MACHINE LEARNING WITH PYTHON

Themistoklis Diamantopoulos

Contents

- Part 1:
  - Introduction to Machine Learning
  - Classification with Decision Trees
  - Classification with Naïve Bayes
  - Classification with SVMs
- Part 2
  - Classification and Regression with kNN
  - Linear and Polynomial Regression
  - Feature Selection and Feature Extraction
  - Centroid-based and Connectivity-based Clustering

INTRODUCTION TO MACHINE LEARNING

What is Machine Learning?

- Subfield of Artificial Intelligence
- Term coined in 1959 by Arthur Samuel

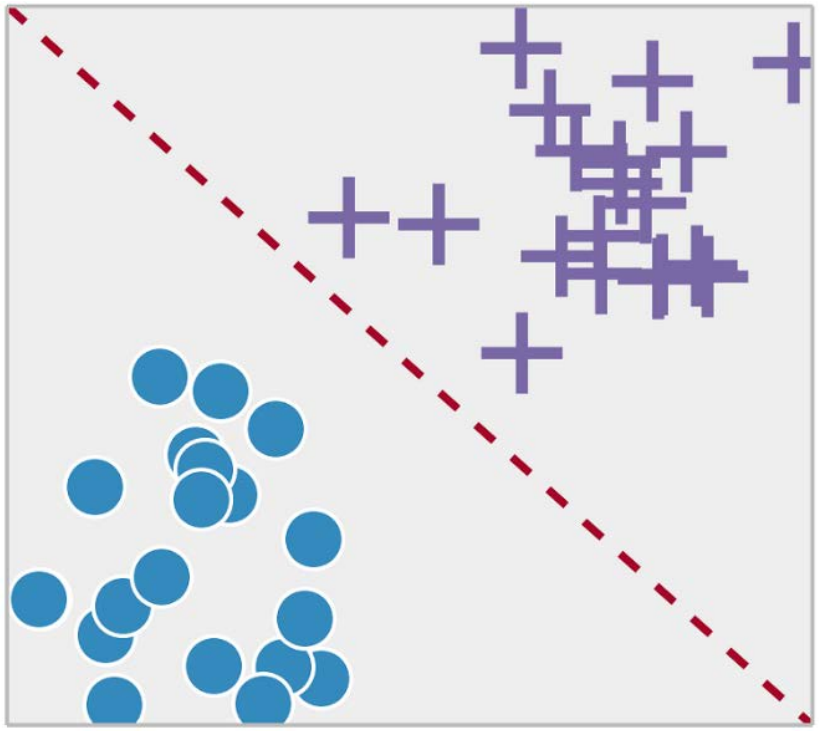
Progressively improve performance on a specific task with data, without being explicitly programmed

Types of Machine Learning tasks

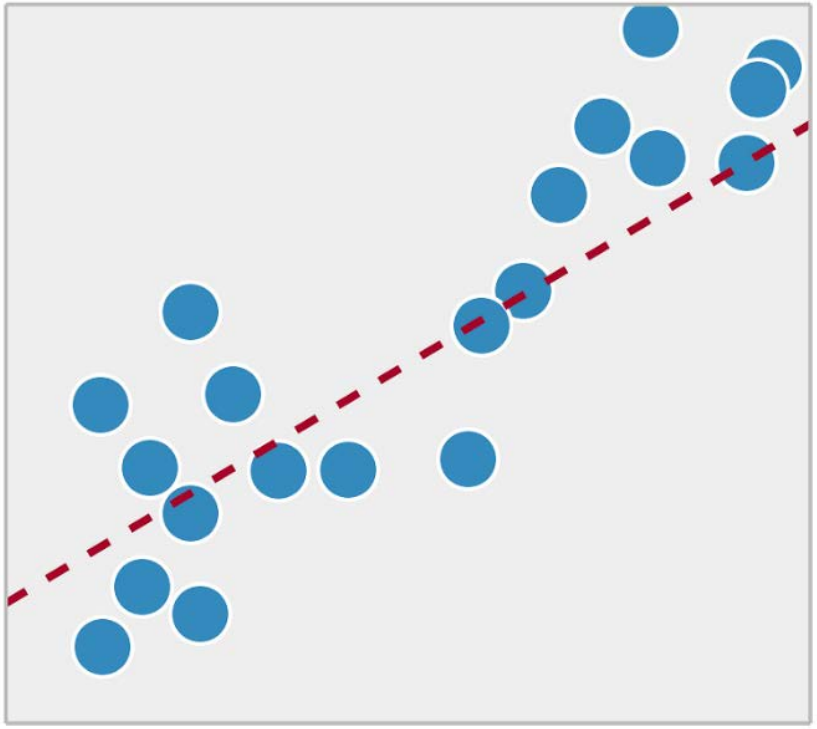
- Supervised Learning
  - Learn output based on input data
- Unsupervised Learning
  - Find structure in given data
- Reinforcement Learning
  - Learn from the environment

Supervised Learning tasks

Classification



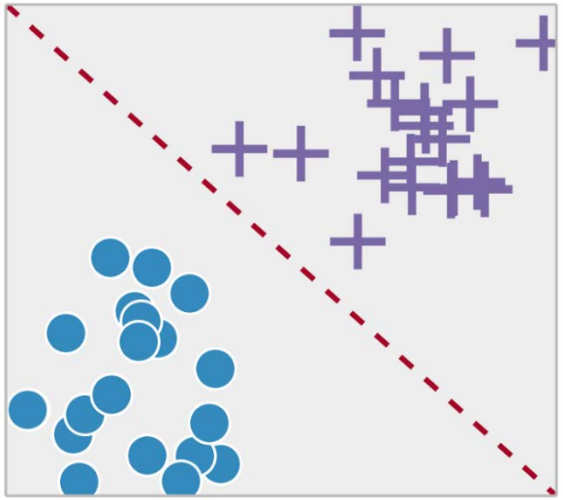
Regression



Source: https://towardsdatascience.com/supervised-vs-unsupervised-learning-14f68e32ea8d

Classification

- Classify data to 1, 2 or more classes
- Confusion Matrix



Predicted Class \ Actual Class	Positive	Negative
Positive	TP	FP
Negative	FN	TN
	P = TP + FN	N = FP + TN

- Evaluation Metrics
  - Accuracy = (TP + TN) / (P + N)
  - Precision = TP / (TP + FP)
  - Recall = TP / (TP + FN)

Regression

- Build a model that fits the data
- Actual (y_i) and predicted values ($\hat{y}_i$)
  - Mean Absolute Error

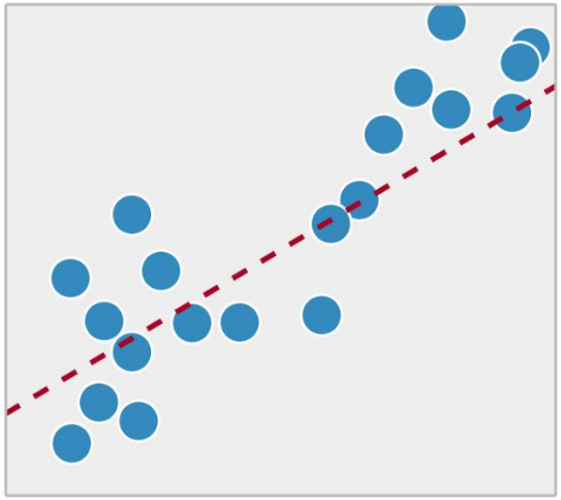
$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i - y_i\right|$$

  - Mean Squared Error

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2$$

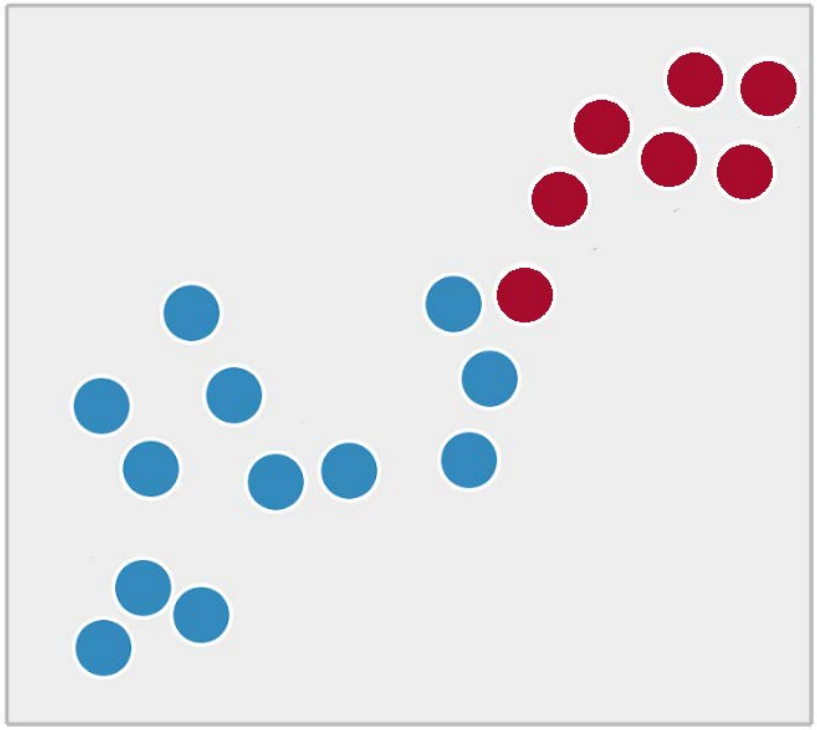
  - Coefficient of Determination

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad \text{where} \quad SS_{res} = \sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \quad \text{and} \quad SS_{tot} = \sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2$$

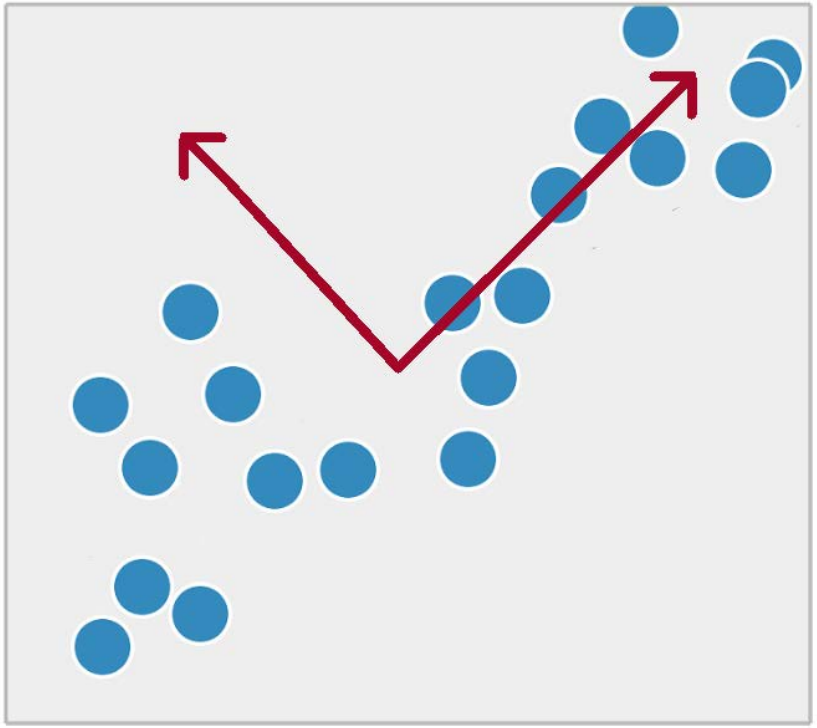


Unsupervised Learning tasks

Clustering



Dimensionality Reduction



Clustering

- Cluster data according to their features
- Evaluation Metrics
  - Cohesion (Within Cluster Sum of Squares)

$$SSE = WSS = \sum_{i} \sum_{x \in C_i} (x - m_i)^2$$

  - Separation (Between Cluster Sum of Squares)

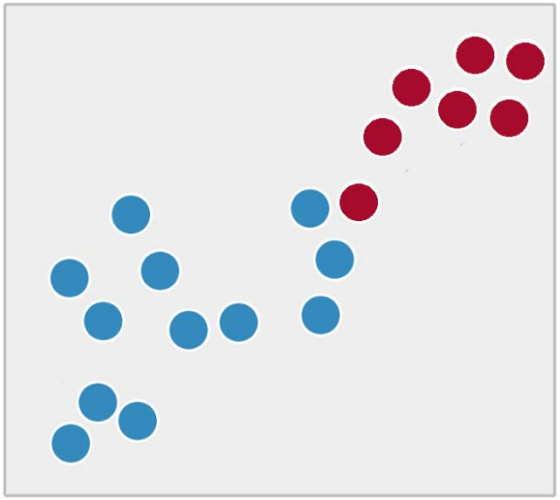
$$BSS = \sum_{i} |C_i| (m - m_i)^2$$

  - Silhouette

$$s = (b - a) / \max(a, b)$$

where a = average distance of i to the points in its cluster

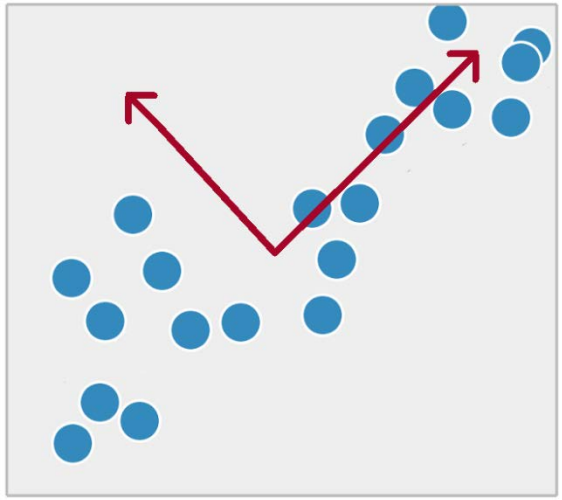
b = min(average distance of i to points in another cluster)

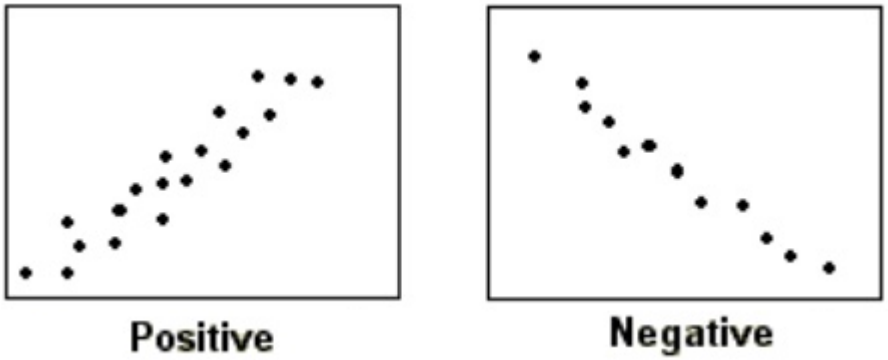


Source: https://www-users.cs.umn.edu/~kumar001/dmbook/slides/chap7_basic_cluster_analysis.pdf

Dimensionality Reduction

- Transform the data to extract useful information
  - Measure correlation
  - Maximize variance





- Evaluation Metrics
  - Percentage of Variance
  - Cumulative Percentage of Variance

Source: https://www.slideshare.net/rickwendell/principal-component-analysys

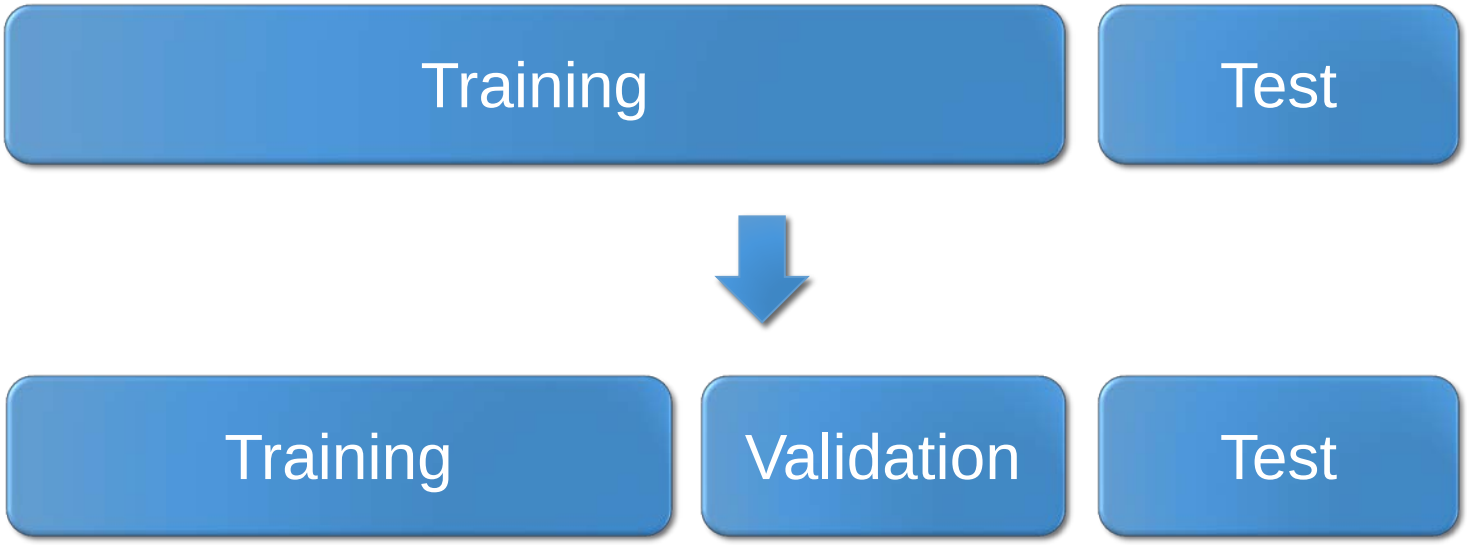
MACHINE LEARNING METHODOLOGY

Machine Learning Steps

- Data cleaning and preprocessing equally important with model selection and training

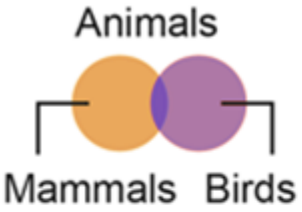
Data Splitting

- Use training data to train the model
  - Some data can be used to validate the model → validation set
  - Use folds of training data for validation → Cross-validation
- Evaluate the model on test data
  - Test set must not overlap with training data



Tribes of Machine Learning

Symbolists



Use symbols, rules, and logic to represent knowledge and draw logical inference

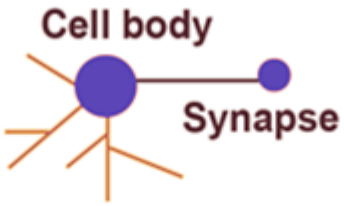
Favored algorithm
Rules and decision trees

Bayesians

Assess the likelihood of occurrence for probabilistic inference

Favored algorithm
Naive Bayes or Markov

Connectionists



Recognize and generalize patterns dynamically with matrices of probabilistic, weighted neurons

Favored algorithm
Neural networks

Evolutionaries

Generate variations and then assess the fitness of each for a given purpose

Favored algorithm
Genetic programs

Analogizers

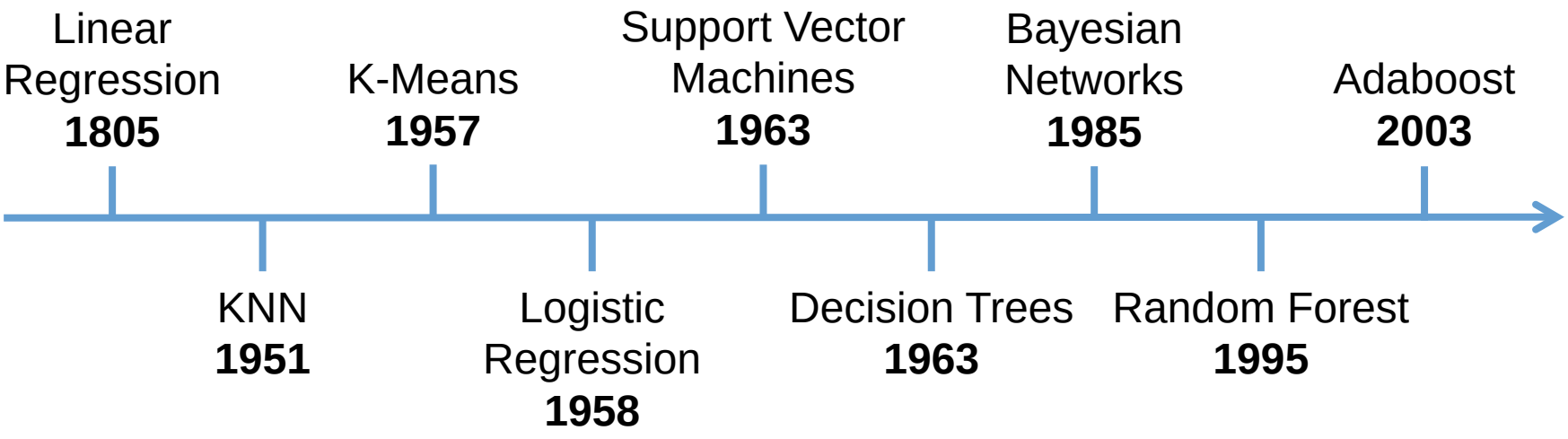
Optimize a function in light of constraints ("going as high as you can while staying on the road")

Favored algorithm
Support vectors

Source: http://www.franksworld.com/2018/09/05/evolution-of-machine-learning-infographic/

A brief history course

- 1960s: golden age of AI
- Predictive Statistical Algorithms



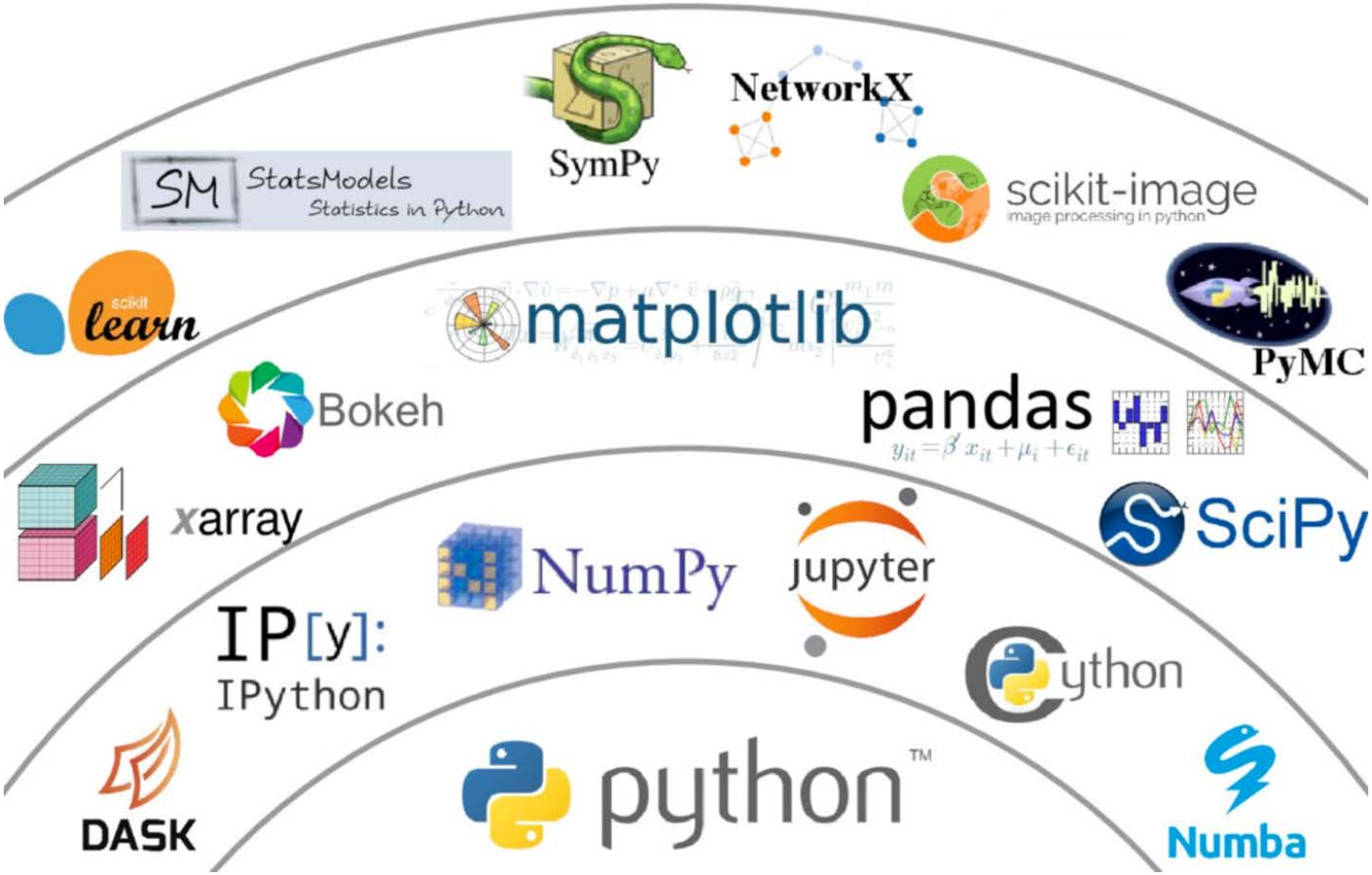
Source: https://governmenttechnologyinsider.com/how-machine-learning-can-save-us-from-cybercriminals/

Machine Learning Applications

Source: https://whatsthebigdata.com/2016/07/22/machine-learning-applications-by-industry/

MACHINE LEARNING WITH PYTHON

Set of Powerful Libraries



Source: https://www.datacamp.com/community/blog/python-scientific-computing-case

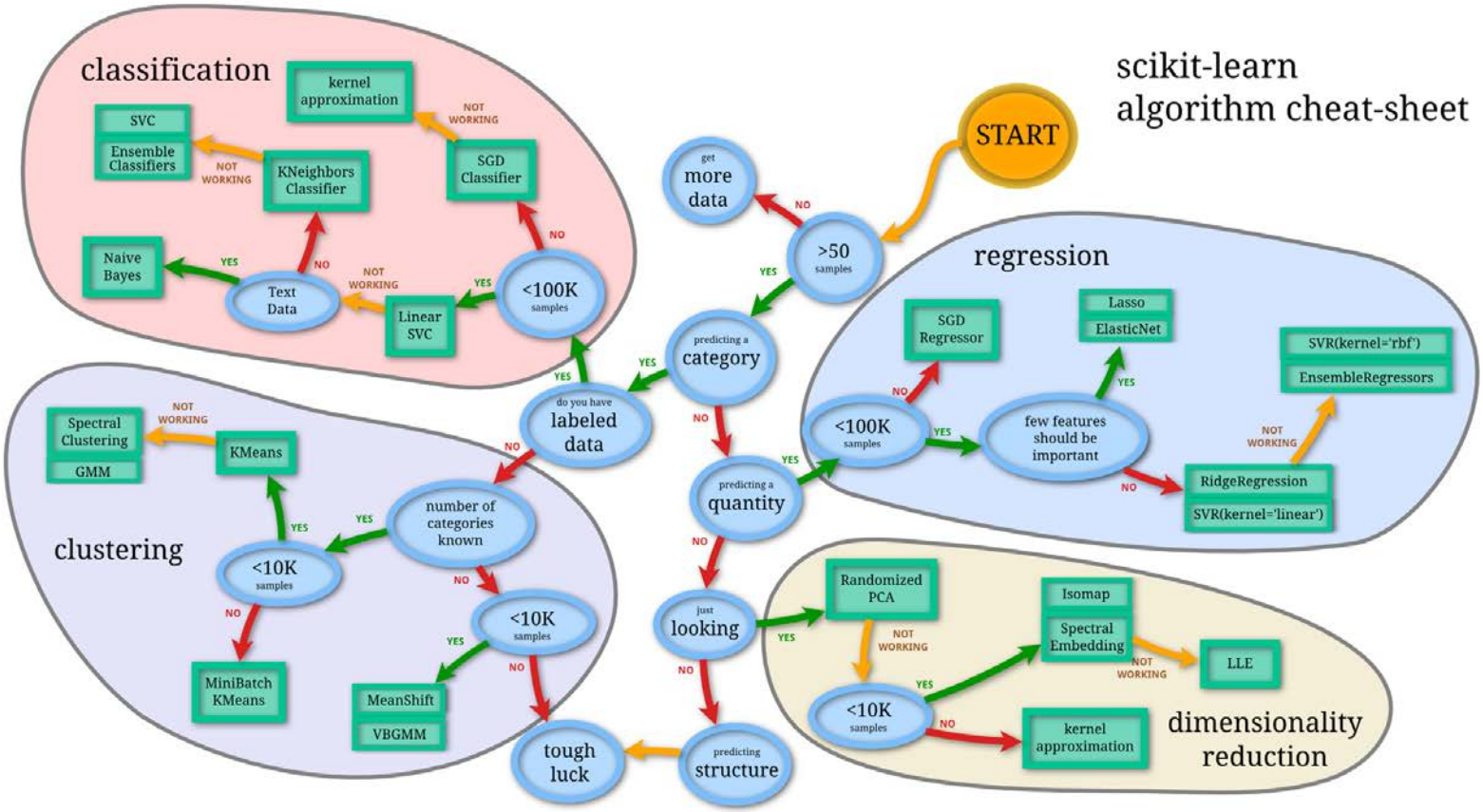
Machine Learning Libraries

- numpy
  - Arrays: universal point of reference in the python ML world
- pandas
  - Data manipulation made easy
- scipy
  - Basis of scientific computing
- scikit-learn
  - (Almost) all machine learning algorithms you will ever need
- matplotlib
  - Plot all of the above

… and all of these are seamlessly connected!

The Scikit-learn Library

- Almost any model you will ever need



Source: http://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

Scikit-learn Examples

- Very easy to use

```
from sklearn.svm import SVC
model = SVC(gamma=0.1)
model.fit(x_train, y_train)
y_pred = model.predict(x_test)
```

Create, Train, and Run Model

- Supports multiple data transformations
- And multiple evaluation metrics

```
from sklearn import metrics
print(metrics.classification_report(y_test, y_pred))
print(metrics.confusion_matrix(y_test, y_pred))
print(metrics.accuracy_score(y_test, y_pred))
```

Classification Evaluation Metrics

Source: https://www.slideshare.net/BenjaminBengfort/introduction-to-machine-learning-with-scikitlearn

Time for hands-on!