

Model building, Interpretability, and Deployment

PRESENTED BY -
Aayush Agrawal



Aayush Agrawal

Education -

- MS in Business Analytics, Carlson School of Management, University of Minnesota, 2017
- B.Tech in Electrical Engineering, Malaviya National Institute of Technology, India, 2013



Experience –

- >5 years in Data science, Currently working as a Data scientist at Land O' Lakes, Inc.
- Moderator and rank 3rd at <https://www.analyticsvidhya.com/>
- Kaggle Expert - <https://www.kaggle.com/aayushmnit>



Reach me out at (@aayushmnit) –

Email - aayushmnit@gmail.com

LinkedIn - <https://www.linkedin.com/in/aayushmnit/>

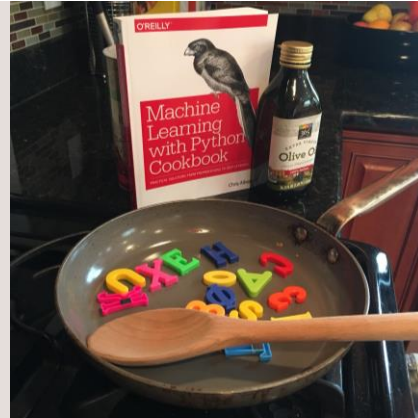
Github - <https://aayushmnit.github.io/>

Medium - <https://medium.com/@aayushmnit>



Agenda

Model building
& Cookbooks



Interpretability using
LIME

Deployment as
API using Flask



Docker
and Cloud

CRISP DM – Best methodology for Data science



Business Understanding

Translation of project objectives and requirements from a business perspective to data Science problem

Data Understanding

Data collection, quality checks, exploration, hypothesis testing

Data preparation

Process of converting raw data to model ready data

Modeling

Building advance analytics/ machine learning model

Evaluation

Model quality assesment and results interpretation

Deployment

Automation of data and modeling pipelines

Use of Cookbooks in Data science

- Many of the data science task are quite repeatable and highly generalizable
- 60% of the time writing code is wasted on functions/program which a data scientist have already written

Reusability

- Less syntax to remember
- DRY (Don't repeat yourself)

Modularity

- Cleaner and well written code
- Reliable performance by multiple re-use in different scenario

Faster execution

- Less code to write (10Xer)
- Consistent output to analyze

Customizable

- Appending/Removing functionality
- Customizable documentation

My cookbooks overview

Link to git repo - <https://github.com/aayushmnit/cookbook>

[Generic Preprocessing](#) - Helper functions to do EDA, missing value analysis & treatment and generic preprocessing like (scaling, encoding etc.)

[Machine learning - Classification](#) - Helper functions to solve classification type problems in machine learning. It contains codes for holdout/cross validation, model explanation codes (LIME and variable importance plot), and codes for general classification algorithms (Xgboost, LightGBM, Extra trees, random forest, logistic regression, decision trees, K-nearest neighbors and SVM)

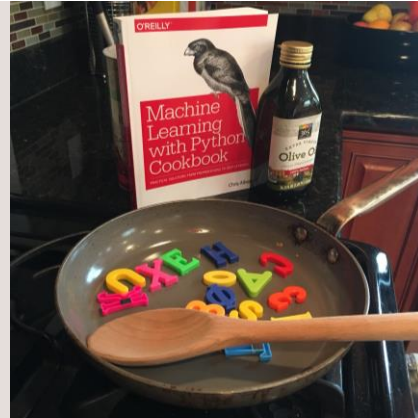
[Machine learning - Regression](#) - Helper functions to solve regression type problems in machine learning. It contains codes for holdout/cross validation, model explanation codes (LIME and variable importance plot), and codes for general regression algorithms (Xgboost, LightGBM, Extra trees, random forest, linear regression, regression trees, K-nearest neighbors and SVM)

[Recommender systems](#) - Helper functions to build recommender systems using Matrix factorization using LightFM package.

[Natural language processing](#) - Helper functions for NLP text processing and analysis like N-grams, word cloud, tokenization, lowercasing, punctuation/stopwords removal, stemmer/lemmatizer, TF-IDF & count vectorizer

Agenda

Model building
& Cookbooks



Interpretability using
LIME

Deployment as
API using Flask



Docker
and Cloud

XKCD: Machine Learning



Why Interpretability

- Typically data scientist are not domain experts and interpretability helps in learning insights about a domain to communicate with domain experts
- Provides ability to users to understand, interact and communicate with complex models with complex algorithms which they don't know about
- Ability to verify that the model is making right prediction and also ability to tweak/influence models to generate predictions which are expected from it
- Establish trust and transparency in a “black-box” system
- Identify models mistakes/blunders

Why Interpretability is important

Trust

How can we trust the predictions are correct?

Being able to interpret the explanations and compare classifiers based on them

Predict

How can we understand and predict the behavior?

Improved prediction of model behavior and time to make that assessment when explanations were provided

Improve

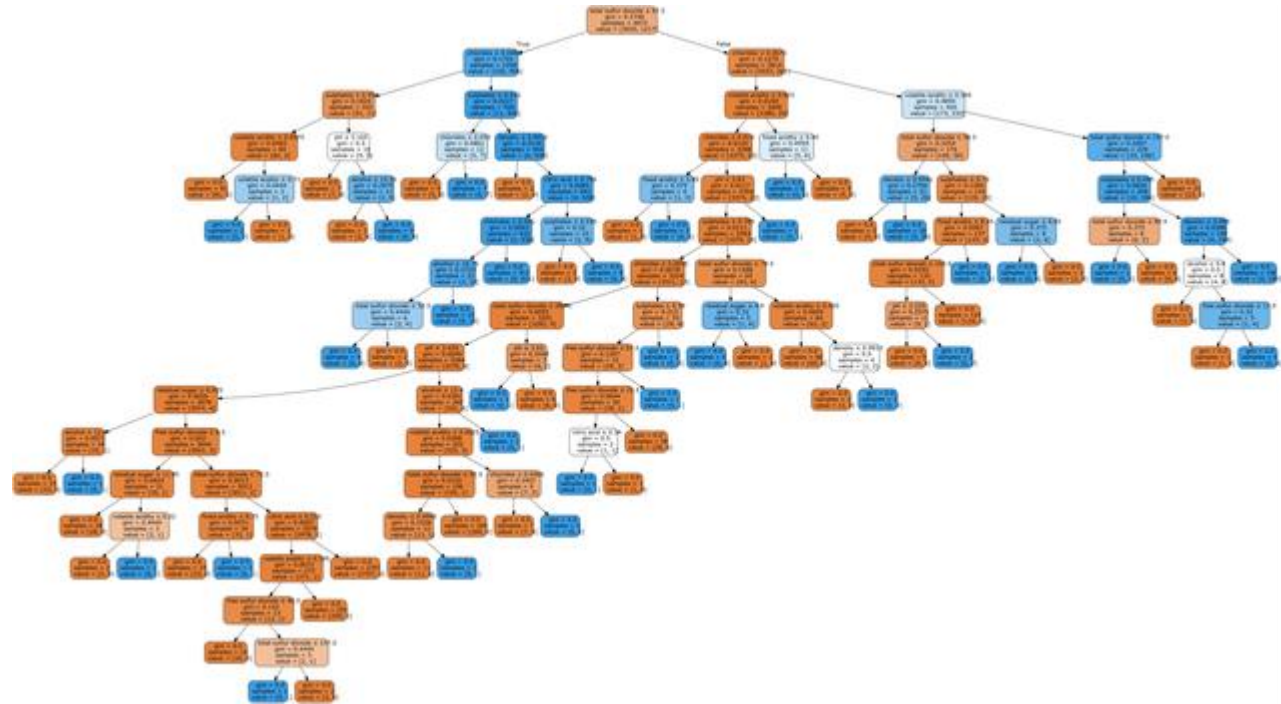
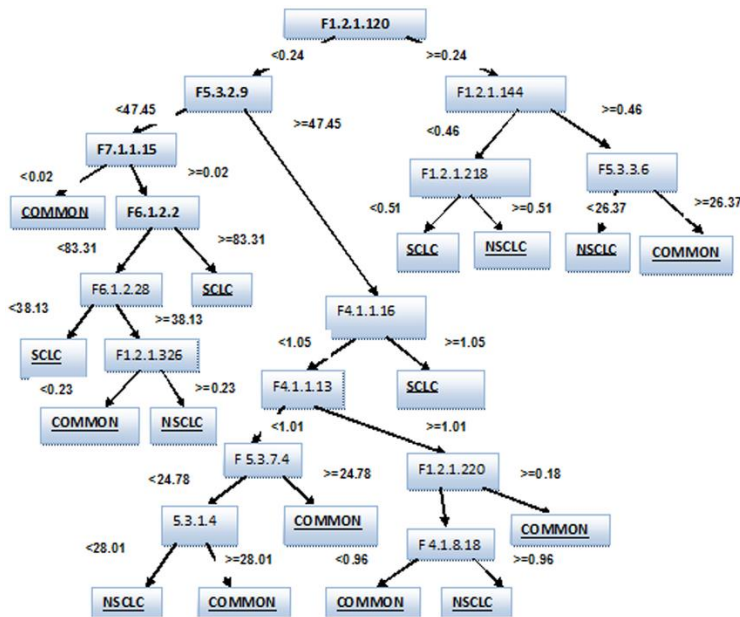
How do we improve it to prevent potential mistakes?

Non-ML experts with explanations

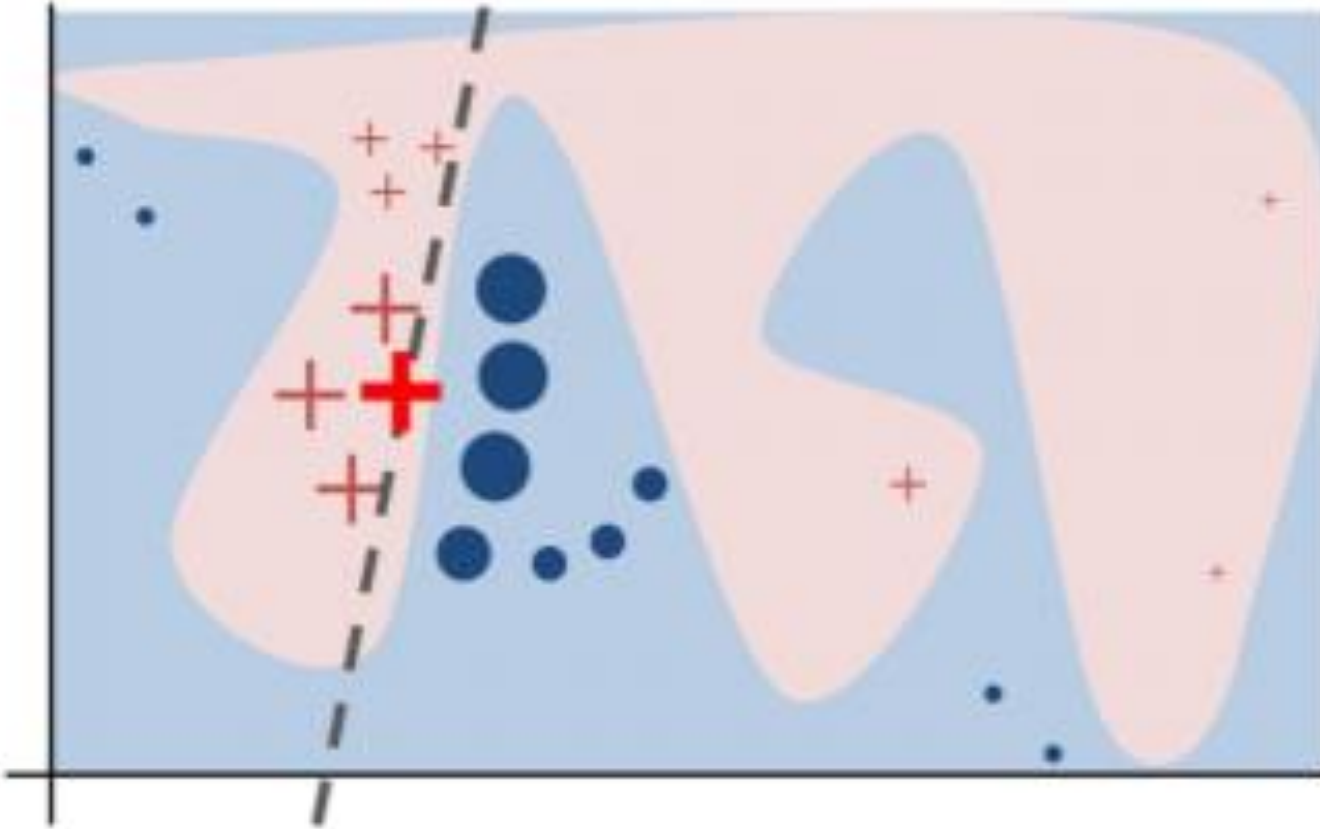
VS

ML experts without explanations

Big data comes with more complexity and dimensions



LIME: Local Interpretable Model-Agnostic Explanations



LIME explanation is an interpretable model that is locally accurate

How LIME works

- 1 – Permute Data*
- 2 – Calculate distance between permutations and original observations*
- 3 – Make predictions on new data using complex model
- 4 – Pick m features best describing the complex model outcome from the permuted data*
- 5 – Fit a sample model to the permuted data with m features and similarity scores as weights*
- 6 – Feature weights from the simple model make explanations for the complex models local behavior

Examples of LIME explanations(Image)

Good Model



(a) Original Image



(b) Explaining *Electric guitar*



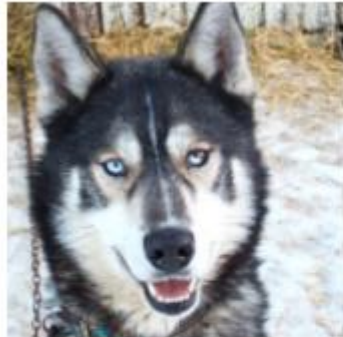
(c) Explaining *Acoustic guitar*



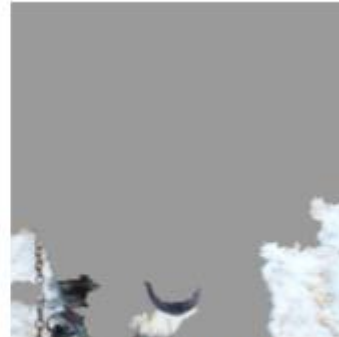
(d) Explaining *Labrador*

Figure 4: Explaining an image classification prediction made by Google's Inception neural network. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$), "Acoustic guitar" ($p = 0.24$) and "Labrador" ($p = 0.21$)

Bad Model



(a) Husky classified as wolf



(b) Explanation

Figure 11: Raw data and explanation of a bad model's prediction in the "Husky vs Wolf" task.

Examples of LIME explanations(Text)

Prediction probabilities



atheism

christian



Text with highlighted words

From: johnchad@triton.unm.edu (jchadwic)
Subject: Another request for Darwin Fish
Organization: University of New Mexico, Albuquerque
Lines: 11
NNTP-Posting-Host: triton.unm.edu

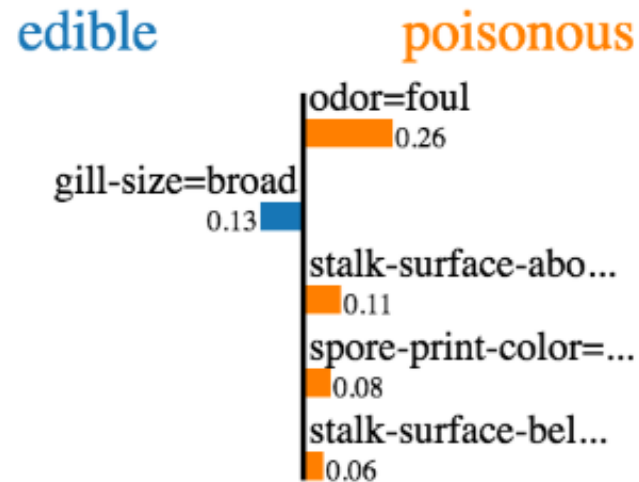
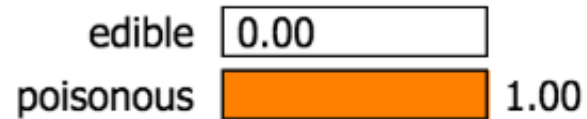
Hello Gang,

There have been some notes recently asking where to obtain the DARWIN fish.
This is the same question I have and I have not seen an answer on the net. If anyone has a contact please post on the net or email me.

Examples of LIME explanations(Tabular data)

Tabular data

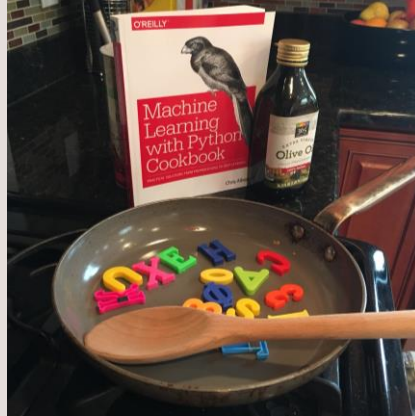
Prediction probabilities



Feature	Value
odor=foul	True
gill-size=broad	True
stalk-surface-above-ring=silky	True
spore-print-color=chocolate	True
stalk-surface-below-ring=silky	True

Agenda

Model building
& Cookbooks



Interpretability using
LIME

Deployment as
API using Flask



Docker
and Cloud

Flask is a micro service web platform framework

Many machine learning models are used real time at the heart of any data product.

Options to implement Machine Learning models –

Option 1: Rewriting the whole code in the same language as product

- Very hard and time consuming process

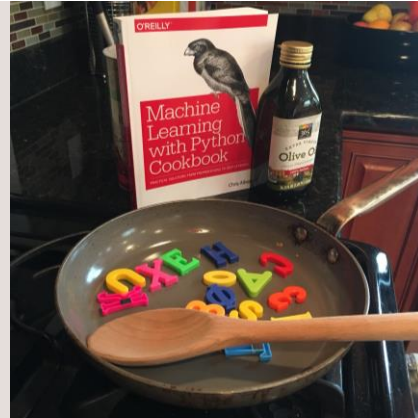
Option 2: API-first approach

- Web APIs have made it easy for cross-language applications to work well
- Software product can just call the API with bunch of parameters to get result regardless of backend language



Agenda

Model building
& Cookbooks



Interpretability using
LIME

Deployment as
API using Flask



Docker
and Cloud

Docker is a containerization platform

Docker is a platform for developers and sysadmins to develop, deploy, and run applications with containers.

Containerization is increasingly popular because containers are:

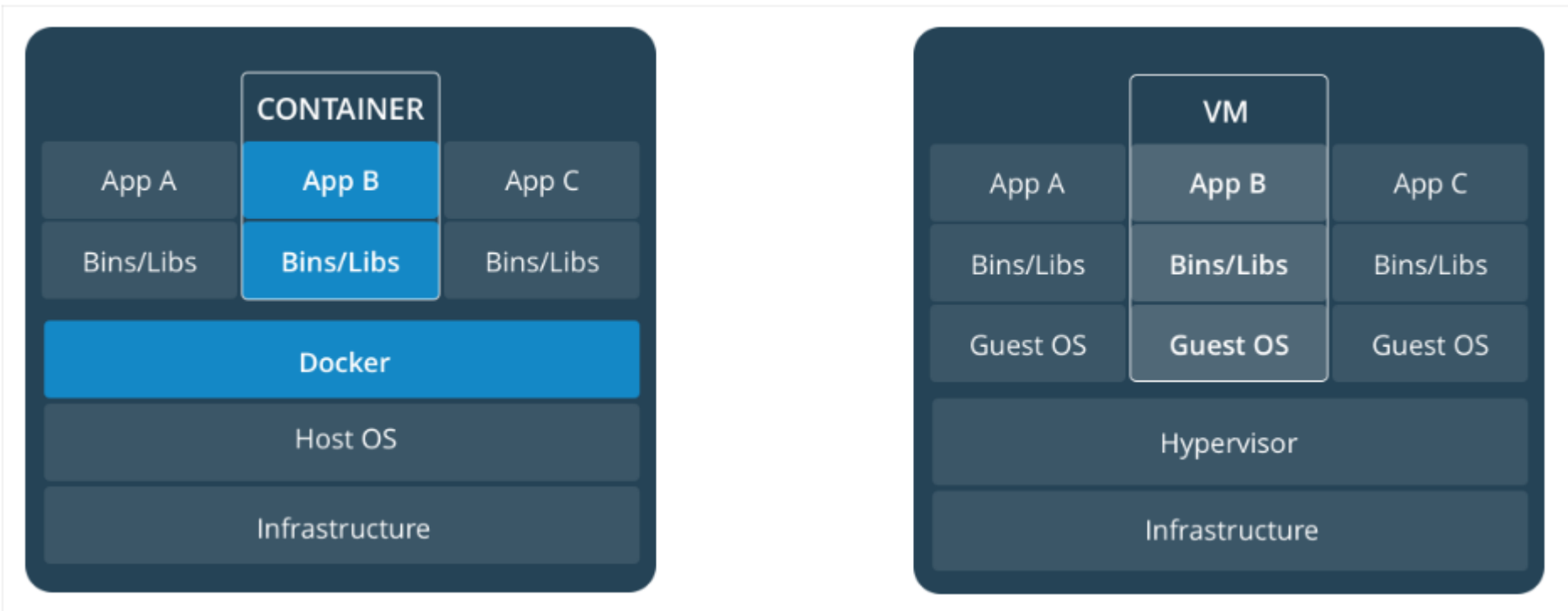
- **Flexible:** Even the most complex applications can be containerized.
- **Lightweight:** Containers leverage and share the host kernel.
- **Interchangeable:** You can deploy updates and upgrades on-the-fly.
- **Portable:** You can build locally, deploy to the cloud, and run anywhere.
- **Scalable:** You can increase and automatically distribute container replicas.
- **Stackable:** You can stack services vertically and on-the-fly.

Image - An image is an executable package that includes everything needed to run an application--the code, a runtime, libraries, environment variables, and configuration files

Container - A container is a runtime instance of an image--what the image becomes in memory when executed



Container is light weight because it doesn't require an OS



Creating your own docker file

<https://docs.docker.com/get-started/part2/#dockerfile>

```
# Use an official Python runtime as a parent image
FROM python:2.7-slim

# Set the working directory to /app
WORKDIR /app

# Copy the current directory contents into the container at /app
ADD . /app

# Install any needed packages specified in requirements.txt
RUN pip install --trusted-host pypi.python.org -r requirements.txt

# Make port 80 available to the world outside this container
EXPOSE 80

# Define environment variable
ENV NAME World

# Run app.py when the container launches
CMD ["python", "app.py"]
```

Some good resources on topic discussed

LIME –

- Lime Github page – [Link](#)
- Why Should I Trust You?: Explaining the Predictions of Any Classifier (Arxiv) - [Link](#)
- Interpretable Machine Learning Using LIME Framework - Kasia Kulma (PhD) - [Link](#)
- Towards interpretable reliable models - Keynote Katharine Jarmul – [Link](#)

Docker –

- Docker official tutorial – [Link](#)
- How Docker Can Help You Become A More Effective Data Scientist – [Hamel Husain](#)
- A Step Towards Reproducible Data Science : Docker for Data Science Workflows – [Analytics Vidhya](#)

API using FLASK –

- Tutorial to deploy Machine Learning models in Production as APIs (using Flask) – [Analytics Vidhya](#)
- How to Deploy Keras Models to Production – [Siraj Raval](#)
- Flask Tutorial – [Link](#)

Thank You!