Hogeschool van Amsterdam

Capstone Big-Data in urban technology

Batting

Klaas Berbee

Thierry Ligeon

supervised by

Dr. J.R. Helmus

November 7, 2017

Contents

1 Capstone Batting **3**
1.1 Context van de dataset . . . 3
1.1.1 Belanghebbende voor deze dataset . . . 3
1.1.2 Hoe de dataset tot stand gekomen is . . . 3
1.1.3 Hedendaagse problemen oplossen m.b.v. deze dataset . . . 3
1.2 Database Theorie . . . 4
1.2.1 Entity-Relationship Diagram . . . 4
1.2.2 Datawarehouse . . . 4
1.3 Extract Transform Load . . . 4
1.3.1 Data importeren via MySQL . . . 4
1.3.2 Data transformeren . . . 5
1.4 Data Exploreren . . . 8

This page is intentionally left blank.

1 Capstone Batting

Klaas Berbee & Thierry Ligeon

Voor de capstone van de minor Big-Data in urban technology aan de Hogeschool van Amsterdam moeten wij de Batting dataset gaan bestuderen. Als eerste beginnen we met het globaal inzicht krijgen van de dataset.

1.1 Context van de dataset

De Batting dataset bevat data over o.a.: teams, spelers en managers. Ook bevat de dataset informatie over hun salarissen en skills. Verder bevat deze dataset informatie over hoe de individuele wedstrijden verlopen zijn, welke activiteiten hierin zijn voorgekomen en door wie deze zijn uitgevoerd.

Het doel is om deze dateset om te zetten naar nuttige informatie en dit vervolgens conclusies uit kunnen trekken. Teams kunnen bijvoorbeeld een inzicht krijgen in welke factoren een grote rol spelen in de prestatie van het team.

1.1.1 Belanghebbende voor deze dataset

Kansspel spelers
: Er kan een voordeel voor kansspelspelers ontstaan zodra zij bezitten over voorspelmodellen voor komende wedstrijden.

Teamcoaches
: De coach kan uit de dataset de sterke en zwakke punten van de spelers inzien. Hierop kan hij zijn trainingen specificeren op bepaalde aspecten voor zijn spelers. Daarnaast kunnen de sterke en zwakke punten van de competitie bekeken worden om zo beter voor te kunnen bereiden oo aankomende wedstrijden.

Junior spelers
: Beginnende spelers kunnen hun idolen en vaardigheden vergelijken en bekijken bij welke club zij getraind hebben. Dit zou een aanleiding kunnen zijn om een club te kiezen

Stadions
: Met deze dataset kan de hoeveelheid bezoekers voorspeld worden. Hierop kunnen prijzen, hoeveelheid beveiliging en promotie op worden aangepast.

1.1.2 Hoe de dataset tot stand gekomen is

Deze dataset is initieel opgezet door Sean Lahman in 1994. De dataset bevat data van 1871 tot 2012[1]. Echter is deze dataset enkele keren herschreven naar een nieuw format. Daarnaast hebben ook meerdere mensen bijgedragen aan deze dataset. Hierdoor zouden er mogelijke imperfecties in de dataset kunnen zijn onstaan.

1.1.3 Hedendaagse problemen oplossen m.b.v. deze dataset

De World Series MLB 2017 is in november afgesloten en gewonnen door de Houston Astros[2]. Er zouden modellen gemaakt kunnen worden op basis van de data van afgelopen jaren, en vergeleken kunnen worden met de uitslag van 2017. Hierop kan een deel van de prestatie van het model worden gebaseerd. Dit zou eveneens met de uitslagen van 2013 tot en met 2016 gedaan worden. Hieruit kan ook afgeleid worden of het model effectief blijft in de toekomst.

1.2 Database Theorie

1.2.1 Entity-Relationship Diagram

De Batting dataset bestaat uit een relationele database. Hieruit kan een ERD worden opgesteld. Enkele problemen die kunnen ontstaan bij het samenvoegen van tabellen:

Door alle tabellen uit de database te bekijken zijn we tot onderstaand ERD gekomen:

1.2.2 Datawarehouse

1.3 Extract Transform Load

Na het opstellen van het ERD en het DWH kunnen we beginnen aan het ETL proces. Des te beter dit proces wordt uitgevoerd, des te makkelijker het daadwerkelijke data analyse proces wordt. De data is opgeslagen in een lokale MySQL Database.

1.3.1 Data importeren via MySQL

Via R-Studio en de package RMySQL kunnen wij een verbinding met een database opzetten. Dit doen wij op de hieronderstaande wijze, Vervolgens bekijken wij welke tabellen de MySQL dataset bevat.

```
my_db <- dbConnect(MySQL(),
                   user='klaasberbee',
                   password='119OKVU',
                   dbname='Batting',
                   host='localhost')

print(dbListTables(my_db))
```

```
##  [1] "AllstarFull"         "Appearances"         "AwardsManagers"
##  [4] "AwardsPlayers"       "AwardsShareManagers" "AwardsSharePlayers"
##  [7] "Batting"             "BattingPost"         "CollegePlaying"
## [10] "Fielding"            "FieldingOF"          "FieldingOFsplit"
## [13] "FieldingPost"        "HallOfFame"          "HomeGames"
## [16] "Managers"            "ManagersHalf"        "Master"
## [19] "Parks"               "Pitching"            "PitchingPost"
## [22] "Salaries"            "Schools"             "SeriesPost"
## [25] "Teams"               "TeamsFranchises"     "TeamsHalf"
```

De database bestaat uit 27 verschillende tabellen, Dit zijn ook de tabellen die wij in onze ERD verwerkt hebben. Uit gemak importeren wij alle tabellen en besluiten wij later in de R-Studio omgeving welke tabellen wij zullen benutten.

```
dbTables <- dbListTables(my_db)

allTables <-
  lapply(dbTables, function(table) {
    dbGetQuery(my_db, paste("select * from", table))
  })
names(allTables) <- paste("db", dbTables, sep = "_")
list2env(allTables, envir = .GlobalEnv)
```

1.3.2 Data transformeren

De opdracht meende dat sommige tabellen variabelen bevatte met het verkeerde type. Ons ERD gaf dit ook al aan. Om dit te verifieren voeren wij een 0e functie uit op het Pitching tabel.

```
glimpse(db_Pitching)
```

```
## Observations: 44,963
## Variables: 30
## $ playerID <chr> "bechtge01", "brainas01", "fergubo01", "fishech01", "...
## $ yearID   <int> 1871, 1871, 1871, 1871, 1871, 1871, 1871, 1871, 1871,...
## $ stint    <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ teamID   <chr> "PH1", "WS3", "NY2", "RC1", "NY2", "TRO", "RC1", "FW1...
## $ lgID     <chr> "NA", "NA", "NA", "NA", "NA", "NA", "NA", "NA", "NA",...
## $ W        <int> 1, 12, 0, 4, 0, 0, 0, 6, 18, 12, 0, 0, 1, 10, 19, 2, ...
## $ L        <int> 2, 15, 0, 16, 1, 0, 1, 11, 5, 15, 0, 2, 0, 17, 10, 0,...
## $ G        <int> 3, 30, 1, 24, 1, 1, 3, 19, 25, 29, 1, 7, 3, 28, 31, 2...
## $ GS       <int> 3, 30, 0, 24, 1, 0, 1, 19, 25, 29, 0, 1, 0, 28, 31, 2...
## $ CG       <int> 2, 30, 0, 22, 1, 0, 1, 19, 25, 28, 0, 1, 0, 22, 22, 2...
## $ SHO      <int> 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0,...
## $ SV       <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 3,...
## $ IPouts   <int> 78, 792, 3, 639, 27, 3, 39, 507, 666, 747, 3, 88, 31,...
## $ H        <int> 43, 361, 8, 295, 20, 1, 20, 261, 285, 430, 1, 50, 10,...
## $ ER       <int> 23, 132, 3, 103, 10, 0, 5, 97, 113, 153, 1, 22, 4, 94...
## $ HR       <int> 0, 4, 0, 3, 0, 0, 0, 5, 3, 4, 0, 4, 0, 9, 2, 0, 7, 0,...
## $ BB       <int> 11, 37, 0, 31, 3, 0, 3, 21, 40, 75, 2, 6, 3, 47, 38, ...
## $ SO       <int> 1, 13, 0, 15, 0, 0, 1, 17, 15, 12, 0, 0, 0, 34, 23, 0...
## $ BAOpp    <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ ERA      <dbl> 7.96, 4.50, 27.00, 4.35, 10.00, 0.00, 3.46, 5.17, 4.5...
## $ IBB      <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ WP       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ HBP      <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ BK       <int> 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ BFP      <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ GF       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ R        <int> 42, 292, 9, 257, 21, 0, 30, 243, 223, 362, 1, 53, 8, ...
## $ SH       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ SF       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ GIDP     <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
```

Zoals hierboven staat, bevat dit tabel zowel variabelen van de classes Intergers, Doubles en Characters. Echter is de class van teamID en lgID nog incorrect, dit zou een factor moeten zijn. Dit betekend dat dit in meerdere tabellen een probleem kan zijn. Kijkend naar de vragen in de opdrachtbeschrijving hebben we besloten om vanaf nu gebruik te maken van de volgende tabellen:

- Pitching
- Batting
- Fielding
- Teams
- Salaries

```
glimpse(db_Batting)
```

```
## Observations: 102,816
```

```
## Variables: 22
## $ playerID <chr> "abercda01", "addybo01", "allisar01", "allisdo01", "a...
## $ yearID   <int> 1871, 1871, 1871, 1871, 1871, 1871, 1871, 1871, 1871,...
## $ stint    <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ teamID   <chr> "TRO", "RC1", "CL1", "WS3", "RC1", "FW1", "RC1", "BS1...
## $ lgID     <chr> "NA", "NA", "NA", "NA", "NA", "NA", "NA", "NA", "NA",...
## $ G        <int> 1, 25, 29, 27, 25, 12, 1, 31, 1, 18, 22, 1, 10, 3, 20...
## $ AB       <int> 4, 118, 137, 133, 120, 49, 4, 157, 5, 86, 89, 3, 36, ...
## $ R        <int> 0, 30, 28, 28, 29, 9, 0, 66, 1, 13, 18, 0, 6, 7, 24, ...
## $ H        <int> 0, 32, 40, 44, 39, 11, 1, 63, 1, 13, 27, 0, 7, 6, 33,...
## $ `2B`     <int> 0, 6, 4, 10, 11, 2, 0, 10, 1, 2, 1, 0, 0, 0, 9, 3, 0,...
## $ `3B`     <int> 0, 0, 5, 2, 3, 1, 0, 9, 0, 1, 10, 0, 0, 0, 1, 3, 0, 0...
## $ HR       <int> 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 3, 0, 0, 0, 1, 0, 0, 0,...
## $ RBI      <int> 0, 13, 19, 27, 16, 5, 2, 34, 1, 11, 18, 0, 1, 5, 21, ...
## $ SB       <int> 0, 8, 3, 1, 6, 0, 0, 11, 0, 1, 0, 0, 2, 2, 4, 4, 0, 0...
## $ CS       <int> 0, 1, 1, 1, 2, 1, 0, 6, 0, 0, 1, 0, 0, 0, 0, 4, 0, 0,...
## $ BB       <int> 0, 4, 2, 0, 2, 0, 1, 13, 0, 0, 3, 1, 2, 0, 2, 9, 0, 0...
## $ SO       <int> 0, 0, 5, 2, 1, 1, 0, 1, 0, 0, 4, 0, 0, 0, 2, 2, 3, 0,...
## $ IBB      <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ HBP      <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ SH       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ SF       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ GIDP     <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
```

```
glimpse(db_Fielding)
```

```
## Observations: 136,815
## Variables: 18
## $ playerID <chr> "abercda01", "addybo01", "addybo01", "allisar01", "al...
## $ yearID   <int> 1871, 1871, 1871, 1871, 1871, 1871, 1871, 1871, 1871,...
## $ stint    <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ teamID   <chr> "TRO", "RC1", "RC1", "CL1", "CL1", "WS3", "RC1", "RC1...
## $ lgID     <chr> "NA", "NA", "NA", "NA", "NA", "NA", "NA", "NA", "NA",...
## $ POS      <chr> "SS", "2B", "SS", "2B", "OF", "C", "1B", "2B", "3B", ...
## $ G        <int> 1, 22, 3, 2, 29, 27, 1, 2, 20, 5, 1, 12, 1, 16, 15, 1...
## $ GS       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ InnOuts  <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ PO       <int> 1, 67, 8, 1, 51, 68, 7, 3, 38, 10, 0, 29, 2, 42, 44, ...
## $ A        <int> 3, 72, 14, 4, 3, 15, 0, 4, 52, 0, 0, 2, 0, 61, 64, 0,...
## $ E        <int> 2, 42, 7, 0, 7, 20, 0, 1, 28, 8, 0, 7, 0, 15, 21, 0, ...
## $ DP       <int> 0, 5, 0, 0, 1, 4, 0, 0, 2, 0, 0, 0, 0, 10, 3, 0, 0, 0...
## $ PB       <chr> "", "", "", "", "", "0", "", "", "", "0", "", "", "",...
## $ WP       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ SB       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ CS       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
## $ ZR       <chr> "", "", "", "", "", "", "", "", "", "", "", "", "", "...
```

```
glimpse(db_Teams)
```

```
## Observations: 2,835
## Variables: 48
## $ yearID         <int> 1871, 1871, 1871, 1871, 1871, 1871, 1871, 1871,...
## $ lgID           <chr> "NA", "NA", "NA", "NA", "NA", "NA", "NA", "NA",...
## $ teamID         <chr> "BS1", "CH1", "CL1", "FW1", "NY2", "PH1", "RC1"...
## $ franchID       <chr> "BNA", "CNA", "CFC", "KEK", "NNA", "PNA", "ROK"...
```

```
## $ divID          <chr> "", "", "", "", "", "", "", "", "", "", "",...
## $ Rank           <int> 3, 2, 8, 7, 5, 1, 9, 6, 4, 2, 9, 6, 1, 7, 8, 3,...
## $ G              <int> 31, 28, 29, 19, 33, 28, 25, 29, 32, 58, 29, 37,...
## $ Ghome          <chr> "", "", "", "", "", "", "", "", "", "", "", "",...
## $ W              <int> 20, 19, 10, 7, 16, 21, 4, 13, 15, 35, 3, 9, 39,...
## $ L              <int> 10, 9, 19, 12, 17, 7, 21, 15, 15, 19, 26, 28, 8...
## $ DivWin         <chr> "", "", "", "", "", "", "", "", "", "", "", "",...
## $ WCWin          <chr> "", "", "", "", "", "", "", "", "", "", "", "",...
## $ LgWin          <chr> "N", "N", "N", "N", "N", "Y", "N", "N", "N", "N...
## $ WSWin          <chr> "", "", "", "", "", "", "", "", "", "", "", "",...
## $ R              <int> 401, 302, 249, 137, 302, 376, 231, 351, 310, 61...
## $ AB             <int> 1372, 1196, 1186, 746, 1404, 1281, 1036, 1248, ...
## $ H              <int> 426, 323, 328, 178, 403, 410, 274, 384, 375, 74...
## $ `2B`           <int> 70, 52, 35, 19, 43, 66, 44, 51, 54, 94, 26, 46,...
## $ `3B`           <int> 37, 21, 40, 8, 21, 27, 25, 34, 26, 35, 6, 10, 3...
## $ HR             <int> 3, 10, 7, 2, 1, 9, 3, 6, 6, 14, 0, 0, 7, 0, 1, ...
## $ BB             <int> 60, 60, 26, 33, 33, 46, 38, 49, 48, 27, 14, 19,...
## $ SO             <int> 19, 22, 25, 9, 15, 23, 30, 19, 13, 28, 29, 24, ...
## $ SB             <int> 73, 69, 18, 16, 46, 56, 53, 62, 48, 35, 8, 17, ...
## $ CS             <chr> "", "", "", "", "", "", "", "", "", "15", "4", ...
## $ HBP            <chr> "", "", "", "", "", "", "", "", "", "", "", "",...
## $ SF             <chr> "", "", "", "", "", "", "", "", "", "", "", "",...
## $ RA             <int> 303, 241, 341, 243, 313, 266, 287, 362, 303, 43...
## $ ER             <int> 109, 77, 116, 97, 121, 137, 108, 153, 137, 173,...
## $ ERA            <dbl> 3.55, 2.76, 4.11, 5.17, 3.72, 4.95, 4.30, 5.51,...
## $ CG             <int> 22, 25, 23, 19, 32, 27, 23, 28, 32, 48, 28, 37,...
## $ SHO            <int> 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 0, 3, 0, 0, 3,...
## $ SV             <int> 3, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0,...
## $ IPouts         <int> 828, 753, 762, 507, 879, 747, 678, 750, 846, 15...
## $ HA             <int> 367, 308, 346, 261, 373, 329, 315, 431, 371, 56...
## $ HRA            <int> 2, 6, 13, 5, 7, 3, 3, 4, 4, 3, 6, 6, 0, 6, 5, 2...
## $ BBA            <int> 42, 28, 53, 21, 42, 53, 34, 75, 45, 63, 24, 19,...
## $ SOA            <int> 23, 22, 34, 17, 22, 16, 16, 12, 13, 0, 0, 0, 0,...
## $ E              <int> 225, 218, 223, 163, 227, 194, 220, 198, 217, 43...
## $ DP             <chr> "", "", "", "", "", "", "", "", "", "", "", "",...
## $ FP             <dbl> 0.838, 0.829, 0.814, 0.803, 0.839, 0.845, 0.821...
## $ name           <chr> "Boston Red Stockings", "Chicago White Stocking...
## $ park           <chr> "South End Grounds I", "Union Base-Ball Grounds...
## $ attendance     <chr> "", "", "", "", "", "", "", "", "", "", "", "",...
## $ BPF            <int> 103, 104, 96, 101, 90, 102, 97, 101, 94, 106, 8...
## $ PPF            <int> 98, 102, 100, 107, 88, 98, 99, 100, 98, 102, 96...
## $ teamIDBR       <chr> "BOS", "CHI", "CLE", "KEK", "NYU", "ATH", "ROK"...
## $ teamIDlahman45 <chr> "BS1", "CH1", "CL1", "FW1", "NY2", "PH1", "RC1"...
## $ teamIDretro    <chr> "BS1", "CH1", "CL1", "FW1", "NY2", "PH1", "RC1"...
```

```
glimpse(db_Salaries)
```

```
## Observations: 26,428
## Variables: 5
## $ yearID   <int> 1985, 1985, 1985, 1985, 1985, 1985, 1985, 1985, 1985,...
## $ teamID   <chr> "ATL", "ATL", "ATL", "ATL", "ATL", "ATL", "ATL", "ATL...
## $ lgID     <chr> "NL", "NL", "NL", "NL", "NL", "NL", "NL", "NL", "NL",...
## $ playerID <chr> "barkele01", "bedrost01", "benedbr01", "campri01", "c...
## $ salary   <int> 870000, 550000, 545000, 633333, 625000, 800000, 15000...
```

```
db_Pitching$teamID = as.factor(db_Pitching$teamID)
db_Pitching$lgID = as.factor(db_Pitching$lgID)

db_Batting$teamID = as.factor(db_Batting$teamID)
db_Batting$lgID = as.factor(db_Batting$lgID)

db_Fielding$teamID = as.factor(db_Fielding$teamID)
db_Fielding$lgID = as.factor(db_Fielding$lgID)
db_Fielding$POS = as.factor(db_Fielding$POS)

db_Teams$teamID = as.factor(db_Teams$teamID)
db_Teams$lgID = as.factor(db_Teams$lgID)
db_Teams$franchID = as.factor(db_Teams$franchID)
db_Teams$divID = as.factor(db_Teams$divID)
db_Teams$teamIDBR = as.factor(db_Teams$teamIDBR)
db_Teams$teamIDlahman45 = as.factor(db_Teams$teamIDlahman45)
db_Teams$teamIDretro = as.factor(db_Teams$teamIDretro)

db_Salaries$teamID = as.factor(db_Salaries$teamID)
db_Salaries$lgID = as.factor(db_Salaries$lgID)
```

Het loading proces hoeven wij vervolgens niet meer te doen aangezien wij de data zullen verwerken in dezelfde R-Studio omgeving.

1.4 Data Exploreren

Om te beginnen met het data exploratie proces moeten wij een tabel gebruiken dat verschillende klassen variabele bevat. Hier moeten we voervolgens de summary functie op uit voeren (1e orde functie). Als voorbeeld nemen wij het tabel Pitching.

```
summary
```

```
## standardGeneric for "summary" defined from package "base"
##
## function (object, ...)
## standardGeneric("summary")
## <environment: 0x505c098>
## Methods may be defined for arguments: object
## Use  showMethods("summary")  for currently available ones.
```

Doormiddel van de onderstaande code kan er uitgezocht worden wat de Primary Keys zijn.

References

[1] Sean Lahman, Batting database explanation 2012,
http://seanlahman.com/files/database/readme2012.txt

[2] AD: Houston Astros verslaan Los Angeles Dodgers, 2017,
https://www.ad.nl/andere-sporten/houston-astros-verslaan-los-angeles-dodgers-met-5-1 a6904dad