EESTech Challenge 2017-2018: Big Data Analysis

Local Round Patras

Welcome to the Patras Local Round of EESTech Challenge (https://eestechchallenge.eestec.net/) 2017-2018, organized by EESTEC LC Patras (https://www.facebook.com/eestecpatras/).

Through an exciting journey in our **solar system** and some of its neighbors in our **galaxy**, you will be able to demonstrate your skills in **Big Data Analysis**!

House Rules

Even though we are not aiming to limit your creativity, for uniformity in your responses we suggest that you use:

- Spark 2.3, using Python 3 language; **if you would like to use R, Scala, or even Java, please let us know!**
  - Spark libraries: SQL, DataFrames, DataSets, MLlib, GraphX
- Python 3.6
  - PyData libraries: NumPy, Pandas, SciPy, SciKit-Learn
- Visualization: display, PixieDust, Matplotlib, Seaborn, ggplot, d3.js, Bokeh, Holoviews, Plotly, Datashader

Given that, this is a Big Data Analysis challenge: Something that will be very seriously taken into consideration, in the evaluation of your responses, is most of the processing to take place in Spark. Fallback to Python only for reporting/visualizing results!

Hackathon Environments

This exercise has been provided to you as an iPython Notebook (.ipynb), the defacto standard notebook format for data engineers and data scientists working in Python. Before we start, here is some information on a few different environments you can use:

A. **Local Anaconda + Spark**

1. Install Java SE 8 (https://java.com/en/download/) or OpenJDK 8 (http://openjdk.java.net/install/)
2. Install Apache Spark 2.3 (https://spark.apache.org/downloads.html)
3. Install Anaconda 5.1 (Python 3.6) (https://www.anaconda.com/download/)
4. Go to the Anaconda Prompt
5. Run `jupyter notebook`
6. Open the notebook file we provided you using *File* > *Open...*
7. Your work is saved to the local filesystem

   Adding packages can be done with `conda` or `pip` in the Anaconda Prompt:

```
conda install pyspark
conda install findspark
conda install bokeh
pip install pixiedust
```

B. **Google Colaboratory:** https://colab.research.google.com/ (https://colab.research.google.com/)

1. Log in with your Google account
2. Review the Welcome to Colaboratory! (https://colab.research.google.com/notebooks/welcome.ipynb) notebook, especially the links under *For more information*
3. Open the notebook file we provided you using *File* > *Upload notebook...*
4. Set it to use Python 3 via *Runtime* > *Change runtime type*
5. Your work is saved in Google Drive

   Move the following into a code block and run it to initialize your environment:

```
!apt-get install openjdk-8-jdk-headless -qq > /dev/null
!wget -q http://apache.forthnet.gr/spark/spark-2.3.0/spark-2.3.0-bin-hadoop
2.7.tgz
!tar xf spark-2.3.0-bin-hadoop2.7.tgz
!pip install -q findspark
!pip install -q bokeh
```

C. **Databricks Community Edition:** https://community.cloud.databricks.com/ (https://community.cloud.databricks.com/)

1. Sign Up then Sign In
2. Review the Gentle Introduction to Apache Spark on Databricks (https://docs.databricks.com/spark/latest/gentle-introduction/gentle-intro.html)
3. Create new cluster (4.0 Runtime, Apache Spark 2.3, Python 3)
4. Import the notebook file we provided you to your *Workspace*
5. Your work is saved in the Databricks Workspace

Most of the capabilities are part of the Databricks environment, but you can add packages via the *Library*.

Python/Spark material and references

- Markdown syntax (https://help.github.com/articles/basic-writing-and-formatting-syntax/) (used in notebook text blocks)
- Python/PyData references
  - Python 3.6 reference (https://docs.python.org/3.6/)
  - iPython/Jypyter docs (https://ipython.org/documentation.html)
  - NumPy reference (https://docs.scipy.org/doc/numpy/reference/)
  - Pandas reference (https://pandas.pydata.org/pandas-docs/stable/)
  - SciPy reference (https://docs.scipy.org/doc/scipy/reference/)
  - SciKit-Learn docs (http://scikit-learn.org/stable/documentation.html)
  - Matplotlib reference (https://matplotlib.org/contents.html)
- KDnuggets' 30 Essential Data Science, Machine Learning & Deep Learning Cheat Sheets (https://www.kdnuggets.com/2017/09/essential-data-science-machine-learning-deep-learning-cheat-sheets.html)
- Apache Spark 2.3 documents (https://spark.apache.org/docs/latest/)
- Spark Guides by Databricks
  - A Gentle Introduction to Apache Spark (eBook) (http://go.databricks.com/gentle-intro-spark)
  - Spark: The Definitive Guide (Excerpts) (https://pages.databricks.com/definitive-guide-spark.html)
  - The Data Scientist's Guide to Apache Spark (eBook) (http://go.databricks.com/data-scientist-spark-guide)

Cells can be executed by hitting `shift+enter` while the cell is selected.

Initialize Spark environment

***Note**: This only applies to environments A and B, since Databricks automatically creates a Spark session.*

In [ ]:
```
import os
os.environ["SPARK_HOME"] = "/spark-2.3.0-bin-hadoop2.7"

import findspark
findspark.init()
```

Create a Spark session

***Note**: This only applies to environments A and B, since Databricks automatically creates a Spark session.*

In [ ]:
```
import pyspark

from pyspark.sql import SparkSession
spark = SparkSession.builder.master("local[*]").getOrCreate()
```

We should have a working Spark session at this point.

In [ ]:
```
spark
```

Let's begin our journey!

Courtesy: https://www.nasa.gov/image-feature/goddard/2018/hubble-exquisite-view-of-a-stellar-nursery (https://www.nasa.gov/image-feature/goddard/2018/hubble-exquisite-view-of-a-stellar-nursery) (NASA)

Part I: Journey in our Solar System

In the first part of the exercise, we will explore the objects of our Solar System. The relevant dataset is provided by NASA's Jet Propulsion Laboratory (JPL) **Solar System Dynamics** web site: https://ssd.jpl.nasa.gov/ (https://ssd.jpl.nasa.gov/). The web site provides information related to the orbits, physical characteristics, and discovery circumstances for most known natural bodies in orbit around our sun. Please check the top-level pages of the site to understand more about the data it offers.

I.1 Load data set

The dataset has been downloaded recently (March 25, 2018) from the NASA JPL Small-Body Database Search Engine (https://ssd.jpl.nasa.gov/sbdb_query.cgi). Please check the particular page to review the list of fields that it contains. Since we don't expect that you will have astronomers in your teams, you shouldn't try to understand each and every field!

Objectives:

1. Load the dataset from the provided CSV. Please note that it includes a header with column names.
2. Let Spark infer the data schema (i.e. the column types). Then print the schema discovered by Spark.
3. Count the number of columns (i.e. fields) and the number of rows (i.e. small objects) in the data set.

Difficulty level: Easy

In [ ]:
```
# I.1.1 Load the dataset from the provided CSV. Please note that it includes a
 header with column names.
```

In [ ]:
```
# I.1.2 Let Spark infer the data schema (i.e. the column types). Then print th
e schema discovered by Spark.
```

In [ ]:
```
# I.1.3 Count the number of columns (i.e. fields) and the number of rows (i.e.
 small objects) in the data set.
```

I.2 Explore data set

Before we perform any analysis, we need to explore the characteristics of the data points, sometimes referred to as observations, included in the dataset.

Objectives:

1. Print 100 sample data points from the small objects dataset in tabular form.
2. Calculate the summary statistics (mean, stddev, min and max) of each column.

If you need to refresh your memory on the definition of the summary statistics, please check Wikipedia: https://en.wikipedia.org/wiki/Summary_statistics (https://en.wikipedia.org/wiki/Summary_statistics).

Difficulty level: Easy-Medium

In [ ]:
```
# I.2.1 Print 100 sample data points from the small objects dataset in tabular
 form.
```

In [ ]:
```
# I.2.2 Calculate the summary statistics (mean, stddev, min and max) of each c
olumn.
```

I.3 Find points of interest

Instead of applying calculations across the entire dataset, as we have done above, we occasionally try to find specific data points, based on certain parameters.

Objective:

1. Find the top 50 asteroids based on diameter.
2. Present them in a table similar to https://en.wikipedia.org/wiki/List_of_exceptional_asteroids (https://en.wikipedia.org/wiki/List_of_exceptional_asteroids).

Please try to include as many columns as you can find that match the ones in the Wikipedia page above.

Difficultv level: Medium

In [ ]:
```
# I.3.1 Find the top 50 asteroids based on diameter.
```

In [ ]:
```
# I.3.2 Present them in a table similar to https://en.wikipedia.org/wiki/List_
of_exceptional_asteroids.
```

I.4 Explore relationships between variables

What is even more important than calculating summary statistics, or finding points of interest, is to explore relationships between the variables (features, in the data science context) of a data set.

Objectives:

1. Produce a histogram of the semi-major axis (`a`) by plotting the number of small objects per histogram interval.
2. Overlay the above with semi-major axis of the bodies (planets) of our Solar System (https://en.wikipedia.org/wiki/Solar_System), including Ceres (https://en.wikipedia.org/wiki/Dwarf_planet). Where are the most asteroids located?
3. Create a scatter plot that explores the relationship between the semi-major axis (`a`) and the orbit period (`per_y`).
4. Train a linear regression model that predicts the orbit period (`per_y`) based on the semi-major axis (`a`). How would you improve model fit?

Difficulty level: Medium-Hard

In [ ]:
```
# I.4.1 Produce a histogram of the semi-major axis (a) by plotting the number
 of small objects per histogram interval.
```

In [ ]:
```
# I.4.2 Overlay the above with semi-major axis of the bodies (planets) of our
 Solar System, including Ceres.
```

Where are the most asteroids located?

In [ ]:
```
# I.4.3 Create a scatter plot that explores the relationship between the se
mi-major axis (a) and the orbit period (per_y).
```

In [ ]:
```
# I.4.4 Train a linear regression model that predicts the orbit period (per_y)
 based on the semi-major axis (a).
```

How would you improve model fit?

I.5 Bonus question

As part of testing Falcon Heavy (http://www.spacex.com/falcon-heavy), SpaceX (http://www.spacex.com/) has launched Elon Musk's personal Tesla roadster (https://www.tesla.com/roadster) to space! According to a Twitter message (https://twitter.com/elonmusk/status/961083704230674438) by Elon Musk replicated by news articles (e.g. see Guardian (https://www.theguardian.com/science/2018/feb/07/elon-musk-space-car-overshoot-mars-asteroid-belt)), the orbit of the Tesla roadster, with its onboard passenger (the Starman), will overshoot Mars and reach the asteroid belt.

Objectives:

1. Use the JPL Horizons Web Interface (https://ssd.jpl.nasa.gov/horizons.cgi) to find the "Tesla", which is officially tracked by NASA!
2. Look for the same object in the dataset we have loaded above. Were you able to find it???
3. Find the date & distance of Starman's closest approaches to Mars & Earth until end of 2020.

Difficulty level: Easy

I.5.1 Use the JPL Horizons Web Interface (https://ssd.jpl.nasa.gov/horizons.cgi) to find the "Tesla", which is officially tracked by NASA!

Include a screenshot of the result. An opportunity to prove that you know some markdown syntax!

In [ ]:
```
# I.5.2 Look for the same object in the dataset we have loaded above. Were you
 able to find it???
```

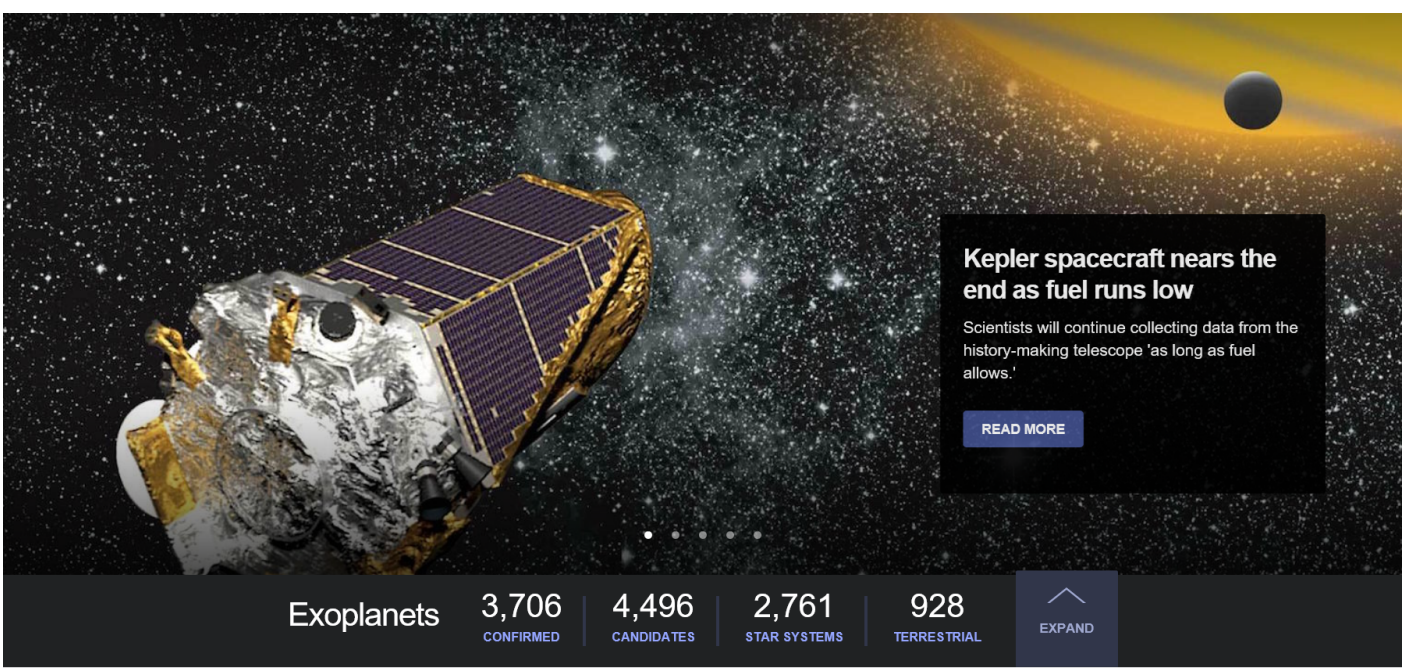
I.5.3 Find the date & distance of Starman's closest approaches to Mars & Earth until end of 2020.

Courtesy: https://apod.nasa.gov/apod/ap180210.html (https://apod.nasa.gov/apod/ap180210.html) (SpaceX)

Part II: Exploring other Solar Systems in our neighborhood

After superficially exploring our solar system, and before a human being manages to even reach Mars (http://www.spacex.com/mars), let's have a look at other solar systems in our vicinity!

NASA runs a an Exoplanet Exploration (https://exoplanets.nasa.gov/) program that analyzes other solar systems in our galaxy and tries to find ones that have planets that may be life-friendly.



Courtesy: https://exoplanets.nasa.gov/ (https://exoplanets.nasa.gov/) (NASA)

II.1 Load and explore data set

As with Part I, the data can be downloaded NASA JPL, but this time from the **Stars and Galaxies** web site: https://www.jpl.nasa.gov/stars-galaxies/ (https://www.jpl.nasa.gov/stars-galaxies/).

The actual location were the dataset can be downloaded from is the NASA **Exoplanet Archive** web site: https://exoplanetarchive.ipac.caltech.edu/ (https://exoplanetarchive.ipac.caltech.edu/).

Objectives:

1. Load the Confirmed Exoplanets (https://exoplanetarchive.ipac.caltech.edu/cgi-bin/TblView/nph-tblView?app=ExoTbls&config=planets) dataset.
2. Understand the dataset fields, as described at the Exoplanet Table Contents (https://exoplanetarchive.ipac.caltech.edu/docs/API_exoplanet_columns.html).
3. Apply what you learnt from Part I (open-ended).

Difficulty level: Easy

II.2 Plot some exoplanets

As you may noticed, the exoplanet dataset of Part II is ~80 times smaller than the small objects dataset of Part I. It can hardly be characterized as big data!

Nevertheless, it is still useful to get more insight on the nature of the data it contains, always to the extent that you don't have any astronomers in the team! It also sets the stage for II.3!

Objectives:

1. Review the pre-generated Exoplanet Plots (https://exoplanetarchive.ipac.caltech.edu/exoplanetplots/).
2. Try reproducing some of the plots above, either using the online Plotting Tool (https://exoplanetarchive.ipac.caltech.edu/cgi-bin/IcePlotter/nph-icePlotInit?mode=demo&set=confirmed), or the visualization tools you feel most comfortable with.

Difficulty level: Medium

II.3 Reproducible research

Now that I have mentioned the word "reproduce"! A concept that has recently raised in importance, particularly in ML/AI research, is the published outcomes of a scientist's research work to be reliably reproducible by a third-party, independent researcher.

Per Andrew Ng (http://www.andrewng.org/) of deeplearning.ai (https://www.deeplearning.ai/) and Jeremy Howard (http://jhoward.fastmail.fm/) of fast.ai (http://www.fast.ai/), reproducing someone else's ML/AI research is the best way to learn, while staying up-to-date with the state of the art. This is what we are going to try!

Not sure if you have heard, but Google has worked with NASA to apply deep learning to the data gathered by Kepler to discover 1-2 exoplanet systems.

- NASA's PoV: Artificial Intelligence, NASA Data Used to Discover Eighth Planet Circling Distant Star (https://www.nasa.gov/press-release/artificial-intelligence-nasa-data-used-to-discover-eighth-planet-circling-distant-star).
- Google's PoV: Earth to exoplanet: Hunting for planets with machine learning (https://www.blog.google/topics/machine-learning/hunting-planets-machine-learning/).

The research work is described extensively at the research publication:

Shallue, C. J., & Vanderburg, A. (2018). Identifying Exoplanets with Deep Learning: A Five-planet Resonant Chain around Kepler-80 and an Eighth Planet around Kepler-90. The Astronomical Journal, 155(2), 94 (http://iopscience.iop.org/article/10.3847/1538-3881/aa9e09/meta),

which can be also downloaded as a PDF (http://iopscience.iop.org/article/10.3847/1538-3881/aa9e09/pdf).

Most importantly, the researchers have also posted their code:
https://github.com/tensorflow/models/tree/master/research/astronet
(https://github.com/tensorflow/models/tree/master/research/astronet)

Objectives:

1. Review the articles/blog posts to understand what they have done.
2. Skim through the research publication (you will not have enough time to read it).
3. Try to add the required libraries to your development environment.
4. Download a subset of the Kepler data (since the entire dataset is 90 GB).
5. Finally, try to follow the instructions for training the AstroNet model.

Difficulty level: The stars is the limit!

If you reached this point, you are ambitious and courageous enough to enlist to the first manned mission to Mars, and beyond! Make us proud!

Quoting Stephen Hawking:

> Remember to look up at the stars and not down at your feet. Try to make sense of what you see and wonder about what makes the universe exist. Be curious. However difficult life may seem, there is always something you can do and succeed at. It matters that you don't just give up.