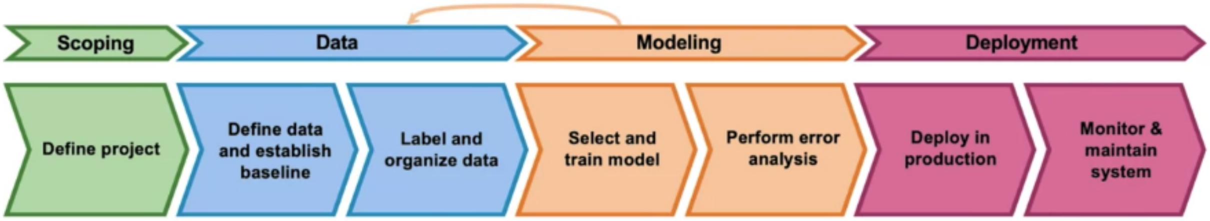
Introduction to Machine Learning in Production

Week 1 - Overview of the ML Lifecycle and Deployment

This week they talked about a proposal for the ML project lifecycle and Deployment, below you can find a screenshot of the whole process.



To detail each stage of the process:

Scoping
- Decide to work on the problem and understand what it's about
- Decide on key metrics, such as: accuracy, latency, throughput, and so on

Data
- Define the data
- Check if the data is labeled consistently
- Perform a normalization in the data collected

Modeling
- Code (algorithm/model) - research/academia
- Hyperparameters - research/academia and product team
- Data - product team

Deployment
- Deploy in production and understand the flow (the API's POST and GET)
- Monitor and maintain the system (concept or data drift)

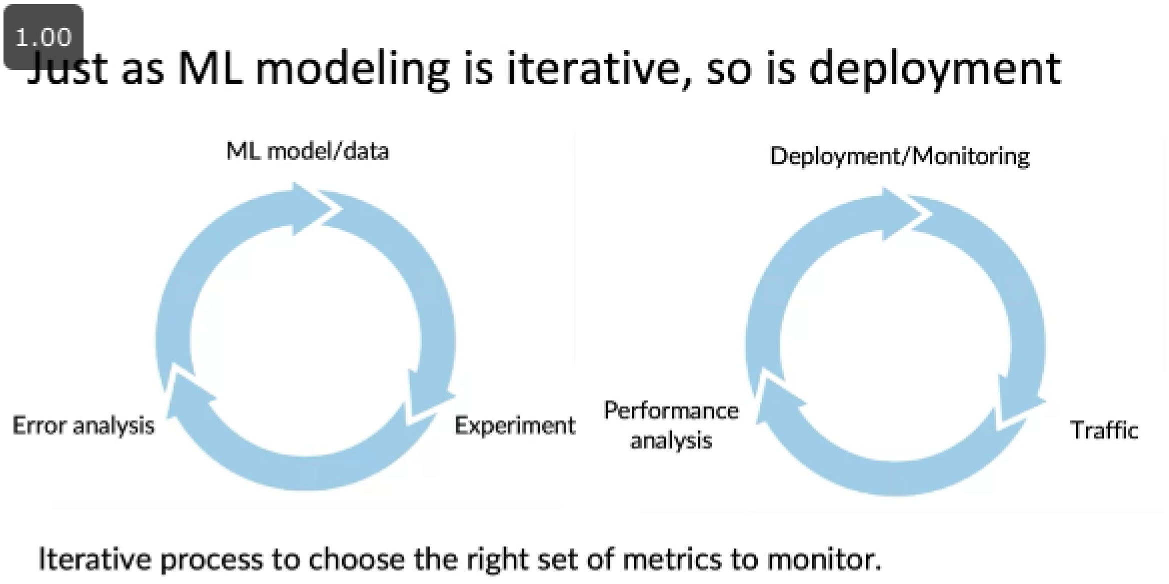
Thus, MLOps is an emerging discipline and comprises a set of tools and principles to support progress through the ML project lifecycle.

For the monitoring, some important definitions:

- Concept drift: the definition of what is y given x changes over time
- Data drift: when the input distribution of the explanatory variables is changing over time

Regarding the different **deployment patterns**:

- Shadow mode: model runs in parallel not making any real decision
- Canary mode: roll out to small fraction of traffic initially (5%, for example) and later on ramp up increasing this percentage
- Blue green mode: old (blue) version, you spin up a new (green) version, and then suddenly you can migrate the flow from the blue to green in one take



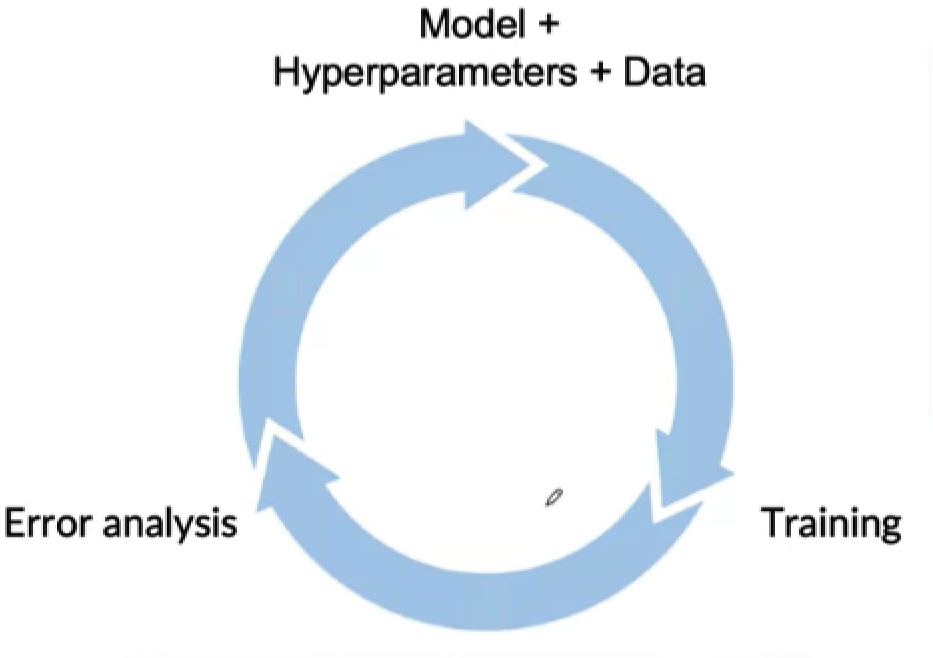
Just as ML modeling is iterative, so is deployment

Iterative process to choose the right set of metrics to monitor.

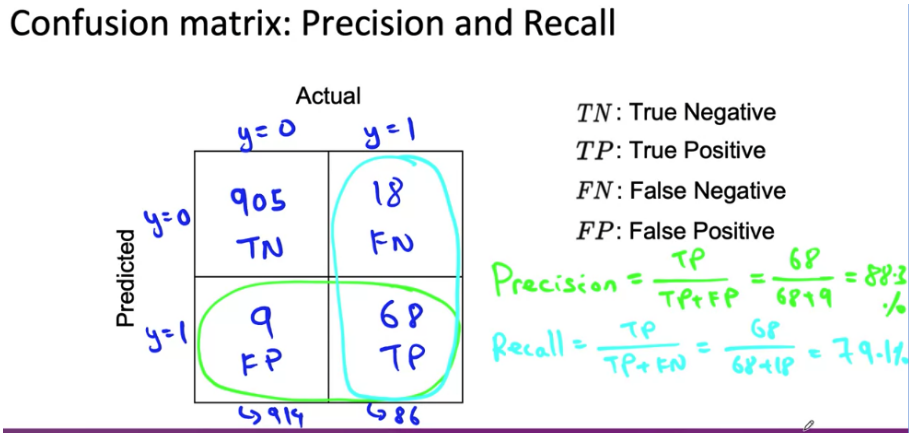
Week 2 - Select and Train a model

This week they talked about how to select and train a model in a data science project. Elaborated several aspects, such as:

- From model-centric to data-centric development: the new trend that is focusing the development of new models in enhancing the data used.
- Model development is an iterative process (shown below).



- Different types of metrics for measuring the performance of the model: accuracy, precision, recall, f1 score, and so on, each one for a different problem space (shown below).



Confusion matrix: Precision and Recall

Actual

Predicted

	y=0	y=1
y=0	905 TN	18 FN
y=1	9 FP	68 TP
	↳914	↳86

TN: True Negative

TP: True Positive

FN: False Negative

FP: False Positive

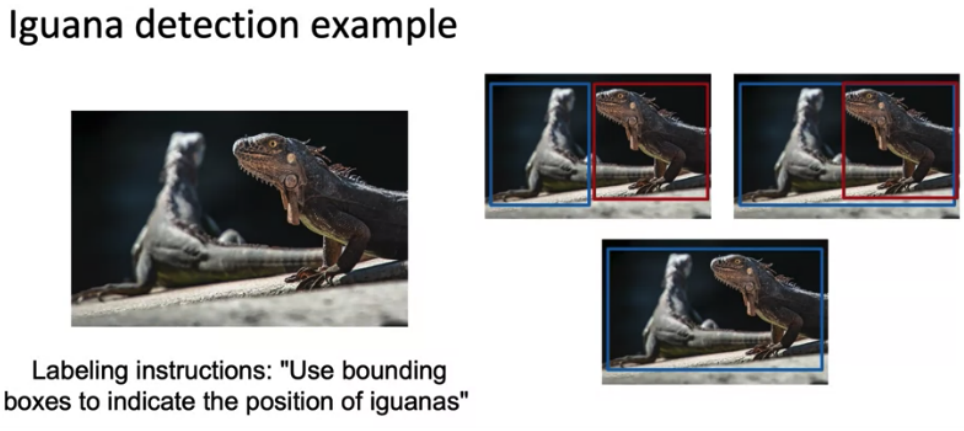
$$\text{Precision} = \frac{TP}{TP+FP} = \frac{68}{68+9} = 88.3\%$$

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{68}{68+18} = 79.1\%$$

Week 3 - Data Definition and Baseline

This week they talked about defining the data and establishing the baseline for comparing models to be used in the training of a model.

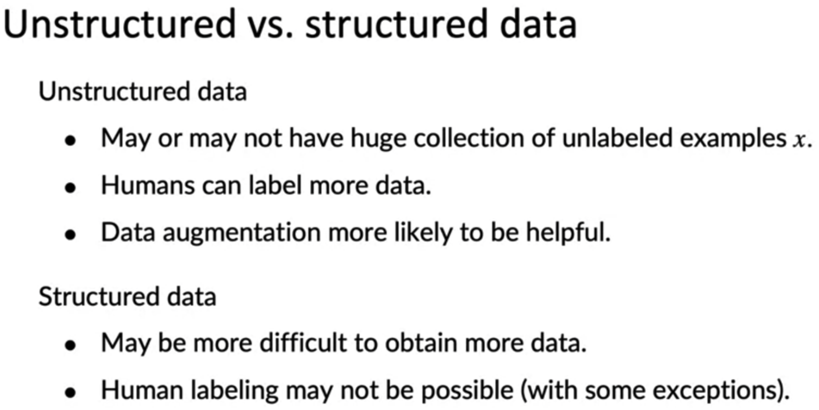
The definition of the data in terms of defining the right label is quite hard, the case can vary and sometimes there is no right/wrong answer, one example is shown below, but there are several different examples of data ambiguity.



Iguana detection example

Labeling instructions: "Use bounding boxes to indicate the position of iguanas"

How unstructured and structured data is treated differently is shown below, and small data (<= 10,000) is fundamental to have clean labels and in big data (> 10,000) is fundamental to have emphasis on the data process.



Unstructured vs. structured data

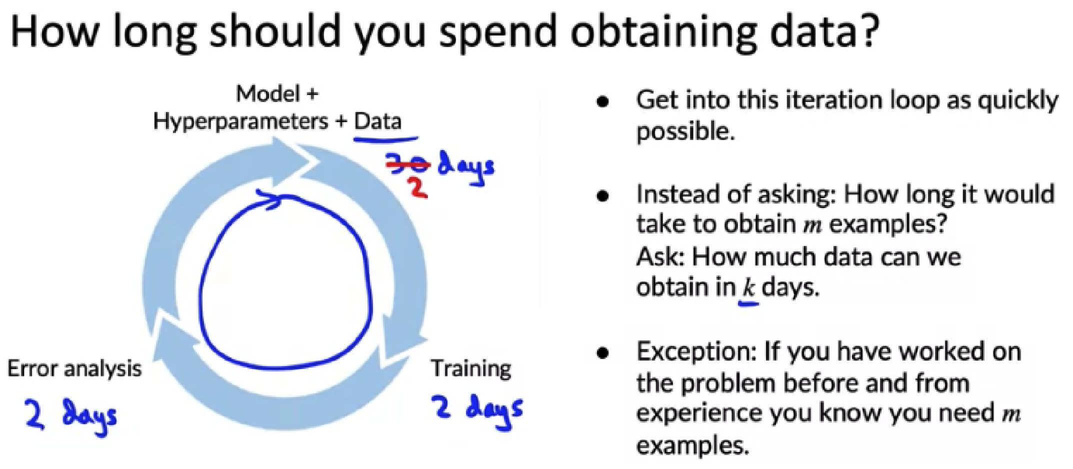
Unstructured data

- May or may not have huge collection of unlabeled examples x.
- Humans can label more data.
- Data augmentation more likely to be helpful.

Structured data

- May be more difficult to obtain more data.
- Human labeling may not be possible (with some exceptions).

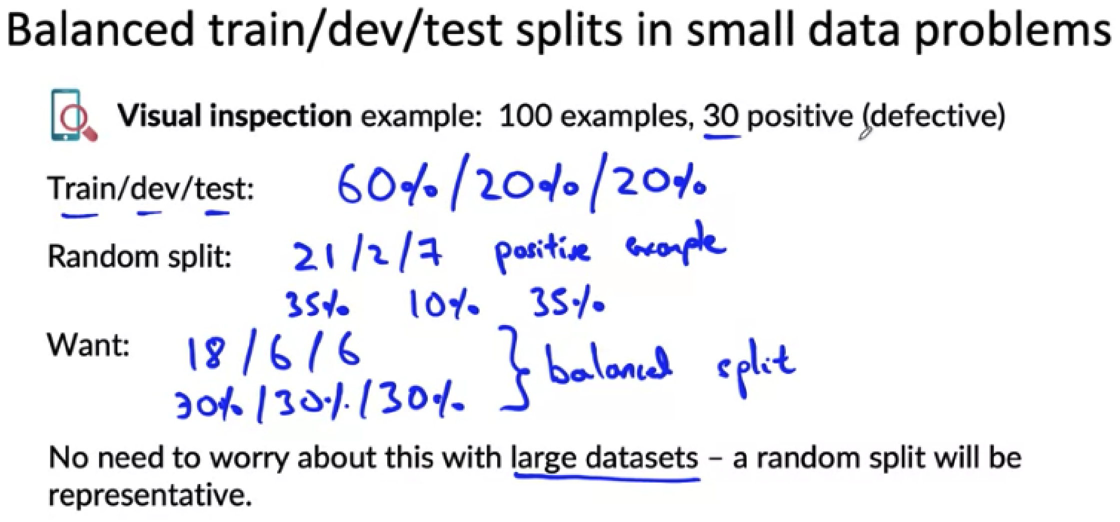
Andrew Ng advocates for taking an approach similar to the one explained in Lean Startup (iterating the process as fast as possible and later on deciding to pivot or persevere) when it comes to obtaining the data and making the training, the flow and estimates for spending time in the process are shown below.



How long should you spend obtaining data?

- Get into this iteration loop as quickly possible.
- Instead of asking: How long it would take to obtain m examples?
  Ask: How much data can we obtain in k days.
- Exception: If you have worked on the problem before and from experience you know you need m examples.

When it comes to splitting the data (usually small data) into training, development (also called validation dataset) and test dataset, they recommend checking the representativeness of each category in each dataset, and in case it's unbalanced, then make a balanced split.



Balanced train/dev/test splits in small data problems

Visual inspection example: 100 examples, 30 positive (defective)

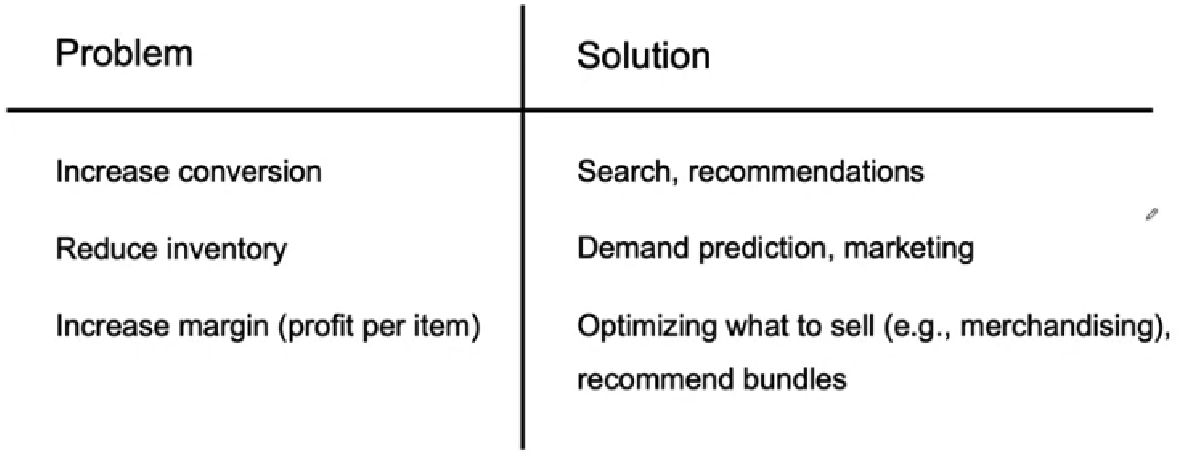
Train/dev/test: 60%/20%/20%

Random split: 21/2/7 positive example
35% 10% 35%

Want: 18/6/6
30%/30%/30% } balanced split

No need to worry about this with large datasets – a random split will be representative.

During the scoping process one should separate the problem identification from the solution per se, below there are some examples.



Problem	Solution
Increase conversion	Search, recommendations
Reduce inventory	Demand prediction, marketing
Increase margin (profit per item)	Optimizing what to sell (e.g., merchandising), recommend bundles

Finally, in diligence on value they touch in the ethical part of the problem as well with some questions such as:

- Is this project creating net positive societal value?
- Is this project reasonably fair and free from bias?
- Have any ethical concerns been openly aired and debated?