




Pós-Graduação em Ciência de Dados e Machine Learning
Persistência de Dados em Bancos NoSQL (Data Storage)

**Trabalho final - Modelagem de
Dados e Desempenho no MongoDB: Um estudo de caso sobre
Predição de Insuficiência Cardíaca**

Aluna: Bruna Alarcão Alves
Professor: Wandré Nunes de Pinho Veloso

1. Cenário de Aplicação

O estudo de caso “Predição de Insuficiência Cardíaca- Registros Clínicos” tem como objetivo prever a sobrevivência ou não do paciente. O conjunto de dados proposto contém os prontuários de 5.000 pacientes que tiveram insuficiência cardíaca, coletados durante o período de seguimento, onde cada perfil de paciente apresenta 13 características clínicas. A base de dados é utilizada em um hospital especializado em cardiologia para monitorar pacientes com insuficiência cardíaca, a fim de melhorar a qualidade do atendimento ao identificar fatores de risco, personalizar tratamentos, prever desfechos clínicos e melhorar a alocação de recursos. Para este trabalho foi utilizado o dataset "Heart Failure Prediction" encontrado na comunidade online de ciência de dados Kaggle.

Dataset pode ser encontrado neste link: [Heart Failure Prediction - Clinical Records](https://www.kaggle.com/datasets/heart-failure-prediction/heart-failure-prediction-clinical-records)
 ([kaggle.com](https://www.kaggle.com))

2. Modelagem de Dados

Informações do atributo:

- idade: idade do paciente (anos)
- anemia: diminuição dos glóbulos vermelhos ou hemoglobina (booleano)
- creatinofosfoquinase (CPK): nível da enzima CPK no sangue (mcg/L)
- diabetes: se o paciente tem diabetes (booleano)
- fração de ejeção: porcentagem de sangue que sai do coração a cada contração (porcentagem)
- pressão alta: se o paciente tem hipertensão (booleano)
- plaquetas: plaquetas no sangue (quiloplaquetas/mL)
- sexo: mulher ou homem (binário)
- creatinina sérica: nível de creatinina sérica no sangue (mg/dL)
- sódio sérico: nível de sódio sérico no sangue (mEq/L)
- tabagismo: se o paciente fuma ou não (booleano)
- Tempo: Período de seguimento (dias)
- DEATH_EVENT: se o paciente faleceu durante o período de seguimento (booleano)

3. Consultas e Indexação

- **Navegando nos diretórios.**

```
C:\Users\bruna.alarcao>cd down*  
C:\Users\bruna.alarcao\Downloads>cd mongo
```

- **Conectando ao MongoDB Atlas usando o 'mongosh'.**

```
C:\Users\bruna.alarcao\Downloads\mongo>mongosh  
"mongodb+srv://cluster0.viaimdz.mongodb.net/" --apiVersion 1 --username brunaalarcao  
Enter password: *****
```

- **Conexão e Informações do Sistema.**

```
Current Mongosh Log ID: 66644b6f6e0592c05b4a2447  
Connecting to:  
mongodb+srv://<credentials>@cluster0.viaimdz.mongodb.net/?appName=mongosh+2.0.0  
Using MongoDB:      7.0.11 (API Version 1)  
Using Mongosh:      2.0.0  
mongosh 2.2.6 is available for download: https://www.mongodb.com/try/download/shell  
For mongosh info see: https://docs.mongodb.com/mongosh-shell/
```

- **Prompt do MongoDB Atlas.**

```
Atlas atlas-i83q1-shard-0 [primary] test> show databases  
Trabalho_Final  8.00 KiB  
sample_mflix   112.16 MiB  
admin           280.00 KiB  
local           2.89 GiB
```

- **Navegando nos diretórios.**

```
PS C:\Users\bruna.alarcao> cd doc*\mongo  
PS C:\Users\bruna.alarcao> cd down*\mongo
```

- **Importando dados para o MongoDB.**

```
PS C:\Users\bruna.alarcao\Downloads\mongo> .\mongoimport --uri  
"mongodb+srv://brunaalarcao:Abc123-@cluster0.viaimdz.mongodb.net/Trabalho_Final"  
--collection HeartFailure --type csv --file HeartFailure.csv --headerline  
2024-06-08T10:13:24.040-0300 connected to:  
mongodb+srv://[**REDACTED**]@cluster0.viaimdz.mongodb.net/Trabalho_Final
```

- **Progresso da importação dos dados e confirmação que o processo foi bem sucedido.**

```
2024-06-08T10:13:27.043-0300 [#####.....] Trabalho_Final.HeartFailure
92.0KB/218KB (42.2%)
2024-06-08T10:13:30.048-0300 [#####.....] Trabalho_Final.HeartFailure
136KB/218KB (62.4%)
2024-06-08T10:13:33.053-0300 [#####.....] Trabalho_Final.HeartFailure
176KB/218KB (80.7%)
2024-06-08T10:13:36.040-0300 [#####.]
Trabalho_Final.HeartFailure 212KB/218KB (97.2%)
2024-06-08T10:13:38.958-0300 [#####]
Trabalho_Final.HeartFailure 218KB/218KB (100.0%)
2024-06-08T10:13:38.959-0300 5000 document(s) imported successfully. 0 document(s)
failed to import.
```

- **Alterando para o banco de dados 'Trabalho_Final'.**

```
Atlas atlas-i83ql1-shard-0 [primary] test> use
use Trabalho_Final use sample_mflix use admin use local
Atlas atlas-i83ql1-shard-0 [primary] test> use Trabalho_Final
switched to db Trabalho_Final
```

- **Executando consulta no banco de dados 'Trabalho_Final'. Esta consulta retorna todos os registros onde o campo 'creatinine_phosphokinase' tem o valor '30'.**

```
Atlas atlas-i83ql1-shard-0 [primary] Trabalho_Final>
db.HeartFailure.find({"creatinine_phosphokinase": 30});
[
  {
    _id: ObjectId("666458f4670a4268f1a4f8c4"),
    age: 40,
    anaemia: 1,
    creatinine_phosphokinase: 30,
    diabetes: 1,
    ejection_fraction: 35,
    high_blood_pressure: 0,
    platelets: 250000,
    serum_creatinine: 1.1,
    serum_sodium: 140,
    sex: 1,
    smoking: 0,
    time: 212,
    DEATH_EVENT: 0
```

```
},
{
  _id: ObjectId("666458f4670a4268f1a4f9fb"),
  age: 85,
  anaemia: 1,
  creatinine_phosphokinase: 30,
  diabetes: 0,
  ejection_fraction: 60,
  high_blood_pressure: 0,
  platelets: 507000,
  serum_creatinine: 3.2,
  serum_sodium: 138,
  sex: 0,
  smoking: 0,
  time: 94,
  DEATH_EVENT: 0
},
{
  _id: ObjectId("666458f7670a4268f1a4fbf0"),
  age: 49,
  anaemia: 1,
  creatinine_phosphokinase: 30,
  diabetes: 0,
  ejection_fraction: 30,
  high_blood_pressure: 1,
  platelets: 427000,
  serum_creatinine: 1,
  serum_sodium: 138,
  sex: 0,
  smoking: 0,
  time: 12,
  DEATH_EVENT: 0
},
{
  _id: ObjectId("666458f7670a4268f1a4fd97"),
  age: 62,
  anaemia: 0,
  creatinine_phosphokinase: 30,
  diabetes: 1,
  ejection_fraction: 60,
  high_blood_pressure: 1,
  platelets: 244000,
  serum_creatinine: 0.9,
  serum_sodium: 139,
  sex: 1,
  smoking: 0,
  time: 117,
  DEATH_EVENT: 0
}
```

```
},
{
  _id: ObjectId("666458f7670a4268f1a4fe1d"),
  age: 62,
  anaemia: 0,
  creatinine_phosphokinase: 30,
  diabetes: 1,
  ejection_fraction: 60,
  high_blood_pressure: 1,
  platelets: 244000,
  serum_creatinine: 0.9,
  serum_sodium: 139,
  sex: 1,
  smoking: 0,
  time: 117,
  DEATH_EVENT: 0
},
{
  _id: ObjectId("666458f7670a4268f1a4ff17"),
  age: 62,
  anaemia: 0,
  creatinine_phosphokinase: 30,
  diabetes: 1,
  ejection_fraction: 60,
  high_blood_pressure: 1,
  platelets: 244000,
  serum_creatinine: 0.9,
  serum_sodium: 139,
  sex: 1,
  smoking: 0,
  time: 117,
  DEATH_EVENT: 0
},
{
  _id: ObjectId("666458f7670a4268f1a4ff93"),
  age: 62,
  anaemia: 0,
  creatinine_phosphokinase: 30,
  diabetes: 1,
  ejection_fraction: 60,
  high_blood_pressure: 1,
  platelets: 244000,
  serum_creatinine: 0.9,
  serum_sodium: 139,
  sex: 1,
  smoking: 0,
  time: 117,
  DEATH_EVENT: 0
}
```

```
},
{
  _id: ObjectId("666458f9670a4268f1a50283"),
  age: 62,
  anaemia: 0,
  creatinine_phosphokinase: 30,
  diabetes: 1,
  ejection_fraction: 60,
  high_blood_pressure: 1,
  platelets: 244000,
  serum_creatinine: 0.9,
  serum_sodium: 139,
  sex: 1,
  smoking: 0,
  time: 117,
  DEATH_EVENT: 0
},
{
  _id: ObjectId("666458fc670a4268f1a503f2"),
  age: 62,
  anaemia: 0,
  creatinine_phosphokinase: 30,
  diabetes: 1,
  ejection_fraction: 60,
  high_blood_pressure: 1,
  platelets: 244000,
  serum_creatinine: 0.9,
  serum_sodium: 139,
  sex: 1,
  smoking: 0,
  time: 117,
  DEATH_EVENT: 0
},
{
  _id: ObjectId("666458fc670a4268f1a50730"),
  age: 62,
  anaemia: 0,
  creatinine_phosphokinase: 30,
  diabetes: 1,
  ejection_fraction: 60,
  high_blood_pressure: 1,
  platelets: 244000,
  serum_creatinine: 0.9,
  serum_sodium: 139,
  sex: 1,
  smoking: 0,
  time: 117,
  DEATH_EVENT: 0
}
```



```
},  
{  
  _id: ObjectId("66645900670a4268f1a507ad"),  
  age: 62,  
  anaemia: 0,  
  creatinine_phosphokinase: 30,  
  diabetes: 1,  
  ejection_fraction: 60,  
  high_blood_pressure: 1,  
  platelets: 244000,  
  serum_creatinine: 0.9,  
  serum_sodium: 139,  
  sex: 1,  
  smoking: 0,  
  time: 117,  
  DEATH_EVENT: 0  
}  
]
```

4. Teste de Desempenho

- **Utilizando 'explain' para obter detalhes sobre como o MongoDB planejou e executou.**

```
Atlas atlas-i83ql1-shard-0 [primary] Trabalho_Final> db.HeartFailure.find({$or:
[{"creatinine_phosphokinase": 30}]}).explain();
{
```

- **Indicando a versão do formato 'explain' utilizado.**
explainVersion: '1',

- **Contém detalhes sobre como o MongoDB planeja executar a consulta.**

```
queryPlanner: {
  namespace: 'Trabalho_Final.HeartFailure',
```

- **Indicando que não há filtros de índice aplicados à consulta.**

```
indexFilterSet: false,
  parsedQuery: { creatinine_phosphokinase: { '$eq': 30 } },
```

- **Identificadores hash que representam a consulta e o plano de execução.**

```
queryHash: '5E7F0BD4',
planCacheKey: '5E7F0BD4',
```

- **Indicadores booleanos mostrando que nenhum dos limites máximos para essas soluções foi atingido.**

```
maxIndexedOrSolutionsReached: false,
maxIndexedAndSolutionsReached: false,
maxScansToExplodeReached: false,
```

- **'COLLSCAN' significa que o MongoDB está executando uma varredura completa da coleção (Collection Scan).**
- **'Filter' mostra o filtro aplicado durante a varredura.**

```
winningPlan: {
  stage: 'COLLSCAN',
  filter: { creatinine_phosphokinase: { '$eq': 30 } },
  direction: 'forward'
},
rejectedPlans: []
},
command: {
```

```

find: 'HeartFailure',
filter: { '$or': [ { creatinine_phosphokinase: 30 } ] },
'$db': 'Trabalho_Final'
},

```

- **Informações sobre o servidor MongoDB que executou a consulta.**

```

serverInfo: {
  host: 'ac-nzyiak0-shard-00-02.viaimdz.mongodb.net',
  port: 27017,
  version: '7.0.11',
  gitVersion: 'f451220f0df2b9dfe073f1521837f8ec5c208a8c'
},

```

- **Parâmetros de configuração do servidor. 'ok: 1' indica que a operação foi bem sucedida.**

```

serverParameters: {
  internalQueryFacetBufferSizeBytes: 104857600,
  internalQueryFacetMaxOutputDocSizeBytes: 104857600,
  internalLookupStageIntermediateDocumentMaxSizeBytes: 16793600,
  internalDocumentSourceGroupMaxMemoryBytes: 104857600,
  internalQueryMaxBlockingSortMemoryUsageBytes: 33554432,
  internalQueryProhibitBlockingMergeOnMongoS: 0,
  internalQueryMaxAddToSetBytes: 104857600,
  internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600,
  internalQueryFrameworkControl: 'trySbeRestricted'
},
ok: 1,

```

- **Informações sobre o tempo de operação e a consistência do cluster.**

```

'$clusterTime': {
  clusterTime: Timestamp({ t: 1717855019, i: 8 }),
  signature: {
    hash: Binary.createFromBase64("tRdLLOd3G1sRoXnXEZ4SiJKM/oM=", 0),
    keyId: Long("7314013539092398081")
  }
},
operationTime: Timestamp({ t: 1717855019, i: 8 })
}

```

- **Listando todos os métodos e propriedades disponíveis para o objeto 'db' no Shell MongoDB.**

```

Atlas atlas-i83ql1-shard-0 [primary] Trabalho_Final> db.
db.__proto__          db.constructor        db.hasOwnProperty
db.isPrototypeOf      db.propertyIsEnumerable db.toLocaleString

```

| | | |
|------------------------------|----------------------------------|--------------------------|
| db.toString | db.valueOf | db.getMongo |
| db.getName | db.getCollectionNames | db.getCollectionInfos |
| db.runCommand | db.adminCommand | db.aggregate |
| db.getSiblingDB | db.getCollection | db.dropDatabase |
| db.createUser | db.updateUser | db.changeUserPassword |
| db.logout | db.dropUser | db.dropAllUsers |
| db.auth | db.grantRolesToUser | db.revokeRolesFromUser |
| db.getUser | db.getUsers | db.createCollection |
| db.createEncryptedCollection | db.createView | db.createRole |
| db.updateRole | db.dropRole | db.dropAllRoles |
| db.grantRolesToRole | db.revokeRolesFromRole | db.grantPrivilegesToRole |
| db.revokePrivilegesFromRole | db.getRole | db.getRoles |
| db.currentOp | db.killOp | db.shutdownServer |
| db.fsyncLock | db.fsyncUnlock | db.version |
| db.serverBits | db.isMaster | db.hello |
| db.serverBuildInfo | db.serverStatus | db.stats |
| db.hostInfo | db.serverCmdLineOpts | db.rotateCertificates |
| db.printCollectionStats | db.getProfilingStatus | db.setProfilingLevel |
| db.setLogLevel | db.getLogComponents | db.commandHelp |
| db.listCommands | db.printSecondaryReplicationInfo | db.getReplicationInfo |
| db.printReplicationInfo | db.watch | db.sql |
| db.HeartFailure | | |

- **Criando o índice com uma ordenação crescente '1'.**

```
Atlas atlas-i83ql1-shard-0 [primary] Trabalho_Final> db.HeartFailure.createIndex
({"creatinine_phosphokinase": 1})
```

- **Nome do índice criado.**

```
creatinine_phosphokinase_1
```

- **Fornece detalhes sobre como o MongoDB planeja executar a consulta.**

```
Atlas atlas-i83ql1-shard-0 [primary] Trabalho_Final> db.HeartFailure.find({$or:
[{"creatinine_phosphokinase": 30}]}).explain();
{
  explainVersion: '1',
  queryPlanner: {
    namespace: 'Trabalho_Final.HeartFailure',
    indexFilterSet: false,
    parsedQuery: { creatinine_phosphokinase: { '$eq': 30 } },
    queryHash: '5E7F0BD4',
    planCacheKey: '37188589',
    maxIndexedOrSolutionsReached: false,
    maxIndexedAndSolutionsReached: false,
```

maxScansToExplodeReached: false,

- **Indica o plano de execução escolhido pelo MongoDB.**

winningPlan: {

- **Indica que os documentos serão buscados na coleção.**

stage: 'FETCH',

- **Indica que o MongoDB usará um índice para buscar os documentos.**

inputStage: {

stage: 'IXSCAN',

keyPattern: { creatinine_phosphokinase: 1 },

indexName: 'creatinine_phosphokinase_1',

isMultiKey: false,

multiKeyPaths: { creatinine_phosphokinase: [] },

isUnique: false,

isSparse: false,

isPartial: false,

indexVersion: 2,

direction: 'forward',

- **Mostra os limites do índice para a busca.**

indexBounds: { creatinine_phosphokinase: ['[30, 30]'] }

}

},

rejectedPlans: []

},

command: {

find: 'HeartFailure',

filter: { '\$or': [{ creatinine_phosphokinase: 30 }] },

'\$db': 'Trabalho_Final'

},

serverInfo: {

host: 'ac-nzyiak0-shard-00-02.viaimdz.mongodb.net',

port: 27017,

version: '7.0.11',

gitVersion: 'f451220f0df2b9dfe073f1521837f8ec5c208a8c'

},

serverParameters: {

internalQueryFacetBufferSizeBytes: 104857600,

internalQueryFacetMaxOutputDocSizeBytes: 104857600,

internalLookupStageIntermediateDocumentMaxSizeBytes: 16793600,

internalDocumentSourceGroupMaxMemoryBytes: 104857600,

internalQueryMaxBlockingSortMemoryUsageBytes: 33554432,

internalQueryProhibitBlockingMergeOnMongoS: 0,

internalQueryMaxAddToSetBytes: 104857600,

internalDocumentSourceSetWindowFieldsMaxMemoryBytes: 104857600,

```
    internalQueryFrameworkControl: 'trySbeRestricted'
  },
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1717855550, i: 1 }),
    signature: {
      hash: Binary.createFromBase64("bOgWFuaTMLPXkY76YRaF1b06GqY=", 0),
      keyId: Long("7314013539092398081")
    }
  },
  operationTime: Timestamp({ t: 1717855550, i: 1 })
}
```

5. Conclusão e Recomendações

A criação do índice no campo `creatinine_phosphokinase` foi realizada para melhorar o desempenho das consultas que envolvem esse campo.

A consulta `find` foi explicada para entender como o MongoDB planeja executá-la, especialmente após a criação do índice.

O MongoDB planeja usar o índice `creatinine_phosphokinase_1` para buscar os documentos onde `creatinine_phosphokinase` é igual a 30, o que resulta em um melhor desempenho de consulta e um sistema mais eficiente.

Otimização da consulta: o estágio 'COLLSCAN' (Collection Scan) indica que o MongoDB está realizando uma varredura completa da coleção para encontrar os documentos que correspondem aos critérios de consulta. As varreduras completas são mais lentas, pois todos os documentos devem ser lidos e filtrados. No caso inicial, a consulta foi processada como uma varredura completa da coleção (COLLSCAN), o que significa que o MongoDB teve que olhar todos os documentos na coleção `HeartFailure` para encontrar aqueles onde `creatinine_phosphokinase` era igual a 30.

Já no 'FETCH' utiliza um índice para encontrar os documentos, o que é mais rápido, pois com o índice reduz o número de documentos que precisam ser lidos. Depois de criar um índice no campo `creatinine_phosphokinase`, a consulta foi processada com um estágio FETCH que seguiu uma busca de índice (IXSCAN). Isso significa que o MongoDB usou o índice `creatinine_phosphokinase_1` para localizar rapidamente os documentos relevantes e, em seguida, recuperou os documentos completos correspondentes a partir da coleção, reduzindo drasticamente o número de documentos que precisam ser examinados diretamente.