# ML for Smart Monkeys

Amit Sethi

Faculty member, IIT Bombay

*Image source: Pixabay.com*

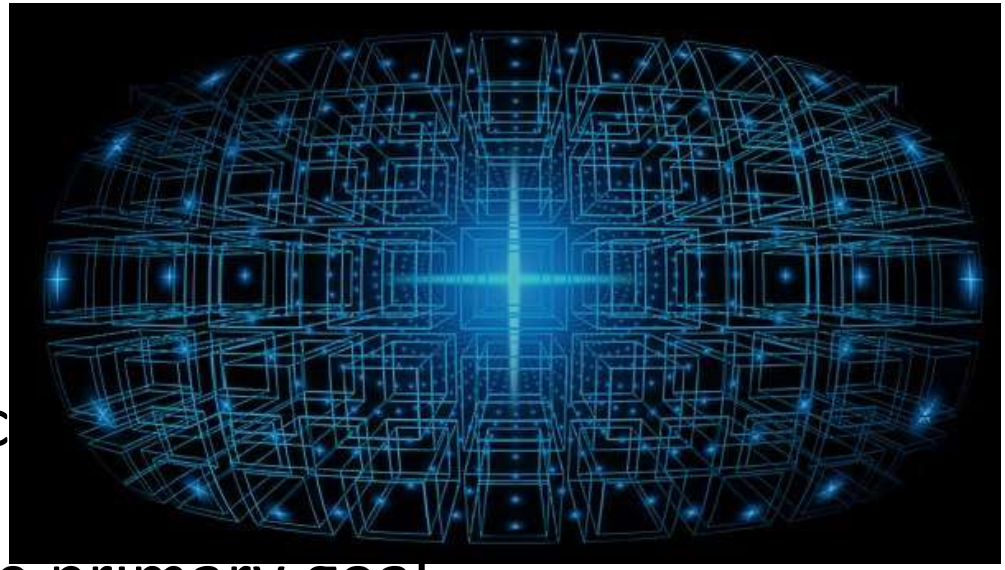
# ML is…

- The practice of automating the use of related data to estimate models that make useful predictions about new data, where the model is too complex for standard statistical analysis

- The practice of improve performance of a machine on a task using experience (Tom Mitchell), e.g.,

  - Improve accuracy of classification of images using labeled images
  - Improve win percentage on alpha-go using several simulated game move sequences and their results
  - Improve the Turing test confusion between human and machine for NLP Q&A using a large sample of text including Q&A

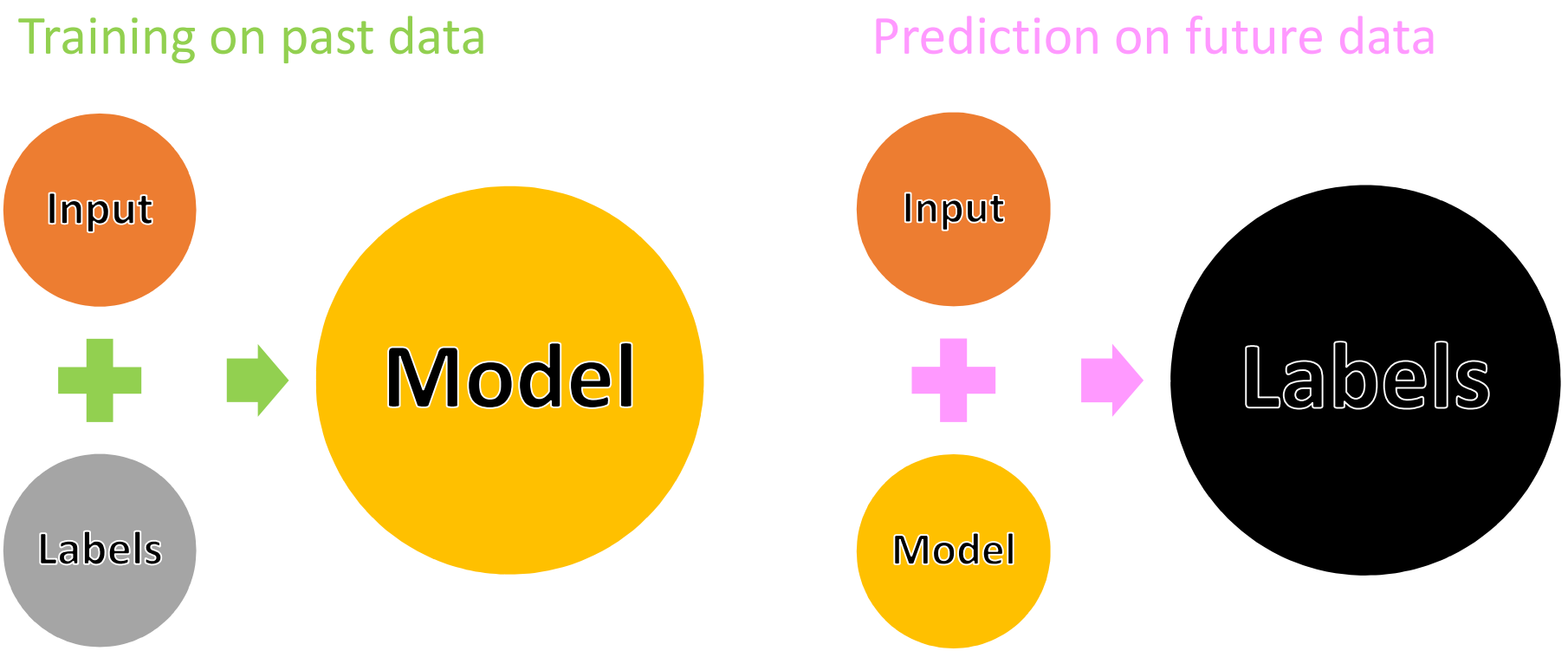
# Sweet spot for ML



- Lots of structured data
- Explainability is not critic
- Prediction accuracy is the primary goal
- Underlying model is complex but stationary

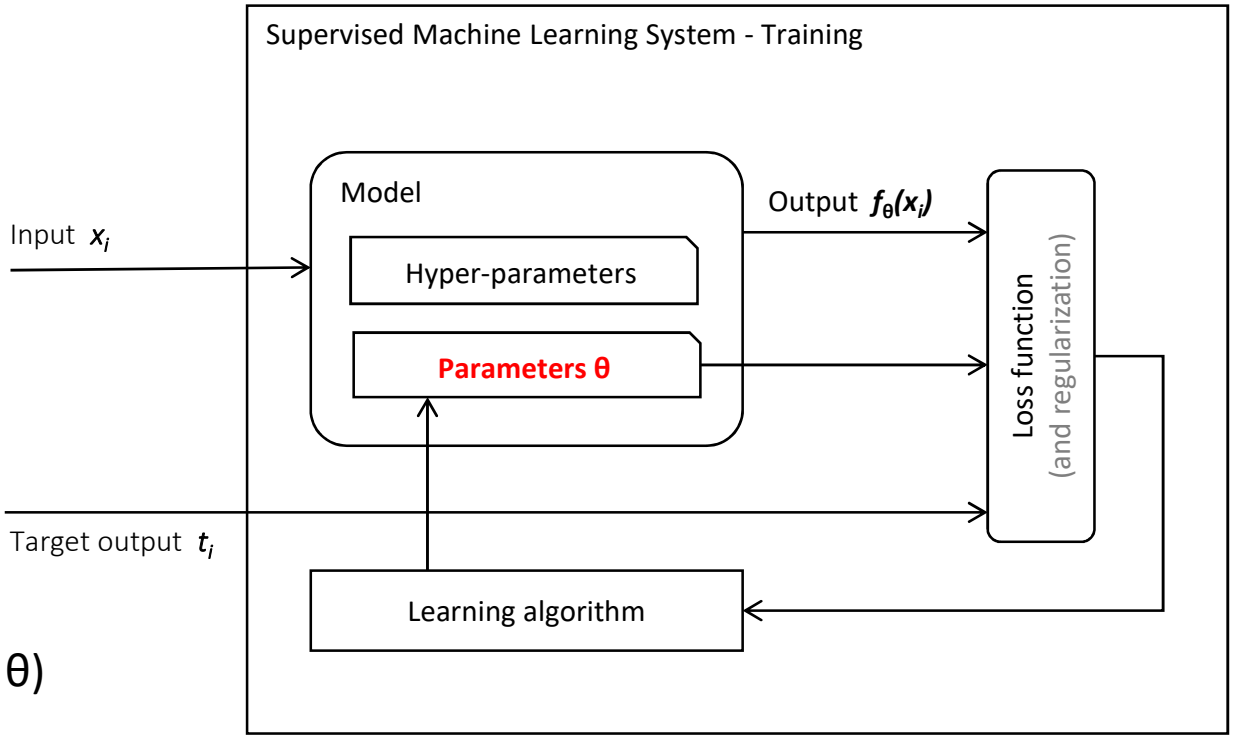
*Image courtesy: Pixabay.com*

# ML model training and deployment

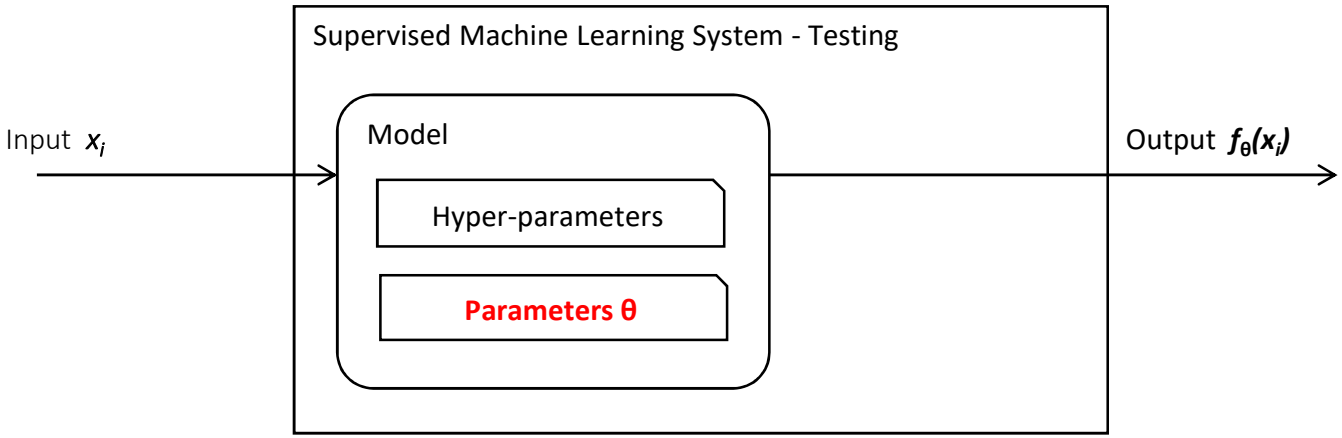


# ML gives a model

- Elements of a model:
  - Input $x_i$
  - Function $f_\theta(x_i)$
- Utility of the model:
  - Target output $t_i$
  - Bring $f_\theta(x_i)$ close to $t_i$
  - Minimize loss $L(t_i, f_\theta(x_i), \theta)$



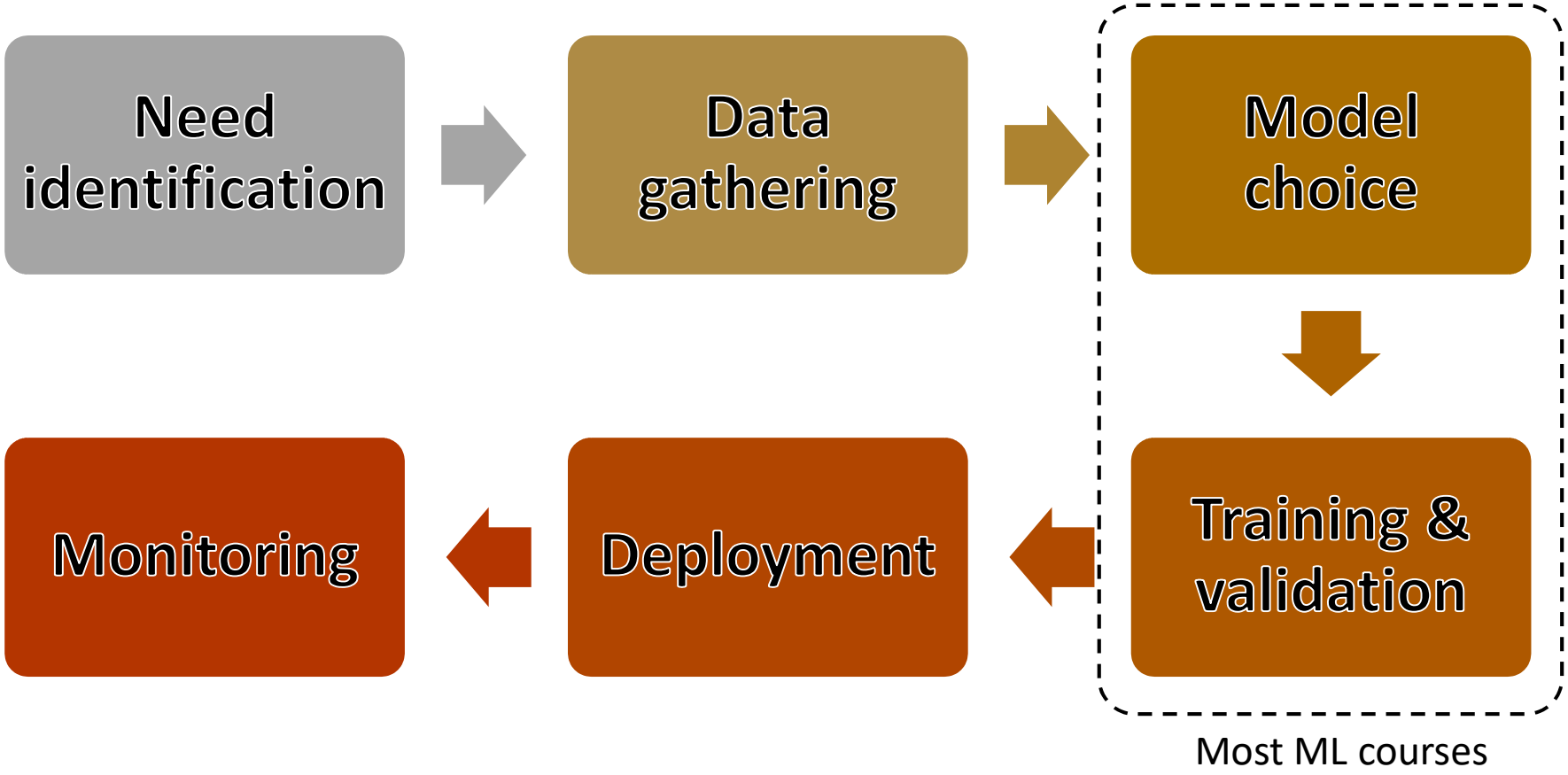
# Components of a Trained ML System



# Mathematically speaking…

- Determine f such that $t_i = f(x_i)$ and $g(T, X)$ is minimized for unseen set $T$ and $X$ pairs, where $T$ is the ground truth that cannot be used
- Form of $f$ is fixed, but some parameters can be tuned:
  - So, $y = f_\theta(x)$, where, $x$ is observed, and $y$ needs to be inferred
  - e.g. $y = 1$, if $mx > c$, $y = 0$ otherwise, so $\theta = (m, c)$
- Machine Learning is concerned with designing algorithms that learn "better" values of θ given "more" $x$ (and $t$) for a given problem

# ML life stages



# Recipe for ML training

- Decide on the type of the ML problem
- Prepare data
- Shortlist ML frameworks
- Prepare training, validation, and test sets
- Train, validate, repeat
- Use test data only once

# Broad types of ML problems

| Output → | Categorical | Ordinal | Continuous |
|---|---|---|---|
| Supervised | Classification | Ranking | Regression |
| (Examples) | {Cats, dogs} | {Low, Med, High} | [-20,+10) |
| Unsupervised | Clustering | | Dimension reduction |

# Preparing data

- Remove useless data
  - No variance
  - Falsely assumed to be available
- Reduce redundancy
  - Correlated
    - Pearson and Spearman
- Handle missing data
  - Impute, if sporadic
  - Drop, if too frequent
- Transform variables
  - Convert discrete to one-hot-bit
  - Normalize continuous variables

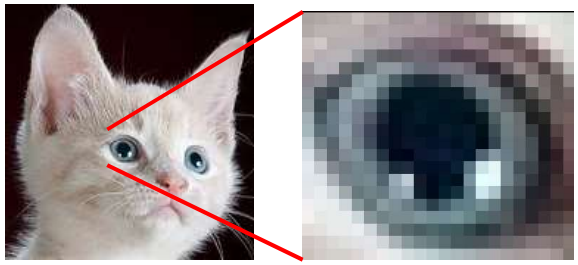
# Examples of structure in the data

- Records

| Product SKU | Price | Margin | Volume |
|---|---|---|---|
| A123ajkhdf | $ 120 | 30% | 1,000,000 |
| B456ddsjh | $200 | 10% | 2,000,000 |

- Temporal order

- Spatial order



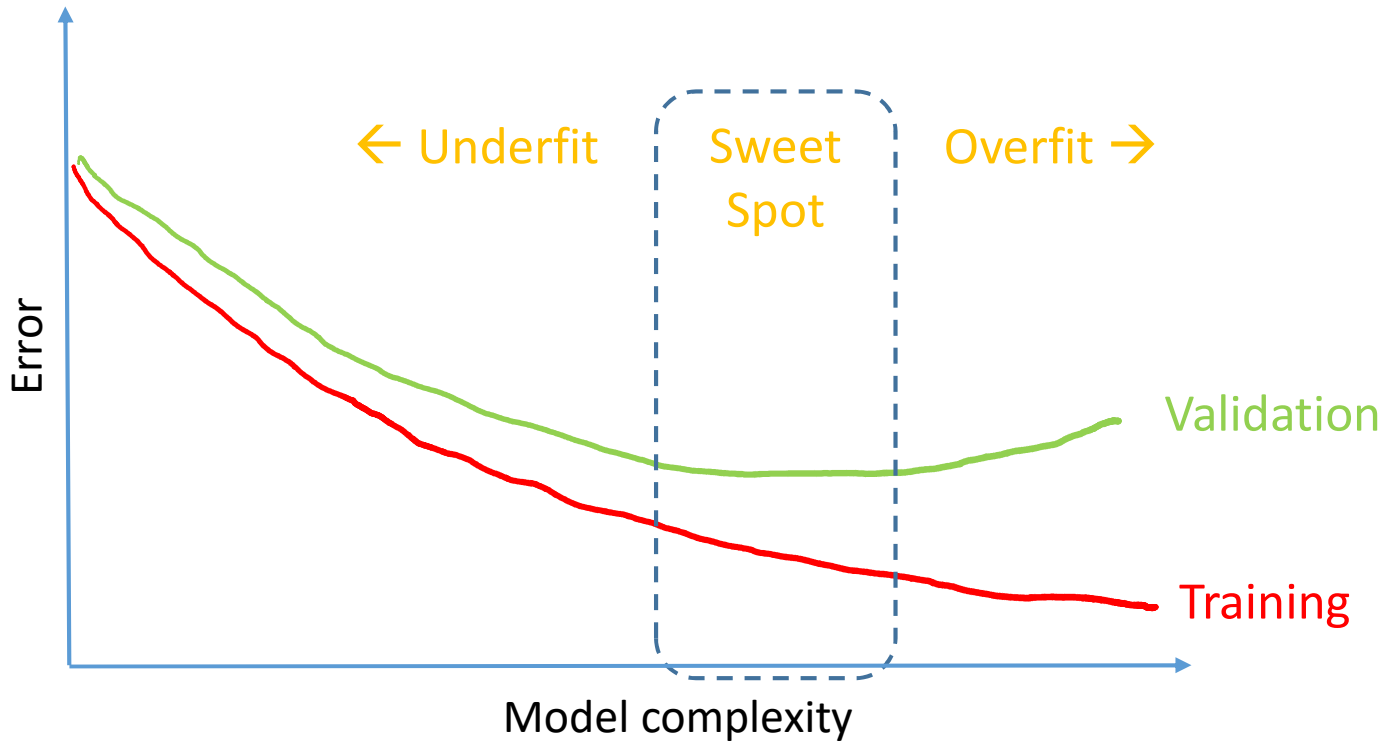
- Web of relationships

*Images courtesy: Pixabay.com*

# Some popular ML frameworks

<table>
<tr><th></th><th>Classification</th><th>Regression</th><th>Clustering</th><th>Dimension reduction</th></tr>
<tr><td rowspan="2">Vector</td><td>Logistic regression</td><td>Linear regression</td><td rowspan="2">K-means, Fuzzy C-means, DB-SCAN</td><td rowspan="2">PCA, k-PCA, LLE, ISOMAP</td></tr>
<tr><td colspan="2">SVM, RF, NN</td></tr>
<tr><td>Series, text</td><td colspan="2">RNN, LSTM, Transformer, 1-D CNN, HMM</td><td></td><td></td></tr>
<tr><td>Images</td><td colspan="2">2-D CNN, MRF</td><td></td><td></td></tr>
<tr><td>Video, MRI</td><td colspan="2">3-D CNN, CNN+LSTM, MRF</td><td></td><td></td></tr>
</table>

# Model choice and rigorous validation are very important



# Preparing data for training and validation

- Data splits:
  - Training → Used to optimize the parameters (e.g. random 70%)
  - Validation → Used to compare models (e.g. random 15%)
  - Testing → One final check after multiple rounds of validation (e.g. random 15%)
- Cross-validation:
  - K-folds: One fold for validation, K-1 folds for training
  - Rotate folds K times
  - Select framework (hyperparameters) best average performance
  - Re-train best framework on entire data
  - Test one final time on held-out data that was not a part of any fold

# ML can fail to perform in deployment

- Lack of training diversity: data had limited confounders
  - Single speaker, author, camera, background, accent, ethnicity, etc.
  - Data imbalance between high-value rare and more common examples
- Proxy label leak during training:
  - E.g. Only speakers A and B provide emotion “anger,” so ML confused their voice characteristics with “anger”
- Too much manual cleansing of training data
- Too little training data, and very complex models
- Concept drift: The assumptions behind training are no longer valid