Best Practices a Data Scientist Should Know

Hande Tuzel
Best Practices an Aspiring Data Scientist Should Know

Hande Tuzel
What do you get when you cross a pirate with a data scientist?

Answer: Someone who specializes in Rrrr

https://datasciencedojo.com/community/jokes/
Virtual Environments
Virtual Environments

• Why? Eliminates the chances of experiencing global installation and/or package collision errors.

• Your main python package directory does not get huge with unnecessary python packages.

• Makes it easier to recreate the environments. Ability to reproduce.

• How: Using venv, virtualenv, Anaconda / conda
Virtual Environments

With Anaconda/conda installed

- On the terminal:  
  $ conda create -n <env-name> python

  or with a specific python version
  $ conda create -n <env-name> python=3.6

- Activate the environment with
  $ conda activate <env-name>

- Deactivate with
  $ conda deactivate  inside the environment.

- List the environments installed:
  $ conda env list
Virtual Environments

Create environment using Anaconda:
Virtual Environments

• Create yaml file using Anaconda:

- On the terminal, inside the environment:

  `$ conda env export --file <env-name>.yml`
### Create an environment from a yaml file

```
$ conda env create -n <env-name> -f </path/to/environment.yml>
```
Add Virtual Environment to Jupyter Lab Kernel

With Anaconda/conda installed

• First install ipykernel package in Anaconda (preferred)

  or with  
  
  $ pip install --user ipykernel

• Then on the terminal, in the environment:

  $ python -m ipykernel install --user --name <env-name>

• See what kernels are installed:  
  
  $ jupyter kernelspec list

• You can also uninstall kernel with  
  
  $ jupyter kernelspec uninstall <env-name>
Work on environment in Jupyter Lab

You can now start a notebook or a console with your environment from your base environment.

No need to start Jupyter Lab inside the environment.
Work on environment in Jupyter Lab

Alternatively, you can start a notebook or a console with your environment from the File Menu.
Work on environment in Jupyter Lab
Work on environment in Jupyter Lab

Choose the environment you want to work with
Work on environment in Jupyter Lab
Try conda install/conda-forge to install Packages First
Try conda install/ conda-forge First

• For consistency, try to use `conda install / conda-forge` on the terminal or Anaconda -> Environments to install packages to your environment before trying `pip install`.

• Although, now you can see the packages you installed with pip under Anaconda Environments it is always a good practice to first try conda install.
Utilize Functions
Utilize Functions

• If you are copy and pasting part of your code more than twice it is time to write a function for that piece of code.

```python
def numsum(a, b):
    return a + b
```

• Using functions makes the overall code more readable and organized.
Utilize Functions: Custom Package

• You can create your own python packages where your functions and / or classes reside.

• This helps keep your code more organized as you can create different files for different purposes.

• For example, you can create one `.py` file for data preprocessing / reading. Another for utility functions etc.
Utilize Functions: Custom Package

- Create your package folder in the same folder where your scripts or Jupyter notebook files reside.
- Create a `__init__.py` file.
- Then write your functions in your package to be imported to your code.
- This way, you can use these packages for your other projects and have ready scripts/ functions to apply with small modifications.
Utilize Functions: Custom Package

• Create your package folder in the same folder where your scripts or jupyter notebook files reside.

• Create a `__init__.py` file.

• Then write your functions in your package to be imported into your code.

• This way, you can use these packages for your other projects and have ready scripts/ functions to apply with small modifications.
Utilize Functions: Custom Package

• Create your package folder in the same folder where your scripts or jupyter notebook files reside.

• Create a `__init__.py` file.

• Then write your functions in your package to be imported into your code.

• This way, you can use these packages for your other projects and have ready scripts to apply with small modifications.
Utilize Functions: Custom Package

• Create your package folder in the same folder where your scripts or jupyter notebook files reside.

• Create a __init__.py file.

• Then write your functions in your package to be imported to your code.

• This way, you can use these packages for your other projects and have ready scripts/functions to apply with small modifications.
Code Documentation
It matters!

First and foremost document your code for yourself!

It will help a coworker understand what your code does and how it works.

For posterity’s sake.
def numsum(a, b):
    """
    Sums a and b and returns the result.
    """

    Parameters:
    __________
    a (float): A number
    b (float): A number

    Returns:
    __________
    a + b
    """
    return a + b
Utilize Function Annotations

```python
def sum(a, b):
    '''
    Sums a and b and returns the result.
    Parameters:
    a (float): A number
    b (float): A number
    Returns:
    a + b
    '''
    return a + b
```

```python
def sum(a: float, b: float) -> float:
    '''
    Sums a and b and returns the result.
    Parameters:
    a : A number
    b : A number
    Returns:
    a + b
    '''
    return a + b
```
All together

```python
[1]: from utils.func import numsum

[2]: numsum(73.5, 6038.08)

[2]: 6111.58

[3]: help(numsum)

Help on function numsum in module utils.func:

numsum(a: float, b: float) -> float
    Sums a and b and returns the result.

Parameters:
-------------------------------
a : A number
b : A number

Returns:
--------
a + b
```
Follow Pep8 Rules
Follow PEP 8 Rules

• It is a good practice to code following PEP 8 rules.

• Lint your code. Linting is an automated way of checking your source code for programmatic and stylistic errors.

• Some linters: Pylint, Flake8, pystylecode

• Check add-on packages you can install to your code editor such as Atom, Visual Studio Code, PyCharm
Version Control
Version Control

For the Aspiring Data Scientist:

• Start using Github/Git for your projects. ([https://lab.github.com](https://lab.github.com))

• Check out this step by step guide on how to quickly start up.

• Do not do one common commit for all your files (Beats the purpose).

• Commit often.

• Use .gitignore to get rid of hidden files.
Version Control - Requirements.txt

• You created your repo and worked on your project.
• Is your work reproducible?
• Two years from now will you be able to run this code?
• You need to document package versions used in your project.
• Usually done by creating a requirements.txt in the main folder of your repo.
• Then one can recreate the correct environment by

$ pip install -r requirements.txt
• You can find the versions of packages in your environment by
  
  ```
  $ pip freeze or $ pip list --format=freeze
  ```
  . These will show the versions of all packages installed in your environment.

• **Better solution:** pipreqs package.

• pipreqs will create a requirements.txt file for you in the location you provided.

• You can install pipreqs with:
  
  ```
  $ conda install -c conda-forge pipreqs
  ```

• Then do
  
  ```
  $ pipreqs project_folder
  ```
  to create a requirements.txt file under project_folder you provided.
Avoid Data Leakage
Avoid Data Leakage

Data leakage can happen in two possible scenarios:

✶ Your training data has some information about your test set. (Ex: Normalizing the data before splitting into train and test)
✶ Your model was trained on features that are not available during production yet.
Avoid Data Leakage

Possible Remedies:

• Use pipelines if using scikit-learn.

• Know your data and what is available during production.

• Do not create features that would not be available during production.

• Do no use features that are available in the database but not during production.

• Some reads on topic:

  https://www.kaggle.com/alexisbcook/data-leakage,
Know Your Metrics
Know Your Metrics

Measuring the performance of your model is important.

- But is the metric you are using to measure the performance of your model the right one?
- For example:
  - Accuracy for binary classification for imbalanced data will not tell you much.
  - If ROI is more important then precision might be more important than recall.
Job Interviews
Some Tips and Advice
Practice Practice Practice Practice!
Practice Practice Practice!

• kaggle.com for ML related projects. There are many other resources for publicly available data to practice on as well.

• Practice on your data visualization skills. Important component of Exploratory Data Analysis (EDA).

• Some examples of places where you can practice coding:
  • https://www.hackerrank.com/,
  • https://www.stratascratch.com/,
  • https://leetcode.com
Practice Practice Practice!

Additionally

• Learn / brush up on statistics!

• Read blogs (but beware of the quality of content).
  - Some examples: medium.com, towardsdatascience.com, kdnuggets.com, company blogs etc.

• Join Meetups. (Also create a Linkedin profile)

• Youtube (supplemental learning / company X interview/data science interview)
  - some channels: ritvikmath / zedstatistics / StatQuest with Josh Starmer
Showcase Your Projects
Showcase Your Projects

- Work on projects/topics that you are most interested in. (Not just basic datasets/very well known projects.)
- Know your work. The why, the how.
- You should be able to explain the problem you are solving!
- A good EDA should be the starting point of your projects.
- Create a repository for the projects to showcase on your resume.
Start Learning SQL
Start Learning SQL

• Most of the time, you will need to gather your own data for your own projects from databases.

• To do that you will need to know SQL.

• Learn about window functions such as RANK, LEAD, LAG, MAX, MIN etc.

• Some examples of places where you can practice:
  • https://www.hackerrank.com/,
  • https://www.stratascratch.com/
Sabre is a software and technology company that powers the global travel industry. With decades of revolutionary firsts, our team of experts drive innovation and ingenuity in the industry.
LEADING TECHNOLOGY PROVIDER TO THE TRAVEL INDUSTRY AND CUSTOMER SET

Retailing, distribution and fulfillment for airlines, hoteliers and agencies

$4B
TOTAL REVENUE*

$260B
GLOBAL TRAVEL SPEND PROCESSED ANNUALLY*

$1B
INVESTED IN TECHNOLOGY ANNUALLY*

$3.7B
TRAVEL SOLUTIONS REVENUE*

$293M
HOSPITALITY SOLUTIONS REVENUE*

*Pre-COVID-19, as of Dec. 2019
Travel Solutions

Sabre Travel Solutions provides support across the entire travel ecosystem

<table>
<thead>
<tr>
<th>TMCs, OTAs, Agencies</th>
<th>Corporations</th>
<th>Hotel rooms</th>
<th>Flight operations</th>
<th>Airline sales</th>
<th>Much more</th>
</tr>
</thead>
<tbody>
<tr>
<td>400+ Airlines</td>
<td>71K+ Travel Agencies</td>
<td>1M+ Property options</td>
<td>355 Tour operators</td>
<td>40 Car rental brands</td>
<td>90+ Rail carriers</td>
</tr>
</tbody>
</table>
Hospitality Solutions

Sabre Hospitality Solutions is the leading technology provider for the hospitality industry, serving hotels, resorts and chains spanning nearly 200 countries.

42,000+ Properties
$25B Total Room Sales
40% Of the world’s leading hotel brands
Provide fast, accurate expert decision making for complex business problems that consist of a large number of variables.

Solve challenging business problems for complex data structures via sophisticated algorithms (e.g. delivered efficient airline shopping search results).

Implement state of the art algorithms, pilots and proofs of concepts (POCs) with Product Development and key airline, hotelier and agency customers.

Create and pilot developer and platform tools that are being adopted across the company by leading product developers.
Positions

• Check https://www.sabre.com/careers/ and search for Data Science Engineer positions.