



Hochschule für
Wirtschaft und Recht Berlin
Berlin School of Economics and Law



NBA 16-17 Regular Season Analysis Enterprise Architectures for Big Data

Oksana Hrytsiv
Laman Mammadova
Fabian Petschke
Adrian Villegas

Project Framework



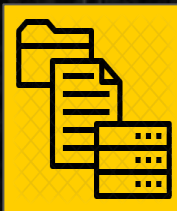
1. Introduction
2. Data Preparation
3. Use Cases
 - a. Scenario 1 (Google BigQuery + Tableau)
 - b. Scenario 2 (Apache Hive + Power BI)
 - c. Scenario 3 (PySpark + Zeppelin)
4. Data Visualization
5. Technologies Evaluation



1. Introduction



Help basketball analysts to evaluate the performance of players and each team



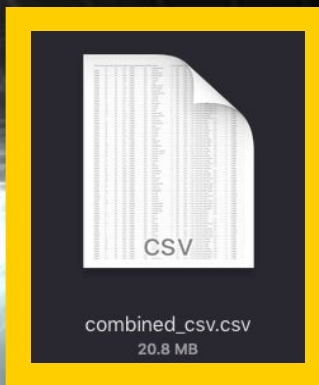
- Game Schedule 16-17-Reg
- Player Regular 16-17 Stats
- Shot log data for each team



Apply big data tools to process and visualize data



2. Data Preparation



Rows: 209176
Columns: 13

| | player_position | home | loc_x | home_team | shot_type | points | away_team | loc_y | time | date | shooter | quarter | outcome |
|---|-----------------|------|-------|-----------|--------------------|--------|-----------|-------|------|------------|----------------|---------|---------|
| 0 | PF | Yes | 107.0 | PHX | Floating Jump Shot | 2 | SAC | 252.0 | 0:51 | 2016-10-26 | Jared Dudley | 1 | SCORED |
| 1 | SG | Yes | 254.0 | PHX | Jump Shot | 3 | SAC | 56.0 | 1:14 | 2016-10-26 | Devin Booker | 1 | MISSED |
| 2 | C | Yes | 52.0 | PHX | Cutting Dunk Shot | 2 | SAC | 250.0 | 1:44 | 2016-10-26 | Tyson Chandler | 1 | SCORED |
| 3 | PG | Yes | 241.0 | PHX | Pullup Jump Shot | 2 | SAC | 359.0 | 2:16 | 2016-10-26 | Eric Bledsoe | 1 | MISSED |
| 4 | SG | Yes | 225.0 | PHX | Jump Shot | 3 | SAC | 447.0 | 2:40 | 2016-10-26 | Devin Booker | 1 | MISSED |



3. Use Cases

1



Google
BigQuery



2



3



Scenario 1 (Google BigQuery + Tableau)



Try the new UI ?

COMPOSE QUERY

Query History

Job History

Scheduled Queries

Transfers

Filter by ID or label ?

NBA BIPM2019

nba_bipm_shots_data

PLAYERS

SHOTS

Public Datasets

bigquery-public-data:hacker_news

bigquery-public-data:noaa_gsod

bigquery-public-data:samples

bigquery-public-data:usa_names

gdelt-bq:hathitrustbooks

gdelt-bq:internetarchivebooks

lookerdata:cdc

nyc-tlc:green

nyc-tlc:yellow

Recent Jobs

Filter jobs

Load

uploaded file to nba-bipm2019:nba_bipm_shots_data.PLAYERS

Repeat Load Job

9:47PM

Job ID

nba-bipm2019:US.bqjob_9d37e62_16bb43a1cbc

Creation Time

Jul 2, 2019, 9:47:17 PM

Start Time

Jul 2, 2019, 9:47:17 PM

End Time

Jul 2, 2019, 9:47:19 PM

User

ksenahr@gmail.com

Destination Table

[nba-bipm2019:nba_bipm_shots_data.PLAYERS](#)

Write Preference

Write if empty

Source Format

CSV

Skip Leading Rows

1

Source URI

uploaded file

Autodetect Schema

true

Repeat Load Job

Cancel Job

Load

gs://nba_shots_bipm_2019/shots_dataframe.csv to nba-bipm2019:nba_bipm_shots_data.SHOTS

Repeat Load Job

9:29PM

Job ID

nba-bipm2019:US.bqjob_58f27863_16bb42a1cbd

Creation Time

Jul 2, 2019, 9:29:47 PM

Start Time

Jul 2, 2019, 9:29:49 PM

End Time

Jul 2, 2019, 9:30:01 PM

User

ksenahr@gmail.com

Destination Table

[nba-bipm2019:nba_bipm_shots_data.SHOTS](#)

Write Preference

Write if empty

Source Format

CSV

Skip Leading Rows

1

Source URI

gs://nba_shots_bipm_2019/shots_dataframe.csv [\(Open in GCS\)](#)

Autodetect Schema

true

Repeat Load Job

Cancel Job



Google BigQuery

Try the new UI



COMPOSE QUERY

Query History

Job History

Scheduled Queries

Transfers

Filter by ID or label



NBA BIPM2019



▼ nba_bipm_shots_data

■ detailed_shots_data

■ PLAYERS

■ SHOTS

▼ Public Datasets

► bigquery-public-data:hacker_news

► bigquery-public-data:noaa_gsod

► bigquery-public-data:samples

► bigquery-public-data:usa_names

► gdelt-bq:hathitrustbooks

► gdelt-bq:internetarchivebooks

► lookedata:cdc

► nyc-tlc:green

► nyc-tlc:yellow

New Query ?

Query Editor

UDF Editor



SQL

```
1 #standardSQL
2 SELECT
3 *
4 FROM
5 `nba_bipm_shots_data.SHOTS` AS a
6 LEFT JOIN
7 `nba_bipm_shots_data.PLAYERS` AS b
8 ON
9 a.shooter = b.shooter_name;
```

Valid: This query will process 19.6 MB when run.

Destination Table

Select Table

nba-bipm2019:nba_bipm_shots_data.detailed_shots_data X

Write Preference

☒ Write if empty

☐ Append to table

☐ Overwrite table

Results Size

☒ Allow Large Results ?

Results Schema

☒ Flatten Results ?

Query Caching

☐ Use Cached Results ?

Query Priority

☒ Interactive

☐ Batch ?

UDF Source URIs

Edit ?

Maximum Bytes Billed

Project Default ?

SQL Dialect

☒ Use Legacy SQL ?

Destination Encryption

Default ?

Processing Location

Unspecified ?

RUN QUERY ▼

Save Query

Save View

Format Query

Schedule Query

Hide Options

Query complete (6.1s elapsed, 19.6 MB processed)



Results

Details

Download as CSV

Download as JSON

Save as Table

Save to Google Sheets

| Row | player_position | home | loc_x | home_team | shot_type | points | away_team | loc_y | time | date | shooter | quarter | outcome | shooter_name |
|-----|-----------------|-------|-------|-----------|-----------|--------|-----------|-------|-------|------------|-----------|---------|---------|--------------|
| 1 | PG | false | 167.0 | DET | Jump Shot | 3 | ATL | 20.0 | 11:17 | 2017-01-18 | Gary Neal | 4 | MISS | null |

Scenario 2 (Apache Hive + Power BI)



```
adrian_villegas_mazo@seneca3:~$ hadoop fs -ls
Found 24 items
drwx----- - adrian_villegas_mazo student      0 2019-07-15 16:00 .Trash
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-29 15:37 .sparkStaging
drwx----- - adrian_villegas_mazo student      0 2019-07-03 23:19 .staging
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-22 19:07 color
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-14 12:45 constitution_hdfs
-rw-r--r-- 3 adrian_villegas_mazo student 2772143 2019-05-22 18:39 diamonds.csv
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 11:23 hadoop
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 17:58 inputdata
-rwxr-xr-x 3 adrian_villegas_mazo student 1640 2019-05-08 17:58 mapper.py
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 11:26 myhdfs
drwxr-xr-x - adrian_villegas_mazo student      0 2019-07-03 21:46 nba_players
drwxr-xr-x - adrian_villegas_mazo student      0 2019-07-03 23:19 nba_shots
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 18:19 outputdata
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 18:38 outputdata2
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 18:59 outputdata3
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 19:13 outputdata4
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 19:27 outputdata5
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 19:35 outputdata6
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-08 19:45 outputdata7
-rw-r--r-- 3 adrian_villegas_mazo student 46806 2019-07-03 21:45 players_dataframe.csv
-rw-r--r-- 3 adrian_villegas_mazo student 2216 2019-05-08 19:44 reducer.py
-rw-r--r-- 3 adrian_villegas_mazo student 16978122 2019-07-03 23:17 shots_dataframe.csv
-rw-r--r-- 3 adrian_villegas_mazo student 183292235 2019-05-14 13:12 wh_visits.txt
drwxr-xr-x - adrian_villegas_mazo student      0 2019-05-22 18:25 words
adrian_villegas_mazo@seneca3:~$ |
```


Scenario 3 (PySpark + Zeppelin)



← → ↻ ⚠ Not secure | 10.50.200.82:8080/#/notebook/2EE6G75FP



Notebook ▾ Job

Search

user1 ▾

NBA_BigDataProject



```
%spark
val sqlContext = new org.apache.spark.sql.hive.HiveContext(sc)
val databases = sqlContext.sql("show databases")
databases.show()
```

FINISHED

warning: there was one deprecation warning; re-run with -d
ereprecation for details

| databaseName |
|----------------------|
| adrian_villegas_mazo |
| ana_maria_cuciuc |
| anneke_lohmann |
| anxhela_merko |
| default |
| diego_conejo |
| ekaterina_diachkova |
| esra_yener |
| fabian_asal |
| fabian_bonnet |

Took 4 sec. Last updated by user1 at July 09 2019, 9:16:00 AM. (outdated)

```
%spark
val sqlContext = new org.apache.spark.sql.hive.HiveContext(sc)
sqlContext.sql("use adrian_villegas_mazo")
val tables = sqlContext.sql("show tables")
tables.show()
```

FINISHED

warning: there was one deprecation warning; re-run with -dep
reprecation for details

| database | tableName | isTemporary |
|----------------------|-------------|-------------|
| adrian_villegas_mazo | nba_players | false |
| adrian_villegas_mazo | nba_shots | false |

```
sqlContext: org.apache.spark.sql.hive.HiveContext = org.apac
he.spark.sql.hive.HiveContext@2f26cc94
tables: org.apache.spark.sql.DataFrame = [database: string,
tableName: string ... 1 more field]
```

Took 1 sec. Last updated by user1 at July 09 2019, 9:16:01 AM. (outdated)

```
%spark.pyspark
from pyspark.sql import HiveContext
hive_context = HiveContext(sc)
df_shots = hive_context.table("adrian_villegas_mazo
.nba_shots")
```

FINISHED

Took 0 sec. Last updated by user1 at July 15 2019, 5:18:24 PM. (outdated)

```
%spark.pyspark
from pyspark.sql import HiveContext
hive_context = HiveContext(sc)
df_players = hive_context.table("adrian_villegas_mazo
.nba_players")
```

FINISHED

Took 0 sec. Last updated by user1 at July 09 2019, 10:45:26 AM. (outdated)

```
%spark.pyspark
nba_players = df_players.toPandas()
nba_shots = df_shots.toPandas()
```

SPARK JOBS FINISHED

Took 15 sec. Last updated by user1 at July 09 2019, 10:45:21 AM. (outdated)



4. Data Visualization and Analysis

Power BI Dashboard



Season 16-17 Shot Location Analysis

Team Name

Warriors

Player Name

Klay Thompson

Home Game

All

Shot Outcome

All

Quarter

All

Game Date

All

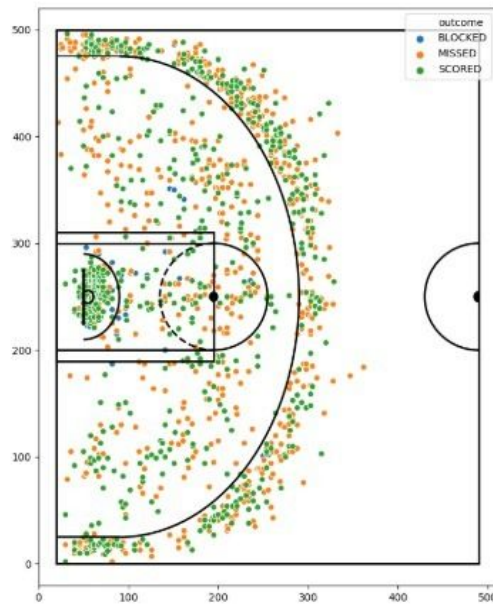


Season Statistics

Conference

Western

| W | L | W/L% |
|-------|-------|------|
| 67 | 15 | 0.8 |
| PA/G | PS/G | SRS |
| 104.3 | 115.9 | 11.4 |

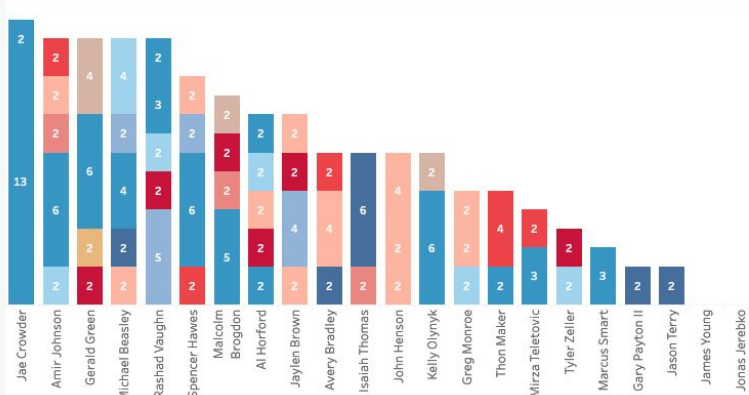


[Click here to see an interactive dashboard](#)

Tableau Dashboard



Player **Points** collection per **Shot Type**



Players Statistics

| Shooter Na.. | Home Team | Max. Poster.. |
|---------------|-----------|---------------|
| Al Horford | BOS | 0.4730667.. |
| Amir Johns.. | BOS | 0.5763029.. |
| Avery Bradl.. | BOS | 0.4631643.. |
| Gary Payto.. | BOS | 0.3722999.. |
| Gerald Green | BOS | 0.4081400.. |
| Greg Monroe | BOS | 0.5327410.. |
| Isaiah Tho.. | BOS | 0.4629687.. |
| Jae Crowder | BOS | 0.4624365.. |
| James Young | BOS | 0.4316385.. |
| Jason Terry | BOS | 0.4322365.. |
| Jaylen Bro.. | BOS | 0.4527820.. |
| John Henson | BOS | 0.5123338.. |
| Jonas Jereb.. | BOS | 0.4352124.. |
| Kelly Olynyk | BOS | 0.5104683.. |
| Malcolm Br.. | BOS | 0.4573621.. |
| Marcus Sm.. | BOS | 0.3599238.. |
| Michael Be.. | BOS | 0.5327652.. |

Select Game Date:

4/12/2017 4/12/2017

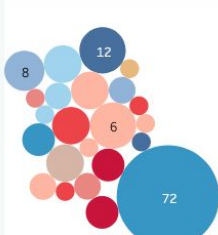
Type Quarter:

Select Team:

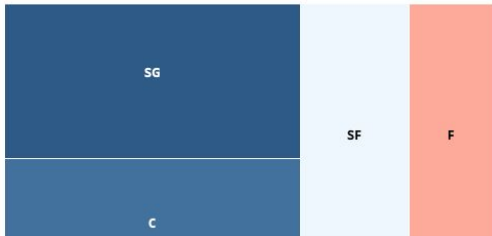
- ☐ (All)
☐ ATL
☒ BOS
☐ BRO
☐ CHA
☐ CHI
☐ CLE
☐ DAL
☐ DEN
☐ DET
☐ GSW
☐ HOU
☐ IND
☐ LAC

Select Sho... (All)

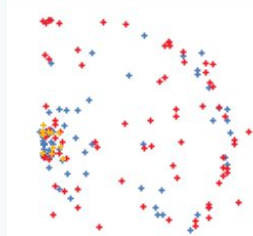
Score per Shot Type



Scored Positions



Shots Map



Outcome

- ☒ BLOCKED
☒ MISSED
☒ SCORED

Select Outcome:

- ☒ (All)
☒ BLOCKED
☒ MISSED
☒ SCORED

[click here to see an interactive dashboard](#)

Zeppelin Notebook



Zeppelin

Notebook ▾ Job

Search

user1 ▾

NBA_BigDataProject



Head ▾



default ▾

NBA_BigDataProject

```
%spark.pyspark
from matplotlib.patches import Circle, Rectangle, Arc
import matplotlib.pyplot as plt

def draw_half_court(ax=None, color='black', lw=2, outer_lines=False):
    # If an axes object isn't provided to plot onto, just get current one
    if ax is None:
        ax = plt.gca()

    backboard1 = Rectangle((50, 225), -1, 50, linewidth=lw, color=color)

    hoop1 = Circle((55, 250), radius=6, linewidth=lw, color=color, fill=False)

    restricted1 = Arc((50, 250), 80, 80, angle=270, theta1=0, theta2=180, linewidth=lw,
                     color=color)

    freethrow1_outer = Arc((195, 250), 100, 120, angle=270, theta1=0, theta2=180,
                          linewidth=lw,
                          color=color)

    freethrow1_inner = Arc((195, 250), 100, 120, angle=270, theta1=180, theta2=0,
                          linewidth=lw,
                          color=color, linestyle='dashed')

    freethrow1_point = Circle((195, 250), radius=4, linewidth=lw, color=color, fill=True)

    threepoint1 = Arc((90, 250), 450, 400, angle=270, theta1=0, theta2=180, linewidth=lw,
                     color=color)

    innerbox1 = Rectangle((20, 200), 175, 100, linewidth=lw, color=color,
                        fill=False)

    outbox1 = Rectangle((20, 190), 175, 120, linewidth=lw, color=color,
                      fill=False)

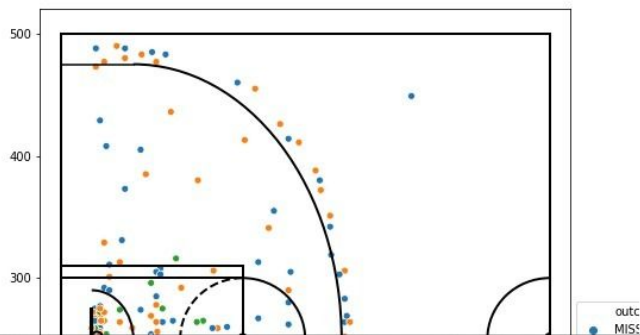
    half_court_circle = Arc((490, 250), 100, 120, angle=270, theta1=180, theta2=0,
```

FINISHED ▶ ⌂ 🔍 ⚙

```
%spark.pyspark
nba_shots_subset = nba_shots[:200]

plt.figure(figsize=(8,10))
sns.scatterplot(x='loc_x', y='loc_y', hue='outcome', data=nba_shots_subset)
draw_half_court(outer_lines=True)
plt.xlim(0,510)
plt.ylim(-20,520)
plt.xlabel('')
plt.ylabel('')
plt.legend(loc='center left', bbox_to_anchor=(1, 0.5), ncol=1)
plt.show()
```

FINISHED ▶ ⌂ 🔍 ⚙





5. Technologies Evaluation

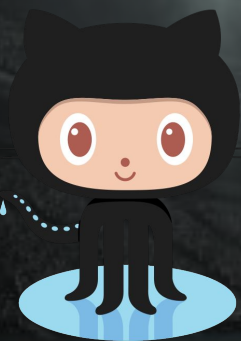


tableau
SOFTWARE



c:\>



Google
BigQuery



hadoop

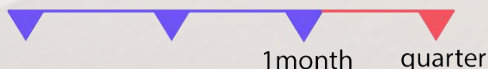


Google Cloud Storage

BigQuery

- ✓ Integration with Tableau
- ✓ On-demand pricing
- ✓ Free processing of small amount of data
- ✓ BigQuery is an asset to analyze billions of rows
- ✓ 3-5 seconds of response time
- ✓ Storing images in Google Storage to be processed by Google BigQuery is not possible

1 TB free querying
10 GB storage



Apache Hive

PySpark

- ✓ Python is slow compared to Scala for Spark Jobs
- ✓ Replaces the Map/Reduce function process with Python language code, which saves time and makes it easier to write
- ✓ Installation of ODBC driver to connect Power BI or Tableau
- ✓ Limited available number of functions built-in into the package



Tableau

- ✓ Integration with BigQuery
- ✓ User-friendly interface
- ✓ Creating visuals using Python is not possible
- ✓ Integration with Hive only through ODBC connector

Power BI

- ✓ Integration with Hive and with several other types of data sources
- ✓ User-friendly interface
- ✓ Allows schema creation regardless of the source with entity-relation models
- ✓ Allows custom Python visuals in the dashboards
- ✓ Free version does not support publishing dashboards with Python visualizations to Web

Zeppelin

- ✓ Integration with Hive
- ✓ User-friendly interface
- ✓ Creating visuals using Python
- ✓ Huge amount of interpreters for different types of data and different types of querying/programming languages
- ✓ Free version does not support publishing dashboards with Python visualizations to Web



Thank You



https://github.com/kseniahr/NBA_BigDataProject



Q&A