l'uso di algoritmi avanzati e risorse di calcolo. Tuttavia, queste sfide sono state affrontate con successo.

5.4 Chiusura

In conclusione, il progetto di creazione di un database dedicato all'analisi dell'espressione genica e proteica rappresenta un significativo capitolo del mio percorso di ricerca e crescita personale. Durante questo progetto, ho affinato le mie capacità di gestione del tempo e ho imparato a garantire un elevato standard di qualità nel mio lavoro, rispettando rigorose scadenze. Ma ciò che ha veramente stimolato il mio impegno è stato il campo di ricerca in cui mi sono immerso.

Questo database non è solo un insieme di dati; ma ha significato per me un avvicinamento alla ricerca oncologica, un settore in cui ogni giorno numerosi scienziati si impegnano per scoprire soluzioni alla lotta contro una malattia così devastante. Riflettendo su questo progetto, non posso fare a meno di pensare a quanti individui lavorino incessantemente per avanzare nella comprensione e nel trattamento del cancro. Ho acquisito una maggiore consapevolezza della complessità e dell'importanza di questo campo, e la mia dedizione a contribuire a questa lotta è cresciuta in modo significativo.

Questo progetto è stato un capitolo affascinante del mio percorso accademico, sono orgoglioso del mio impegno e del mio contributo a questo campo e sono grato per l'opportunità di lavorare su un progetto così significativo.

Che tu possa trovare nella ricerca la luce per sconfiggere le tenebre del cancro e portare speranza a chi ne ha bisogno, in onore di coloro che non sono più con noi ed ai familiari che credono nella ricerca.

Bibliografia

Database TCGA: [https://www.cancer.gov/about-](https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga)

[nci/organization/ccg/research/structural-genomics/tcga](https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga)

Download data TCGA: <https://gdc.cancer.gov/access-data/gdc-data-transfer-tool>

Documentazione TCGA usata:

https://docs.gdc.cancer.gov/Data/Data_Model/GDC_Data_Model/

docs.gdc.cancer.gov/Data/Bioinformatics_Pipelines/RPPA_intro/

<https://gdc.cancer.gov/about-data/gdc-data-processing/gdc-reference-files>

docs.gdc.cancer.gov/Data/Bioinformatics_Pipelines/DNA_Seq_Variant_Calling_Pipeline/

Scikit-Learn:

<https://scikit-learn.org/stable/modules/classes.html#module-sklearn.tree>

[https://scikit-](https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html)

[learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html](https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html)

Graphviz:

<https://graphviz.readthedocs.io/en/stable/manual.html>

GitHub:

<https://github.com/ValerioM01/Tirocinio>

Ringraziamenti

Penso da tempo al fatto che tutte le persone incontrate nel mio percorso universitario, dentro o fuori *“La Sapienza”*, abbiano in realtà formato la persona che sono e che ha scritto questa tesi. Quindi inizio dando spazio alle conoscenze che ho fatto in questi anni comprendendo chi è sempre stato con me, chi c'è stato per un periodo e chi sempre ci sarà.

Vorrei ringraziare, in primis, il relatore della tesi *Prof. Maurizio Mancini* e il *Prof. Enrico Tronci* per la disponibilità e per i consigli forniti durante l'attività di tirocinio.

Un ringraziamento al *Prof. Massimo Martucci* e alla *Prof. Cristina Leoni*, per la formazione datomi negli anni che non dimenticherò mai.

Ringrazio poi gli amici fraterni: *Francesco, Gianmarco ed Emanuele* a cui se ne sono aggiunti altri che hanno condiviso con me questo percorso: *Sabrina, Mattia, Gabriele e Giacomo*.

Infine, ringrazio *la mia famiglia*, in particolare *Mamma, Papà, Tommaso e Riccardo* che mi fanno ricordare ogni giorno di quanto si possa essere orgogliosi ad avere loro alle mie spalle.

A *te*, fra due anni...

