


Data Management Plan

General Information

Creator

Biagio Licari, s276019@ds.units.it

ORCID iD

 0000-0002-2291-1782

Affiliation

University of Trieste

Project Title

Nba season statistics management

Project Acronym

NBASODM

Project Abstract

The main goal of this experiment is to create a data management system related to nba team statistical data, in order to facilitate future statistical analysis on such data via ML or other types of data analysis

Raw data was collected via api and web scraping on different official sources related to nba team data.

Level of Distribution

This DMP is licensed under a Creative Commons Attribution 4.0 International License.

It is attributed to Biagio Licari and published at Zenodo. DOI: 10.5281/zenodo.276019



Keywords

Nba, nba-stats, stats, season-nba, nba-bet, nba-players, nba-coach, player-stats, coach-stats, nba-data, nba-season-analysis

Contents

1	Data Description and Collection or Re-Use of Existing Data	3
1.1	Data Creation and Re-Use	3
1.2	Produced Data Files.....	3
2	Documentation and Data Quality	4
2.1	FAIR Data	4
2.2	Organization, Metadata and Documentation	4
2.3	Data Quality Control.....	5
3	Storage and Backup During the Research Process	5
3.1	Data Storage and Backup	5
3.2	Data Security and Protection	5
4	Legal and Ethical Requirements, Codes of Conduct	6
4.1	Personal Data	6
5	Data Sharing and Long-Term Preservation	6
5.1	Modalities of Data Sharing, Restrictions	6
5.2	Data Preservation	6
5.3	Software Required for Using and Reproducing the Results.....	7
5.4	Persistent Identifiers	7

1 Data Description and Collection or Re-Use of Existing Data

1.1 Data Creation and Re-Use

The research data will be produced by utilization of a Python notebook specifically conceived for this purpose. This has the advantage that changes are quickly adopted without the need of a separate compilation step. Furthermore, it is a relatively lightweight, platform independent technology, which inherently provides support for all popular operating systems and machines with reasonably modern hardware. The module dependencies of the Python script have been kept to a minimum for facilitated interoperability, namely pandas, ponyORM and other open tool.

The data processing part of this project is based on 3 publicly available raw datasets:

1. <https://www.basketball-reference.com/leagues>
2. [https://www.basketball-reference.com/leagues/NBA 2022 per game.html](https://www.basketball-reference.com/leagues/NBA_2022_per_game.html)
3. [https://en.wikipedia.org/wiki/List of National Basketball Association single-game scoring leaders](https://en.wikipedia.org/wiki/List_of_National_Basketball_Association_single-game_scoring_leaders)
4. <https://www.kaggle.com/nathanlauga/nba-games>

1.2 Produced Data Files

The result of this project consists of many artifacts.

1. Cleaned and processed data related to statistics obtained from datasets, including data related to coaches and for each team. Additional statistical data was calculated and added to supplement the raw data information. The dataset is provided in an open format as is the CSV format and in a XML format, with a well-documented XSD schema.
2. Database

Since the nature of the experiment's question is very specific, there are no community guidelines or standards as to how such data is preferably formatted. In theory, it is within the realms of possibility to generate an ontological representation of the output data instead of or in addition to the CSV files and xml files.

2 Documentation and Data Quality

2.1 FAIR Data

The data and source code produced over the course of this experiment will adhere to the FAIR principles.

The artifacts will be findable because they are published to Zenodo, an open-access repository. They are equipped with a permanent and persistent identifier, a DOI¹. In addition to the Zenodo record, the data is also indexed and publicly findable in a GitHub repository³ that links to the Zenodo record via DOI. Both repository entries provide metadata information that facilitate finding the experiment – for more information on metadata, see also subsection 2.2.

All (digital) information related to the project is available in data repositories that are accessible by means of searching the World Wide Web. They are open, i.e. there are no access restrictions – neither to the source datasets, the result datasets nor the experiment’s source code.

Interoperability is ensured by the employment of standardized and open formats such as CSV, PDF and XML, i.e. no proprietary software is needed to read or modify the files produced. Furthermore, the documentation, which is also part of the contents indexed in the repositories, contains explanations as to the meaning of the data fields (“columns”) of the raw output data file. This knowledge, in combination with the well-known specification of the CSV format, allows for further processing in a (semi-)automatic way. It is also worth mentioning that Zenodo supports exporting the provided metadata in machine-actionable formats (e.g. DataCite, Dublin Core, DCAT)

The reusability of result data is ensured by the extensive documentation and additional metadata (see subsection 2.2) that accompany the data related to the experiment. This also entails information on the conditions that may apply for reusing data or code conceived for the purpose of this project.

2.2 Organization, Metadata and Documentation

The data is organized in files and folders in the file system. The structure and naming convention is explained in the documentation that will be attached to the project files. Changes are tracked automatically through the use of git. The release of versions denoting specific project milestones are designated with the help of git’s concept of *tags*.

Even though EML specializes in earth and environmental sciences, it is well-suited and increasingly often used to describe metadata in other disciplines as well. Information such as creator, contact details, keywords, descriptions of the produced data files (e.g. the meaning of fields in the CSV file) will be serialized into an XML file and will form part of the

documentation attached to the other project-related files. For facilitated creation of the metadata file, the tool ezEML² will be used. Standard compliance of the produced XML file will be verified using the online version of the official *EML Validity Parser*³.

2.3 Data Quality Control

Since, by nature of this project, the input and output data are static, there is no need for (re-)calibration or repeated measurements and data reviews. However, consistency and correctness checks will be performed on a sample basis, as well as code refactorings to improve maintainability.

3 Storage and Backup During the Research Process

3.1 Data Storage and Backup

Changes to any files immediately relevant to the project during the development process are tracked as a result of employing the git version control system. The files themselves are primarily hosted on GitHub. Thus, the most recent state of the project files including differences to previous versions are always accessible via GitHub. Selected, important snapshots, so-called *releases*, are also published to and archived by Zenodo.

Additionally, there are two self-maintained backup strategies. Firstly, a manual backup is performed twice a week and stored on an external drive in the possession of the project creator. Secondly, the local development environment (representing the files that are also hosted on GitHub) is mirrored, i.e. automatically synchronized, with the personal Amazon Cloud instance of the project creator.

It is acknowledged that keeping copies in the form of standalone hard drives is not an ideal backup strategy. However, considering the manageable size of the project, this approach – in combination with the other storage mediums mentioned above – achieves enough data redundancy to cover for potential data loss at one medium.

3.2 Data Security and Protection

As data is stored on multiple, completely independent, storage mediums (GitHub, Zenodo, HDD, Amazon Cloud), the likelihood of an all-encompassing incident is extremely low. Thus, temporary or permanent loss of data on one or even multiple mediums is not a major complication because, in almost any case, there exists another copy of the data somewhere else. This holds true especially since GitHub and Zenodo for themselves already have measures in place that aim to prevent data loss.

² <https://ezeml.edirepository.org/>

³ <https://knb.ecoinformatics.org/emlparser> ⁸<https://www.metadata2go.com/>

Write-access to the data, i.e. modification rights, are exclusive to the project creator. This includes account credentials and privileges to GitHub, Zenodo, Amazon as well as physical access to the development machine and backup drive. All storage mediums that are accessed over a network are only addressed via TLS, i.e. over a secure communication channel which obstructs the interception of usernames and passwords.

The project neither processes nor produces sensitive or personal data. Hence, no privacy-related actions to protect any of the data need to be taken.

As the experiment is conducted in the context of Vienna University of Technology, the institutional data protection policy⁴⁵ applies.

4 Legal and Ethical Requirements, Codes of Conduct

4.1 Personal Data

This project does not deal with personal data. As a consequence, there is no requirement for explicitly defined measures to comply with the legislation on personal data and security.

5 Data Sharing and Long-Term Preservation

5.1 Modalities of Data Sharing, Restrictions

All changes to the project – be it alterations in the input/output data or the source code – will be made public immediately in the GitHub repository mentioned in subsection 2.1. Specific milestones in the project are designated by *releases* which can be observed on GitHub, too. Additionally, for every release (i.e. substantial parts of the project have changed, not only metadata), a new DOI is generated and accessible on the general-purpose data repository Zenodo – see also the linked target DOI URL in subsection 2.1.

There is no embargo period nor any other restriction on the re-use of data: All data related to this project will be made available to everyone immediately.

5.2 Data Preservation

Due to the small file sizes of the datasets consumed and produced, every piece of input and output data, the documentation and the source code will be archived. The attached documentation contains a detailed list of the specific files that are contained in this enumeration. This enables reproducing the results in any case, but also provides the actual results of the specific project version right away. No “runtime” such as a Python virtual

⁴ [https://www.tuwien.at/index.php?eID=dms&s=4&path=Documents/Data%20Protection%](https://www.tuwien.at/index.php?eID=dms&s=4&path=Documents/Data%20Protection%20Guidelines/Data%20Protection%20Policy.pdf)

⁵ [Guidelines/Data%20Protection%20Policy.pdf](https://www.tuwien.at/index.php?eID=dms&s=4&path=Documents/Data%20Protection%20Guidelines/Data%20Protection%20Policy.pdf)

environment will be archived since it is not strictly related to the experiment data. For this reason, there are instructions for creating such an environment in the documentation.

GitHub does not explicitly state the preservation time of their code repositories, but claims that it “(...) *intends to keep your public repositories available unless you remove them*”⁶. Zenodo is more specific and promises a retention period of its own lifetime, which amounts to at least the next 20 years

The target audience of the data produced might be researchers with a focus on recent developments in the nba analysis industry. It is also possible that the results are relevant for people interested in betting.

5.3 Software Required for Using and Reproducing the Results

In order to use the results of the experiment, no special software is needed. The data can be accessed from one of the repositories – GitHub, Zenodo – with an ordinary web browser over the internet. The data themselves can be used on every major platforms without any special or proprietary software. The chosen formats CSV for the raw and processed data and PDF for all the thing are universally supported with software that is almost always part of the operating system.

As for the reproduction of the observed results, there are mainly two options. The first one was already hinted at in subsection 5.2 above. Since the experiment is implemented as a Python script, it can be executed using a local Python interpreter – detailed instructions are given in the documentation.

5.4 Persistent Identifiers

A DOI is assigned to the source code using the Zenodo and GitHub integration (see subsection 2.1). Additionally, a DOI is assigned to the data produced in the experiment.

Table 1 provides a summary of the persistent identifiers that have been reserved for the project and the location of the source code repository on GitHub. The DOI badges always resolve to the latest version available.

Data Object	DOI or Location
Source Code Repository	https://github.com/biagio7xD/Nba-Season-Stats-Management.git

⁶ <https://docs.github.com/en/github/creating-cloning-and-archiving-repositories/aboutarchiving-content-and-data-on-github>





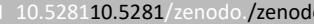
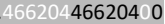




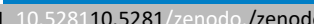
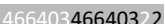
Source Code (Including Produced Data)	  
Produced Data (Standalone)	  
Data Management Plan	  
Machine-Actionable Data Management Plan	  

Table 1: DOIs and Locations of Project Components