

Using FeatureExtraction (Korean)

Jeon Ga Bin & Martijn J. Schuemie

2025-04-16

1

```
(cohort) (feature) .  
, , , .  
.  
.  
.  
FeatureExtraction CohortMethod PatientLevelPrediction  
FeatureExtraction .
```

2

- 1.
- 2.
- 3.

```
(cohort start) — , .  
ID ( 1) ( 2) ( 3)  
ID : ID ID (: Charlson Comorbidity ). ID  
ID ID ,
```

2.1

:

```
settings <- createDefaultCovariateSettings()
```

:

```
settings <- createDefaultCovariateSettings(
  excludedCovariateConceptIds = 1124300,
  addDescendantsToExclude = TRUE
)
```

1124300() (,) .

2.2

createCovariateSettings .

createCovariateSettings

:

```
settings <- createCovariateSettings(
  useDemographicsGender = TRUE,
  useDemographicsAgeGroup = TRUE,
  useConditionOccurrenceAnyTimePrior = TRUE
)
```

, (5) () — .

- : 365 .
- : 180 .
- : 30 .

:

```
settings <- createCovariateSettings(
  useConditionEraLongTerm = TRUE,
  useConditionEraShortTerm = TRUE,
  useDrugEraLongTerm = TRUE,
  useDrugEraShortTerm = TRUE,
  longTermStartDays = -180,
  shortTermStartDays = -14,
  endDays = -1
)
```

180 () 14 () .
ID .

```
settings <- createCovariateSettings(
  useConditionEraLongTerm = TRUE,
  useConditionEraShortTerm = TRUE,
  useDrugEraLongTerm = TRUE,
  useDrugEraShortTerm = TRUE,
  longTermStartDays = -180,
```

```
shortTermStartDays = -14,  
endDays = -1,  
excludedCovariateConceptIds = 1124300,  
addDescendantsToExclude = TRUE  
)
```

2.3

SQL

```
settings <- createCovariateSettings(useConditionEraLongTerm = TRUE)  
settings2 <- convertPrespecSettingsToDetailedSettings(settings)  
settings2$analyses[[1]]
```

```
## $analysisId  
## [1] 202  
##  
## $sqlFileName  
## [1] "DomainConcept.sql"  
##  
## $parameters  
## $parameters$analysisId  
## [1] 202  
##  
## $parameters$analysisName  
## [1] "ConditionEraLongTerm"  
##  
## $parameters$startDay  
## [1] -365  
##  
## $parameters$endDay  
## [1] 0  
##  
## $parameters$subType  
## [1] "all"  
##  
## $parameters$domainId  
## [1] "Condition"  
##  
## $parameters$domainTable  
## [1] "condition_era"  
##  
## $parameters$domainConceptId  
## [1] "condition_concept_id"  
##  
## $parameters$domainStartDate  
## [1] "condition_era_start_date"  
##  
## $parameters$domainEndDate  
## [1] "condition_era_end_date"  
##  
## $parameters$description
```

```

## [1] "One covariate per condition in the condition_era table overlapping with any part of the long te
##
##
## $includedCovariateConceptIds
## list()
##
## $includedCovariateIds
## list()
##
## $addDescendantsToInclude
## [1] FALSE
##
## $excludedCovariateConceptIds
## list()
##
## $addDescendantsToExclude
## [1] FALSE

```

```

analysisDetails <- createAnalysisDetails(
  analysisId = 1,
  sqlFileName = "DemographicsGender.sql",
  parameters = list(
    analysisId = 1,
    analysisName = "Gender",
    domainId = "Demographics"
  ),
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)

settings <- createDetailedCovariateSettings(list(analysisDetails))

```

2.4

```

( : , , ) .
. , 365 . .
ID ID . CohortMethod .
PatientLevelPrediction .

```

```

settings <- createDefaultTemporalCovariateSettings()

```

```
settings <- createTemporalCovariateSettings(
  useConditionOccurrence = TRUE,
  useMeasurementValue = TRUE
)
```

condition_occurrence , CDM measurement
365 () 7

```
settings <- createTemporalCovariateSettings(
  useConditionOccurrence = TRUE,
  useMeasurementValue = TRUE,
  temporalStartDays = seq(-364, -7, by = 7),
  temporalEndDays = seq(-358, -1, by = 7)
)
```

```
analysisDetails <- createAnalysisDetails(
  analysisId = 1,
  sqlFileName = "MeasurementValue.sql",
  parameters = list(
    analysisId = 1,
    analysisName = "MeasurementValue",
    domainId = "Measurement"
  ),
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

```
settings <- createDetailedTemporalCovariateSettings(list(analysisDetails))
```

3

, diclofenaca cellecoxi .

3.1

R . CohortMethod createConnectionDetails DatabaseConnector
(DBMS) createConnectionDetails . PostgreSQL .

```
connectionDetails <- createConnectionDetails(
  dbms = "postgresql",
  server = "localhost/ohdsi",
  user = "joe",
  password = "supersecret"
)
```

```
cdmDatabaseSchema <- "my_cdm_data"
resultsDatabaseSchema <- "my_results"
```

```
cdmDatabaseSchema resultsDatabaseSchema . CDM , R
. Microsoft SQL Server cdmDatabaseSchema <- "my_cdm_data.dbo"
```

3.2

```
FeatureExtraction . SQL 1
. 365 . FeatureExtraction .
```

```

/*****
File cohortsOfInterest.sql
*****/

IF OBJECT_ID('@resultsDatabaseSchema.cohorts_of_interest', 'U') IS NOT NULL
  DROP TABLE @resultsDatabaseSchema.cohorts_of_interest;

SELECT first_use.*
INTO @resultsDatabaseSchema.cohorts_of_interest
FROM (
  SELECT drug_concept_id AS cohort_definition_id,
         MIN(drug_era_start_date) AS cohort_start_date,
         MIN(drug_era_end_date) AS cohort_end_date,
         person_id
  FROM @cdmDatabaseSchema.drug_era
  WHERE drug_concept_id = 1118084 -- celecoxib
         OR drug_concept_id = 1124300 -- diclofenac
  GROUP BY drug_concept_id,
           person_id
) first_use
INNER JOIN @cdmDatabaseSchema.observation_period
  ON first_use.person_id = observation_period.person_id
  AND cohort_start_date >= observation_period_start_date
  AND cohort_end_date <= observation_period_end_date
WHERE DATEDIFF(DAY, observation_period_start_date, cohort_start_date) >= 365;
```

```
Sql SqlRender . SQL . SQL CDM . SQL
. SQL . SqlRender SQL .
```

```
library(SqlRender)
sql <- readSql("cohortsOfInterest.sql")
sql <- render(sql,
  cdmDatabaseSchema = cdmDatabaseSchema,
  resultsDatabaseSchema = resultsDatabaseSchema
)
sql <- translate(sql, targetDialect = connectionDetails$dbms)

connection <- connect(connectionDetails)
executeSql(connection, sql)
```

SQL . . . SQL . . . DBMS . . . ,

```
sql <- paste(
  "SELECT cohort_definition_id, COUNT(*) AS count",
  "FROM @resultsDatabaseSchema.cohorts_of_interest",
  "GROUP BY cohort_definition_id"
)
sql <- render(sql, resultsDatabaseSchema = resultsDatabaseSchema)
sql <- translate(sql, targetDialect = connectionDetails$dbms)

querySql(connection, sql)
```

```
## cohort_concept_id count
## 1 1124300 240761
## 2 1118084 47293
```

3.3 1

1 .

```
covariateSettings <- createDefaultCovariateSettings()

covariateData <- getDbCovariateData(
  connectionDetails = connectionDetails,
  cdmDatabaseSchema = cdmDatabaseSchema,
  cohortDatabaseSchema = resultsDatabaseSchema,
  cohortTable = "cohorts_of_interest",
  cohortIds = c(1118084),
  rowIdField = "subject_id",
  covariateSettings = covariateSettings
)

summary(covariateData)
```

3.3.1

covariateData .

covariateData\$covariates

- rowId . getDbCovariateData rowIdField = "subject_id" rowId
- subject_id . rowId
- covariate covariates cohortData\$covariateRef .
- covariateValue .

3.3.2

`saveCovariateData(covariateDataobjects, filename, connectionDetails, cdmDatabaseSchema)`

```
saveCovariateData(covariateData, "covariates")
```

```
loadCovariateData(filename, connectionDetails, cdmDatabaseSchema)
```

3.3.3

1

```
tidyCovariateData(covariateData)
```

1. `tidyCovariateData` returns a `tidyCovariates` object.
2. `tidyCovariates` has a `metaData` slot containing information about the covariates.
3. `tidyCovariates` has a `deletedCovariateIds` slot containing the IDs of covariates that were deleted.

```
tidyCovariates <- tidyCovariateData(covariateData,
  minFraction = 0.001,
  normalize = TRUE,
  removeRedundancy = TRUE
)
```

```
tidyCovariates$metaData
```

```
deletedCovariateIds <- tidyCovariates$metaData$deletedInfrequentCovariateIds
head(deletedCovariateIds)
```

```
tidyCovariates$metaData$deletedRedundantCovariateIds
```

```
deletedCovariateIds <- tidyCovariates$metaData$deletedRedundantCovariateIds
head(deletedCovariateIds)
```

3.4

1. `aggregateCovariates` aggregates covariates into a single covariate.
1. `aggregateCovariates` returns a `tidyCovariates` object.

```
covariateData2 <- aggregateCovariates(covariateData)
```

```
covariateSettings <- createDefaultCovariateSettings()
```

```
covariateData2 <- getDbCovariateData(
  connectionDetails = connectionDetails,
  cdmDatabaseSchema = cdmDatabaseSchema,
  covariateSettings = covariateSettings
)
```



```

cohortDatabaseSchema = resultsDatabaseSchema,
cohortTable = "cohorts_of_interest",
cohortIds = c(1118084),
covariateSettings = covariateSettings,
aggregated = TRUE
)
summary(covariateData2)

```

```

aggregated = TRUE      .      personId      rowIdField      .

```

3.4.1

```

covariateData      .

```

```

covariateData2$covariates

```

```

covariateData2$covariatesContinuous

```

covariates

- covariateId , cohortData\$covariateRef
- sumValue . , 1 .
- averageValue . , 1

covariatesContinuous

- covariateId , cohortData\$covariateRef
- countValue ()
- minValue,maxValue,averageValue,standardDeviation,medianValue,p10Value,p25Value,p75Value,p90Value
 . (: Charlson comorbidity index) 0 0 (:) 0
 . covariateData\$analysisRef missingMeansZero .

3.5 1

```

      .      ' 1'      .      1      .

```

```

result <- createTable1(
  covariateData1 = covariateData2,
  output = "one column"
)
print(result, row.names = FALSE, right = FALSE)

```

```

      ,      ,      365      . createTable1      .
getDefaultValueSpecifications      .      ID      ID      ID      ID      .      1      .
      covariateData      1      .      createTable1CovariateSettings

```

```

covariateSettings <- createTable1CovariateSettings()

covariateData2b <- getDbCovariateData(
  connectionDetails = connectionDetails,
  cdmDatabaseSchema = cdmDatabaseSchema,
  cohortDatabaseSchema = resultsDatabaseSchema,
  cohortTable = "cohorts_of_interest",
  cohortIds = c(1118084),
  covariateSettings = covariateSettings,
  aggregated = TRUE
)
summary(covariateData2b)

```

4.

1

```

settings <- createTable1CovariateSettings(
  excludedCovariateConceptIds = c(1118084, 1124300),
  addDescendantsToExclude = TRUE
)

covCelecoxib <- getDbCovariateData(
  connectionDetails = connectionDetails,
  cdmDatabaseSchema = cdmDatabaseSchema,
  cohortDatabaseSchema = resultsDatabaseSchema,
  cohortTable = "cohorts_of_interest",
  cohortIds = c(1118084),
  covariateSettings = settings,
  aggregated = TRUE
)

covDiclofenac <- getDbCovariateData(
  connectionDetails = connectionDetails,
  cdmDatabaseSchema = cdmDatabaseSchema,
  cohortDatabaseSchema = resultsDatabaseSchema,
  cohortTable = "cohorts_of_interest",
  cohortIds = c(1124300),
  covariateSettings = settings,
  aggregated = TRUE
)
std <- computeStandardizedDifference(covCelecoxib, covDiclofenac)

```

celecoxib (1118084) diclofenac (1124300)

```
head(std)
```

```
stdDiff
```

1

```
result <- createTable1(  
  covariateData1 = covCelecoxib,  
  covariateData2 = covDiclofenac,  
  output = "two columns"  
)  
print(result, row.names = FALSE, right = FALSE)
```