CS526 DIVA PROJECT

# MONITORING AND ANALYSIS OF PyPI DOWNLOADS

*Nitin Reddy*

*Ravi Vasani*

*Xuan Wang*

# OUTLINE

- Project Description
- Motivation
- Data
- Architecture
- Interface
- Findings
- Future Scope
- References and Resources
- Demo

# PROJECT DESCRIPTION

*Description*

*Motivation*

*Data Collection*

# DESCRIPTION

*Analyse the python package downloads on live stream and offline data*

## STREAMING DATA

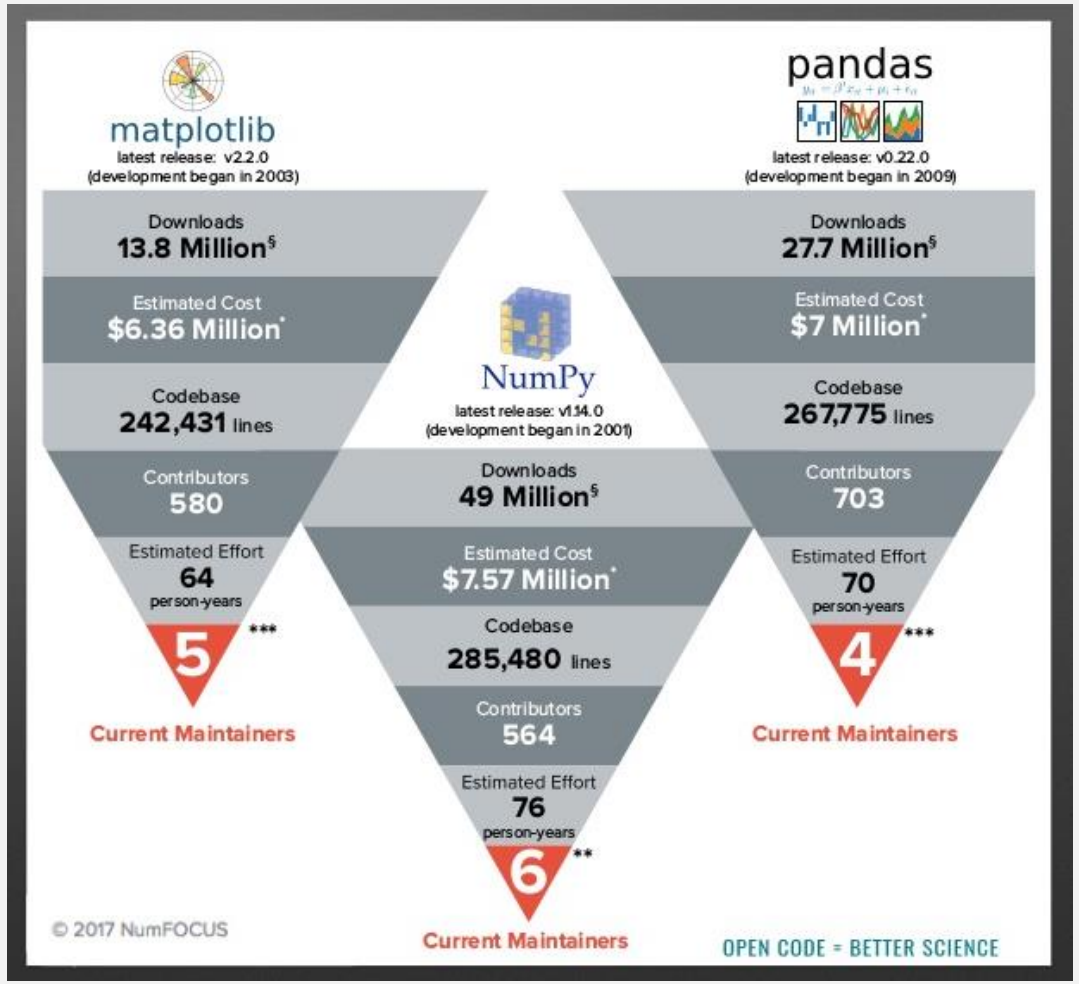
- Build an interface that would help in monitoring the python package downloads from PyPi
- Understand the packages currently being downloaded
- Monitor the rate of download of packages across the world

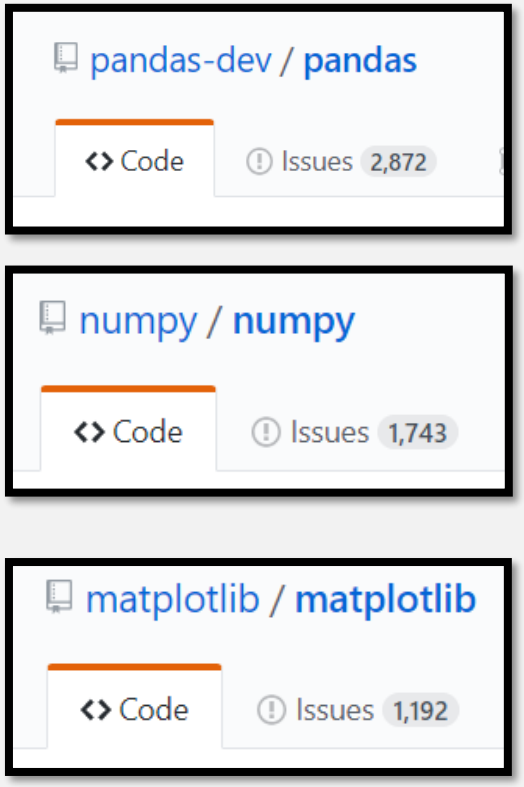
## OFFLINE DATA – BIG DATA

- Ask questions to the data
- Look out for any trends and information in the data
- Use of visualization techniques to interpret the data

# MOTIVATION

1. Not many have analyzed this data
2. There are lot of users and maintained by very few people - So, we wanted to check how does the download data look like.




Source : https://www.slideshare.net/KelleCruz/collaborations-in-the-extreme-the-rise-of-open-code-development-in-the-scientific-community

# DATA


*Data is available on Google Big Query*

- Data Generated : 15-40 GB/day on average
- Packages Downloaded (rows) : 50 – 100+ millions every day

# DATA ATTRIBUTES

| Column | Description | Values |
|---|---|---|
| TimeStamp | Time stamp of the download | 2019-02-12 23:45:23 UTC |
| File.Project | Name of the Package | numpy, pandas |
| File.Name | Package name with version | mozfile-2.0.0-py2-py3-none-any.whl |
| Version | Version of Python | 2..7, 3.6, 3.3 |
| SystemName | Version of System | Linux, Windows, Darwin |
| CPUDetails | Details of CPU | X86, x64 |
| CountryCode | Country downloaded from | US, CH, IN |
| InstallerVersion | Version of Installer | 19.0.2 |

# BIG DATA?

*Amount of data we mined for this dashboard*

Google BigQuery

>57
Terabytes

3
Years

Processed

45
Megabytes

USEFUL DATA

100
Popular Packages

# DASHBOARD

*Architecture*

*Interface*

# ARCHITECTURE

Data Generation


Data Storage

Google BigQuery

Interface

Monitor Live Data

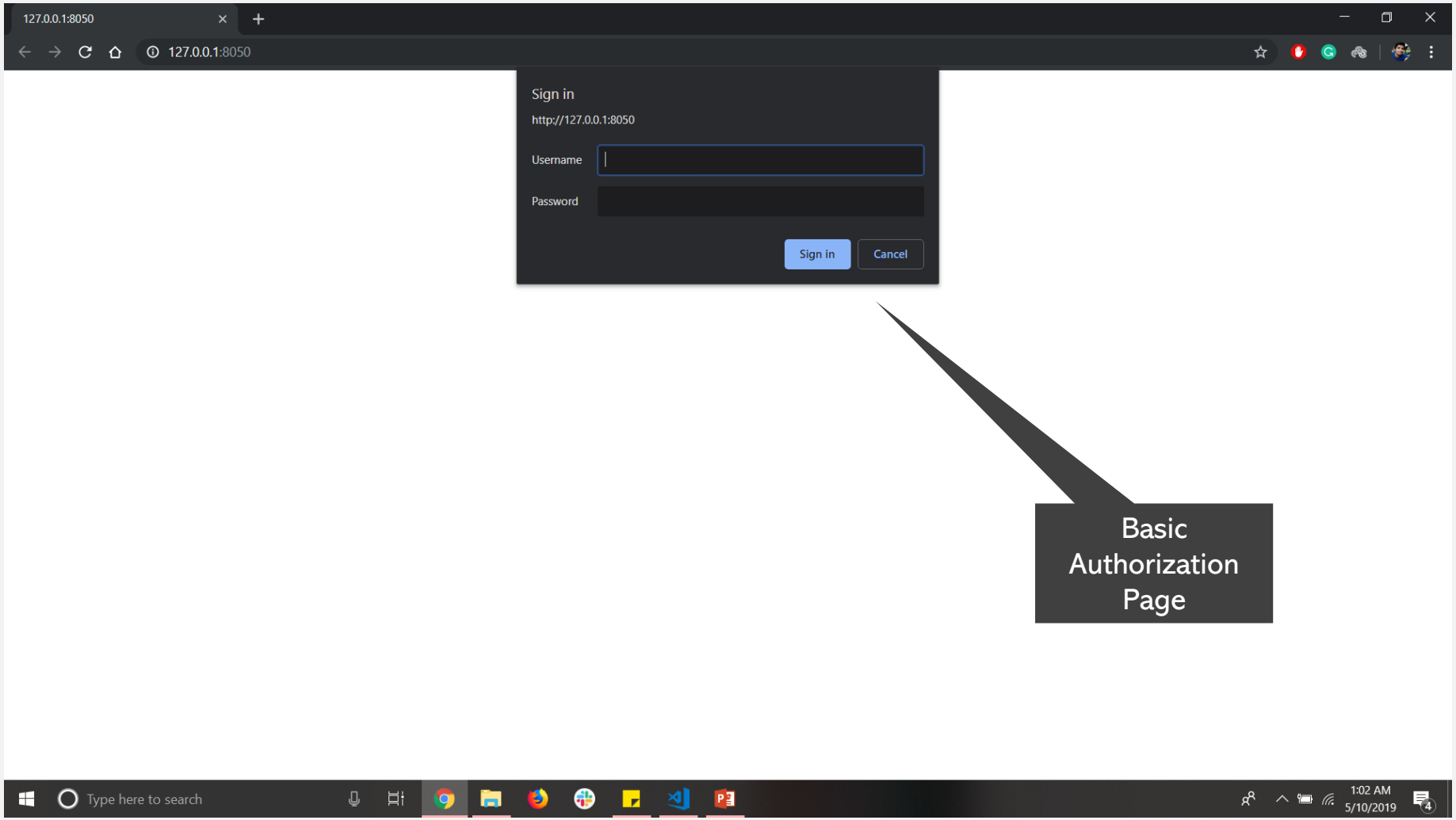
Analyzed Data

User

1) Developers
2) Maintainers
3) Industry Decision Takers

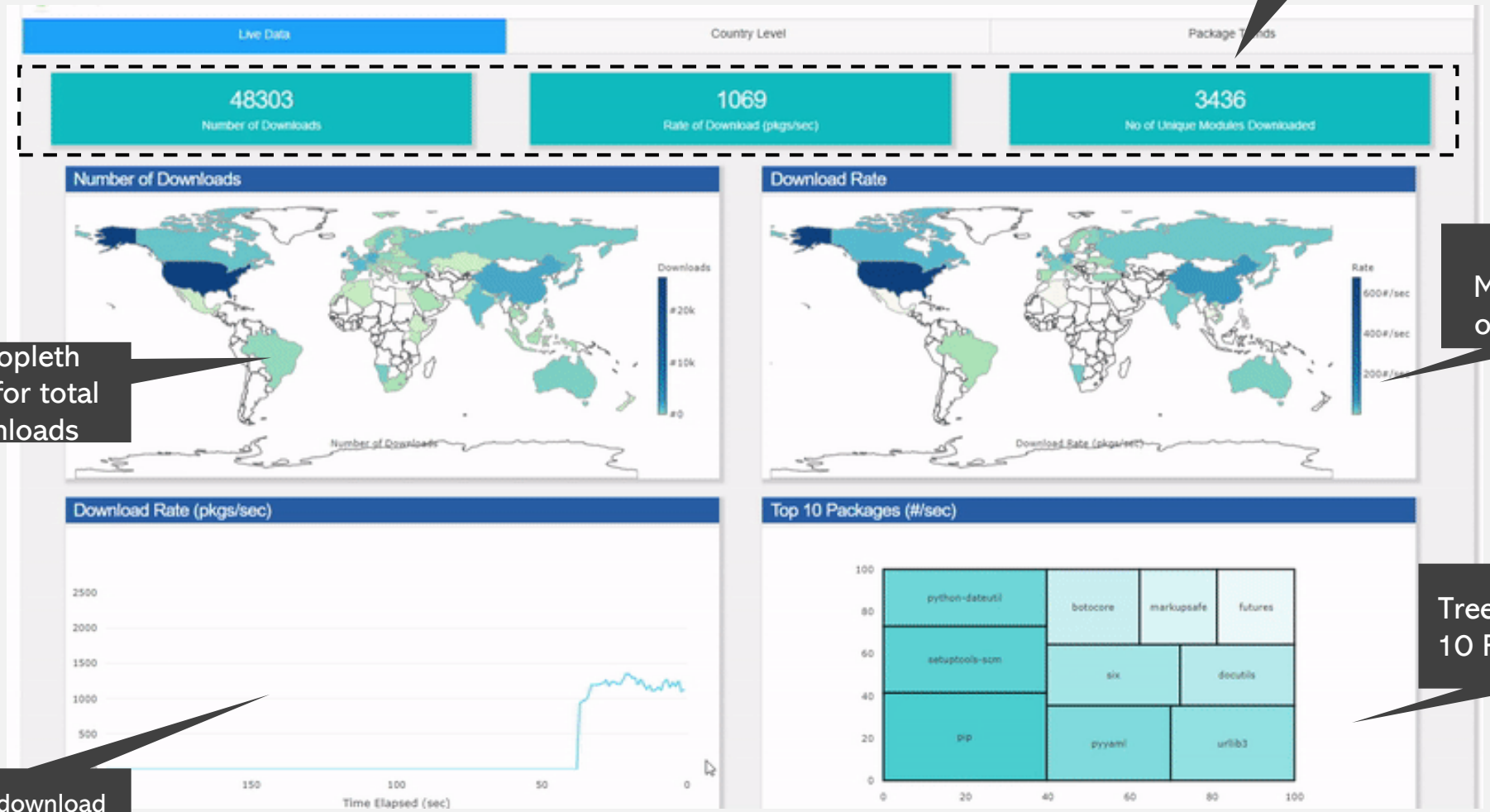
# INTERFACE – LOGIN PAGE

*Dashboard to have two pages –Live data and Historical data*



# INTERFACE – LIVE DATA

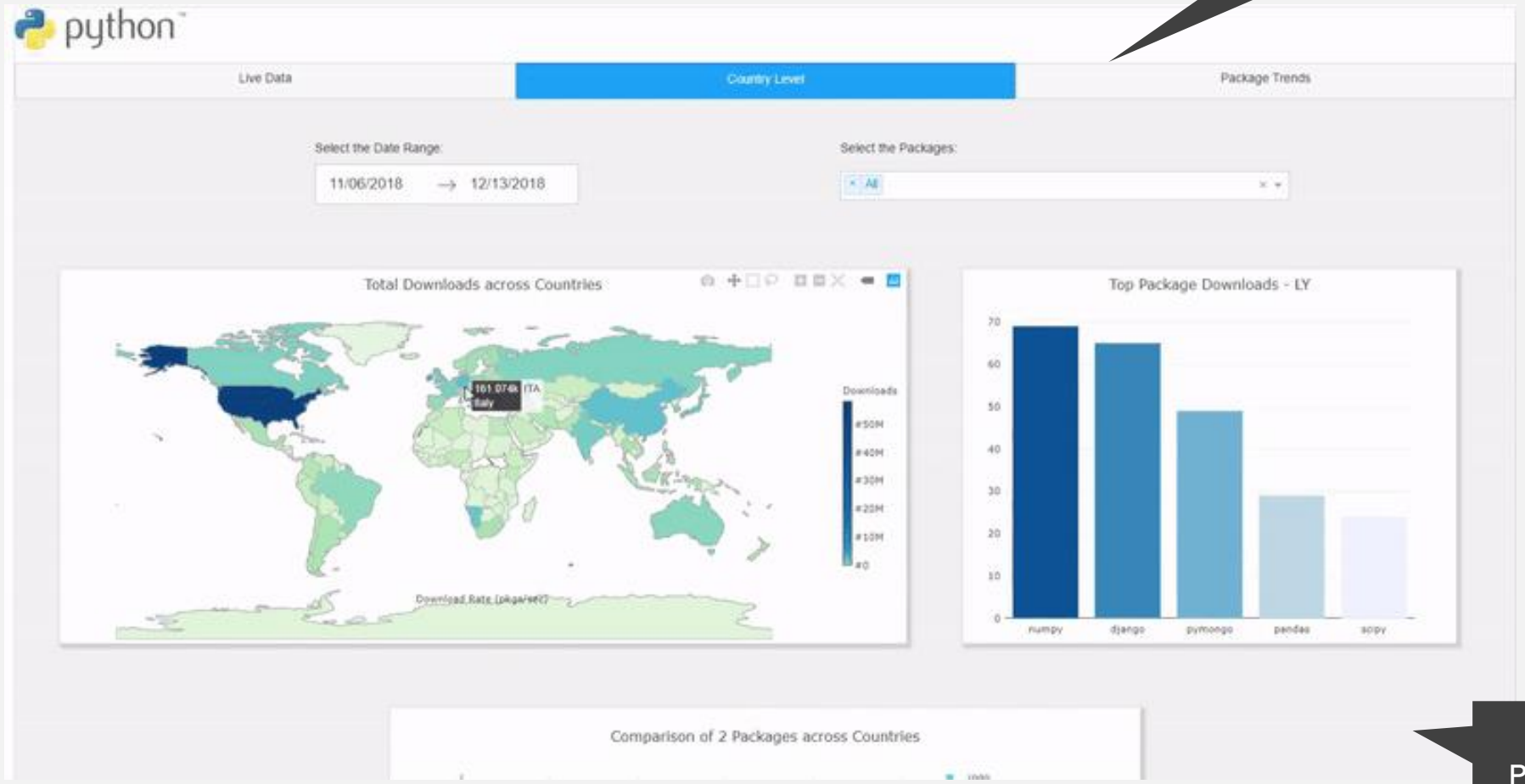
*First page of the Dashboard – Monitoring the Downloads*



# INTERFACE – HISTORICAL DATA

*Second page of Dashboard – Country level Analysis*

Inputs/ Interaction

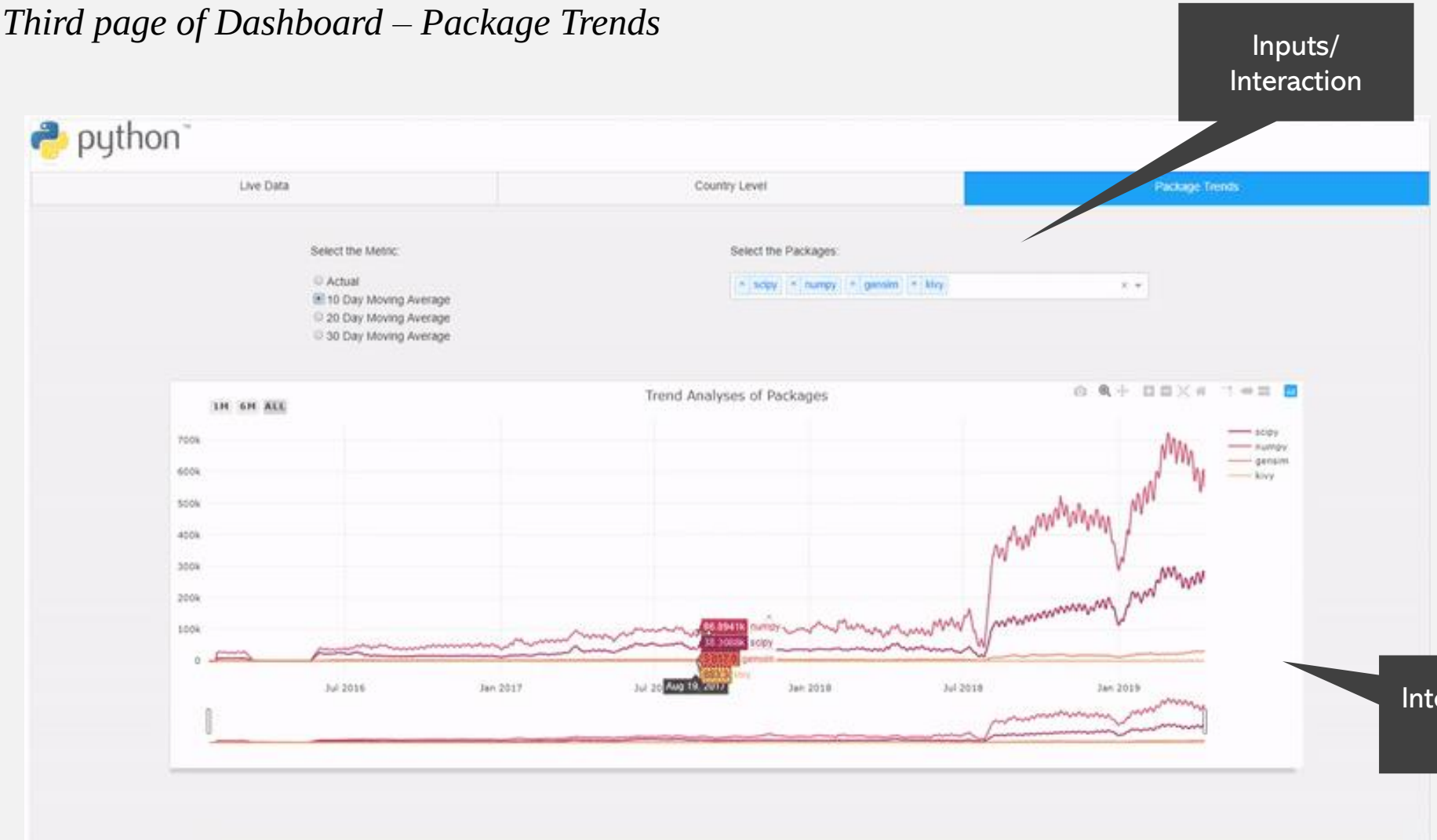



Bar Chart Showing Top Downloads in a Country

Package1 vs Package 2 across Countries

# INTERFACE – HISTORICAL DATA

*Third page of Dashboard – Package Trends*



# FINDINGS

*Questions*

*Findings from the Data*

# QUESTIONS

## SIMPLE

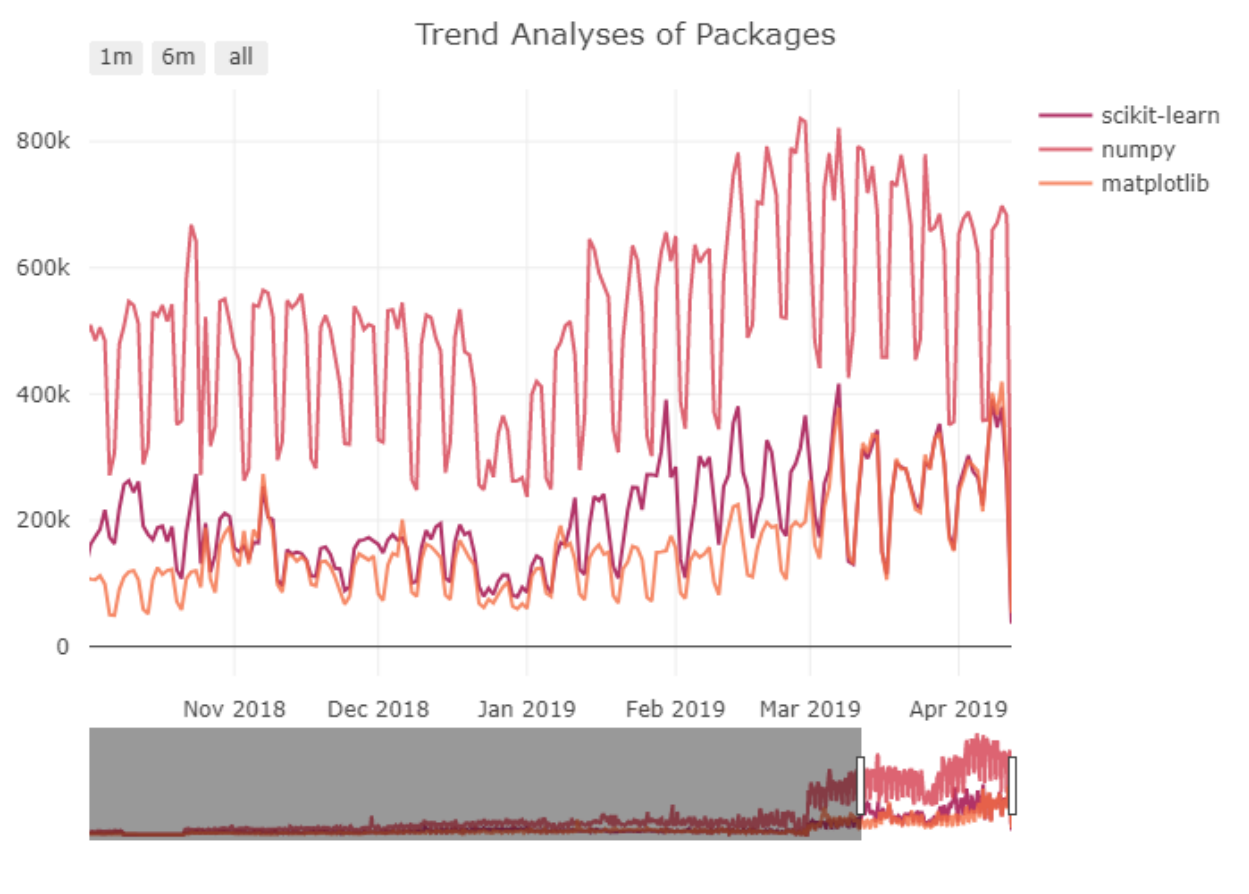
- How many packages are being downloaded in any Country X?
- What is the rate of download ?
- How many packages are being downloaded?
- What is the top downloaded package now and their distribution?
- How does the rate of download look like over time?
- For any period, any country what are the top packages downloaded?
- How does download of one package compare to other over time?

## INTERESTING

- How does each country compare to each other with respect to downloads?
- Which deep learning platform is most preferred?
- Does package download represent any trends of technology?

# FINDING 1

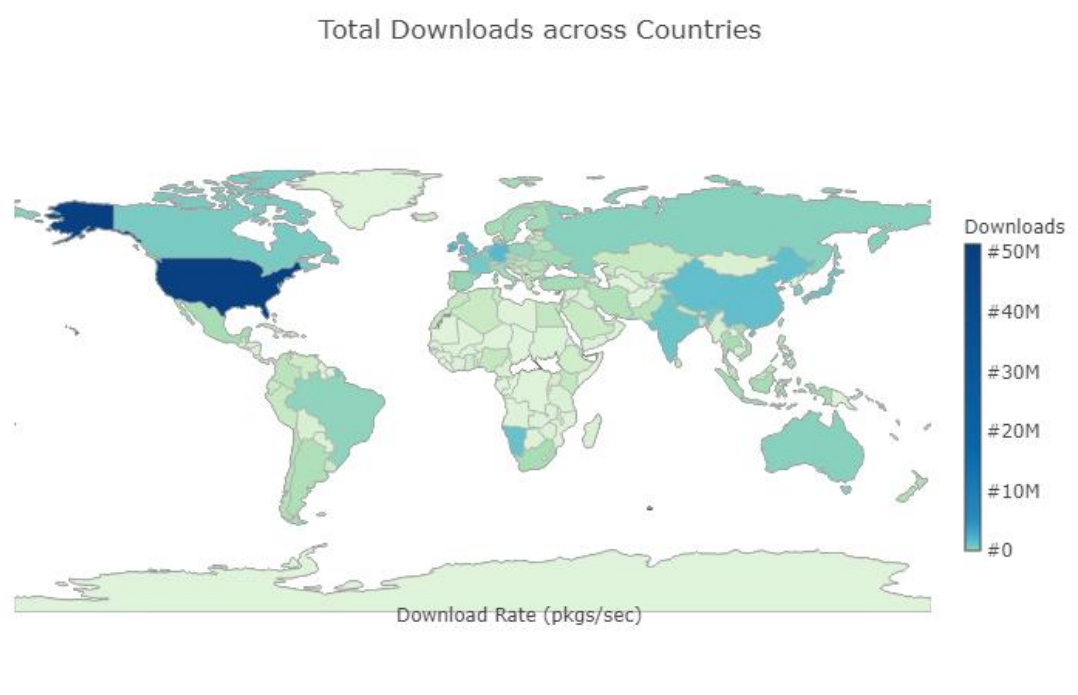
*A constant dip occurred once in week and then rises for rest of the week for all the packages.*

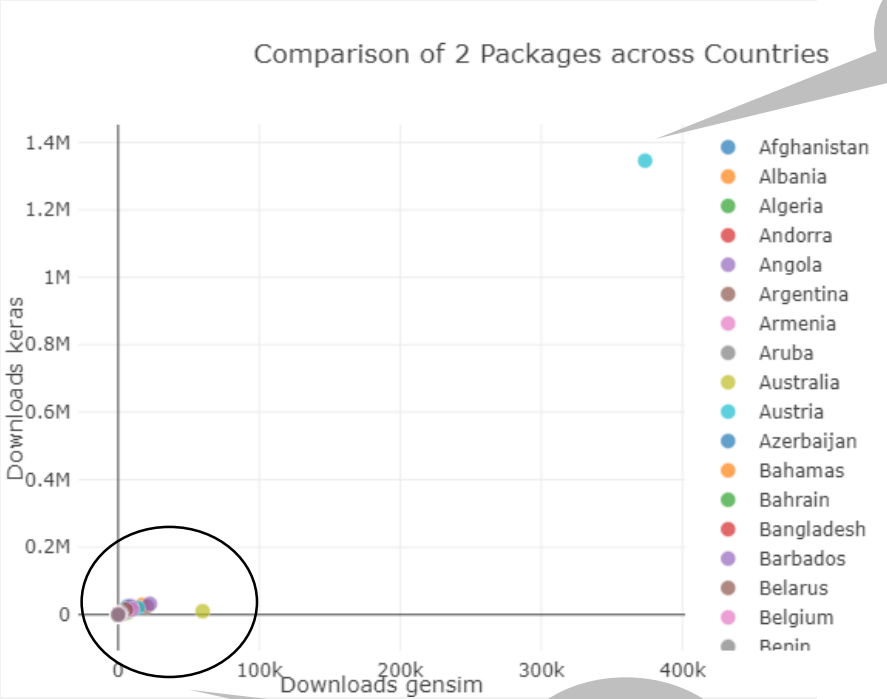



**Possible Reason –** Some of python servers could have possible downtime in week, or Application might have setup to update their packages every week.

# FINDING 2

*US undoubtedly dominated the downloads for any package.*




US

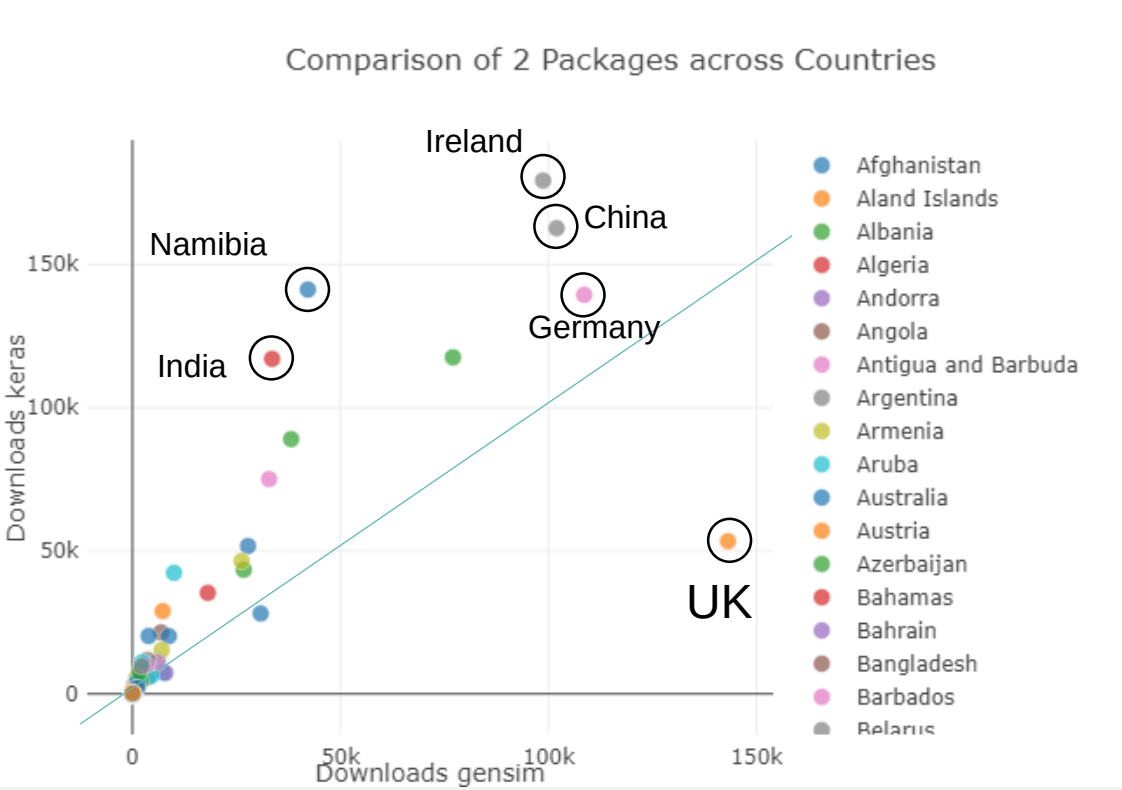
Rest of the world

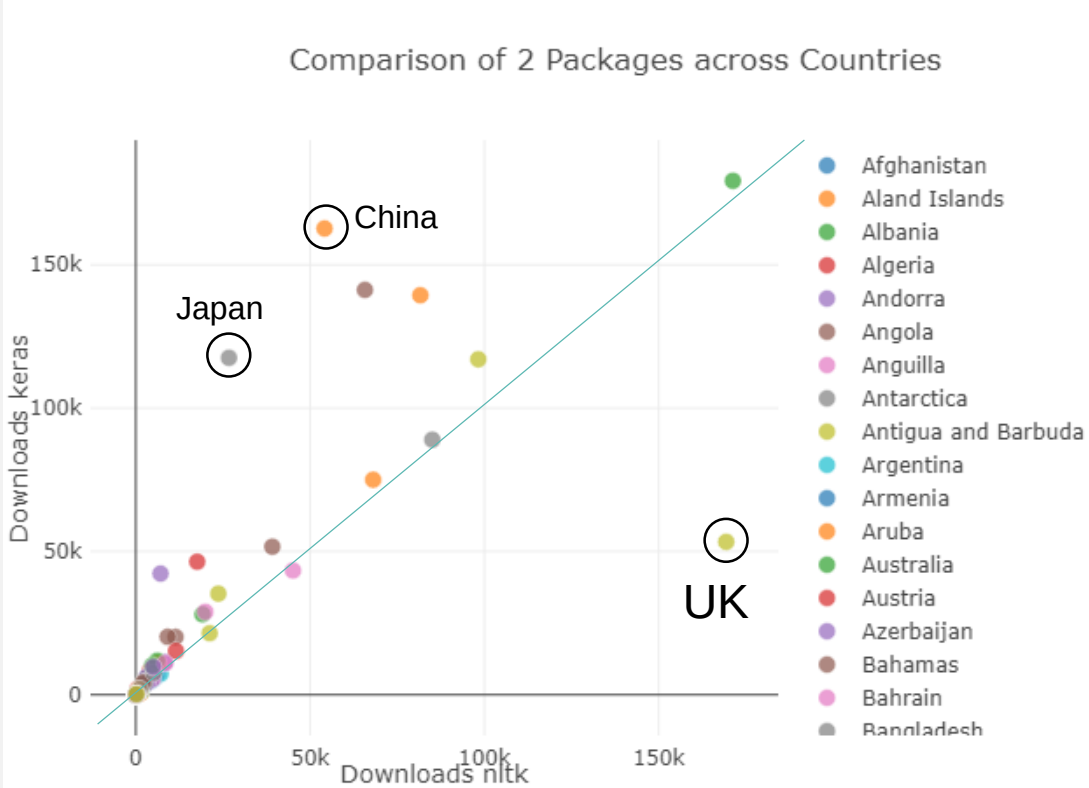
**Possible Reason –** Could be most of the application using python might have setup servers in US or when downloading show their location as US.

Dates - 01 Sep 2018 – 15 Feb 2019

# FINDING 3

*Does regions have any preferences ?*



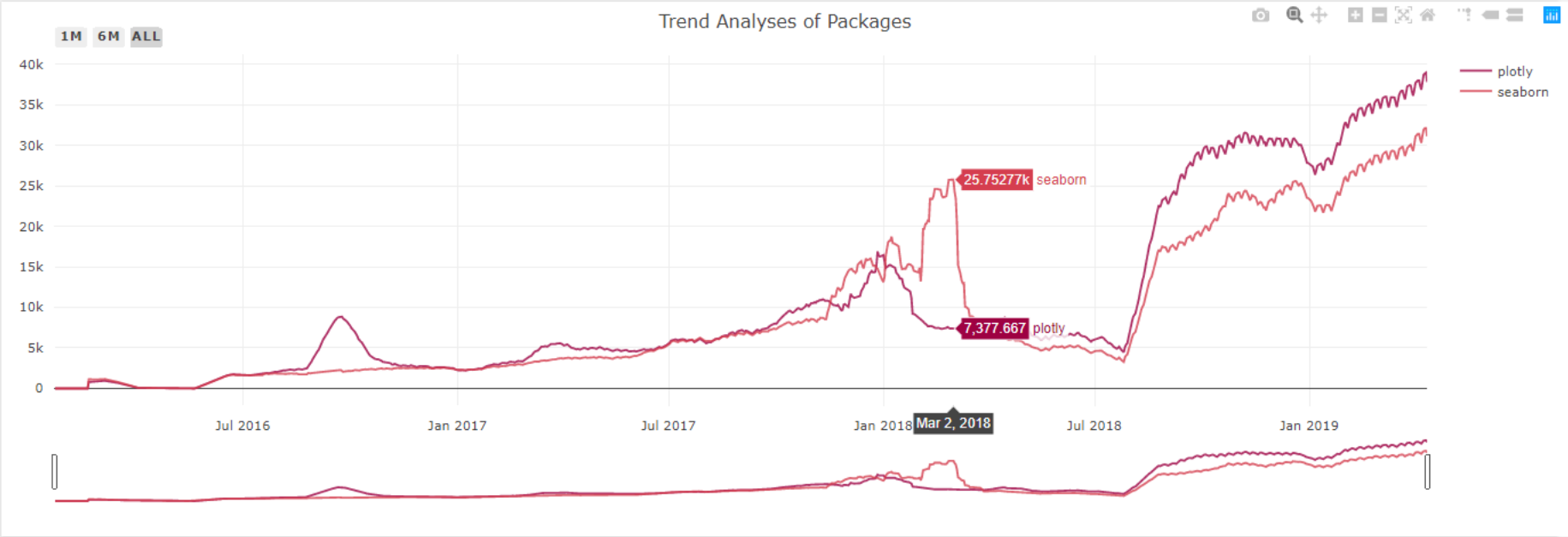




UK is more inclined towards Natural Language Processing than deep learning when compared with rest of the world.

**Possible Reason –** Since a part of EU, required to work with different languages spoken in EU

# FINDING 4

*Was Plot.ly always preferred over Seaborn?*




Seaborn was most preferred until March 2018, then plot.ly surpassed in total downloaded post that.

# CONCLUSION

*Future Scope*

*References*

*Resources*

*Demo*

# FUTURE SCOPE

- Adding Google trends along with package downloads can lead to better insights
- Integrating with Stack Over Flow data to understand demands and usage better

# REFERENCES


- DATA - *https://bigquery.cloud.google.com/table/the-psf:pypi.downloads20190212?tab=preview*
- API - *https://pypistats.org/*
- Dash - *https://dash.plot.ly/*
- Master Book - *http://www.vismaster.eu/wp-content/uploads/2010/11/VisMaster-book-lowres.pdf*
- NumFocus - *https://numfocus.org/wp-content/uploads/2018/07/NumFOCUS-Corporate-Sponsorship-Brochure.pdf*
- Data Visualization: A Successful Design Process – by Andy Kirk


# RESOURCES & TOOLS

- Python
- Dash
- Plot.ly
- Flask
- D3.js
- HTML
- Google BigQuery
- SQL
- JavaScript
- CSS


**Special thanks to Prof. James Abello, Teaching Assistant - Alireza for all the guidance and support.**

# DEMO

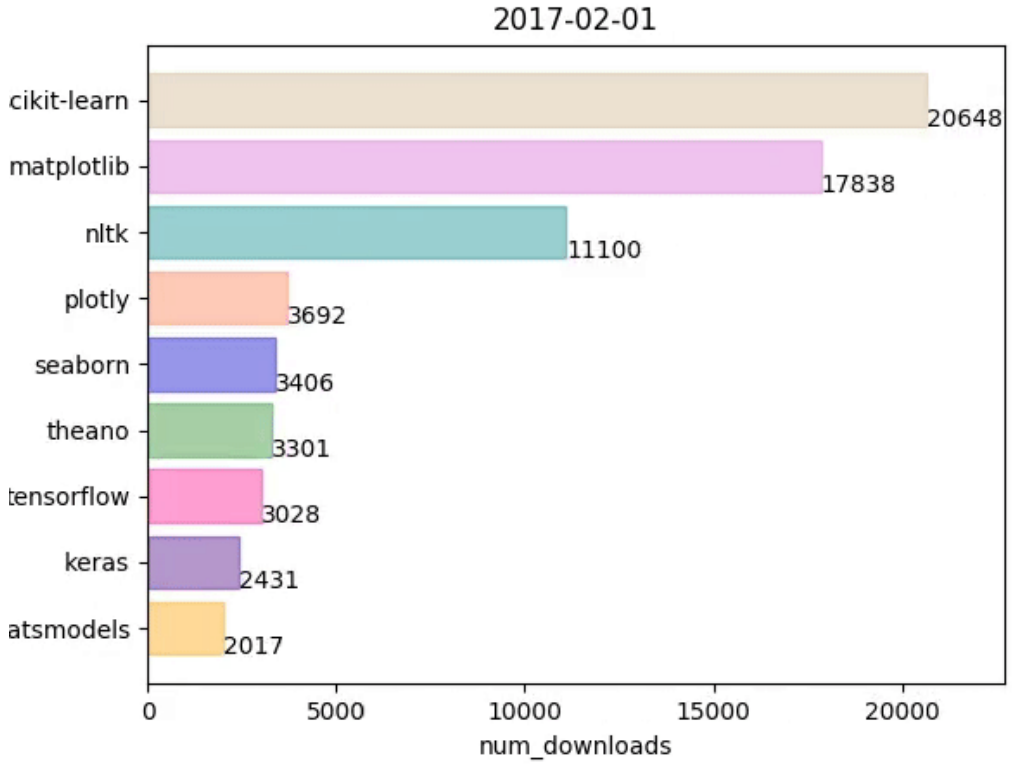
## VIDEO

https://youtu.be/AcX-obAXoqM

### ALSO APPLICATION IS LIVE AT

http://47.88.49.150:8080/

## Download Trend Over Time – JUST FOR FUN !




# THANK YOU


Nitin Reddy Karolla ✉ *nitinreddy.k@rutgers.edu*

Ravi Vasani ✉ *rkv12@scarletmail.rutgers.edu*

Xuan Wang ✉ *xw285@scarletmail.rutgers.edu*