

Machine Learning

Machine learning using scikit-learn

Contents

- Use cases
- What is ML
- Pipeline steps
- Learning?
- Many ML models
- Evaluating
- Examples
- Hackathon suggestions

What can machine learning do?

- Automate - Save \$\$\$



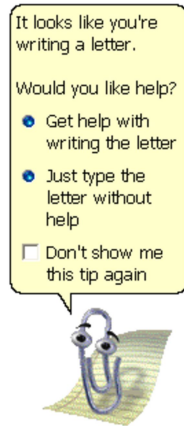
What can machine learning do?

- Detect anomalies (security, fraud, ...)



What can machine learning do?

- Healthcare - Save lives
- GeriMedica - Second Opinion

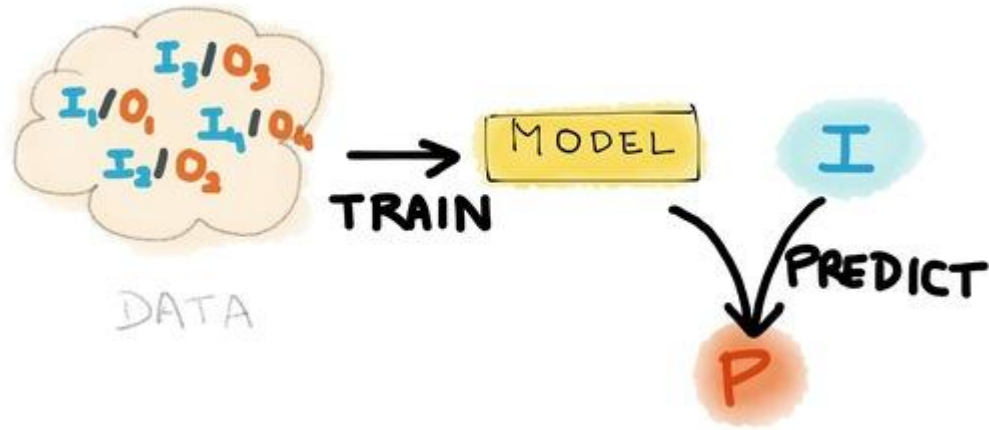


What can machine learning do?

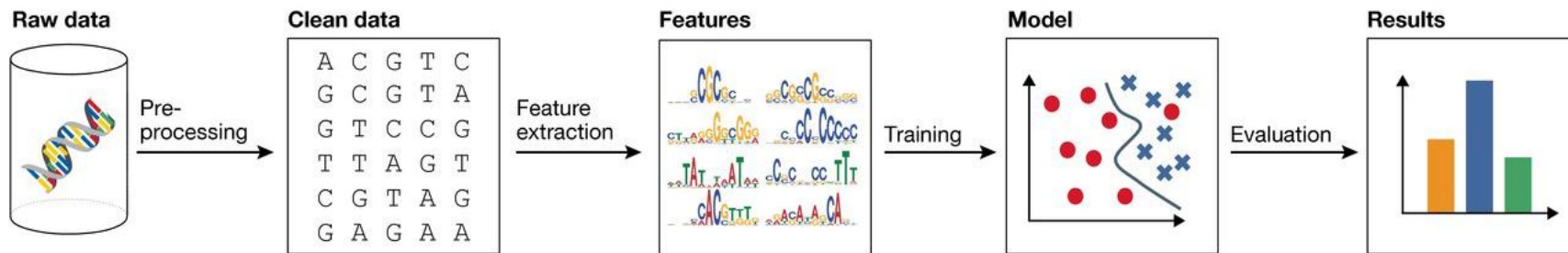
- Customer segmentation - targeted marketing - make more \$\$\$



What is machine learning?




Machine learning pipeline



Cleaning data

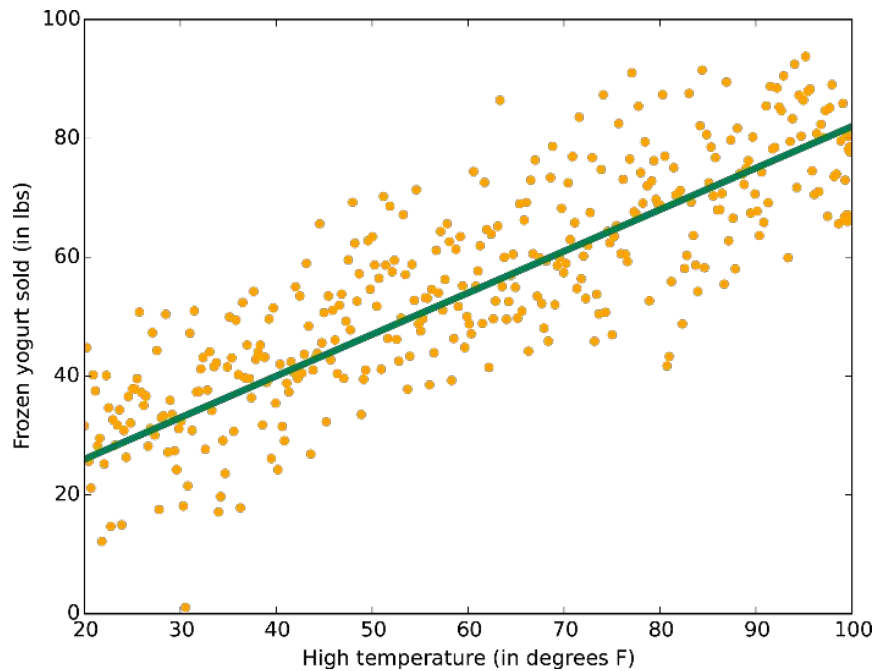
- Clean data
 - Fill missing values
 - Drop incomplete rows



	I	J	K	L	M	N	O	P	h.s
1	kitchen	meal	pc	dog	nose	newspaper	tail	to	h.s
2		5	5	1	1	5	1	5	2
3		5	5	3	5	4	1	5	1
4		5	5	4	5	5	4	5	2
5		5	5	5	5	5	1	5	1
6		5	5	2	5	5	3		2
7		2	2	1	3	2	1		1
8		5	5	5		5	3		1
9		2	2	2	3	2	2		2
10		3	5	1	3	5	1		1
11		5	5	2	5	5	1		2
12		1	3	1	5	5	1		1
13		4	5	4	5	5	2		2
14		5	5	1	4	5	1		1
15		5	5	3		5	1	5	1
16		5	5	1		5	1	5	2
17		5	5	3	3	5	2	5	1
18		5	4	1	5	5	1	5	1
19		5	3	2	4	5	2	5	2
20		4	4	4		3	1	4	1

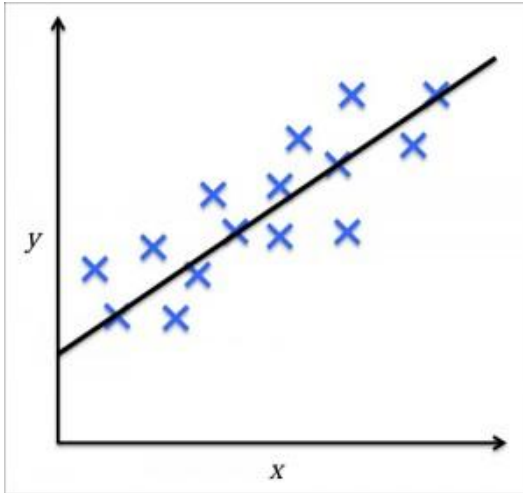
Learning a function

- Learn function $f(x) \rightarrow y$

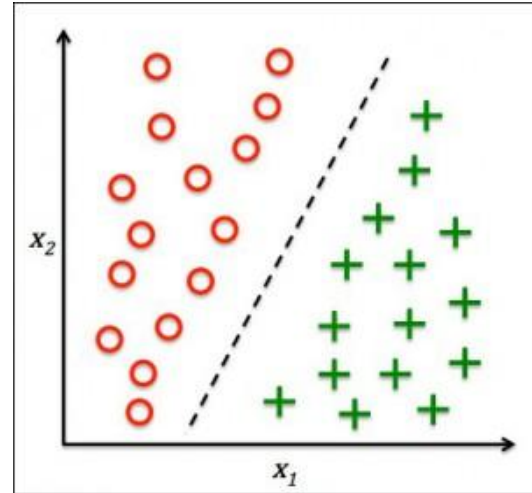


Types of algorithms

Regression

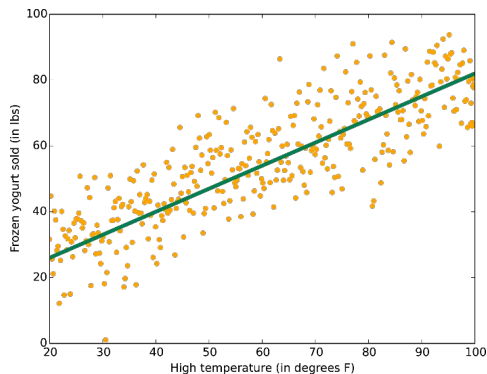
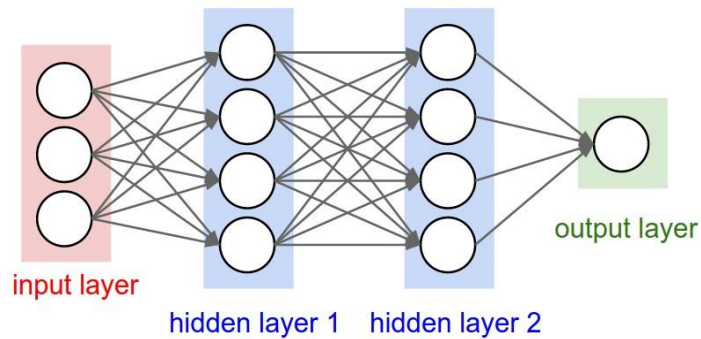
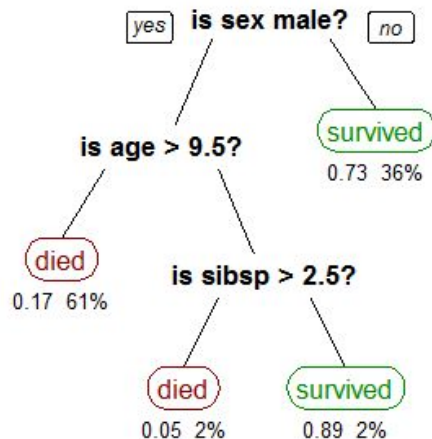


Classification



Types of algorithms

- Different approaches (models)



How does learning work?

- $y = \text{Intercept} + \text{Slope} * x$
 - Learn *Intercept*, *Slope*

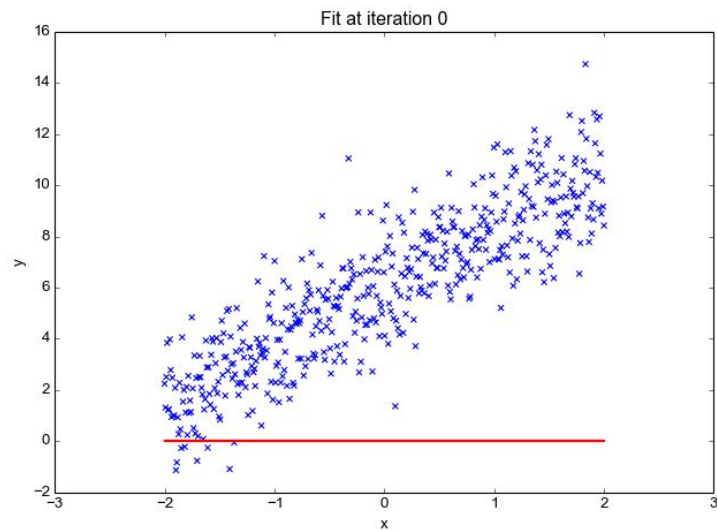
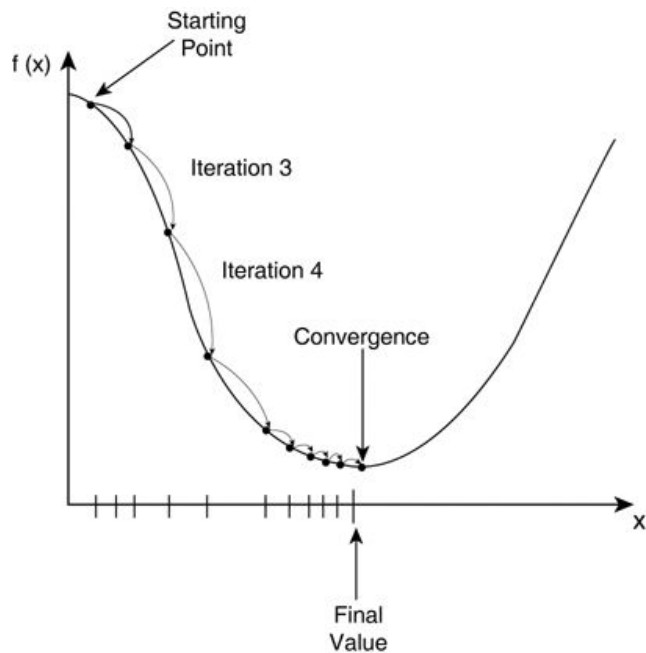
- Loss - e.g. MSE

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \tilde{y}_i)^2$$



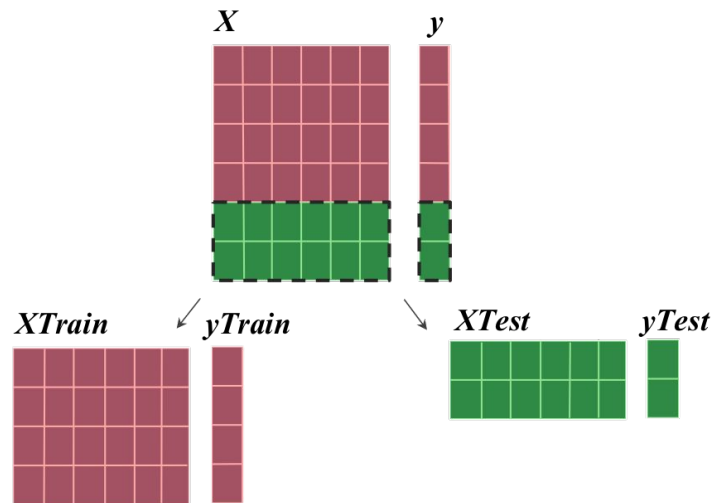
How does learning work?

- Training : Gradient descent



Evaluating

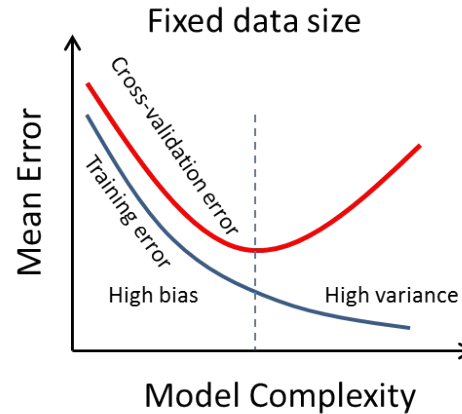
Simple: train/test split



Better: KFold Cross validation



What could possibly go wrong?



Python & scikit-learn

```
from sklearn import datasets
from sklearn.tree import DecisionTreeClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

X,y = datasets.load_iris(return_X_y=True)

X_train, X_test, y_train, y_test = train_test_split(X, y)

model = DecisionTreeClassifier(min_samples_split=8, min_samples_leaf=4)

model.fit(X_train, y_train)

y_predicted = model.predict(X_test)

print("Score: %.4f " % accuracy_score(y_test, y_predicted))
```

Example

- [Analysis, features & model NYC taxi trips](#)
- [Analysis & model cycle share](#)

Hackathon suggestions

- Find research questions / hypotheses
 - Predict number of trips (weather?)
 - Predict trip duration (weather, sex, age, ...)
 - Predict popular routes (week/weekend, ...)
- Prepare data
- Train & evaluate models
- Try, learn & improve !

Ready ?

- Let's start hacking !