

Query (Read performance)

After Loading the data successfully, the next steps involves writing and performing queries that can be performed on the data and measuring the execution time for each database to measure the write performance.

Queries:

Query1: Calculate the number of people whose start date is greater than '1991-09-10' and end date is less than '2010-03-07' and are dead.

Query2: Calculate the percentage of data that is censored (individuals for whom the exact death time is unknown).

Query3: Calculate the average duration of observations for individuals who are not censored (i.e., those who completed the observation period).

Kaplan-Meier Estimate:

Time-to-Event:

"time-to-event" is the difference between "age_end" and "age_start_observed."

Event Status:

If "is_dead" is True, it means an event occurred (death), so set the event status to 1.

If "is_censored" is True and "is_dead" is False, it means the observation was censored (the event did not occur within the observation period), so set the event status to 0.

LogRank

The log rank test is a popular test to test the null hypothesis of no difference in survival between two or more independent groups.

Groups : is_dead and age_end

Queries

Cox Proportional Hazards (CoxPH) analysis:

I have selected the 'age_end' column as the duration column, which represents the time until an event (death) occurs. I have selected the 'is_dead' column as the event column, indicating whether an individual is deceased (True) or not (False).

Key elements to look at in the model summary:

coef: This column provides the coefficients for each variable in the model. It indicates the direction and strength of the relationship between each variable and the hazard. A positive coefficient means a higher hazard (increased risk of the event), while a negative coefficient means a lower hazard.

exp(coef): This is the hazard ratio (HR), which represents the multiplicative effect of a one-unit change in the predictor variable on the hazard. An HR greater than 1 indicates an increased risk, while an HR less than 1 indicates a decreased risk.

p: The p-value associated with each coefficient. It helps determine the statistical significance of each variable. Lower p-values suggest stronger evidence that a variable has an impact on the hazard.

95% CI (lower and upper): Confidence intervals for the hazard ratio. This helps you assess the range of plausible values for the HR.

Dataset1 : Data generation

Dataset1: 1 million rows (10^6)

```
PS D:\Aarsh\SJSU\CS298\Project\data_generation> python generate_data.py --n 1000000 --database timescaledb
Dataset creation took 10.94 seconds.
Data saved in timescaledb format.
```

Dataset 1: Data Loading

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python mongo_load.py --database project1 --collection data --json_file data1.json
890155 documents inserted into data in project1.
Data loading took 14.59 seconds.
```

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python questdb_load.py http://localhost:9000 project1 data1.csv
Data loaded into project1
Data loading took 1.61 seconds.
```

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python cassandra_load.py --keyspace project1 --table data --csv_file cassandra_data1.csv
localhost
Using 7 child processes

Starting copy of project1.data with columns [uiud, age_end, age_start_observed, date_end_observed, date_start_observed, is_censored, is_dead, is_truncated].
Processed: 890155 rows; Rate: 11465 rows/s; Avg. rate: 16820 rows/s
890155 rows imported from 1 files in 0 day, 0 hour, 0 minute, and 52.923 seconds (0 skipped).
Data loading took 55.08 seconds.
```

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python timescale_load.py --database aarsh --table data1 --csv_file data1.csv --username postgres --password aarsh
Data loaded into data1 in aarsh.
Data loading took 3.65 seconds.
```

Query 1

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query1> python .\query1_questdb.py  
"count"  
1254
```

Execution Time: 0.08 seconds

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query1> python .\query1_timescaledb.py  
(1254,)
```

Execution Time: 0.10 seconds

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query1> python .\query1_cassandra.py  
Row(count=1254)
```

Execution Time: 4.52 seconds

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query1> python .\query1_mongodb.py  
Count: 1254
```

Execution Time: 0.78 seconds

Query 2

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query2> python .\query2_questdb.py
Percentage of True values: 43.89%
Execution Time: 0.06 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query2> python .\query2_timescaledb.py
Percentage of True values: 43.89%
Execution Time: 0.30 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query2> python .\query2_cassandra.py
Percentage of True values: 43.89%
Execution Time: 23.78 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query2> python .\query2_mongodb.py
Percentage of True values: 43.89%
Execution Time: 1.17 seconds
```

Query3

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query3> python .\query3_questdb.py  
"average_duration_of_observation"  
52.547430364648
```

```
Execution Time: 0.24 seconds
```

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query3> python .\query3_timescaledb.py  
(Decimal('52.5474303646481245'),)
```

```
Execution Time: 0.14 seconds
```

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query3> python .\query3_cassandra.py  
Row(average_duration_of_observation=-52)
```

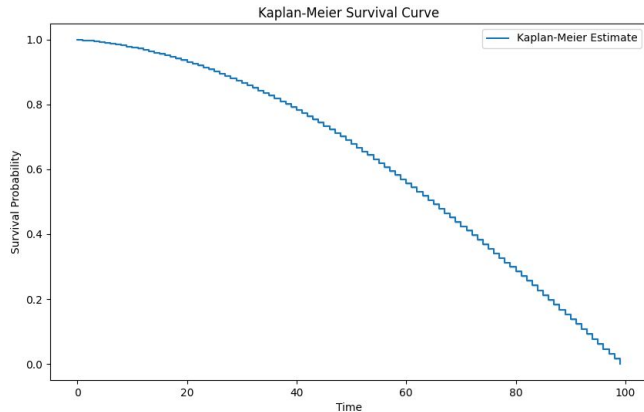
```
Execution Time: 8.76 seconds
```

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query3> python .\query3_mongodb.py  
average_duration_of_observation: 52.54743036464812
```

```
Execution Time: 1.56 seconds
```


Kaplan-Meier

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier> python .\Kaplan_questdb.py
Median Survival Time: 65.0
Execution time: 2.39 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier> python .\Kaplan_timescaledb.py
D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier\Kaplan_timescaledb.py:23: UserWarning: pandas only supports SQLAlchemy connectable (engine/connection) or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not tested. Please consider using SQLAlchemy.
  data = pd.read_sql_query(query, conn)
Median Survival Time: 65.0
Execution time: 1.56 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier> python .\Kaplan_cassandra.py
Median Survival Time: 65.0
Execution time: 12.17 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier> python .\Kaplan_mongo.py
Median Survival Time: 65.0
Execution time: 8.79 seconds
```



LogRank

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank> python .\logrank_questdb.py
<lifelines.StatisticalResult: logrank_test>
      t_0 = -1
  null_distribution = chi squared
degrees_of_freedom = 1
      test_name = logrank_test

---
test_statistic      p  -log2(p)
      1812886.77 <0.005      inf
Execution time: 2.26 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank> python .\logrank_timescaledb.py
D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank\logrank_timescaledb.py:23: UserWarning: pandas only supports SQLAlchemy connectable (engine/connection)
or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not tested. Please consider using SQLAlchemy.
  data = pd.read_sql_query(query, conn)
<lifelines.StatisticalResult: logrank_test>
      t_0 = -1
  null_distribution = chi squared
degrees_of_freedom = 1
      test_name = logrank_test

---
test_statistic      p  -log2(p)
      1812886.77 <0.005      inf
Execution time: 1.20 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank> python .\logrank_cassandra.py
<lifelines.StatisticalResult: logrank_test>
      t_0 = -1
  null_distribution = chi squared
degrees_of_freedom = 1
      test_name = logrank_test

---
test_statistic      p  -log2(p)
      1812886.77 <0.005      inf
Execution time: 10.03 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank> python .\logrank_mongo.py
<lifelines.StatisticalResult: logrank_test>
      t_0 = -1
  null_distribution = chi squared
degrees_of_freedom = 1
      test_name = logrank_test

---
test_statistic      p  -log2(p)
      1812886.77 <0.005      inf
Execution time: 8.83 seconds
```

CoX-QuestDB

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Cox> python .\cox_questdb.py
<lifelines.CoxPHFitter: fitted with 890155 total observations, 499477 right-censored observations>
    duration col = '1'
    event col = '2'
    baseline estimation = breslow
    number of observations = 890155
    number of events observed = 390678
    partial log-likelihood = -4761971.72
    time fit was run = 2023-11-06 20:44:04 UTC

---

```

| | coef | exp(coef) | se(coef) | coef lower 95% | coef upper 95% | exp(coef) lower 95% | exp(coef) upper 95% |
|-----------|-------|-----------|----------|----------------|----------------|---------------------|---------------------|
| covariate | | | | | | | |
| 0 | -0.01 | 0.99 | 0.00 | -0.01 | -0.01 | 0.99 | 0.99 |
| 3 | 0.04 | 1.04 | 0.01 | 0.03 | 0.05 | 1.03 | 1.05 |

| | cmp to | z | p | -log2(p) |
|-----------|--------|--------|--------|----------|
| covariate | | | | |
| 0 | 0.00 | -36.43 | <0.005 | 962.60 |
| 3 | 0.00 | 7.19 | <0.005 | 40.48 |

```
---
Concordance = 0.54
Partial AIC = 9523947.44
log-likelihood ratio test = 2663.53 on 2 df
-log2(p) of ll-ratio test = inf

Execution time: 23.89 seconds
```

CoX-TimescaleDB

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Cox> python .\cox_timescaledb.py
D:\Aarsh\SJSU\CS298\Project\query_execution\Cox\cox_timescaledb.py:23: UserWarning: pandas only supports SQLAlchemy connectable
(engine/connection) or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not tested. Please consider
using SQLAlchemy.
  data = pd.read_sql_query(query, conn)
<lifelines.CoxPHFitter: fitted with 890155 total observations, 499477 right-censored observations>
  duration col = 'age_end'
  event col = 'is_dead'
  baseline estimation = breslow
  number of observations = 890155
number of events observed = 390678
  partial log-likelihood = -4761971.72
  time fit was run = 2023-11-06 20:42:13 UTC

----

```

| | coef | exp(coef) | se(coef) | coef lower 95% | coef upper 95% | exp(coef) lower 95% | exp(coef) upper 95% |
|--------------------|-------|-----------|----------|----------------|----------------|---------------------|---------------------|
| covariate | | | | | | | |
| age_start_observed | -0.01 | 0.99 | 0.00 | -0.01 | -0.01 | 0.99 | 0.99 |
| is_truncated | 0.04 | 1.04 | 0.01 | 0.03 | 0.05 | 1.03 | 1.05 |

| | cmp to | z | p | -log2(p) |
|--------------------|--------|--------|--------|----------|
| covariate | | | | |
| age_start_observed | 0.00 | -36.43 | <0.005 | 962.60 |
| is_truncated | 0.00 | 7.19 | <0.005 | 40.48 |

```
----
Concordance = 0.54
Partial AIC = 9523947.44
log-likelihood ratio test = 2663.53 on 2 df
-log2(p) of ll-ratio test = inf

Execution time: 23.21 seconds
```

CoX-MongoDB

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Cox> python .\cox_mongo.py
<lifelines.CoxPHFitter: fitted with 890155 total observations, 499477 right-censored observations>
    duration col = 'age_end'
    event col = 'is_dead'
    baseline estimation = breslow
    number of observations = 890155
    number of events observed = 390678
    partial log-likelihood = -4761971.72
    time fit was run = 2023-11-06 20:44:53 UTC

---

```

| | coef | exp(coef) | se(coef) | coef lower 95% | coef upper 95% | exp(coef) lower 95% | exp(coef) upper 95% |
|--------------------|-------|-----------|----------|----------------|----------------|---------------------|---------------------|
| covariate | | | | | | | |
| age_start_observed | -0.01 | 0.99 | 0.00 | -0.01 | -0.01 | 0.99 | 0.99 |
| is_truncated | 0.04 | 1.04 | 0.01 | 0.03 | 0.05 | 1.03 | 1.05 |

| | cmp to | z | p | -log2(p) |
|--------------------|--------|--------|--------|----------|
| covariate | | | | |
| age_start_observed | 0.00 | -36.43 | <0.005 | 962.60 |
| is_truncated | 0.00 | 7.19 | <0.005 | 40.48 |

```
---
Concordance = 0.54
Partial AIC = 9523947.44
log-likelihood ratio test = 2663.53 on 2 df
-log2(p) of ll-ratio test = inf

Execution time: 27.17 seconds
```

CoX-Cassandra

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Cox> python .\cox_cassandra.py
<lifelines.CoxPHFitter: fitted with 890155 total observations, 499477 right-censored observations>
    duration col = 'age_start_observed'
    event col = 'is_truncated'
    baseline estimation = breslow
    number of observations = 890155
    number of events observed = 390678
    partial log-likelihood = -4761971.72
    time fit was run = 2023-11-06 20:46:05 UTC

---

```

| | coef | exp(coef) | se(coef) | coef lower 95% | coef upper 95% | exp(coef) lower 95% | exp(coef) upper 95% |
|-------------|-------|-----------|----------|----------------|----------------|---------------------|---------------------|
| covariate | | | | | | | |
| age_end | -0.01 | 0.99 | 0.00 | -0.01 | -0.01 | 0.99 | 0.99 |
| is_censored | 0.04 | 1.04 | 0.01 | 0.03 | 0.05 | 1.03 | 1.05 |

```


```

| | cmp to | z | p | -log2(p) |
|-------------|--------|--------|--------|----------|
| covariate | | | | |
| age_end | 0.00 | -36.43 | <0.005 | 962.60 |
| is_censored | 0.00 | 7.19 | <0.005 | 40.48 |

```

---
Concordance = 0.54
Partial AIC = 9523947.44
log-likelihood ratio test = 2663.53 on 2 df
-log2(p) of ll-ratio test = inf

Execution time: 31.77 seconds
```

Dataset 2 : Data generation

Dataset2: 10 million rows (10^7)

```
PS D:\Aarsh\SJSU\CS298\Project\data_generation> python generate_data.py --n 10000000 --database timescaledb
Dataset creation took 109.00 seconds.
Data saved in timescaledb format.
```

Dataset 2: Data Loading

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python mongo_load.py --database project2 --collection data --json_file data2.json
8902269 documents inserted into data in project2.
Data loading took 192.43 seconds.
```

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python questdb_load.py http://localhost:9000 project2 data2.csv
Data loaded into project2
Data loading took 22.78 seconds.
```

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python cassandra_load.py --keyspace project2 --table data --csv_file cassandra_data2.csv
localhost
Using 7 child processes

Starting copy of project2.data with columns [uiid, age_end, age_start_observed, date_end_observed, date_start_observed, is_censored, is_dead, is_truncated].
Processed: 8902269 rows; Rate: 27956 rows/s; Avg. rate: 16030 rows/s
8902269 rows imported from 1 files in 0 day, 0 hour, 9 minutes, and 15.360 seconds (0 skipped).
Data loading took 558.61 seconds.
```

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python timescale_load.py --database aarsh --table data2 --csv_file data2.csv --username postgres --password aarsh
Data loaded into data2 in aarsh.
Data loading took 34.44 seconds.
```


Query 1

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query1> python .\query1_questdb.py  
"count"  
12253  
  
Execution Time: 1.70 seconds  
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query1> python .\query1_timescaledb.py  
(12253,)  
Execution Time: 1.83 seconds  
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query1> python .\query1_cassandra.py  
Row(count=12253)  
Execution Time: 29.80 seconds  
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query1> python .\query1_mongodb.py  
Count: 12260  
Execution Time: 7.99 seconds
```

Query 2

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query2> python .\query2_questdb.py
Percentage of True values: 43.95%
Execution Time: 0.16 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query2> python .\query2_timescaledb.py
Percentage of True values: 43.95%
Execution Time: 0.95 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query2> python .\query2_cassandra.py
Percentage of True values: 43.95%
Execution Time: 106.73 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query2> python .\query2_mongodb.py
Percentage of True values: 43.95%
Execution Time: 5.61 seconds
```

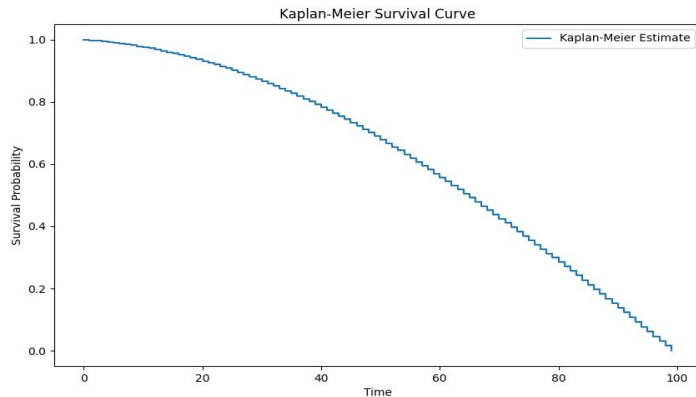
Query3

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query3> python .\query3_questdb.py
"average_duration_of_observation"
52.619327898403

Execution Time: 0.52 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query3> python .\query3_timescaledb.py
(Decimal('52.6193278984039485'),)
Execution Time: 0.67 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query3> python .\query3_cassandra.py
Row(average_duration_of_observation=-52)
Execution Time: 46.04 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\query3> python .\query3_mongodb.py
average_duration_of_observation: 52.619327898403945
Execution Time: 5.17 seconds
```

Kaplan-Meier

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier> python .\Kaplan_questdb.py
Median Survival Time: 65.0
Execution time: 16.20 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier> python .\Kaplan_timescaledb.py
D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier\Kaplan_timescaledb.py:23: UserWarning: pandas only supports SQLAlchemy connectable (engine/connection) or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not tested. Please consider using SQLAlchemy.
  data = pd.read_sql_query(query, conn)
Median Survival Time: 65.0
Execution time: 13.80 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier> python .\Kaplan_cassandra.py
Median Survival Time: 65.0
Execution time: 131.38 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Kaplan-Meier> python .\Kaplan_mongo.py
Median Survival Time: 65.0
Execution time: 176.33 seconds
```



LogRank

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank> python .\logrank_questdb.py
<lifelines.StatisticalResult: logrank_test>
      t_0 = -1
  null_distribution = chi squared
degrees_of_freedom = 1
      test_name = logrank_test

---
test_statistic      p  -log2(p)
18125791.12 <0.005      inf
Execution time: 16.14 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank> python .\logrank_timescaledb.py
D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank\logrank_timescaledb.py:23: UserWarning: pandas only supports SQLAlchemy connectable (engine/connection)
or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not tested. Please consider using SQLAlchemy.
  data = pd.read_sql_query(query, conn)
<lifelines.StatisticalResult: logrank_test>
      t_0 = -1
  null_distribution = chi squared
degrees_of_freedom = 1
      test_name = logrank_test

---
test_statistic      p  -log2(p)
18125791.12 <0.005      inf
Execution time: 12.95 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank> python .\logrank_cassandra.py
<lifelines.StatisticalResult: logrank_test>
      t_0 = -1
  null_distribution = chi squared
degrees_of_freedom = 1
      test_name = logrank_test

---
test_statistic      p  -log2(p)
18125791.12 <0.005      inf
Execution time: 116.05 seconds
PS D:\Aarsh\SJSU\CS298\Project\query_execution\LogRank> python .\logrank_mongo.py
<lifelines.StatisticalResult: logrank_test>
      t_0 = -1
  null_distribution = chi squared
degrees_of_freedom = 1
      test_name = logrank_test

---
test_statistic      p  -log2(p)
18125791.12 <0.005      inf
Execution time: 133.78 seconds
```

CoX-QuestDB

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Cox> python .\cox_questdb.py
<lifelines.CoxPHFitter: fitted with 8.90227e+06 total observations, 4.98968e+06 right-censored observations>
    duration col = '1'
    event col = '2'
    baseline estimation = breslow
    number of observations = 8.90227e+06
number of events observed = 3.91259e+06
    partial log-likelihood = -56696051.25
    time fit was run = 2023-11-06 20:20:33 UTC

---

```

| | coef | exp(coef) | se(coef) | coef lower 95% | coef upper 95% | exp(coef) lower 95% | exp(coef) upper 95% |
|-----------|-------|-----------|----------|----------------|----------------|---------------------|---------------------|
| covariate | | | | | | | |
| 0 | -0.01 | 0.99 | 0.00 | -0.01 | -0.01 | 0.99 | 0.99 |
| 3 | 0.04 | 1.04 | 0.00 | 0.03 | 0.04 | 1.03 | 1.04 |

```


```

| | cmp to | z | p | -log2(p) |
|-----------|--------|---------|--------|----------|
| covariate | | | | |
| 0 | 0.00 | -113.21 | <0.005 | inf |
| 3 | 0.00 | 21.04 | <0.005 | 323.91 |

```

---
Concordance = 0.54
Partial AIC = 113392106.49
log-likelihood ratio test = 26246.24 on 2 df
-log2(p) of ll-ratio test = inf

Execution time: 317.06 seconds
```

CoX-TimescaleDB

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Cox> python .\cox_timescaledb.py
D:\Aarsh\SJSU\CS298\Project\query_execution\Cox\cox_timescaledb.py:23: UserWarning: pandas only supports SQLAlchemy connectable (engine/connection) or database string URI or sqlite3 DBAPI2 connection. Other DBAPI2 objects are not tested. Please consider using SQLAlchemy.
  data = pd.read_sql_query(query, conn)
<lifelines.CoxPHFitter: fitted with 8.90227e+06 total observations, 4.98968e+06 right-censored observations>
  duration col = 'age_end'
  event col = 'is_dead'
  baseline estimation = breslow
  number of observations = 8.90227e+06
  number of events observed = 3.91259e+06
  partial log-likelihood = -56696051.25
  time fit was run = 2023-11-06 19:52:02 UTC

---

```

| | coef | exp(coef) | se(coef) | coef lower 95% | coef upper 95% | exp(coef) lower 95% | exp(coef) upper 95% |
|--------------------|-------|-----------|----------|----------------|----------------|---------------------|---------------------|
| covariate | | | | | | | |
| age_start_observed | -0.01 | 0.99 | 0.00 | -0.01 | -0.01 | 0.99 | 0.99 |
| is_truncated | 0.04 | 1.04 | 0.00 | 0.03 | 0.04 | 1.03 | 1.04 |

| | cmp to | z | p | -log2(p) |
|--------------------|--------|---------|--------|----------|
| covariate | | | | |
| age_start_observed | 0.00 | -113.21 | <0.005 | inf |
| is_truncated | 0.00 | 21.04 | <0.005 | 323.91 |

```
---
Concordance = 0.54
Partial AIC = 113392106.49
log-likelihood ratio test = 26246.24 on 2 df
-log2(p) of ll-ratio test = inf

Execution time: 306.76 seconds
```

CoX-MongoDB

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Cox> python .\cox_mongo.py
<lifelines.CoxPHFitter: fitted with 8.90227e+06 total observations, 4.98968e+06 right-censored observations>
    duration col = 'age_end'
    event col = 'is_dead'
    baseline estimation = breslow
    number of observations = 8.90227e+06
    number of events observed = 3.91259e+06
    partial log-likelihood = -56696051.25
    time fit was run = 2023-11-06 20:06:40 UTC

---

```

| | coef | exp(coef) | se(coef) | coef lower 95% | coef upper 95% | exp(coef) lower 95% | exp(coef) upper 95% |
|--------------------|-------|-----------|----------|----------------|----------------|---------------------|---------------------|
| covariate | | | | | | | |
| age_start_observed | -0.01 | 0.99 | 0.00 | -0.01 | -0.01 | 0.99 | 0.99 |
| is_truncated | 0.04 | 1.04 | 0.00 | 0.03 | 0.04 | 1.03 | 1.04 |

| | cmp to | z | p | -log2(p) |
|--------------------|--------|---------|--------|----------|
| covariate | | | | |
| age_start_observed | 0.00 | -113.21 | <0.005 | inf |
| is_truncated | 0.00 | 21.04 | <0.005 | 323.91 |

```
---
Concordance = 0.54
Partial AIC = 113392106.49
log-likelihood ratio test = 26246.24 on 2 df
-log2(p) of ll-ratio test = inf

Execution time: 333.39 seconds
```


CoX-Cassandra

```
PS D:\Aarsh\SJSU\CS298\Project\query_execution\Cox> python .\cox_cassandra.py
<lifelines.CoxPHFitter: fitted with 8.90227e+06 total observations, 4.98968e+06 right-censored observations>
    duration col = 'age_start_observed'
    event col = 'is_truncated'
    baseline estimation = breslow
    number of observations = 8.90227e+06
    number of events observed = 3.91259e+06
    partial log-likelihood = -56696051.25
    time fit was run = 2023-11-06 20:29:51 UTC

---

```

| | coef | exp(coef) | se(coef) | coef lower 95% | coef upper 95% | exp(coef) lower 95% | exp(coef) upper 95% |
|-------------|-------|-----------|----------|----------------|----------------|---------------------|---------------------|
| covariate | | | | | | | |
| age_end | -0.01 | 0.99 | 0.00 | -0.01 | -0.01 | 0.99 | 0.99 |
| is_censored | 0.04 | 1.04 | 0.00 | 0.03 | 0.04 | 1.03 | 1.04 |

| | cmp to | z | p | -log2(p) |
|-------------|--------|---------|--------|----------|
| covariate | | | | |
| age_end | 0.00 | -113.21 | <0.005 | inf |
| is_censored | 0.00 | 21.04 | <0.005 | 323.91 |

```
---
Concordance = 0.54
Partial AIC = 113392106.49
log-likelihood ratio test = 26246.24 on 2 df
-log2(p) of ll-ratio test = inf

Execution time: 403.29 seconds
```

Results : Data Loading(Write Performance)

| | QuestDB | TimescaleDB | Cassandra | MongoDB |
|-----------|---------|-------------|-----------|---------|
| Dataset 1 | 1.61 | 3.65 | 55.08 | 14.59 |
| Dataset 2 | 22.76 | 34.44 | 558.61 | 192.43 |

Results : Query Execution(Read Performance)

| Dataset1 | Query1 | Query2 | Query3 | Kaplan | LogRank | CoX |
|-------------|--------|--------|--------|--------|---------|-------|
| QuestDB | 0.08 | 0.06 | 0.24 | 2.39 | 2.26 | 23.89 |
| TimescaleDB | 0.10 | 0.30 | 0.14 | 1.56 | 1.20 | 23.21 |
| Cassandra | 4.52 | 23.78 | 8.76 | 12.17 | 10.03 | 31.77 |
| MongoDB | 0.78 | 1.17 | 1.56 | 8.79 | 8.83 | 27.17 |

Results : Query Execution(Read Performance)

| Dataset2 | Query1 | Query2 | Query3 | Kaplan | LogRank | CoX |
|-------------|--------|--------|--------|--------|---------|--------|
| QuestDB | 1.70 | 0.16 | 0.52 | 16.20 | 16.14 | 317.06 |
| TimescaleDB | 1.83 | 0.95 | 0.67 | 13.80 | 12.96 | 306.76 |
| Cassandra | 29.80 | 106.73 | 46.04 | 131.38 | 116.05 | 403.29 |
| MongoDB | 7.99 | 5.61 | 5.17 | 176.33 | 133.78 | 333.39 |