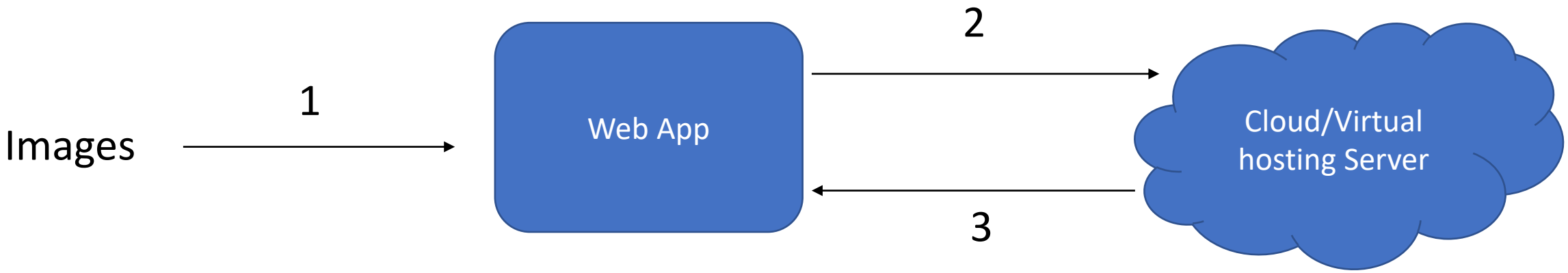
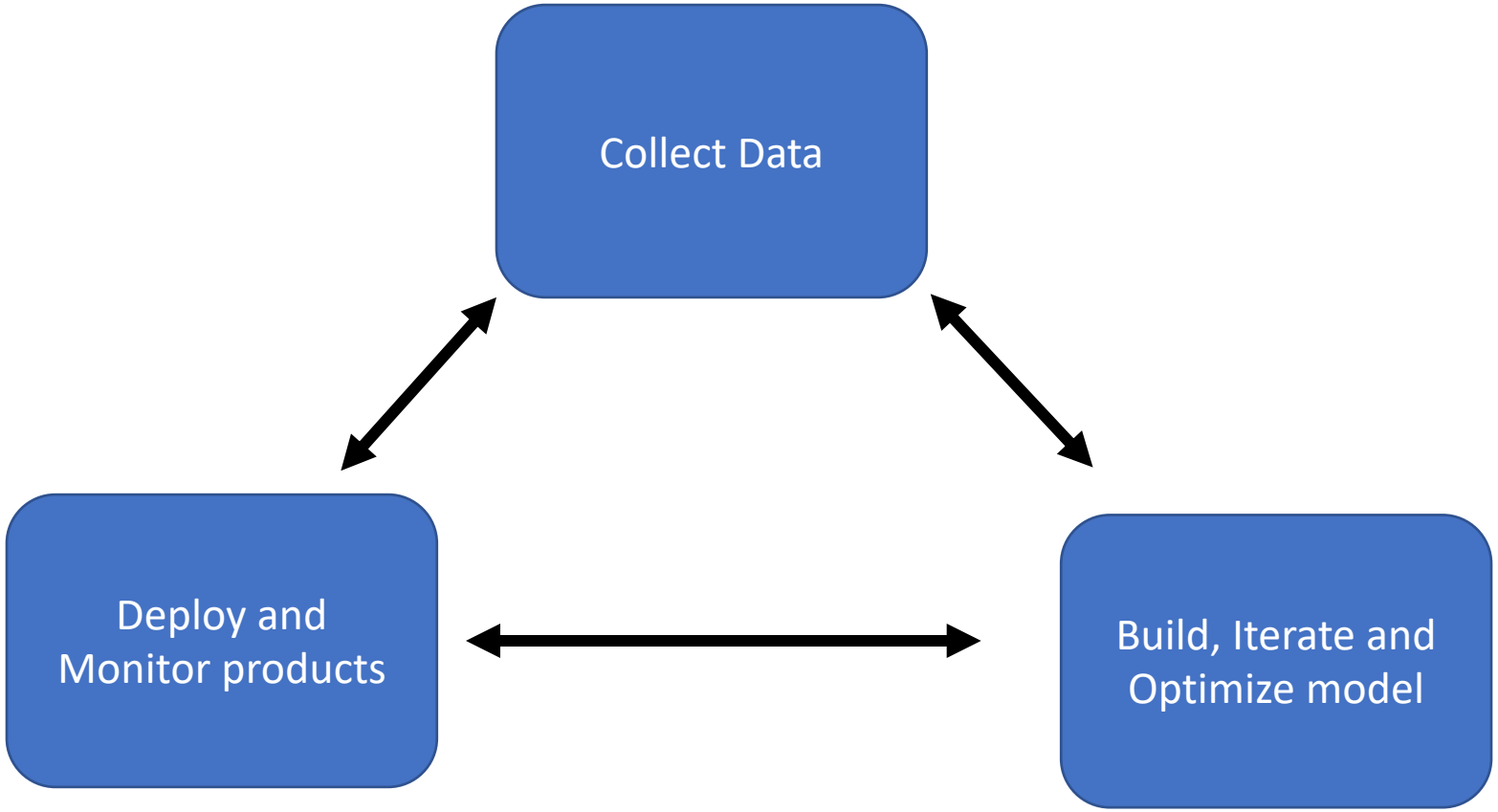
Full-Stack ML:
Sneaker Brand Classification & Price Prediction Web App

Main Use Case



1. Sneaker image uploaded to web app
2. Request sent to server for classification and prediction task
3. Response from server with classification and prediction result

Development Phase



I. Collect Data

- **Objective**: Build a dataset that contains **sneaker images** and their corresponding **brands** and **prices**
- **Result and Achievement**:
  - ✓ Python script to scrape image links, brands and prices
  - ✓ Python script to automatically download images from link
- **Improvement and future work**:
  - ✓ Diversify data sources to avoid bias (public dataset, other websites ….)
  - ✓ Multi-threading to speed up the downloading process

	A	B	C	D
1	Product_name	Brand_name	Img_source	Price_tag
2	Benassi JDI Slide By Nike	Nike	https://m.media-amazon.com/images/I/61KtX5bZY0L._AC_SX510_.jpg	19.95
3	Kawa Slide By Nike	Nike	https://m.media-amazon.com/images/I/71dZQUitQTL._AC_SX510_.jpg	24.95
4	adissage By adidas	adidas	https://m.media-amazon.com/images/I/61IRs+N+R1L._AC_SX510_.jpg	30
5	Benassi JDI Slide By Nike	Nike	https://m.media-amazon.com/images/I/61mpybX3b-L._AC_SX510_.jpg	19.95
6	Revolution 5 By Nike	Nike	https://m.media-amazon.com/images/I/71h9NwLcGML._AC_SX510_.jpg	65
7	Air Zoom Pegasus 36 By Nike	Nike	https://m.media-amazon.com/images/I/71y68z8KGtL._AC_SX510_.jpg	90
8	GEL-Venture¬Æ 7 By ASICS	ASICS	https://m.media-amazon.com/images/I/71H5TBDullL._AC_SX510_.jpg	52.5

II. Build, Iterate and Optimize model performance

- **Objective**: Build a model that can take the input as the **sneaker image** and then map out its **brand**

- **Result and Achievement**:
  - A Jupyter notebook file detailing all the data pre-processing, fitting and evaluating for the model
  - A classification model achieve more than 80% accuracy score for predicting the brand

- **Lesson, Improvement and Future Work**:
  - Start simple (quick to implement, understandable, deployable) --> benchmark for improvement
  - Debug model:
    - **Wiring**: flow of data from pictures to prediction → SW good practices
    - **Learning**: underfitting → more sophisticated model, data sampling techniques, quality, …
    - **Generalizing:** overfitting → regularization, augmentation, …
  - Once workflow clears → CLI or simple tools to speed up development process

II. Build, Iterate and Optimize model performance

Preprocessing:
- Resize
- Scale
- Normalize …

Dimensionality Reduction/ Feature Engineering:
- PCA
- Image Augmentation

Fit Model and Evaluate:
- Supervised techniques
- CV and Debugging

Save as .pkl file

Training/Evaluation Pipeline

II. Build, Iterate and Optimize model performance

Preprocessing

Dimensionality Reduction/ Feature Engineering:

Saved pkl model

Prediction

Testing Pipeline

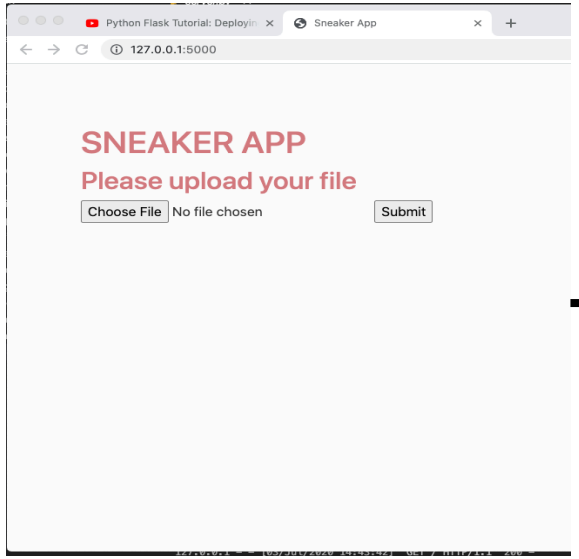
III. Deploy and Monitor

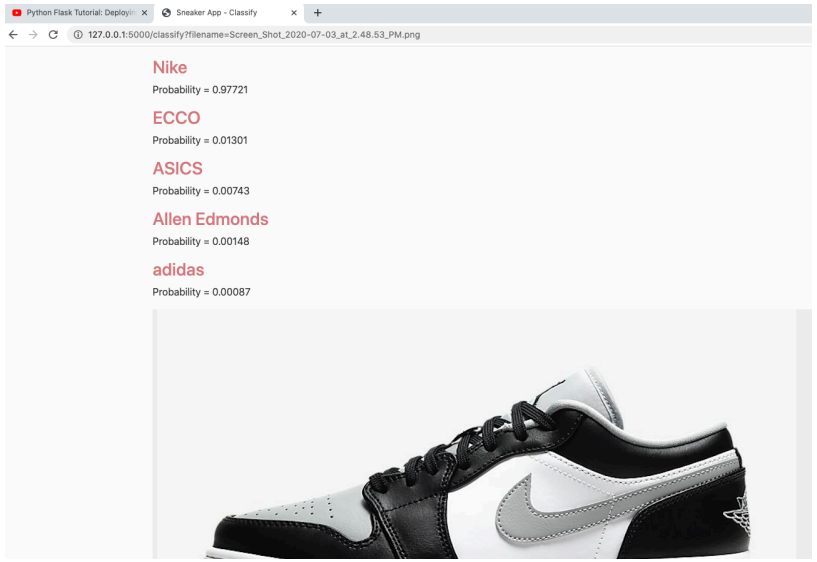
- **Objective:** Productionize our model by building a web app hosted on a cloud server (AWS, Linode or Heroku)

- **Result and Achievement:**
  - A complete Web App (built with python Flask framework) fully tested on local machine
  - A ready-to-deployed Linux environment for the Web App

- **Lesson, Improvement and Future Work:**
  - Safeguards for application → estimate when your application/model fail and engineer around it
  - Monitoring and Maintaining not only software and but also model
  - Many tools and technology to speed up process → choose wisely

III. Deploy and Monitor





https://youtu.be/vyIXO6fwjWY

Overall Conclusion and Future Work

1) Technical Skills

- Finish up the deployment process and maybe obtain a domain name
- The development for price prediction model is similar → A couple of approach:
  - Separate model for regression problem
  - Multi-task learning

2) Soft Skills

- Machine Learning Engineer is the glue/sweetspot of Data Scientist and Software Engineering
- Not just tools/ technology → Teamwork, Critical Thinking and Communication is the key