# Data Science Case Studies with python

## IIIT Vadodara

Falak Shah
Lead Research Scientist
Infocusp Innovations Pvt. Ltd.

# About me

---

I’m passionate about application oriented research (and meditation :). I have 5 years of experience in using Machine learning/ Data Science for a wide array of applications.

I grew up in Ahmedabad, am an avid reader, and love travelling to mountains.

Questions? Raise hand/ speak up!

Your background in python/ data science?

# Background/ About Infocusp

---

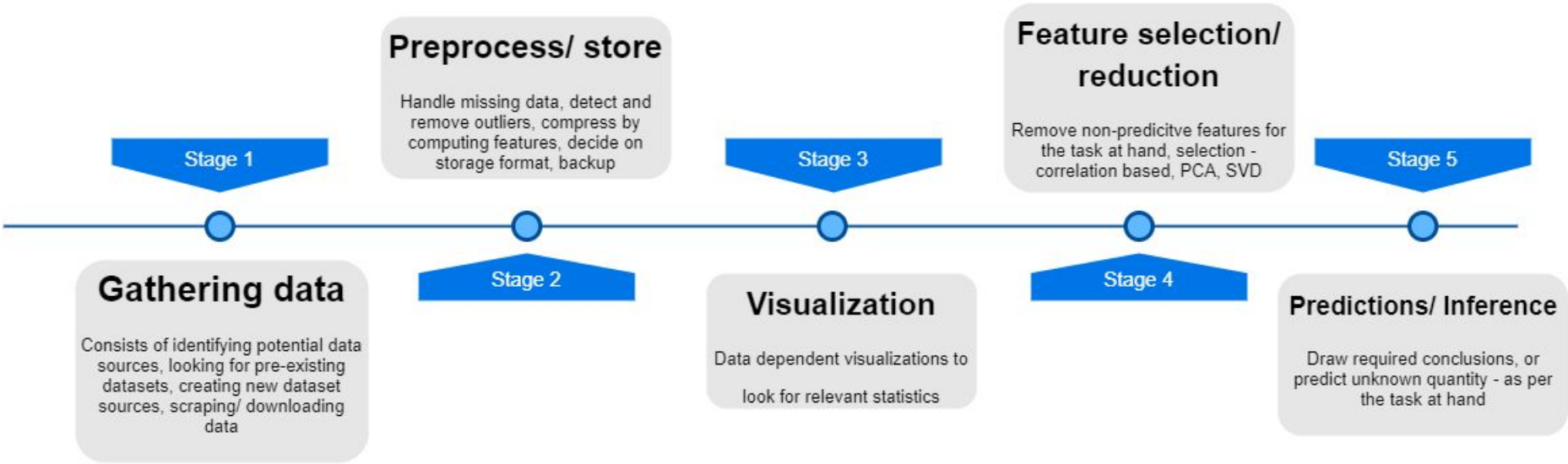
- Financial modelling
- Sifting through legal documents
- Biomedical applications - mix of DSP/ ML
- Object tracking/ detection (Computer Vision)
- Optimizing the food supply chain
- Research (most exciting!) - Music generation, Visualizing CNNs, Kaggle DS for good, LSTMs (writing like Shakespeare), optimizers and many more

# Today’s session

---

- Data Science pipeline
- Useful libraries for that in python (Pointers)
- Hands on Pandas
- Industry case studies
- Kaggle contest examples
- Hands on visualization
- Practical tips for applying DS to actual problems

# Data Science Pipeline



# Gathering/ Loading data

---

Gathering data is very much problem dependent - images/ webpages/ news/ text/ speech/ videos

Most commonly used formats: csv, Hierarchical Data Format (HDF5), sql, json, images, html, mp3, mp4

Pandas has functions to read most tabular formats directly into dataframes

```
pd.read_csv
pd.read_excel
pd.read_json
pd.read_hdf

import zipfile
archive = zipfile.ZipFile('T.zip', 'r')
df = archive.read('train.csv')

XML:
import xml.etree.ElementTree as ET
tree = ET.parse('train.xml')

HTMLs: Use beautifulsoup

Images:
Import cv2
cv2.imread(‘image.png’)
```

# Preprocess/ Cleaning the data

---

- Connect related columns from different files
- Fill NaNs
- Remove outliers
- Fix noisy samples (incorrect data type)
- Get Rid of Extra Spaces
- Fill in blank data (mean/ median/ zeros)
- Convert numbers stored as text to floats/ int
- Categorical data to numbers

**Quality data beats fancy algorithms**

# Storage

---

- Mostly cloud based
- Typical configurations of local machines: 2 TB, 64 GB, 4/8/16 core
- Can connect to different instances
- Buckets
- Databases- MongoDB, SQL
- CSVs
- Different formats for video/ images
- TFrecords- for DL

# Visualizing

---

Horde of options:

Matplotlib

Seaborn

Plotly

QT (for real time), Tensorboard

Bokeh

And many more - learn from kaggle and debug using SO

# Examples

---

Kaggle notebooks

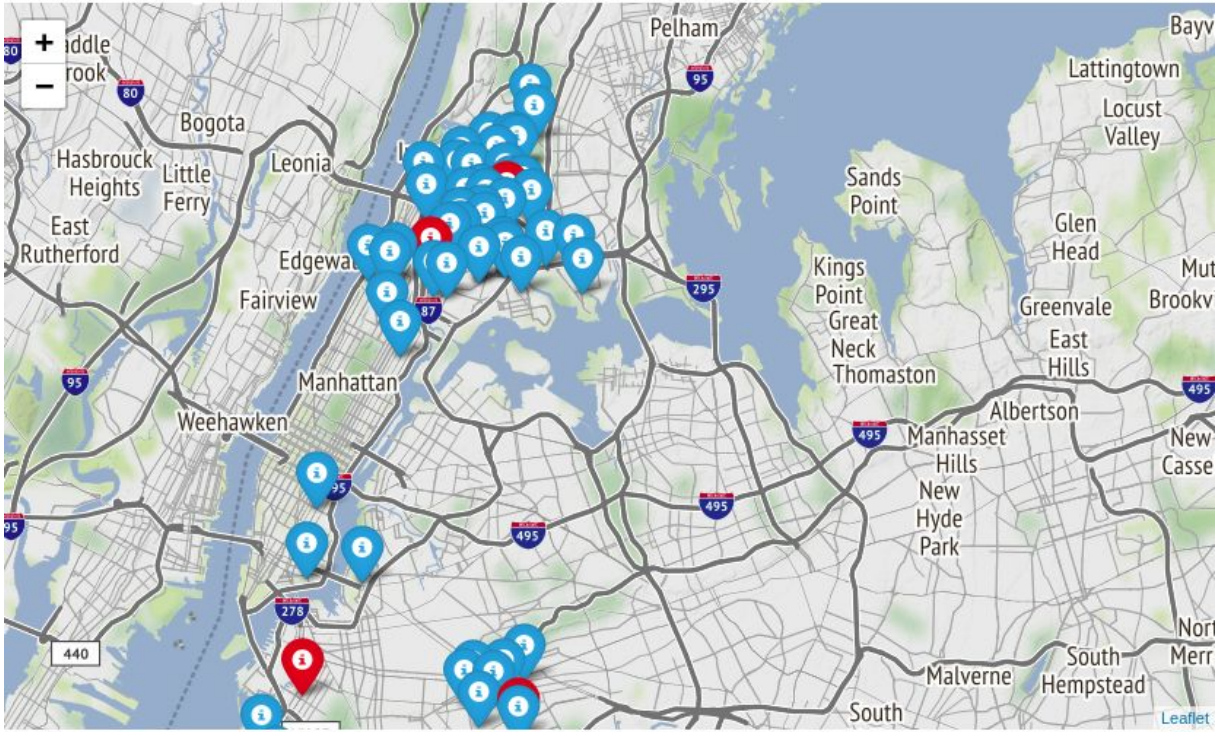
https://github.com/InFoCusp/lstm_deepdive

https://github.com/InFoCusp/tf_cnnvis (Just images)

Gradient descent based optimizers to visualize loss

TSNE: Link

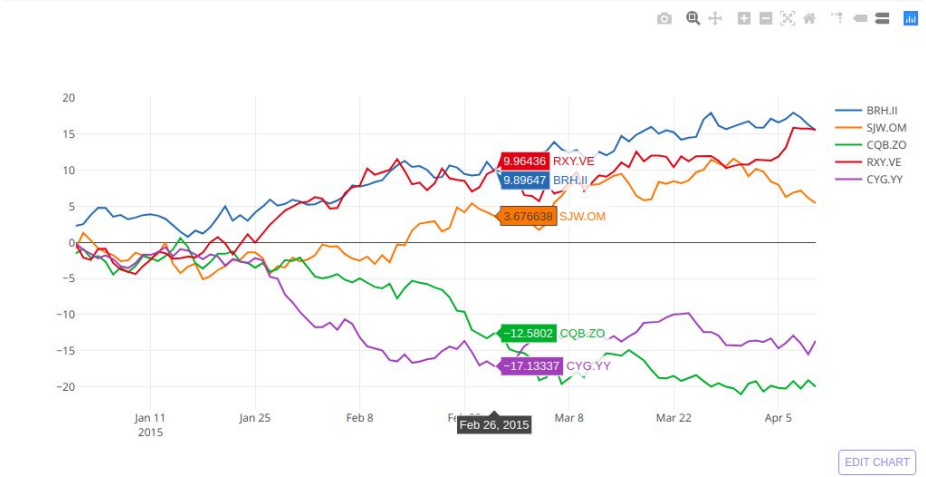
# Not just plots either

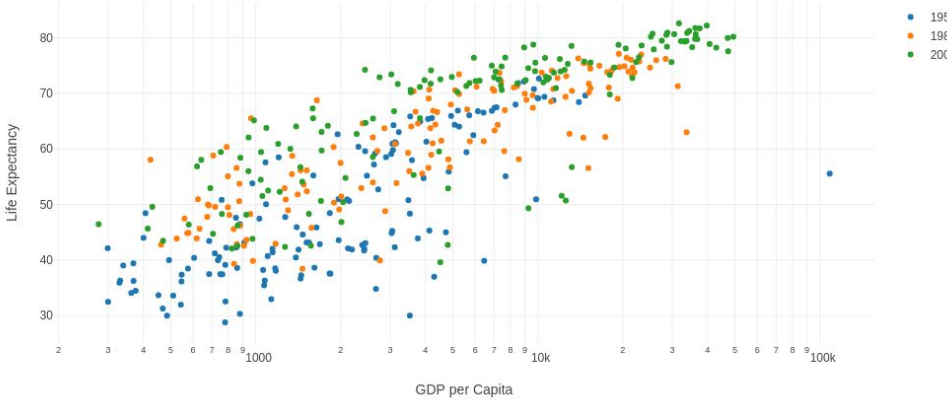


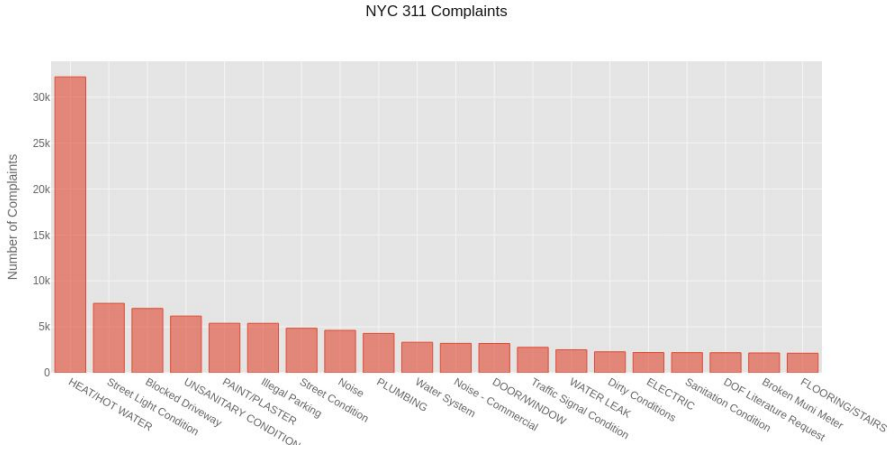

Folium Magic

Fair warning: Not easy to use, but the results look good.

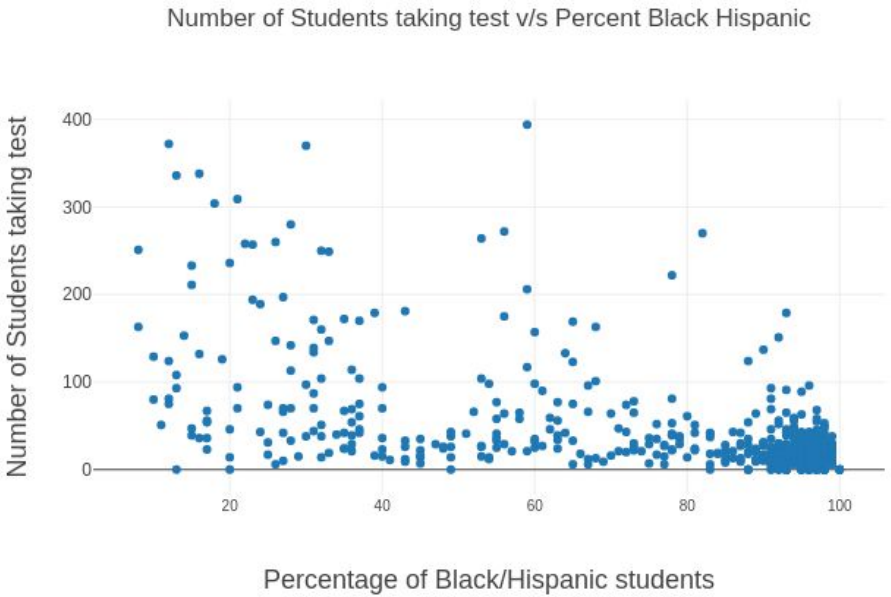
Link

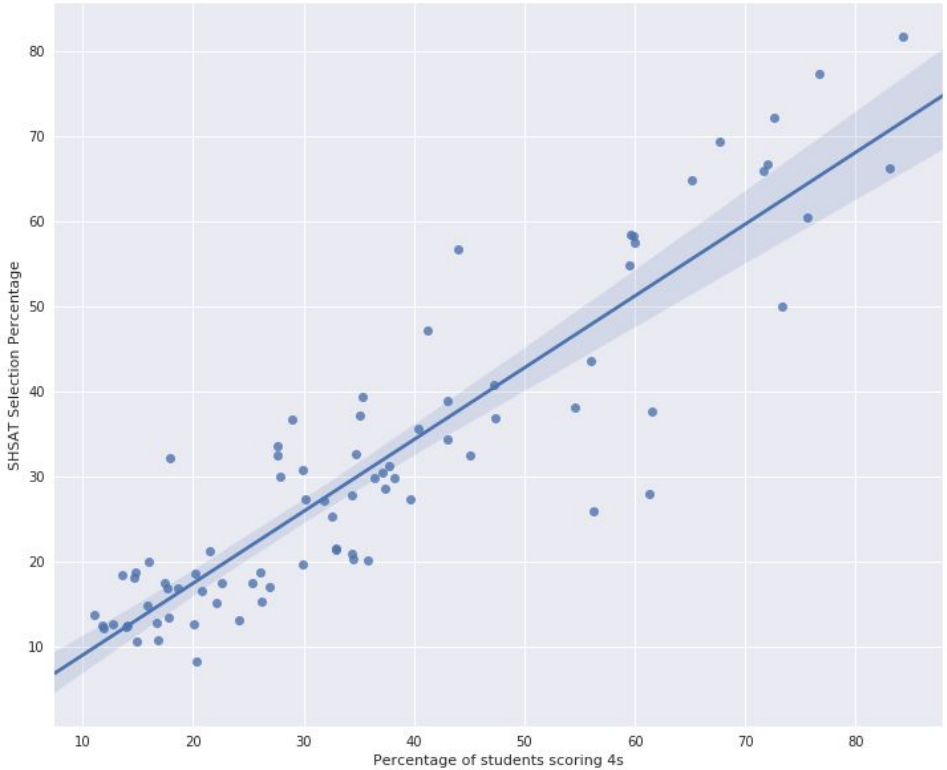



# Interactive plots with plotly




**Visualizations should be self explanatory**

# Feature Selection/ Reduction

---

Variance Based

Correlation Based (between features and to target)

Recursive Feature Elimination

PCA/ SVD

# Predictions

---

- Start with simpler methods (linear/ logistic regression benchmarks)
- Start with some known benchmarks if available
- Bias/ variance tradeoff
- Deep learning tools only if you have large amount of data

Data Science hands on:
https://colab.research.google.com/drive/1wvWbtjcxIqmxFcMgJFxehZlXAg7NrWQb

Visualizations:
https://colab.research.google.com/drive/1PnQEAirJDHpgtZQV055D63taKf9--YQk

# Industry Case Study 1

---

- Finance domain data
- Get data from paid platforms (Each with their own API)
- Scrape data from websites which allow it
- Handle unavailability of data
- Compute features
- Compute strategies
- Test new strategies on all past data!
- 60-70% of the task consists of getting the data right

# Challenges

---

- Scale of data (time x stocks x fields)
- Changing nature of the data
- Time constraints
- Handle holidays/ correctness of the data
- Run the models on the data (with possibly some missing values)
- All infrastructure on the cloud

# Industry Case Study 2

---

- Food supply chain (company food outlets)
- No prior work/ don’t know what data to collect
- Cannot ask for intervention from people managing it
- Capture videos/ images
- Tag videos (manual)
- Gather enough data to automate it
- Analyse what is being wasted where
- Different outlets have some form of inventory
- Different menus/ items at different places

# PASSNYC Data Science for good challenge

https://www.kaggle.com/infocusp/holographic-view-of-underperforming-schools

https://www.kaggle.com/infocusp/recommendations-to-passnyc-based-on-data-analysis

Purely data science - no ML

# Careervillage: Mapping professionals to likely questions

---

https://www.kaggle.com/infocusp/deepdive-into-careervillage/data

Not just about coding either: https://www.kaggle.com/danielbecker/careervillage-org-recommendation-engine

But once you have the idea clear, python to the rescue

# General Practical Tips

# 1. Know how to formulate

---

Formulating the problem is the key.

- Memory / compute size constraints
- Cost constraints
- Class balance
- When will the features be available
- Missing data?
- How reliable the inputs are?
- Kaggle type of problems are not real enough! The real world problems are not well-defined. The onus is on you to define it.

# 2. Know the context

---

However important part ML may be for the system, it is not an isolated part

Instead, ML is a part of some system, some product.

Get sense of timing, get sense of what customers want. Accuracy is not everything!

Examples:

Cancer detection

Google search

# 3. Know the data and domain

---

Knowing the process of data generation and creation is important. It decides which models we can apply to the problem (more importantly, which ones we cannot!)

Is the training data distribution same as real-life testing distribution?

Having domain knowledge about the field also helps

Examples:

- All projects at Infocusp

# 4. Know the layout

---

- No single tool offers everything necessary. There are pros and cons.
- Fixating on a single tool/technology can be disastrous.
- Exploring other options stretches how far one can go
- Evolving technologies

# 5. Be a good communicator

---

- ML and data science is extremely interdisciplinary.
- You will meet people with lack of belief and lack of knowledge.
- Eventually, data mining and machine learning practitioners report to someone.
- Important to back your claims with sound analysis.
- Important to present the findings effectively, being a good visual communicator helps.

# 6. Spend time knowing the data

The model is only as good as the data is.

- You cannot change the raw data, BUT
- you can clean it.
- you can transform it into powerful features.

Examples :

- Speech processing
- Face recognition
- Face detection
- Deep learning is changing the landscape … but only partially.

# 7. Occam’s razor

---

- Occam’s Razor : It is your friend!
- Clarification : It does not mean, complex algorithms should not be used!
- It means that simpler models are preferred given all the other things being the same.

# GitHub (open source) and use VCS

- An excellent platform to showcase your projects
- Get suggestions/ ideas from the community
- Make sure your work is being used by the people and it comes in handy (Write about it)
- Look at pending issues in popular repositories

VCS

- Prevent multiple copies of source code
- Merge changes from different users seamlessly
- Prevent code from breaking repeatedly
- Always have a working copy in case of inadvertent changes
- Changes are modular and you know whom to blame :)


# Contact Details

**Falak Shah**
falak@infocusp.in
https://github.com/falaktheoptimist