



UNIVERSITÀ⁹ DEGLI STUDI DI
NAPOLI FEDERICO II

Scuola Politecnica e delle Scienze di Base
Corso di Laurea Magistrale in Ingegneria Informatica

Homework 1 - Big Data Engineering

*Clinical Trials Dataset (PySpark
Analytics)*

Anno Accademico 2024/2025

Professore
prof. Vincenzo Moscato

Candidati
Vincenzo Luigi Bruno matr. M63001670
Salvatore Cangiano matr. M63001647

Contents

1	Dataset e Preprocessing	1
1.1	Descrizione del Dataset	1
1.2	Preprocessing	4
1.2.1	Conversione Dizionari	5
1.2.2	Conversione Liste	5
1.2.3	Conversione età	6
2	Analytics	8
2.1	Numero di studi avviati per anno	8
2.1.1	Implementazione della query	8
2.1.2	Risultati	8
2.2	Numero medio di partecipanti per studio	11
2.2.1	Implementazione della query	11
2.2.2	Risultati	11
2.3	Top 10 condizioni mediche più studiate	12
2.3.1	Implementazione della Query	12
2.3.2	Risultati	13
2.4	Nazione con la media di partecipanti più alta, per tipo di studio . .	14
2.4.1	Implementazione della query	14
2.4.2	Risultati	15
2.5	Città che hanno trattato maggiormente una determinata condizione	17
2.5.1	Implementazione della query	17
2.5.2	Risultati	18

2.6	Nazioni che sponsorizzano il maggior numero di studi clinici rivolti alle donne	19
2.6.1	Implementazione della query	19
2.6.2	Risultati	19
2.7	Nazioni che sponsorizzano il maggior numero di studi che coinvolgono minori	20
2.7.1	Implementazione della query	20
2.7.2	Risultati	21
2.8	Città che ricercano più attivamente uno specifico tipo di cancro . .	22
2.8.1	Implementazione della query	22
2.8.2	Risultati	23
2.9	Studi clinici che si sono conclusi, con relative condizioni mediche trattate	25
2.9.1	Implementazione della query	25
2.9.2	Risultati	26
2.10	Studi clinici con maggiore visibilità mediatica (Altmetric score) . .	27
2.10.1	Implementazione della query	27
2.10.2	Risultati	27
2.11	Condizioni con media più alta di Altmetric Score	29
2.11.1	Implementazione della query	29
2.11.2	Risultati	29
2.12	Nazioni delle organizzazioni finanziarie che hanno sponsorizzato il maggior numero di studi clinici	31
2.12.1	Implementazione della query	31
2.12.2	Risultati	32
2.13	Ricercatori/Contatti più attivi con relativa patologia più studiata .	33
2.13.1	Implementazione della query	33
2.13.2	Risultati	34
2.14	Top collaborazioni dell'Università di Napoli Federico II con enti finanziatori	36

CONTENTS

2.14.1 Implementazione della query	36
2.14.2 Risultati	37
2.15 Word Cloud dei Titoli Brevi	39
2.15.1 Implementazione della word cloud	39
2.15.2 Risultati	39
2.16 Top collaborazione per ente finanziatore	41
2.16.1 Implementazione della query	41
2.16.2 Risultati	41

Chapter 1

Dataset e Preprocessing

1.1 Descrizione del Dataset

Il dataset utilizzato in questo esercizio proviene da *Dimensions.ai*, una piattaforma che aggrega dati relativi alla ricerca scientifica a livello globale. Ogni riga del dataset rappresenta uno *studio clinico* (clinical trial), contenente informazioni strutturate riguardanti l'identificativo, i dettagli dello studio, i soggetti coinvolti, le condizioni mediche trattate, nonché dati su sponsor, finanziatori e metriche di impatto.

I campi del dataset sono:

1. **Rank**: posizione numerica o classifica assegnata a ciascun trial.
2. **Trial ID**: identificatore univoco del trial clinico.
3. **Title**: titolo completo del trial
4. **Brief title**: titolo breve o sintetico del trial.
5. **Acronym**: acronimo associato al trial (se disponibile).
6. **Abstract**: sommario o sinossi dello studio, con obiettivi, metodologia e risultati principali (se disponibili).

7. **Start date:** data di inizio del trial clinico.
8. **Start Year:** anno in cui è iniziato il trial.
9. **End Date:** data di fine prevista o effettiva dello studio.
10. **Completion Year:** anno di completamento del trial.
11. **Phase:** fase dello studio clinico (1-4).
12. **Study Type:** tipo di studio (es. Interventistico, Osservazionale).
13. **Study Design:** descrizione del progetto sperimentale.
14. **Conditions:** elenco delle patologie o condizioni mediche esaminate.
15. **Recruitment Status:** stato del reclutamento dei partecipanti (es. in corso, completato, terminato anticipatamente).
16. **Number of Participants:** numero previsto o reale di partecipanti coinvolti.
17. **Intervention:** descrizione dell'intervento.
18. **Gender:** genere dei partecipanti target (maschi, femmine, entrambi).
19. **Age:** fasce d'età dei partecipanti coinvolti.
20. **Registry:** nome del registro dove è registrato il trial (es. ClinicalTrials.gov).
21. **Investigators/Contacts:** informazioni sui principali ricercatori e/o contatti associati allo studio.
22. **Sponsors/Collaborators:** enti o organizzazioni che sponsorizzano e/o collaborano allo studio.
23. **City of Sponsor/Collaborator:** città associata allo sponsor o collaboratore.

24. **State of Sponsor/Collaborator:** stato o provincia dello sponsor/collaboratore.
25. **Country of Sponsor/Collaborator:** paese dello sponsor/collaboratore.
26. **Collaborating Funders:** altri enti finanziatori che collaborano al trial.
27. **Funder Group:** gruppi o categorie di appartenenza degli enti finanziatori.
28. **Funder Country:** paesi di origine degli enti finanziatori.
29. **Source Linkout:** link esterno alla fonte originale o scheda ufficiale dello studio.
30. **Altmetric Attention Score:** punteggio che misura l'attenzione mediatica ricevuta (social media, news, blog).
31. **Dimension URL:** link alla piattaforma Dimensions.ai (per metadati e metriche bibliometriche).
32. **Fields of Research (ANZSRC 2020):** aree di ricerca secondo la classificazione ANZSRC 2020 (Australian & New Zealand Standard Research Classification).
33. **RCDC Categories:** categorie di ricerca secondo il sistema RCDC (Research, Condition, and Disease Categorization) del NIH.
34. **HRCS HC Categories:** categorie sanitarie secondo il sistema HRCS (UK Health Research Classification System) – Health Categories.
35. **HRCS RAC Categories:** Categorie di allocazione delle risorse secondo HRCS – Research Activity Codes.
36. **Cancer Types:** tipi di cancro studiati (se applicabile).
37. **CSO Categories:** categorie secondo il sistema CSO (Common Scientific Outline), usato nella ricerca sul cancro.

38. **AHC**: informazioni relative ai Centri Accademici di Salute (Academic Health Centers), se disponibili.

Questo dataset consente un'analisi multidimensionale della ricerca clinica globale, includendo aspetti scientifici, organizzativi e sociali dei trial.

Il dataset presenta quindi **38 colonne e 15990 righe**.

1.2 Preprocessing

Molte colonne del dataset si presentano in un formato **pseudo-strutturato**. Ad esempio il campo **Condition** è una stringa di molte condizioni separate dal carattere ";". Di fatto le colonne come Condition, non sono esplicitamente formattate con il Type che esse vogliono rappresentare. PySpark però mette a disposizione la possibilità di assegnare campi strutturati come **ArrayList** per le liste e **MapType** per i dizionari, pertanto per semplificare il processo di analisi e implementazione delle query si è deciso di convertire il tipo di queste colonne.

Data Type	Field Name
<i>Dictionary</i>	Study Design
	Conditions
	Investigators/Contacts
	Sponsors/Collaborators
	City of Sponsor/Collaborator
	State of Sponsor/Collaborator
	Country of Sponsor/Collaborator
	Fields of Research (ANZSRC 2020)
<i>List</i>	Collaborating Funders
	Funder Group
	Funder Country
	RCDC Categories
	HRCS HC Categories
	HRCS RAC Categories
	Cancer Types
	CSO Categories
<i>List of two elements</i>	Age

1.2.1 Conversione Dizionari

L'unica colonna che è un dizionario nello schema è **Study Design**, la rappresentiamo in pyspark con il tipo **MapType** (chiave valore). In particolare la stringa iniziale si presenta come una lista di coppie chiave valore, le coppie sono separate da ";" mentre nella coppia la chiave e il valore sono separati dal carattere ":". Il processo di conversione è il seguente:

1. Usiamo la funzione di pyspark **udf** (User defined function), ci permette di creare una logica personalizzata da applicare alle colonne
2. Creiamo la funzione **string_to_map** che:
 - creiamo un dizionario vuoto **result**
 - separiamo in parti la stringa utilizzando come separatore ";"
 - iteriamo sulle parti così create e separiamo ulteriormente con ":"
 - inseriamo le chiavi e i valori estratti all'interno di **result**
3. Passiamo l'oggetto richiamabile creato alla funzione udf, passando anche il tipo di ritorno che vorremmo, ossia **MapType**

```
def string_to_map(text):
    if text is None:
        return None
    result = {}
    parts = text.split(';')
    for part in parts:
        if part:
            key, value = part.split(':', 1)
            result[key.strip()] = value.strip()
    return result

string_to_map_udf = udf(string_to_map, MapType(StringType(), StringType()))
ctDS = ctDS.withColumn("Study Design", string_to_map_udf(col("Study Design")))
```

Figure 1.1: Implementazione conversione dizionari a MapType

1.2.2 Conversione Liste

Similmente ai dizionari, le colonne che sono pseudo-liste sono stringhe di valori separati dal carattere ;. Tutte le colonne con questo comportamento sono state

elencate nella precedente tabella degli pseudo tipi. Il processo di conversione è il seguente:

1. Una lista di stringhe, dichiarata columnsList, contiene le colonne interessate
2. creiamo la logica applicativa, la funzione **string_to_array** da applicare a ciascuna colonna
3. La funzione divide in parti delimitate da ";", mantenendo valori **None** quando il valore è mancante.
4. Applichiamo la trasformazione a tutta la columnsList

```
# All columns that are pseudo-lists
columnsList = ["Conditions","Investigators/Contacts","Sponsors/Collaborators","City of Sponsor/Collaborator","State of
Sponsor/Collaborator","Country of Sponsor/Collaborator","Collaborating Funders","Fields of Research (ANZSRC 2020)","RCDC Categories","HRGS
HC Categories","HRGS RAC Categories","Cancer Types","ESD Categories","Funder Group","Funder Country"]

def string_to_array(text):
    if text is None:
        return None
    items = text.split(';')
    return [item.strip() if item.strip() != '' else None for item in items]

string_to_array_udf = udf(clean_split_with_none, ArrayType(StringType()))

for column in columnsList:
    ctDS = ctDS.withColumn(column, string_to_array_udf(col(column)))
```

Figure 1.2: Implementazione conversione Liste a ArrayType

1.2.3 Conversione età

Il campo età ha subito ha una struttura peculiare che ha necessitato di una trasformazione più complessa. Si presenta come una stringa, che rappresenta l'**età minima** per partecipare al clinical trial e l'**età massima**. Il problema principale è che possiamo avere situazioni in cui il minimo e il massimo sono entrambi in formato **Years**, ma anche in formato **Month** oppure **Days**. Si è deciso quindi di includere nella trasformazione anche un processo di conversione di tutti formati al singolo formato **Years**. In particolare:

1. Definiamo la funzione **age_preprocessing** da passare a udf
2. La funzione divide in parti la stringa passata utilizzando come delimitatore il carattere **"-"**

3. Il preprocessing del formato età è il seguente:

- Se una delle due parti contiene "N/A" sostituiamo con valore **None**
- Altrimenti estraiamo un numero con una funzione regex e lo convertiamo in anni, tenendo conto dell'unità di misura iniziale (e.g. mesi, ore).

4. Registriamo la udf con il tipo di ritorno **ArrayType**, in particolare un array di **FloatType**

```
g(text):

    """
    If 'text' contains 'N/A - N/A', return an array with None, None
    "N/A - N/A":
    [None, None]
    """

    [None] #Initialization with [None, None]
    ["-"]

    | have exactly two parts
    :
    | minimum value (parts[0])
    | s[0];
    | None
    |
    | number
    | re.search(r'\d+(\.\d+)?', parts[0])
    | :
    | t(number_match.group())
    |
    | years
    | in parts[0].lower() or "months" in parts[0].lower():
    |     ] = value / 12 # Convert months to years
    | in parts[0].lower() or "hours" in parts[0].lower():
    |     ] = value / (24 * 365) # Convert hours to years
    | in parts[0].lower() or "days" in parts[0].lower():
    |     ] = value / 365 # Convert days to years
    | in parts[0].lower() or "weeks" in parts[0].lower():
    |     ] = value / 52 # Convert weeks to years
    |
    | ] = value # We assume that it is already in years

    maximum value (parts[1])
    | s[1]:
    | None
    |
    | = re.search(r'\d+(\.\d+)?', parts[1])
    | tch:
    | float(number_match.group())
    |
    | in parts[1].lower() or "months" in parts[1].lower():
    |     t[1] = value / 12
    | in parts[1].lower() or "hours" in parts[1].lower():
    |     t[1] = value / (24 * 365)
    | in parts[1].lower() or "days" in parts[1].lower():
    |     t[1] = value / 365
    | in parts[1].lower() or "weeks" in parts[1].lower():
    |     t[1] = value / 52
    |
    | t[1] = value

f = udf(age_preprocessing, ArrayType(FloatType()))

# The "Age" column.
array("Age", age_preprocessing_udf(col("Age")))

```

Figure 1.3: Implementazione conversione età in formato ArrayType di FloatType

Chapter 2

Analytics

2.1 Numero di studi avviati per anno

2.1.1 Implementazione della query

L'obiettivo della query è analizzare quanti **studi clinici** sono stati **iniziat i per ogni anno**.

In particolare:

- Si raggruppano i dati per anno di inizio (*Start Year*).
- Si conteggiano gli studi per ciascun anno.
- Si ordina il risultato in ordine decrescente.

2.1.2 Risultati

Dalla tabella risultante è possibile osservare alcuni trend:

- Il maggior numero di studi si è registrato negli anni 2019-2021.
- Vi è un calo nel numero di studi invece nel periodo successivo (2023-2025) che può essere parzialmente spiegato dal fatto che i dati potrebbero essere **incompleti o ancora in aggiornamento**.



```

studiesPerYear = ctDS.select("Start Year") \
    .filter(ctDS["Start Year"].isNotNull()) \
    .groupBy(ctDS["Start Year"]) \
    .count() \
    .withColumnRenamed("count","NumStudies per Year") \
    .orderBy(col("NumStudies per Year").desc())

studiesPerYear.show(50,truncate = False)

```

Figure 2.1: Numero di studi avviati per anno

- Anni precedenti al 2000: pochissimi studi, probabilmente perché la raccolta digitale dei dati era limitata o ancora assente.

Questa analitica è utile poiché permette di identificare periodi di maggiore attività nella ricerca clinica e valutare l'impatto degli eventi globali, come ad esempio, l'incremento negli anni 2020-2021 che può essere legato alla pandemia COVID-19 e alla corsa alla sperimentazione clinica.

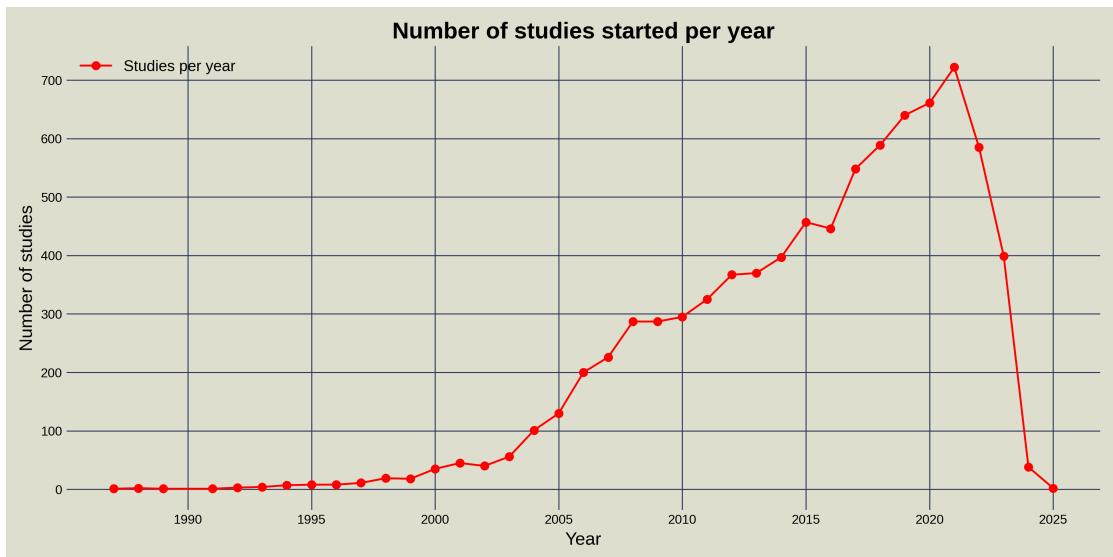


Figure 2.2: Line Plot

Start Year	NumStudies per Year
2021.0	722
2020.0	661
2019.0	640
2018.0	589
2022.0	585
2017.0	548
2015.0	457
2016.0	446
2023.0	399
2014.0	397
2013.0	370
2012.0	367
2011.0	325
2010.0	295
2009.0	287
2008.0	287
2007.0	226
2006.0	200
2005.0	130
2004.0	101
2003.0	56
2001.0	45
2002.0	40
2024.0	38
2000.0	35
1998.0	19
1999.0	18
1997.0	11
1995.0	8
1996.0	8
1994.0	7
1993.0	4
1992.0	3
1988.0	2
2025.0	2
1987.0	1
1989.0	1
1991.0	1

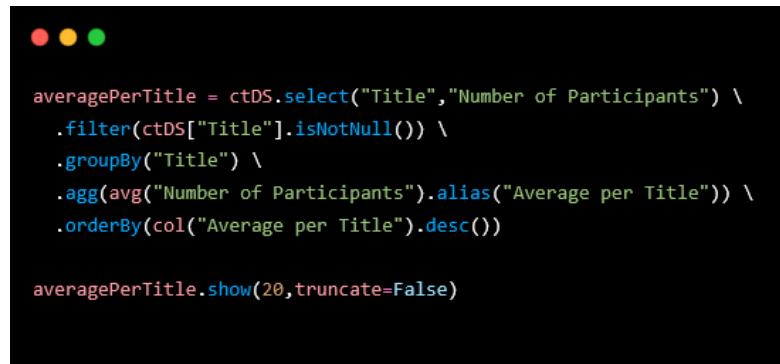
2.2 Numero medio di partecipanti per studio

2.2.1 Implementazione della query

L'obiettivo di questa analitica è calcolare il **numero medio di partecipanti** per ogni **studio** (identificato dal suo *Titolo*)

In particolare:

- Si raggruppano i dati per titolo dello studio (*Title*)
- Si calcola la **media** del numero di partecipanti per ciascun titolo
- Si ordinano i risultati in modo decrescente.



```
averagePerTitle = ctDS.select("Title", "Number of Participants") \
    .filter(ctDS["Title"].isNotNull()) \
    .groupBy("Title") \
    .agg(avg("Number of Participants").alias("Average per Title")) \
    .orderBy(col("Average per Title").desc())

averagePerTitle.show(20, truncate=False)
```

Figure 2.3: Numero medio di partecipanti per studio

2.2.2 Risultati

Questa analitica è utile per individuare gli studi di grande impatto, dato che spesso gli studi con un elevato numero medio di partecipanti sono associati a **grande rilevanza clinica** ed **impatto sanitario**.

In particolare osserviamo che i primi due studi che hanno più partecipanti in media sono legati alla pandemia di COVID-19, in particolare sono studi osservazionali di vasta scala e monitoraggi post-vaccinazione.

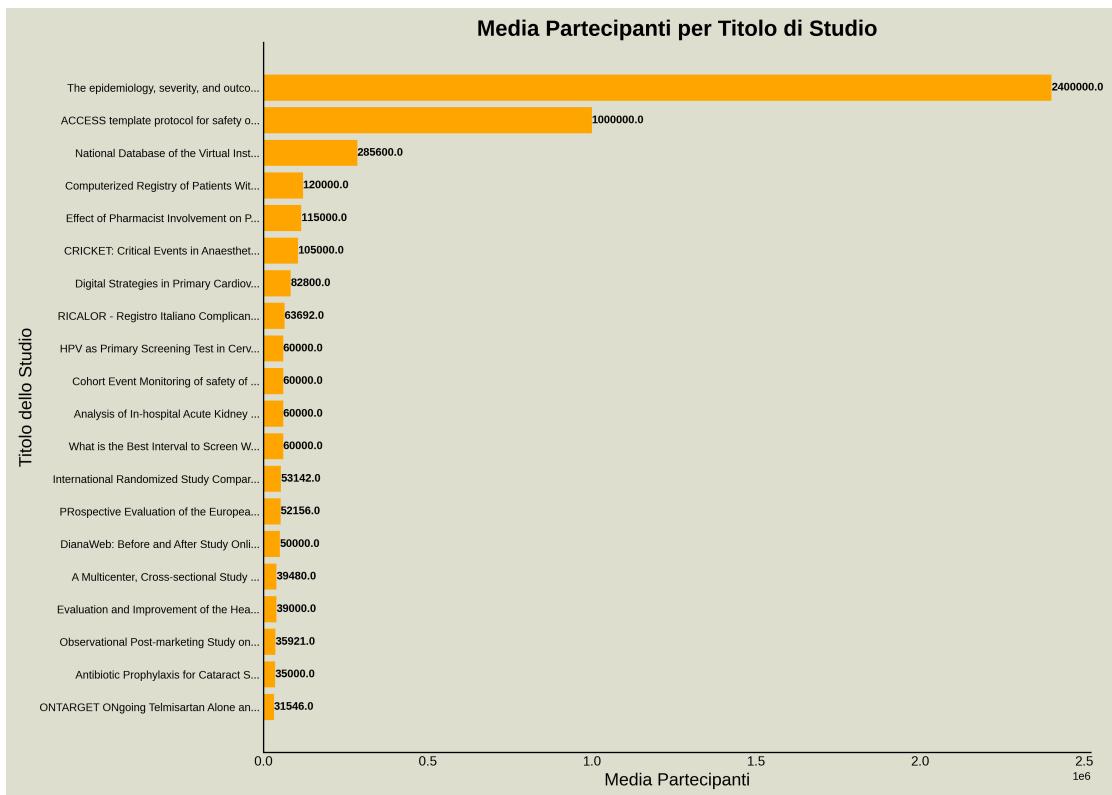


Figure 2.4: Bar Chart Orizzontale

Title	Average per Title
The epidemiology, severity, and outcomes of children presenting to emergency departments across Europe during the SARS-CoV-2 pandemic: the EPISODES study	2400000
ACCESS template protocol for safety of COVID-19 vaccines	1000000
National Database of the Virtual Institute of Cerebrovascular Diseases	285600
Computerized Registry of Patients With Venous Thromboembolism (RIETE)	120000
Effect of Pharmacist Involvement on Patient Reporting of Adverse Drug Reactions: A Multiregional Italian Study	115000
CRICKET: Critical Events in Anaesthetised Kids Undergoing Tracheal Intubation	105000
Digital Strategies in Primary Cardiovascular Prevention in the Italian Population	82800
RICALOR - Registro Italiano Complicanze Anesthesia LOCO Regionale - Italian Registry for Complications During Regional Anesthesia	63692
HPV as Primary Screening Test in Cervical Cancer	60000
Cohort Event Monitoring of safety of ...	60000
Analysis of In-hospital Acute Kidney Injury Epidemiology, Treatment and Outcomes	60000

2.3 Top 10 condizioni mediche più studiate

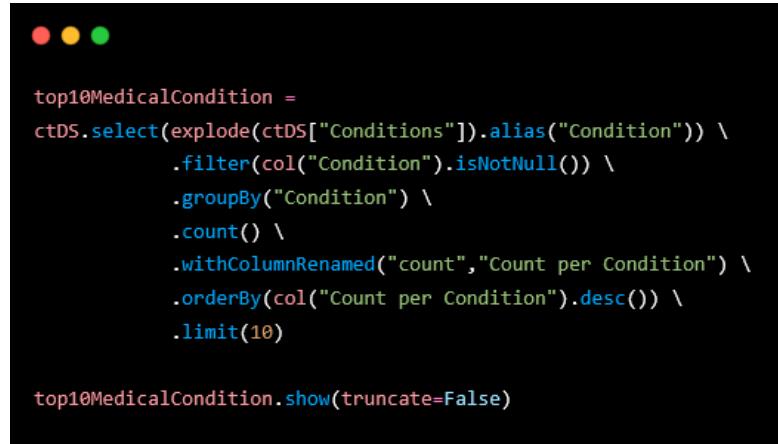
2.3.1 Implementazione della Query

Questa analitica serve a determinare le **10 condizioni mediche più frequentemente studiate**.

In particolare:

- Si "esplode" la lista di condizioni mediche (nel campo *Conditions*) in righe singole: se uno studio riguarda più condizioni, ognuna avrà una riga separata.
- Si raggruppano le righe per condizioni mediche.

- Si conta quante volte appare ciascuna condizione.
- Si ordina in ordine decrescente di frequenza.
- Si prendono solo le 10 condizioni più frequenti.



```
top10MedicalCondition =  
    ctDS.select(explode(ctDS["Conditions"]).alias("Condition")) \  
        .filter(col("Condition").isNotNull()) \  
        .groupBy("Condition") \  
        .count() \  
        .withColumnRenamed("count","Count per Condition") \  
        .orderBy(col("Count per Condition").desc()) \  
        .limit(10)  
  
top10MedicalCondition.show(truncate=False)
```

Figure 2.5: Top 10 condizioni mediche più studiate

2.3.2 Risultati

Notiamo come ci sia forte interesse clinico e di ricerca nelle condizioni di *cancro al seno* o *mieloma multiplo* e che ben 6 delle 10 condizioni rientrano nell'ambito oncologico:

- *Breast, ovarian, lung, colorectal, melanoma, prostate.*

Ciò conferma la centralità del **cancro** nella ricerca biomedica per impatto, complessità ed investimento.

Quest'analitica è utile ai ricercatori e enti finanziatori come supporto alla pianificazione strategica della ricerca, dato che è molto semplice identificare le aree di alta concentrazione di studi in maniera tale da evitare duplicazioni ed investire in settori sottorappresentati.

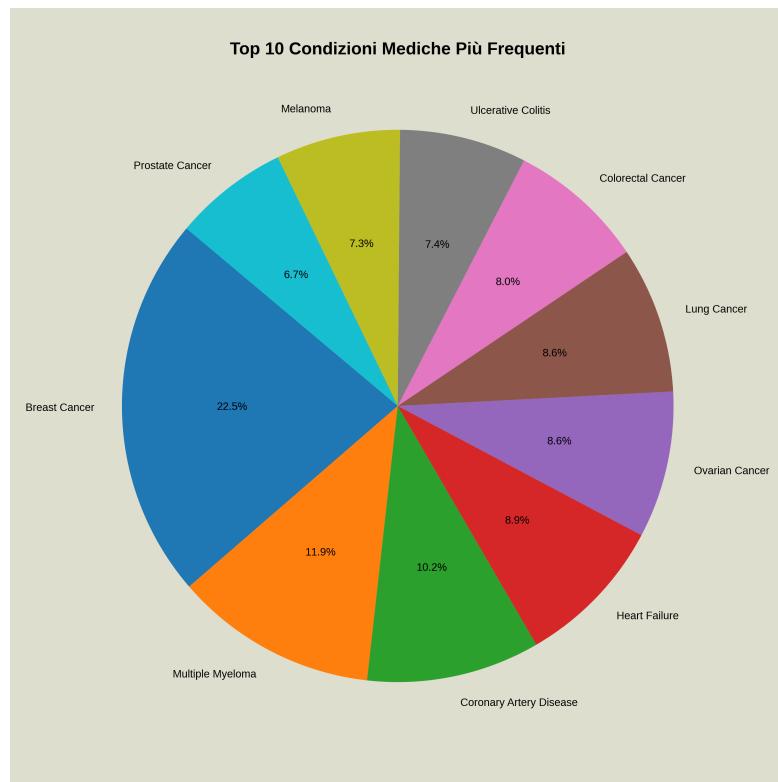


Figure 2.6: Top 10 condizioni mediche più studiate

Condition	Count per Condition
Breast Cancer	157
Multiple Myeloma	83
Coronary Artery Disease	71
Heart Failure	62
Ovarian Cancer	60
Lung Cancer	60
Colorectal Cancer	56
Ulcerative Colitis	52
Melanoma	51
Prostate Cancer	47

2.4 Nazione con la media di partecipanti più alta, per tipo di studio

2.4.1 Implementazione della query

Questa analitica serve ad identificare i paesi con la più alta media di partecipanti ai trial, per ogni categoria di studio nel dataset. In particolare:

- Crea una nuova colonna contenente solo i paesi unici coinvolti per ogni studio.
- Si genera una riga per ogni coppia (Tipo di studio, Paese), mantenendo anche il numero di partecipanti dello studio.
- Si calcola la media dei partecipanti per ciascuna combinazione di tipo di studio e paese.
- Si crea una finestra per classificare i paesi all'interno di ogni tipo di studio, ordinati per media decrescente.
- Si assegna un ranking e si seleziona solo il primo paese (con la media più alta) per ogni tipo di studio.

```

● ● ●

allAvgParticipants = ctDS.withColumn("Unique_Countries", array_distinct(col("Country of Sponsor/
Collaborator"))) \
    .select("Study Type", explode(col("Unique_Countries")).alias("Country"), "Number of Participants") \
    .filter(col("Study Type").isNotNull() & col("Country").isNotNull() & col("Number of
Participants").isNotNull()) \
    .groupBy("Study Type", "Country") \
    .agg(avg("Number of Participants").alias("Average per Type/Country")) \
    .orderBy(col("Average per Type/Country").desc()) \
    .windowSpec = Window.partitionBy("Study Type").orderBy(col("Average per Type/Country").desc())

maxCountryAvgPerStudyType = allAvgParticipants.withColumn("rank", row_number().over(windowSpec)) \
    .filter(col("rank")==1) \
    .select("Study Type", "rank", "Country", "Average per Type/Country")

```

Figure 2.7: Nazione con la media di partecipanti più alta, per tipo di studio

2.4.2 Risultati

Questa analitica è utile ad individuare paesi leader per specifiche modalità di ricerca clinica.

Dai risultati osserviamo che l'Islanda domina in due categorie:

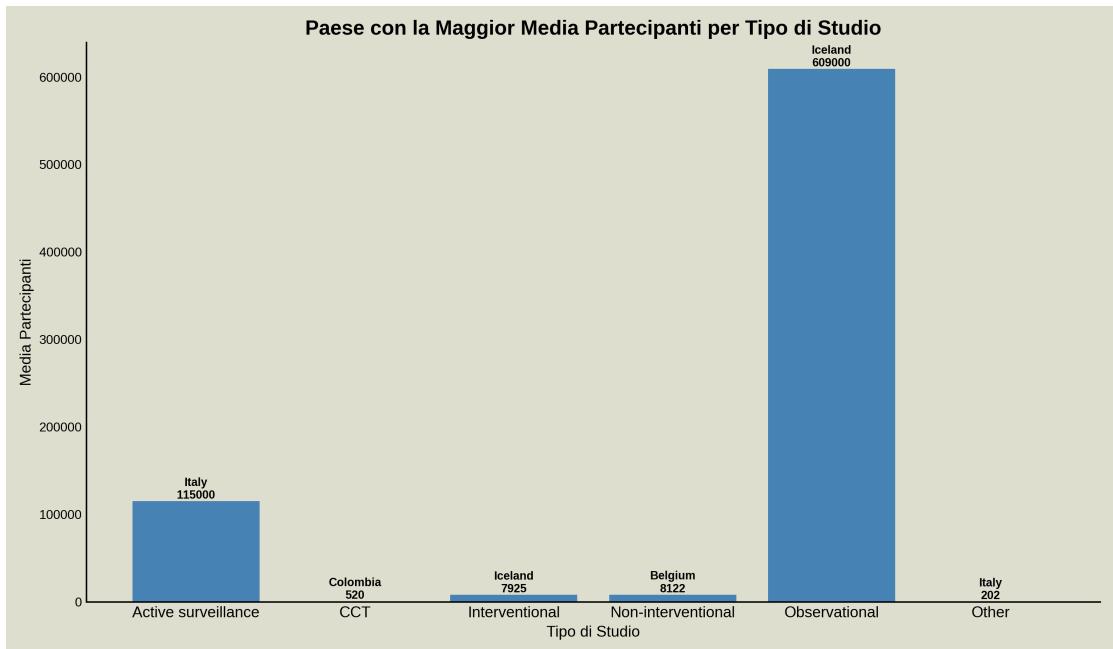


Figure 2.8: Bar chart - Nazione con la media di partecipanti più alta, per tipo di studio

Study Type	Country	Average per Type/Country
Active surveillance	Italy	115000.0
CCT	Colombia	520.0
Interventional	Iceland	7924.888888888889
Non-interventional	Belgium	8122.5
Observational	Iceland	609000.0
Other	Italy	202.0

- *Observational* con una media gigantesca (609.000): probabilmente uno studio nazionale su larga scala.
- *Interventional* con quasi 8.000.

2.5 Città che hanno trattato maggiormente una determinata condizione

2.5.1 Implementazione della query

Questa analitica serve ad identificare per ogni condizione medica, la città che ha condotto il maggior numero di studi clinici.

In particolare:

- Si "esplodono" le città e le condizioni mediche per ogni studio, così da trattarle separatamente. Usiamo inoltre la funzione *array_distinct* per evitare città duplicate all'interno dello stesso record.
- Per ogni condizione, si ordinano le città per numero di studi clinici in modo decrescente. Questo ci servirà a prendere soltanto la città al primo posto per una determinata condizione.
- Si conta il numero di studi per ogni coppia città-condizione.
- Si filtra solo per il primo classificato (cioè la città con più studi per quella condizione).
- Si scartano i risultati con un solo studio o con città nulla.



```
cityCondition = cityConditionDS.select("City", "Condition") \
    .groupBy("City", "Condition") \
    .count() \
    .withColumnRenamed("count", "NumStudies per City/Condition") \
    .withColumn("rank", row_number().over(windowSpecCityCondition)) \
    .filter((col("rank") == 1) & (col("City").isNotNull()) &
    (col("NumStudies per City/Condition") > 1)) \
    .select("City", "Condition", "NumStudies per City/Condition")
```

Figure 2.9: Città che hanno trattato maggiormente una determinata condizione

2.5.2 Risultati

Questo tipo di analitica è molto utile per individuare quelli che sono i centri di eccellenza. Infatti le città evidenziate possono essere considerate hub clinici per specifiche patologie.

Es.:

- Roma per *Leucemia Mieloide Acuta* (29 studi)

City	Condition	NumStudies per City/Condition
Pavia	AL Amyloidosis	6
Milan	ALS	3
Naples	Acid Maltase Deficiency	2
Orlando	Acquired Immunodeficiency Syndrome	2
Yorkville	Acral Lentiginous Melanoma	2
Brno	Acute Ischemic Stroke	2
Rome	Acute Myeloid Leukemia	29
Brussels	Acute Severe Respiratory Failure	2
Barcelona	Acute-On-Chronic Liver Failure	2
Ann Arbor	Adenocarcinoma	2

Vengono mostrati solo i primi 10 risultati per motivi di spazio.

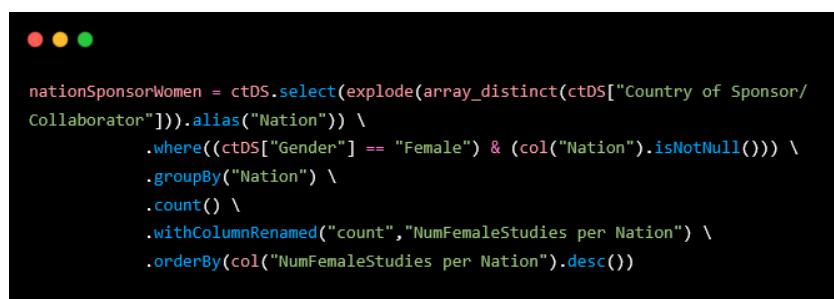
2.6 Nazioni che sponsorizzano il maggior numero di studi clinici rivolti alle donne

2.6.1 Implementazione della query

L'obiettivo di questa analitica è determinare quali nazioni sponsorizzano più studi clinici dedicati al genere femminile (*Gender == "Female"*), basandosi sulle informazioni relative agli sponsor/collaboratori.

In particolare:

- Si estraggono le nazioni "esplodendo" ogni voce della lista di paesi sponsor (*Country of Sponsor/Collaborator*) evitando duplicati.
- Si selezionano solo gli studi che indicano come target il genere femminile.
- Si raggruppa per nazioni e si conta quanti studi per il genere femminile sono sponsorizzati da ogni nazione.
- Si ordinano le nazioni in base al numero di studi in modo decrescente.



```
nationSponsorWomen = ctDS.select(explode(array_distinct(ctDS["Country of Sponsor/Collaborator"])).alias("Nation")) \
    .where((ctDS["Gender"] == "Female") & (col("Nation").isNotNull())) \
    .groupBy("Nation") \
    .count() \
    .withColumnRenamed("count","NumFemaleStudies per Nation") \
    .orderBy(col("NumFemaleStudies per Nation").desc())
```

Figure 2.10: Nazioni che sponsorizzano il maggior numero di studi clinici rivolti alle donne

2.6.2 Risultati

Analizzando i risultati, notiamo che in questo dataset l'Italia è leader nella sponsorizzazione di studi clinici focalizzati sulle donne, con un numero significativamente più alto rispetto agli altri paesi.

Nation	NumFemaleStudies per Nation
Italy	582
United States	176
Spain	133
Belgium	109
France	106
Germany	102
United Kingdom	102
Canada	75
Poland	71
Netherlands	59

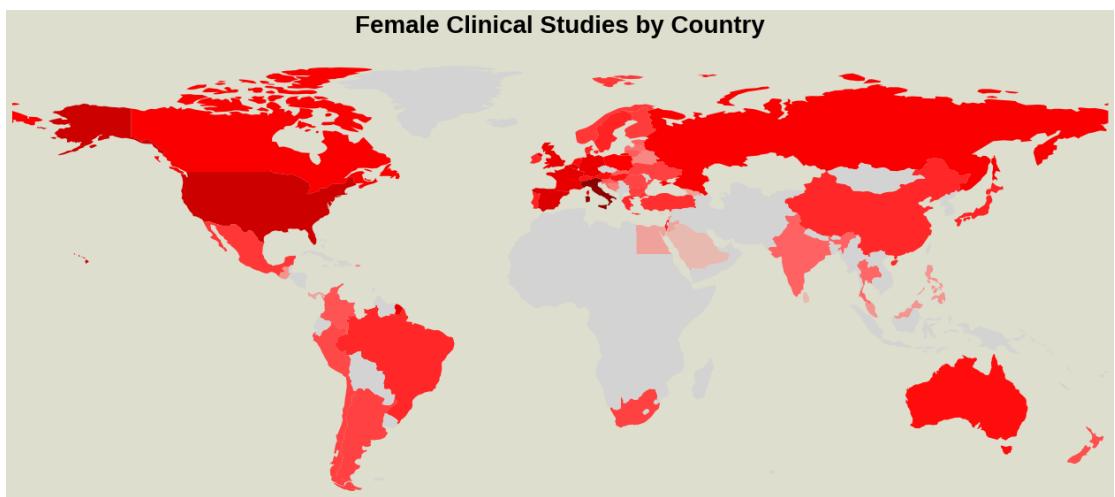


Figure 2.11: Map view - Nazioni che sponsorizzano il maggior numero di studi clinici rivolti alle donne

2.7 Nazioni che sponsorizzano il maggior numero di studi che coinvolgono minori

2.7.1 Implementazione della query

L'obiettivo dell'analitica è identificare i paesi che partecipano al maggior numero di studi clinici che coinvolgono minori (età < 18 anni), in qualità di sponsor o collaboratori.

In particolare:

- Si estraggono le nazioni "esplodendo" ogni voce della lista di paesi sponsor

(*Country of Sponsor/Collaborator*) evitando duplicati.

- Si filtrano gli studi clinici selezionando soltanto quelli il cui primo valore dell'intervallo di età (*age*) è minore di 18. Si escludono inoltre anche le righe senza età definita o senza nazione.
- Si aggrega per nazione e si conta il numero totale di studi.
- Si ordina in maniera decrescente per numero di studi.



```

● ● ●

nationMinorsCollaborators = ctDS.select(explode(array_distinct("Country of Sponsor/
Collaborator"))).alias("Country") \
    .filter((col("Age").isNotNull()) & (col("Age")[0] < 18) &
    (col("Country").isNotNull())) \
    .groupBy("Country") \
    .count() \
    .withColumnRenamed("count","Count of Studies per Country") \
    .orderBy(col("Count of Studies per Country").desc())

```

Figure 2.12: Nazioni che sponsorizzano il maggior numero di studi che coinvolgono minori

2.7.2 Risultati

L'analisi dei risultati mostra che anche in questo caso l'Italia è in cima, con il maggior numero di studi clinici che coinvolgono minori. Ciò indica la presenza di centri pediatrici di eccellenza e politiche favorevoli alla ricerca su minori.

Country	Count of Studies per Country
Italy	876
United States	496
Spain	372
France	365
United Kingdom	351
Germany	350
Belgium	274
Canada	246
Poland	228
Netherlands	224



Figure 2.13: Map View - Nazioni che sponsorizzano il maggior numero di studi che coinvolgono minori

2.8 Città che ricercano più attivamente uno specifico tipo di cancro

2.8.1 Implementazione della query

L’obiettivo dell’analitica è identificare la città che conduce il maggior numero di studi clinici per ciascun tipo di cancro.

In particolare:

- Si elaborano i dati assicurandoci che ogni città e ogni tipo di cancro siano considerati in righe separate e senza duplicati. Questo crea una vista dove ogni riga rappresenta una coppia distinta (Città, Tipo di Cancro).
- Si aggrega su queste coppie e si conta il numero di studi clinici per ciascuna coppia.
- Si usa una finestra partizionata per tipo di cancro, ordinata per numero di studi decrescente e si seleziona solo la città top per ciascun tipo di cancro ($rank==1$), eliminando eventuali valori nulli.
- Si ordina in maniera decrescente per numero di studi.



```

# Preprocessing, because select can't include more than one explode at time
cityCancerDS = ctDS.withColumn("City",explode(array_distinct(col("City of Sponsor/
Collaborator")))) \
    .withColumn("Cancer Type",explode(ctDS["Cancer Types"]))

# Defining the windowSpec
windowSpecCityCancer = Window.partitionBy("Cancer Type").orderBy(col("NumStudies per
City/Cancer").desc())

# Here we could filter for a specific city, for instance where City == Bologna
topCityCancer = cityCancerDS.select("City","Cancer Type")\
    .groupBy("City","Cancer Type") \
    .count() \
    .withColumnRenamed("count","NumStudies per City/Cancer") \
    .withColumn("rank",row_number().over(windowSpecCityCancer)) \
    .filter((col("rank") == 1) & (col("City").isNotNull())) \
    .orderBy(col("NumStudies per City/Cancer").desc())

```

Figure 2.14: Città che ricercano più attivamente uno specifico tipo di cancro

2.8.2 Risultati

I risultati di questa analitica sono utili per localizzare i poli di eccellenza clinica oncologica. Possono quindi essere utili a guidare pazienti e ricercatori verso centri più attivi e per guidare collaborazioni internazionali focalizzate su specifici tipi di tumori.

Dall’analisi dei risultati emerge ad esempio che Milano domina nella maggior parte dei tipi di cancro, risultando quindi il primo centro di ricerca oncologica clinica in italia.

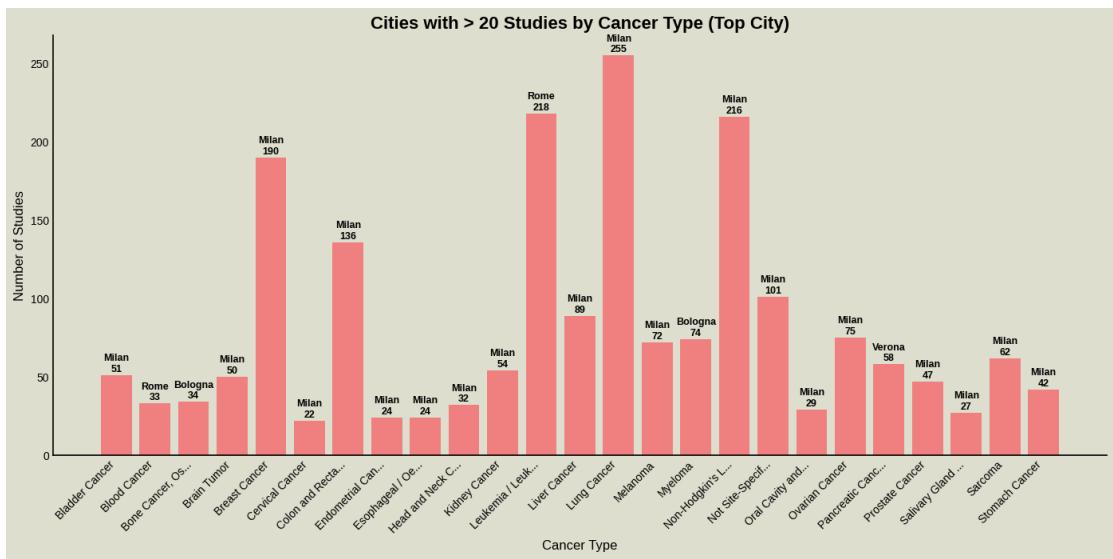


Figure 2.15: Bar chart - Città che ricercano più attivamente uno specifico tipo di cancro

City	Cancer Type	NumStudies per City/Cancer
Milan	Lung Cancer	255
Rome	Leukemia / Leukaemia	218
Milan	Non-Hodgkin's Lymphoma	216
Milan	Breast Cancer	190
Milan	Colon and Rectal Cancer	136
Milan	Not Site-Specific Cancer	101
Milan	Liver Cancer	89
Milan	Ovarian Cancer	75
Bologna	Myeloma	74
Milan	Melanoma	72
Milan	Sarcoma	62
Verona	Pancreatic Cancer	58
Milan	Kidney Cancer	54
Milan	Bladder Cancer	51
Milan	Brain Tumor	50
Milan	Prostate Cancer	47
Milan	Stomach Cancer	42
Bologna	Bone Cancer	34
Rome	Blood Cancer	33

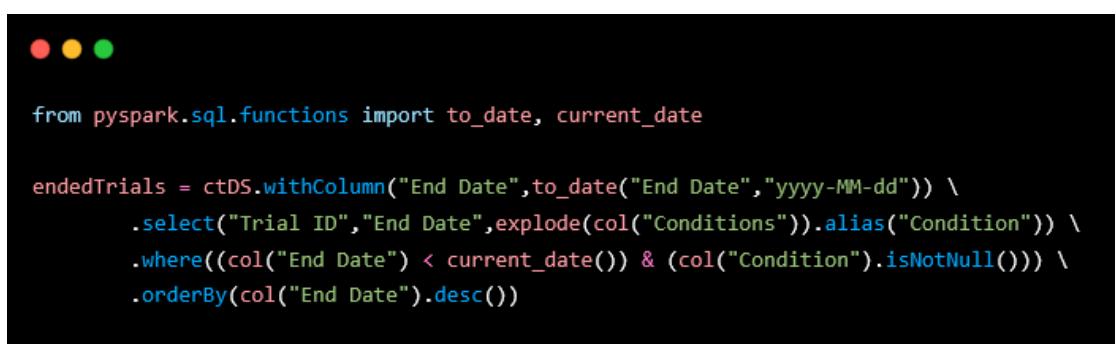
2.9 Studi clinici che si sono conclusi, con relative condizioni mediche trattate

2.9.1 Implementazione della query

L'obiettivo di questa analitica è trovare tutti gli studi clinici che si sono conclusi (cioè la cui data di fine è anteriore alla data odierna) e identificare le condizioni mediche trattate in ciascuno di essi.

In particolare:

- Si converte la colonna *End Date* da stringa a DateType per poter confrontare correttamente la data con quella fornita dalla funzione *current_date()*.
- Si "esplode" la lista delle condizioni trattate in ciascuno studio in righe separate
- Si filtrano le righe includendo solo gli studi già conclusi e che trattano condizioni non nulle.
- Si ordinano gli studi dal più recentemente concluso al più vecchio.



```
from pyspark.sql.functions import to_date, current_date

endedTrials = ctDS.withColumn("End Date", to_date("End Date", "yyyy-MM-dd")) \
    .select("Trial ID", "End Date", explode(col("Conditions")).alias("Condition")) \
    .where((col("End Date") < current_date()) & (col("Condition").isNotNull())) \
    .orderBy(col("End Date").desc())
```

Figure 2.16: Studi clinici che si sono conclusi, con relative condizioni mediche trattate

2.9.2 Risultati

Questa analitica è utile poichè fornisce una visione storica ed immediata degli ultimi sviluppi clinici.

Trial ID	End Date	Condition
NCT05754502	5/2/2025	Breast Cancer
NCT01340430	5/1/2025	HER-2 Positive Breast Cancer
NCT01397682	5/1/2025	Aging
NCT01397682	5/1/2025	Inpatients
NCT02855476	5/1/2025	Huntington's Disease
NCT03603184	5/1/2025	Endometrial Cancer
NCT03666741	5/1/2025	Aortic Valve Stenosis
NCT04597125	5/1/2025	Metastatic Castrate Resistant Prostate Cancer (mCRPC)
NCT04617925	5/1/2025	AL Amyloidosis
NCT04745234	5/1/2025	Cutaneous T-Cell Lymphoma, Relapsed

Vengono mostrate solo le prime 10 righe per motivi di spazio.

2.10 Studi clinici con maggiore visibilità mediatica (Altmetric score)

2.10.1 Implementazione della query

L'obiettivo di questa analitica è identificare gli studi con maggiore impatto mediatico, misurato tramite l'Altmetric Attention Score, e associarli alle condizioni mediche trattate.

In particolare:

- Si "esplode" la lista delle condizioni mediche in righe separate.
- Si includono solo gli studi che hanno ricevuto visibilità mediatica (Altmetric Score non nullo) con almeno una condizione trattata non nulla.
- Si ordinano i risultati in ordine crescente di visibilità.



```
topAltmetricScore = ctDS.select("Trial ID", "Title", "Altmetric Attention Score", explode(col("Conditions")).alias("Condition")) \
    .where(col("Altmetric Attention Score").isNotNull() & \
    col("Condition").isNotNull()) \
    .orderBy(col("Altmetric Attention Score").desc())
```

Figure 2.17: Studi clinici con maggiore visibilità mediatica

2.10.2 Risultati

Questa analitica è interessante poichè mostra gli studi più discussi nel panorama scientifico e mediatico e quindi le condizioni cliniche che attirano maggiore attenzione pubblica.

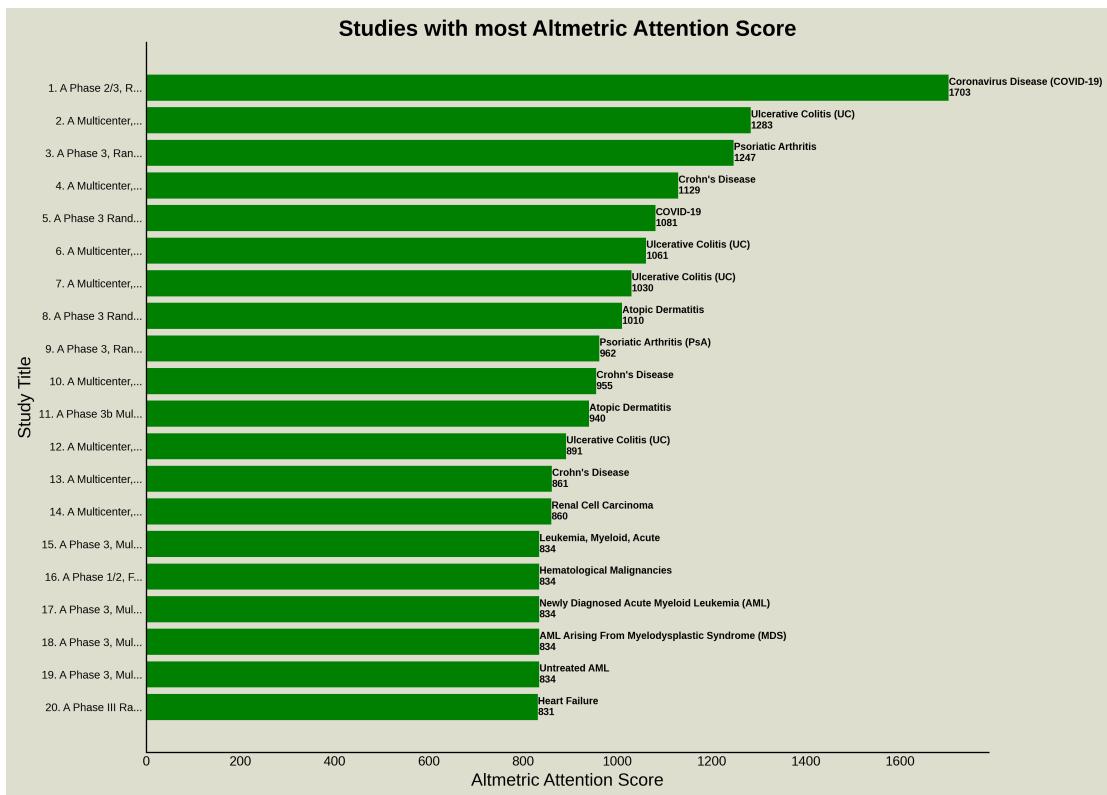


Figure 2.18: Bar chart orizzontale - Studi clinici con maggiore visibilità mediatica

Trial ID	Altmetric Attention Score	Condition
NCT04575597	1703	Coronavirus Disease (COVID-19)
NCT02819635	1283	Ulcerative Colitis (UC)
NCT03104400	1247	Psoriatic Arthritis
NCT03105128	1129	Crohn's Disease
NCT04292899	1081	COVID-19
NCT03398148	1061	Ulcerative Colitis (UC)
NCT03398135	1030	Ulcerative Colitis (UC)
NCT03569293	1010	Atopic Dermatitis
NCT03671148	962	Psoriatic Arthritis (PsA)
NCT03345836	955	Crohn's Disease

Table 2.1: La colonna relativi ai Titoli è omessa per leggibilità

Notiamo che lo studio che ha ricevuto la maggior attenzione mediatica è relativo alla condizione di COVID-19.



```

conditionHighestScore =
    ctDS.select(explode(col("Conditions")).alias("Condition"), "Altmetric
Attention Score") \
    .filter(col("Condition").isNotNull()) \
    .groupBy("Condition") \
    .agg(avg("Altmetric Attention Score").alias("Average Attention Score")) \
    .orderBy(col("Average Attention Score").desc())

```

Figure 2.19: Condizioni con media più alta di Altmetric Score

2.11 Condizioni con media più alta di Altmetric Score

2.11.1 Implementazione della query

Questa analitica ha l'obiettivo di mostrare le condizioni mediche associate agli studi clinici, che in media, hanno ottenuto la maggior visibilità sui media, secondo il punteggio Altmetric Attention Score.

In particolare:

- Si "esplode" la lista delle condizioni mediche.
- Si raggruppa per condizione.
- Si fa la media degli Altmetric Attention Score.
- Si ordina in maniera decrescente per punteggio.

2.11.2 Risultati

Dai risultati notiamo che anche in questo caso il COVID-19 è la condizione che ha ricevuto più attenzione mediatica.

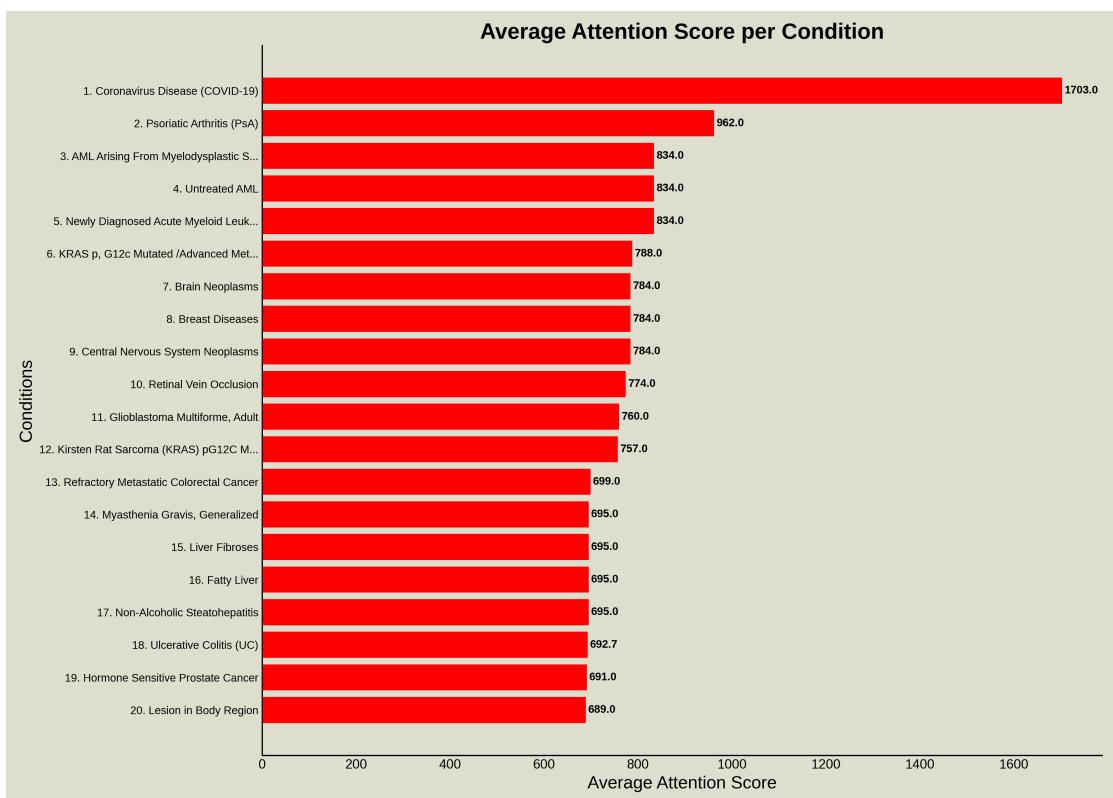


Figure 2.20: Bar chart orizzontale Condizioni con media più alta di Altmetric Score

Condition	Average Attention Score
Coronavirus Disease (COVID-19)	1703
Psoriatic Arthritis (PsA)	962
Untreated AML	834
Newly Diagnosed Acute Myeloid Leukemia (AML)	834
AML Arising From Myelodysplastic Syndrome (MDS)	834
KRAS p, G12c Mutated /Advanced Metastatic NSCLC	788
Central Nervous System Neoplasms	784
Breast Diseases	784
Brain Neoplasms	784
Retinal Vein Occlusion	774
Glioblastoma Multiforme, Adult	760
Kirsten Rat Sarcoma (KRAS) pG12C Mutation	757
Refractory Metastatic Colorectal Cancer	699
Liver Fibroses	695
Myasthenia Gravis, Generalized	695
Non-Alcoholic Steatohepatitis	695
Fatty Liver	695
Ulcerative Colitis (UC)	692.7142857
Hormone Sensitive Prostate Cancer	691
Lesion in Body Region	689

2.12 Nazioni delle organizzazioni finanziarie che hanno sponsorizzato il maggior numero di studi clinici

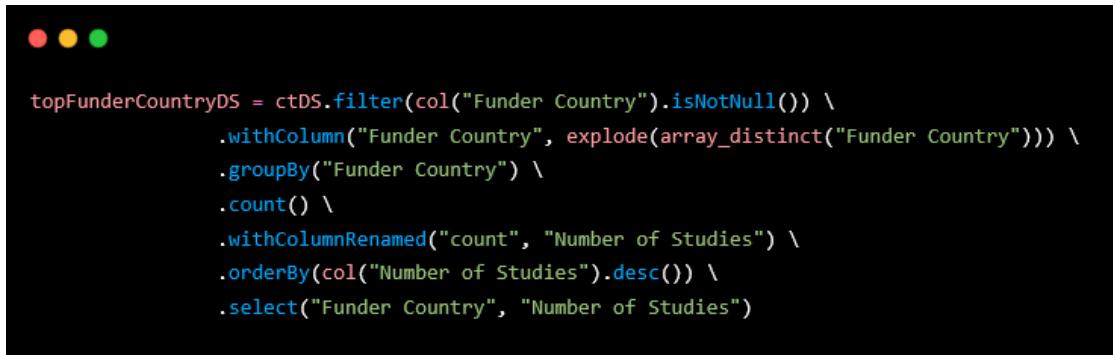
2.12.1 Implementazione della query

L'obiettivo è identificare i paesi delle organizzazioni finanziarie che hanno sponsorizzato il maggior numero di studi clinici.

In particolare:

- Si esplode la lista dei paesi dei finanziatori in righe individuali, evitando le duplicazioni.
- Si raggruppano i dati per paese e si conta il numero di studi associati a ciascun paese.

- Si ordina in maniera decrescente in base al numero di studi.



```
topFunderCountryDS = ctDS.filter(col("Funder Country").isNotNull()) \
    .withColumn("Funder Country", explode(array_distinct("Funder Country"))) \
    .groupBy("Funder Country") \
    .count() \
    .withColumnRenamed("count", "Number of Studies") \
    .orderBy(col("Number of Studies").desc()) \
    .select("Funder Country", "Number of Studies")
```

Figure 2.21: Nazioni delle organizzazioni finanziarie che hanno sponsorizzato il maggior numero di studi clinici

2.12.2 Risultati

Questa analitica può essere utile per enti di ricerca e università ad identificare paesi con elevata attività di finanziamento con cui collaborare.

I risultati mostrano che l'Italia emerge come il paese con il maggior numero di studi sponsorizzati (1592), superando anche gli Stati Uniti (1529), tradizionalmente leader nel settore della ricerca clinica.

Funder Country	Number of Studies
Italy	1592
United States	1529
Germany	277
Japan	191
Belgium	183
United Kingdom	182
Switzerland	119
Netherlands	83
France	79
Spain	47
Canada	39
Denmark	36
Sweden	32
Australia	26
Austria	20
Norway	14
Portugal	12
Ireland	11
Finland	10
Brazil	8

2.13 Ricercatori/Contatti più attivi con relativa patologia più studiata

2.13.1 Implementazione della query

L'obiettivo di questa analitica è identificare i ricercatori/contatti più attivi negli studi clinici e determinare per ciascuno la condizione medica più frequentemente studiata.

In particolare:

- Si filtrano gli studi con almeno un contatto o ricercatore associato e si "esplosione" la lista dei contatti in righe individuali.
- Si normalizzano i nomi per evitare duplicati dovuti a maiuscole/minuscole.
- Si conta il numero di studi associati a ciascun ricercatore.

- Si esegue inoltre un conteggio delle occorrenze di ogni condizione per ciascun ricercatore.
- Si definisce una finestra di partizione per ordinare le condizioni per ricercatori in ordine decrescente di frequenza. Questo servirà ad estrarre per ogni ricercatore soltanto la condizione più studiata.
- Si effettua un *join* tra il conteggio degli studi per ricercatore/contatto e la loro condizione principale.
- Si ordinano i risultati per numero decrescente di studi.

2.13.2 Risultati

Questa analitica è utile per identificare esperti in base all'area terapeutica, così da proporre collaborazioni di ricerca.

Investigator	Number of Studies	Top Condition	Top Condition Count
abbvie inc	89	crohn's disease	9
francesco perrone	52	ovarian cancer	9
boehringer ingelheim	52	carcinoma, non-small-cell lung	4
sandro pignata	27	ovarian cancer	14
antonio carroccio	25	non-celiac wheat sensitivity	11
nicola cascavilla	25	acute myeloid leukemia	5
filippo de braud	24	breast cancer	4
ciro gallo	24	ovarian cancer	6
alessandro rambaldi	23	chronic myeloid leukemia	3
lisa licitra	22	head and neck cancer	4
nicola di renzo	21	acute myeloid leukemia	3
davide chiumello	21	acute respiratory distress syndrome	4
michele spina	20	diffuse large b-cell lymphoma	3
francesco zaja	19	mantle cell lymphoma	3
renato bassan	18	acute lymphoblastic leukemia	4
bryan a. faller	18	stage iiib lung cancer ajcc v8	3
luca arcaini	18	follicular lymphoma	4
cesare gridelli	18	advanced non-small cell lung cancer	4
fabrizio pane	18	anemia	2
giuseppe tarantini	17	coronary artery disease	3

Dai risultati notiamo che le prime posizioni sono occupate da grandi aziende farmaceutiche (es. Abbvie Inc., Boehringer Ingelheim), seguite da ricercatori accademici o ospedalieri. Inoltre notiamo che molti ricercatori mostrano un'elevata



```

topInvestigatorsDs = ctDS.filter(col("Investigators/Contacts").isNotNull()) \
    .withColumn("Investigator", explode("Investigators/Contacts")) \
    .withColumn("Investigator", trim(lower(col("Investigator")))) \
    .groupBy("Investigator") \
    .count() \
    .withColumnRenamed("count", "Number of Studies")

conditionCountDs = ctDS.filter(col("Investigators/Contacts").isNotNull() &
    col("Conditions").isNotNull()) \
    .withColumn("Investigator", explode("Investigators/Contacts")) \
    .withColumn("Investigator", trim(lower(col("Investigator")))) \
    .withColumn("Condition", explode("Conditions")) \
    .withColumn("Condition", trim(lower(col("Condition")))) \
    .groupBy("Investigator", "Condition") \
    .count() \
    .withColumnRenamed("count", "Condition Count")

# Apply a window function to identify the top condition for each Investigator.
windowSpecTopCondition = Window.partitionBy("Investigator").orderBy(col("Condition
Count").desc())

topConditionPerInvestigator = conditionCountDs.withColumn("rank",
row_number().over(windowSpecTopCondition)) \
    .filter(col("rank")==1) \
    .select("Investigator", "Condition", "Condition Count") \
    .withColumnRenamed("Condition", "Top Condition")

#Join
topInvestigatorWithCondition = topInvestigatorsDs.join(topConditionPerInvestigator,
on="Investigator", how="left") \
    .orderBy(col("Number of Studies").desc()) \
    .withColumnRenamed("Condition Count", "Top Condition Count")

```

Figure 2.22: Ricercatori/Contatti più attivi con relativa patologia più studiata

specializzazione in un campo specifico. Ad esempio Sandro Pignata ha 14 studi su 27 incentrati sul cancro ovarico.

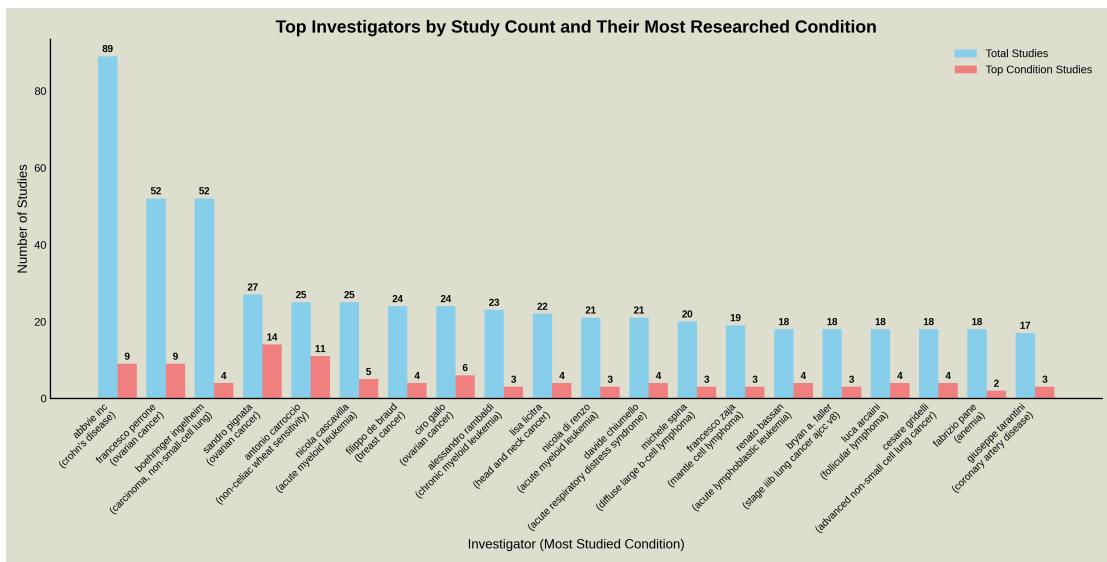


Figure 2.23: Bar chart - Ricercatori/Contatti più attivi con relativa patologia più studiata

2.14 Top collaborazioni dell'Università di Napoli Federico II con enti finanziatori

2.14.1 Implementazione della query

L'obiettivo di questa analitica è determinare quali enti finanziatori o collaboratori hanno partecipato più frequentemente a sperimentazioni cliniche in partnership con l'AOU Federico II.

In particolare:

- Si filtrano i trial clinici in cui l'AHC (Academic Health Center) è AOUSSN_FEDERICOII.
- Si "esplode" la lista di sponsor/collaboratori, in modo che ogni voce diventi una riga distinta.
- Si aggrega per ciascun collaboratore e si conta quante volte compare nei trial della Federico II.
- Si ordinano i risultati per numero di collaborazioni.



```
federicoII_collabSponsor = ctDS.filter(col("AHC") == "AUSSN_FEDERICOII") \
    .filter(col("Sponsors/Collaborators").isNotNull()) \
    .withColumn("Sponsors/Collaborators", explode("Sponsors/Collaborators")) \
    .groupBy("Sponsors/Collaborators") \
    .count() \
    .withColumnRenamed("count", "Number of Collaborations") \
    .withColumn("AHC", lit("Federico II")) \
    .select("AHC", "Sponsors/Collaborators", "Number of Collaborations") \
    .orderBy(col("Number of Collaborations").desc())
```

Figure 2.24: Top collaborazioni dell'Università di Napoli Federico II con enti finanziatori

2.14.2 Risultati

Questa analitica è utile per valutare l'impatto e la visibilità scientifica della Federico II nel contesto nazionale ed internazionale.

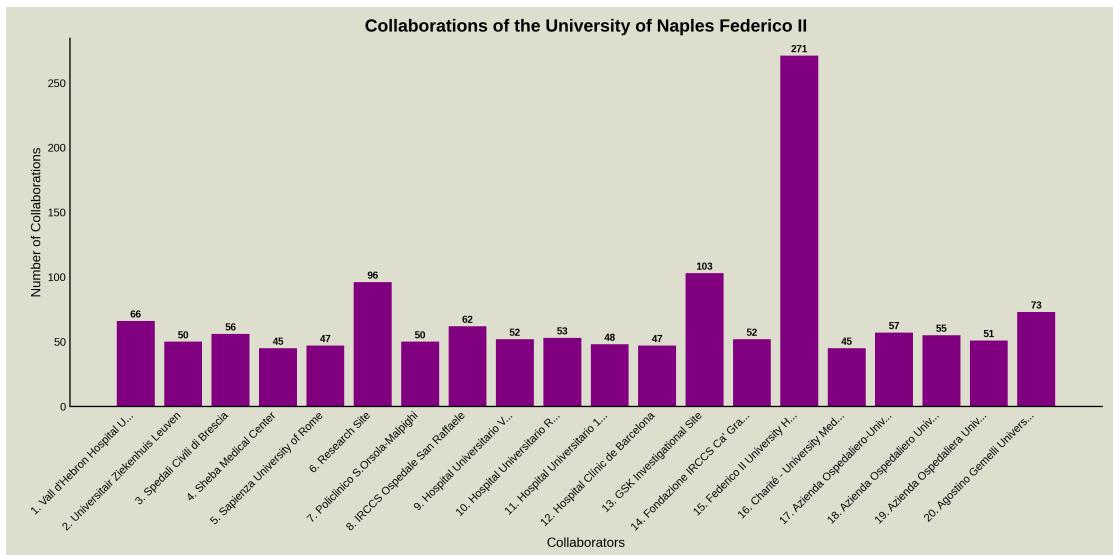


Figure 2.25: Bar chart - Top collaborazioni dell'Università di Napoli Federico II con enti finanziatori

Dai risultati è possibile osservare che il primo collaboratore è "Federico II University Hospital" stesso: questo conferma che molte sperimentazioni sono registrate con l'ospedale come centro promotore e come centro partecipante. Tuttavia

CHAPTER 2. ANALYTICS

AHC	Sponsors/Collaborators	Number of Collaborations
Federico II	Federico II University Hospital	271
Federico II	GSK Investigational Site	103
Federico II	Research Site	96
Federico II	Agostino Gemelli University Polyclinic	73
Federico II	Vall d'Hebron Hospital Universitari	66
Federico II	IRCCS Ospedale San Raffaele	62
Federico II	Azienda Ospedaliero-Universitaria Careggi	57
Federico II	Spedali Civili di Brescia	56
Federico II	Azienda Ospedaliero Universitaria Ospedali Riuniti	55
Federico II	Hospital Universitario Ramón y Cajal	53
Federico II	Hospital Universitario Virgen del Rocío	52
Federico II	Fondazione IRCCS Ca' Granda Ospedale Maggiore Policlinico	52
Federico II	Azienda Ospedaliera Universitaria Pisana	51
Federico II	Universitair Ziekenhuis Leuven	50
Federico II	Policlinico S.Orsola-Malpighi	50
Federico II	Hospital Universitario 12 de Octubre	48
Federico II	Hospital Clínic de Barcelona	47
Federico II	Sapienza University of Rome	47
Federico II	Sheba Medical Center	45
Federico II	Charité - University Medicine Berlin	45

vediamo comparire anche molti altri ospedali italiani ed internazionali.

2.15 Word Cloud dei Titoli Brevi

2.15.1 Implementazione della word cloud

Questa word cloud generata dai titoli brevi degli studi clinici ha l'obiettivo di ottenere una visione sintetica e visuale dei temi più ricorrenti nelle sperimentazioni.

In particolare:

- Si seleziona la colonna *Brief Title* e si rimuovono eventuali valori nulli.
- Si costruisce un'unica stringa concatenando tutti i titoli.
- Si pulisce il testo rimuovendo simboli e numeri, trasformando il tutto in minuscolo per uniformità.
- Si crea una lista di stopwords personalizzate (aggiunte alle classiche della lingua inglese) per far emergere i concetti clinici reali.
- Si genera la word cloud dalle parole rimanenti.

2.15.2 Risultati

La word cloud ci permette di identificare rapidamente le aree terapeutiche principali, tipologie di intervento etc...

```
from pyspark.sql.functions import concat_ws
from wordcloud import STOPWORDS, WordCloud
import matplotlib.pyplot as plt

briefTitlesDF = ctDS.select("Brief Title").na.drop()
briefTitles_RDD = briefTitlesDF.rdd.map(lambda row: row[0])
briefTitlesText = " ".join(briefTitles_RDD.collect())

cleanedText = re.sub(r'[^A-Za-z\s]', '', briefTitlesText)
cleanedText = cleanedText.lower()

custom_stopwords = set(STOPWORDS)
custom_stopwords.update([
    "patient", "patients", "study", "subject", "subjects", "clinical", "trial", "randomized",
    "placebo", "group", "efficacy", "treatment", "participant", "participants", "versus",
    "evaluate", "evaluation", "combination", "effect", "without"])

wordcloud = WordCloud(width=800, height=400, background_color='white',
                      stopwords=custom_stopwords, max_words=100).generate(cleanedText)
```

Figure 2.26: Generazion Word Cloud



Figure 2.27: Word Cloud dei Titoli Brevi

2.16 Top collaborazione per ente finanziatore

2.16.1 Implementazione della query

L'obiettivo di questa analitica è trovare, per ciascun finanziatore principale (*Main Funder*), l'organizzazione finanziatrice con cui collabora più frequentemente (*Collaborating Funder*) in progetti clinici, mostrando il numero di collaborazioni.

In particolare:

- Si filtra il dataset per tenere solo gli studi che hanno entrambe le informazioni: finanziatori principali e collaboratori.
- Si "esplodono" le liste di finanziatori e collaboratori in righe separate, così da generare tutte le combinazioni possibili per ogni studio.
- Si eliminano i casi in cui il finanziatore principale ed il collaboratore coincidono.
- Si conta l'occorrenza di ogni collaborazione.
- Si crea una finestra di partizione per ordinare i collaboratori in base al numero di collaborazioni per ogni ente principale.
- Si filtra il dataset per ottenere solo il collaboratore più frequente per ogni finanziatore principale.
- Si ordina il risultato finale in ordine decrescente di collaborazioni.

2.16.2 Risultati

Questa analitica è utile per mettere in luce relazioni forti e consolidate tra enti finanziatori. Ciò può essere di supporto ad enti di ricerca che possono utilizzare queste informazioni per migliorare le proprie possibilità di finanziamento orientandosi verso network già esistenti.

```

● ● ●

collabFunderPreDS = ctDS.filter(col("Funder Group").isNotNull() & col("Collaborating
Funders").isNotNull()) \
    .withColumn("Main Funder", explode("Funder Group")) \
    .withColumn("Collaborating Funder", explode("Collaborating Funders")) \
    .filter(col("Main Funder") != col("Collaborating Funder")) \
    .groupBy("Main Funder", "Collaborating Funder") \
    .count() \
    .withColumnRenamed("count", "Number of Collaborations")

windowSpecCollab = Window.partitionBy("Main Funder").orderBy(col("Number of
Collaborations").desc())

topCollaborators = collabFunderPreDS.withColumn("rank", row_number().over(windowSpecCollab)) \
    .filter(col("rank")==1) \
    .select("Main Funder", "Collaborating Funder", "Number of Collaborations") \
    .orderBy(col("Number of Collaborations").desc())

```

Figure 2.28: Top collaborazione per ente finanziatore

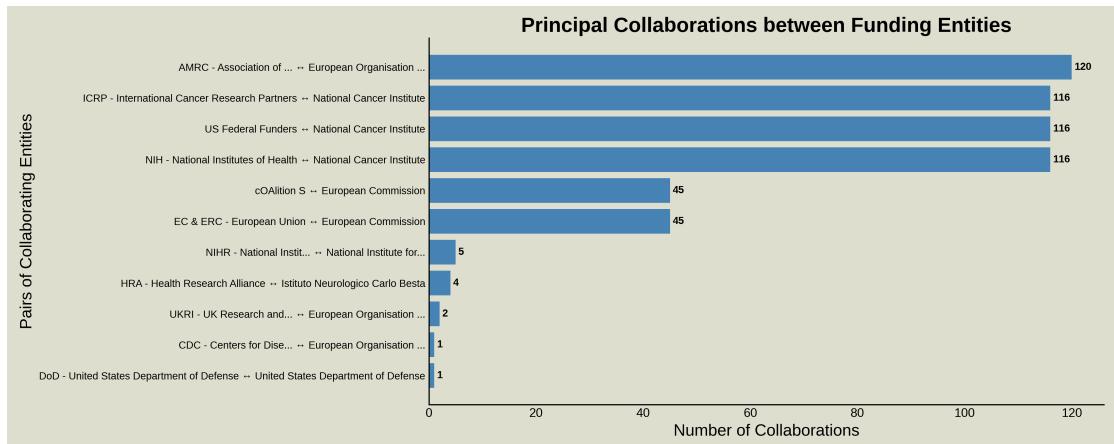


Figure 2.29: Bar chart - Top collaborazione per ente finanziatore

Main Funder	Collaborating Funder	Number of Collaborations
AMRC - Association of Medical Research Charities	European Organisation for Research and Treatment of Cancer	120
ICRP - International Cancer Research Partnership	National Cancer Institute	116
NIH - National Institutes of Health	National Cancer Institute	116
US Federal Funders	National Cancer Institute	116
EC & ERC - European Union	European Commission	45
cOALition S	European Commission	45
NIHR - National Institute for Health Research	National Institute for Health and Care Research	5
HRA - Health Research Alliance	Istituto Neurologico Carlo Besta	4
UKRI - UK Research and Innovation	European Organisation for Research and Treatment of Cancer	2
CDC - Centers for Disease Control and Prevention	European Organisation for Research and Treatment of Cancer	1
DoD - United States Department of Defense	United States Department of Defense	1