Learning scikit learn

– An Introduction to Machine Learning in Python

PyData Chicago 2016
Chicago, The University of Illinois • August 26, 2016

Sebastian Raschka

Links & Info

Tutorial Material on GitHub:

https://github.com/rasbt/pydata-chicago2016-ml-tutorial

Contact:

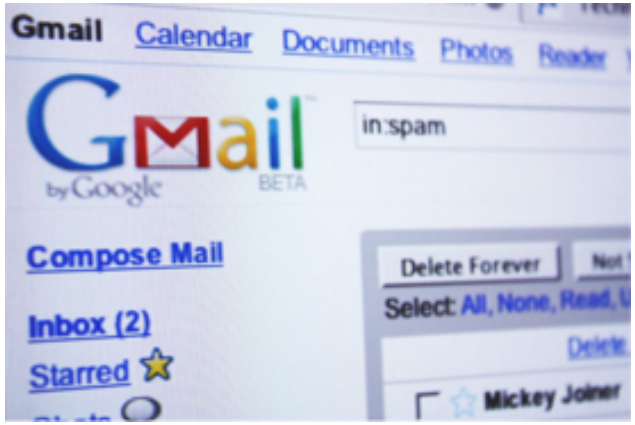
- E-mail: mail@sebastianraschka.com
- Website: http://sebastianraschka.com
- Twitter: @rasbt
- GitHub: rasbt

Let's Not Stress!

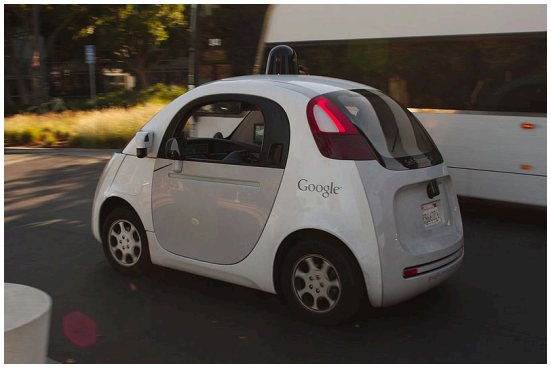
This is an introductory tutorial, and we are here to learn!

Please ask questions!

What can Machine Learning do for us?



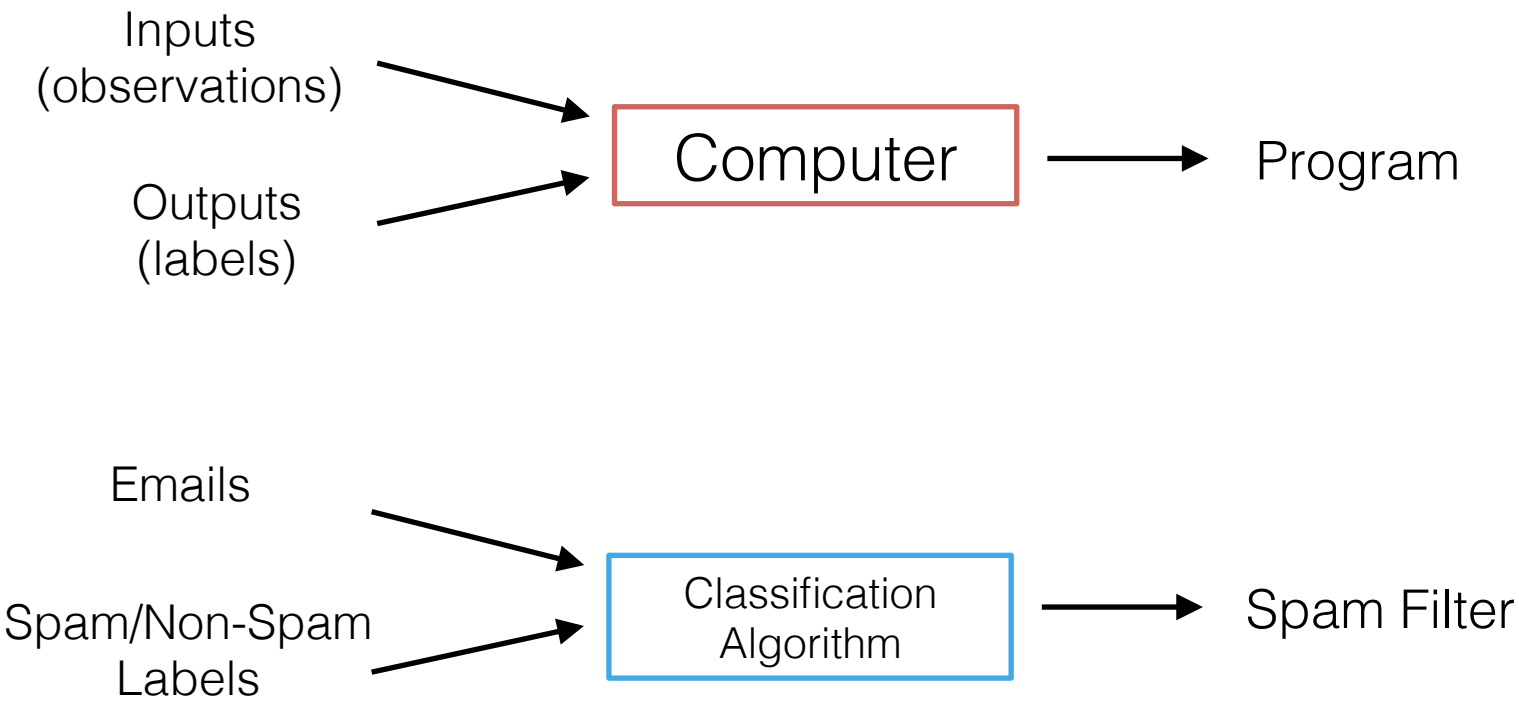
https://flic.kr/p/5BLW6G [CC BY 2.0]



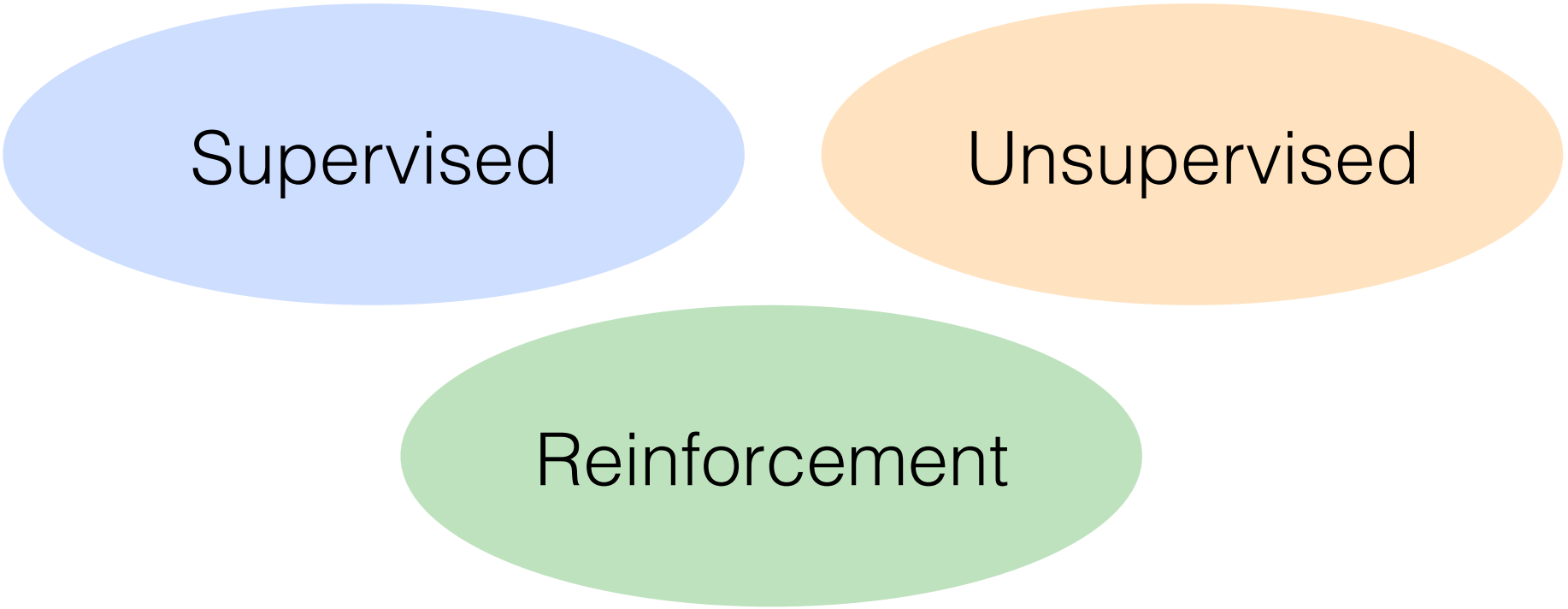
https://commons.wikimedia.org/wiki/File:Google_self_driving_car_at_the_Googleplex.jpg

Photo by Michasel Shick, CC BY-SA 4.0 lit

What is Machine Learning?



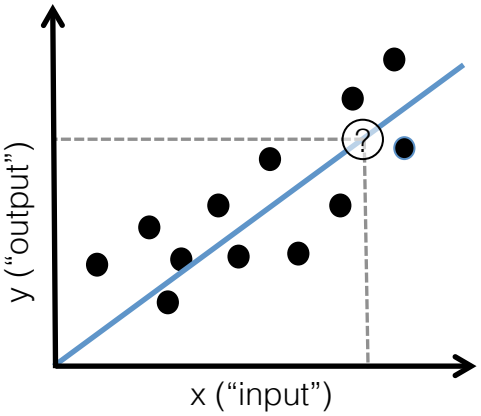
3 Types of Learning



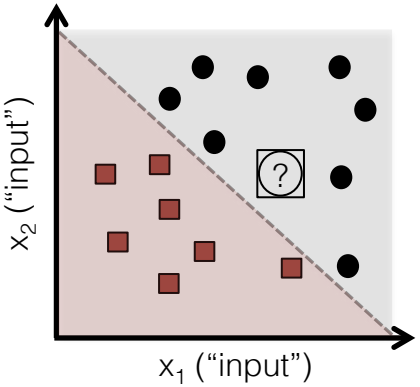
Working with Labeled Data

Supervised Learning

Regression



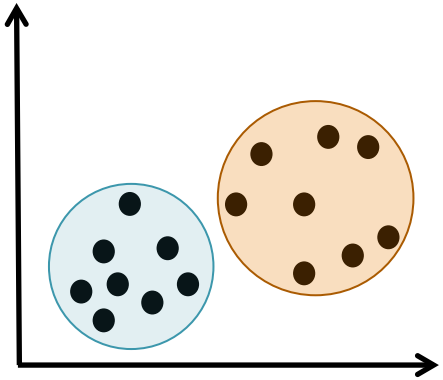
Classification



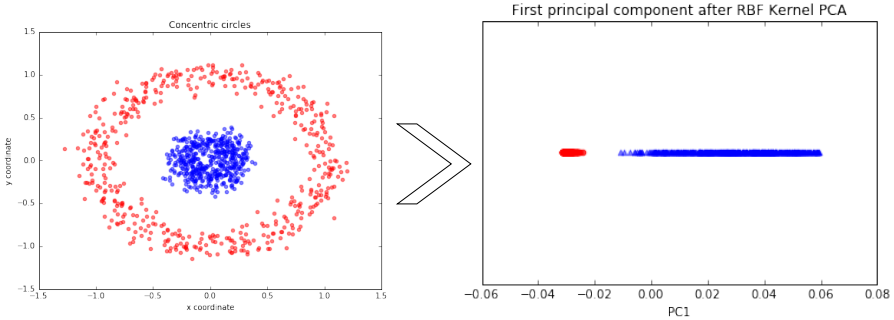
Working with Unlabeled Data

Unsupervised Learning

Clustering



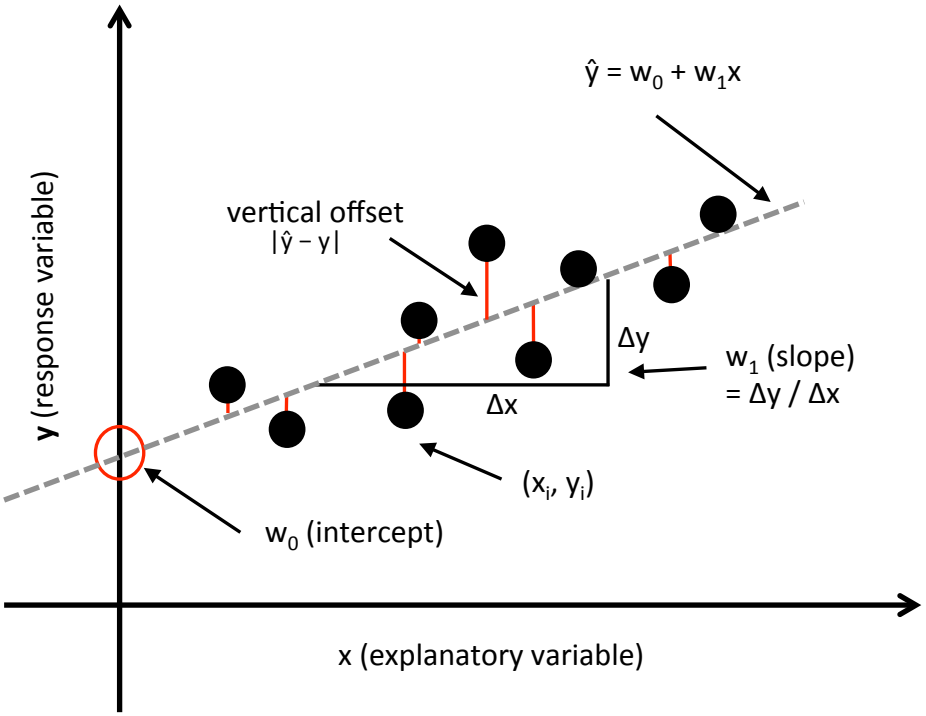
Compression



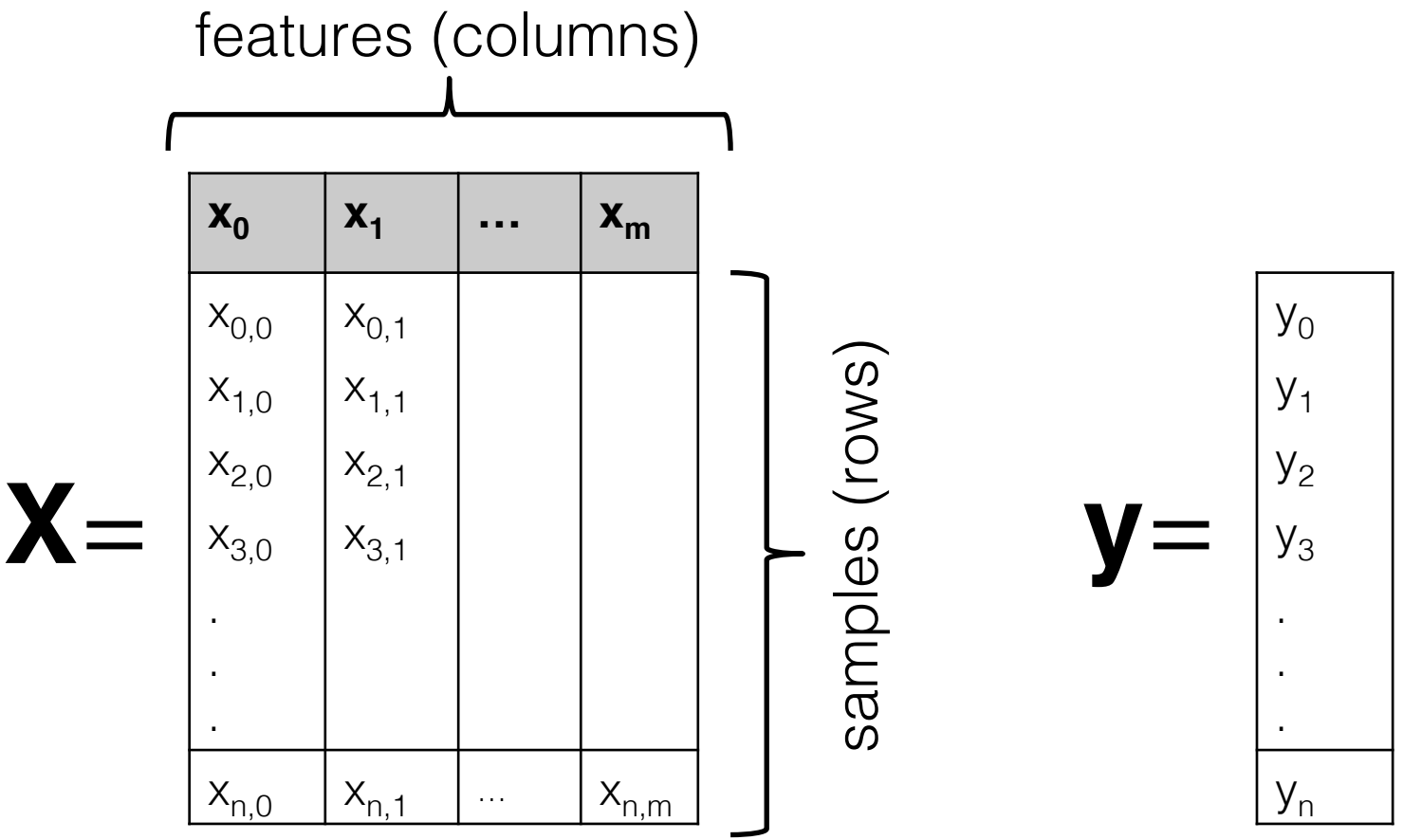
Topics

1. Introduction to Machine Learning
2. **Linear Regression**
3. Introduction to Classification
4. Feature Preprocessing & scikit-learn Pipelines
5. Dimensionality Reduction: Feature Selection & Extraction
6. Model Evaluation & Hyperparameter Tuning

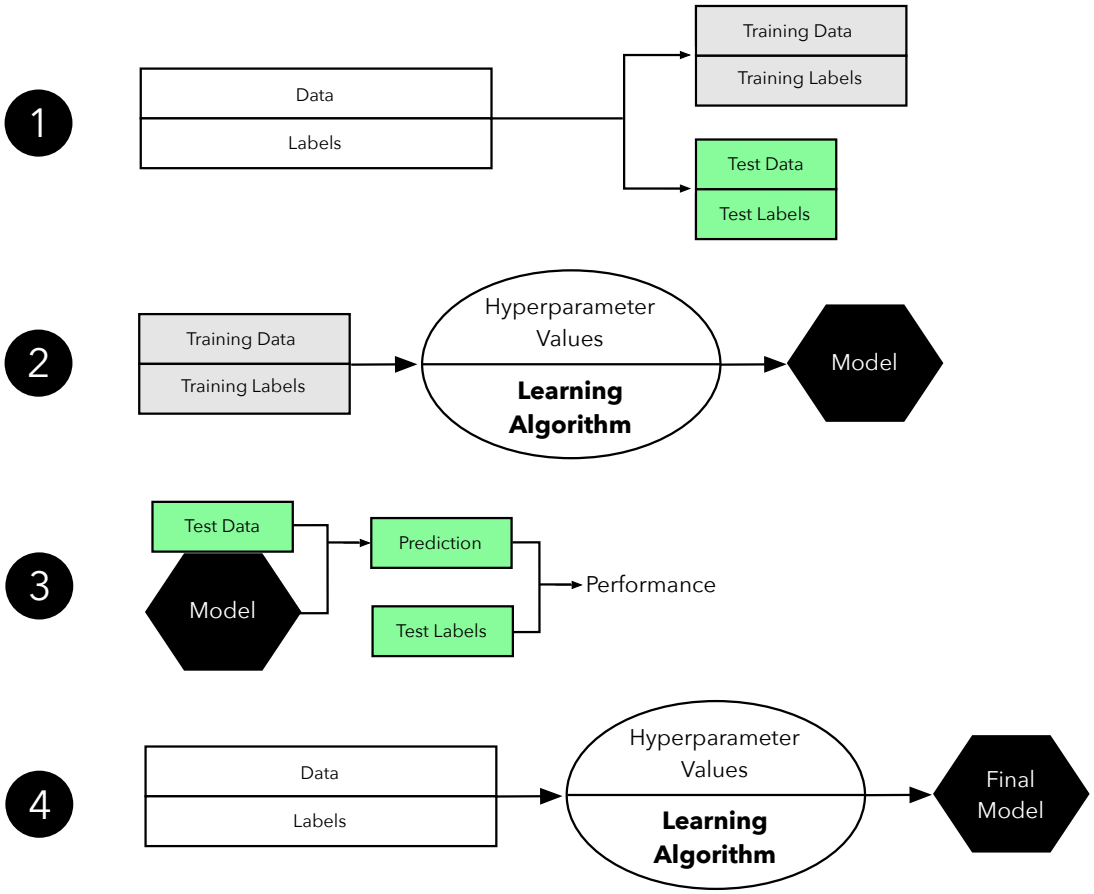
Simple Linear Regression



Data Representation



“Basic” Supervised Learning Workflow



Code Examples

➡ Jupyter Notebook

Topics

1. Introduction to Machine Learning
2. Linear Regression
3. **Introduction to Classification**
4. Feature Preprocessing & scikit-learn Pipelines
5. Dimensionality Reduction: Feature Selection & Extraction
6. Model Evaluation & Hyperparameter Tuning

Scikit-learn API

```
class SupervisedEstimator(...):
    def __init__(self, hyperparam, ...):
        ...
    def fit(self, X, y):
        ...
        return self
    def predict(self, X):
        ...
        return y_pred
    def score(self, X, y):
        ...
        return score
    ...
```

Iris Dataset

Iris-Setosa

Iris-Setosa

Iris-Versicolor

Iris Dataset

features (columns)

X=

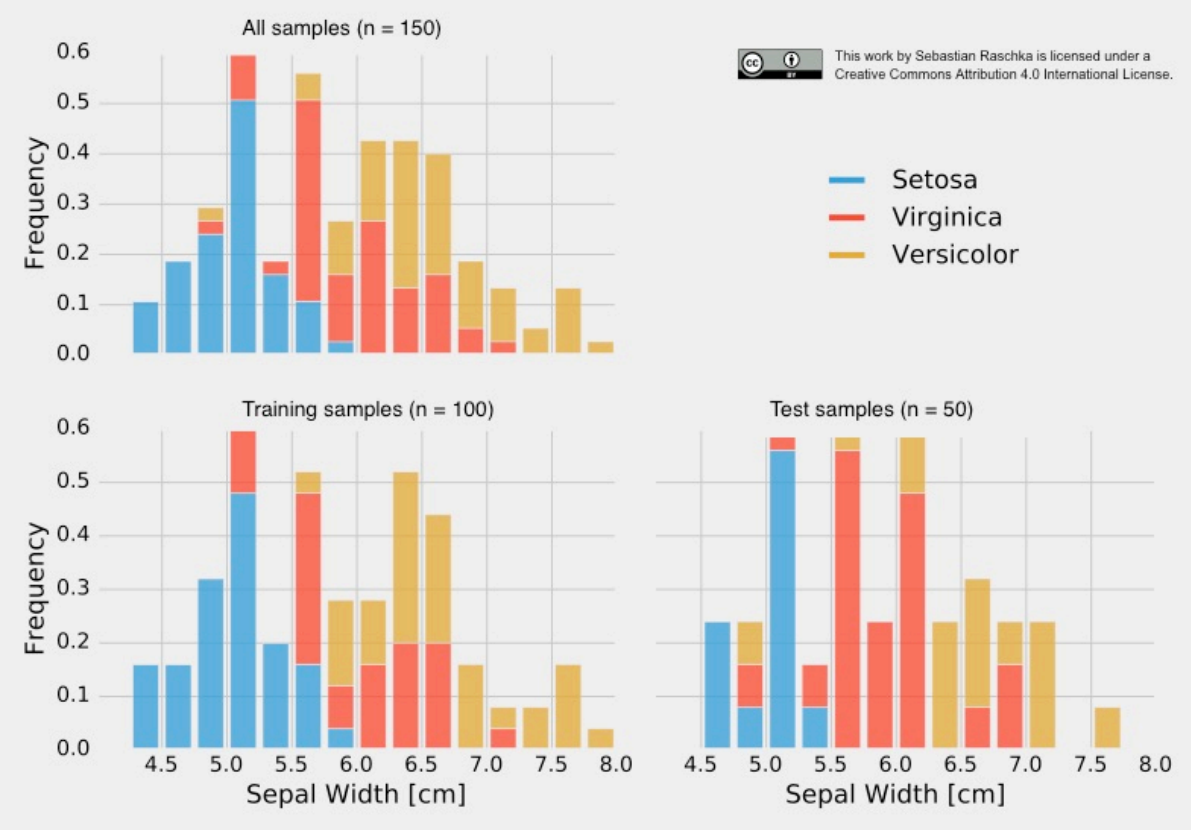
	sepal length [cm]	sepal width [cm]	petal length [cm]	petal width [cm]
1	5.1	3.5	1.4	0.2
2	4.9	3.0	1.4	0.2
50	6.4	3.5	4.5	1.2
	.			
	.			
	.			
150	5.9	3.0	5.0	1.8

samples (rows)

y=

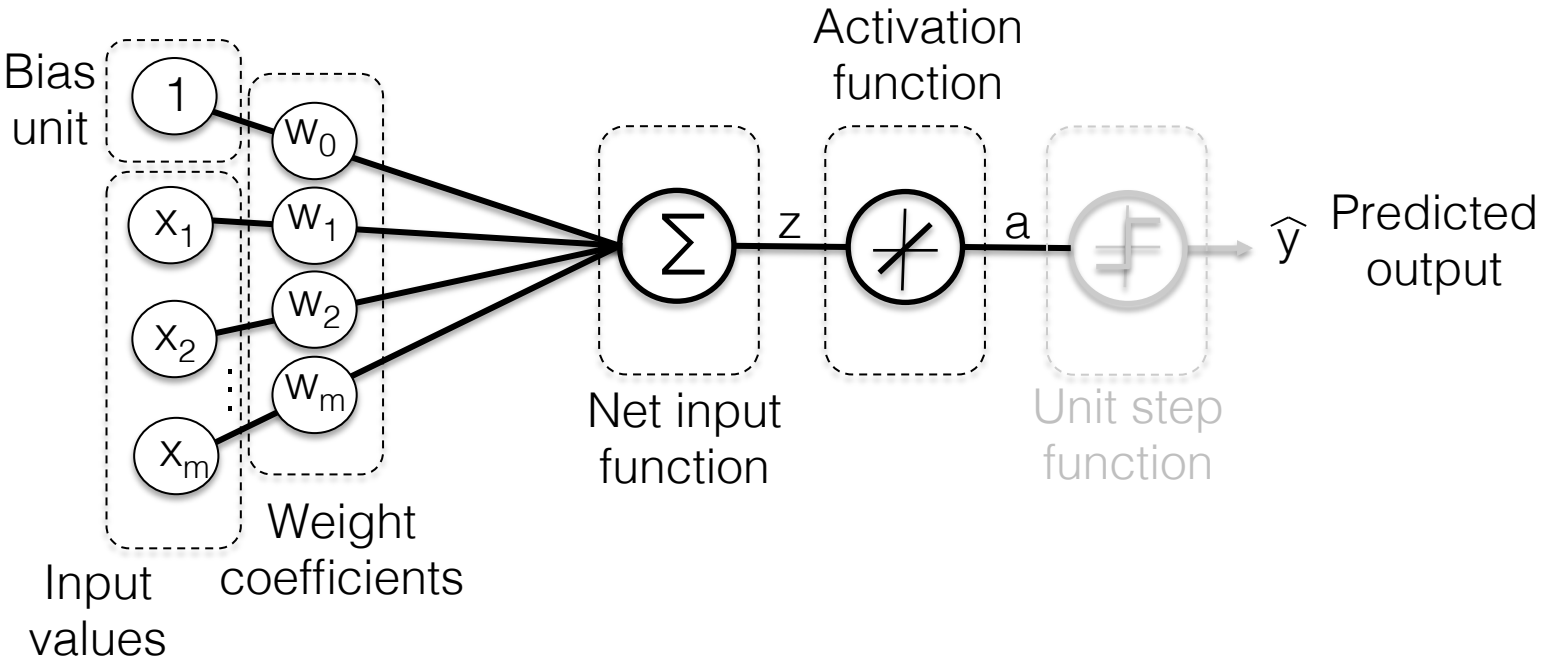
setosa
setosa
versicolor
.
.
.
virginica

Note about Non-Stratified Splits

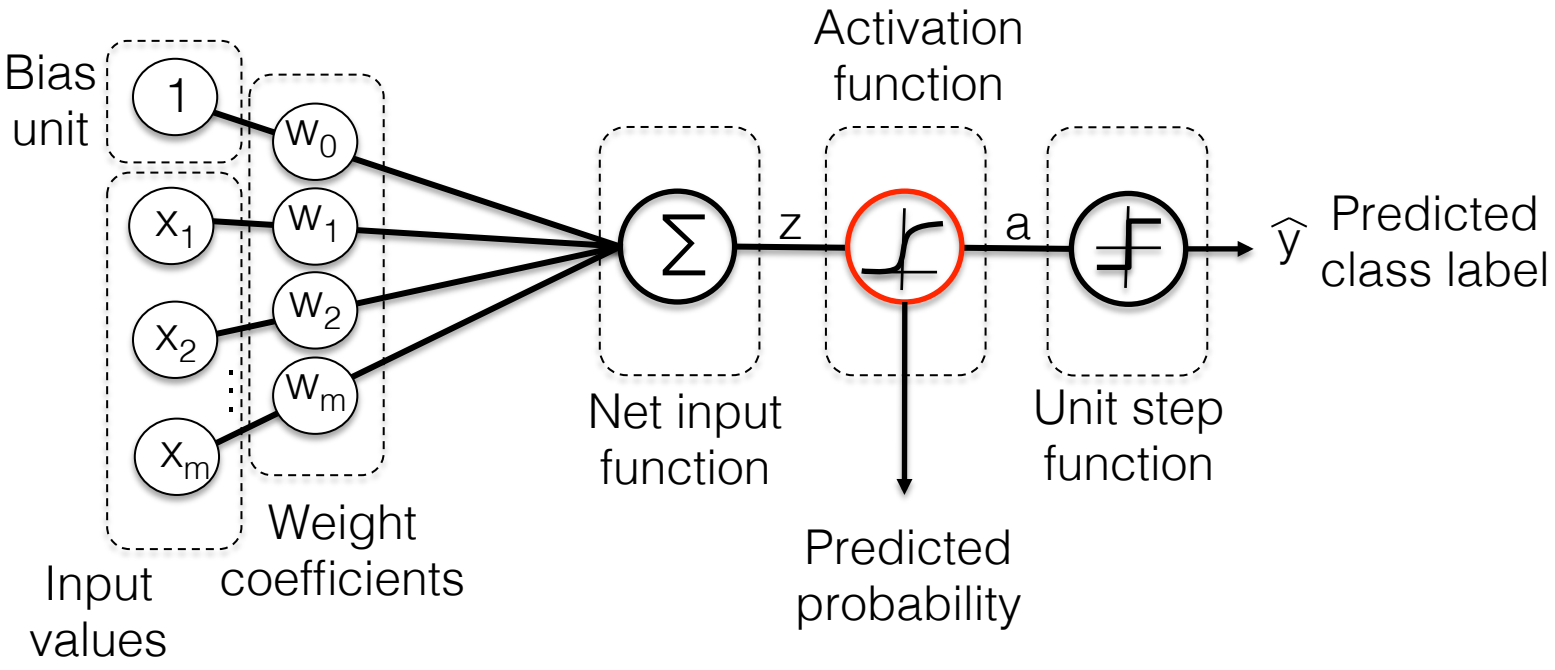


- training set → 38 x Setosa, 28 x Versicolor, 34 x Virginica
- test set → 12 x Setosa, 22 x Versicolor, 16 x Virginica

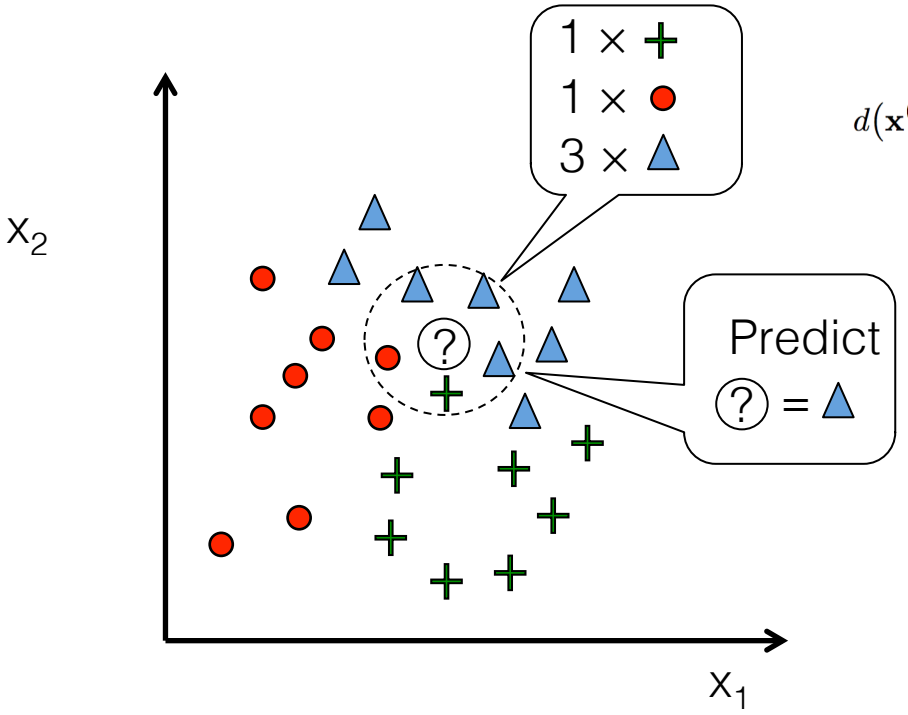
Linear Regression Recap



Logistic Regression, a Generalized Linear Model



A “Lazy Learner:” K-Nearest Neighbors Classifier



$$d(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \sqrt[p]{\sum_k |x_k^{(i)} - x_k^{(j)}|^p}$$

Code Examples

➡ Jupyter Notebook

Topics

1. Introduction to Machine Learning
2. Linear Regression
3. Introduction to Classification
4. **Feature Preprocessing & scikit-learn Pipelines**
5. Dimensionality Reduction: Feature Selection & Extraction
6. Model Evaluation & Hyperparameter Tuning

Categorical Variables

color	size	price	class label
red	M	$10.49	0
blue	XL	$15.00	1
green	L	$12.99	1

Encoding Categorical Variables

color	size	price	class label
red	M	$10.49	0
blue	XL	$15.00	1
green	L	$12.99	1

red	blue	green
1	0	0
0	1	0
0	0	1

size
0
2
1

Feature Normalization

Min-max scaling

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Z-score standardization

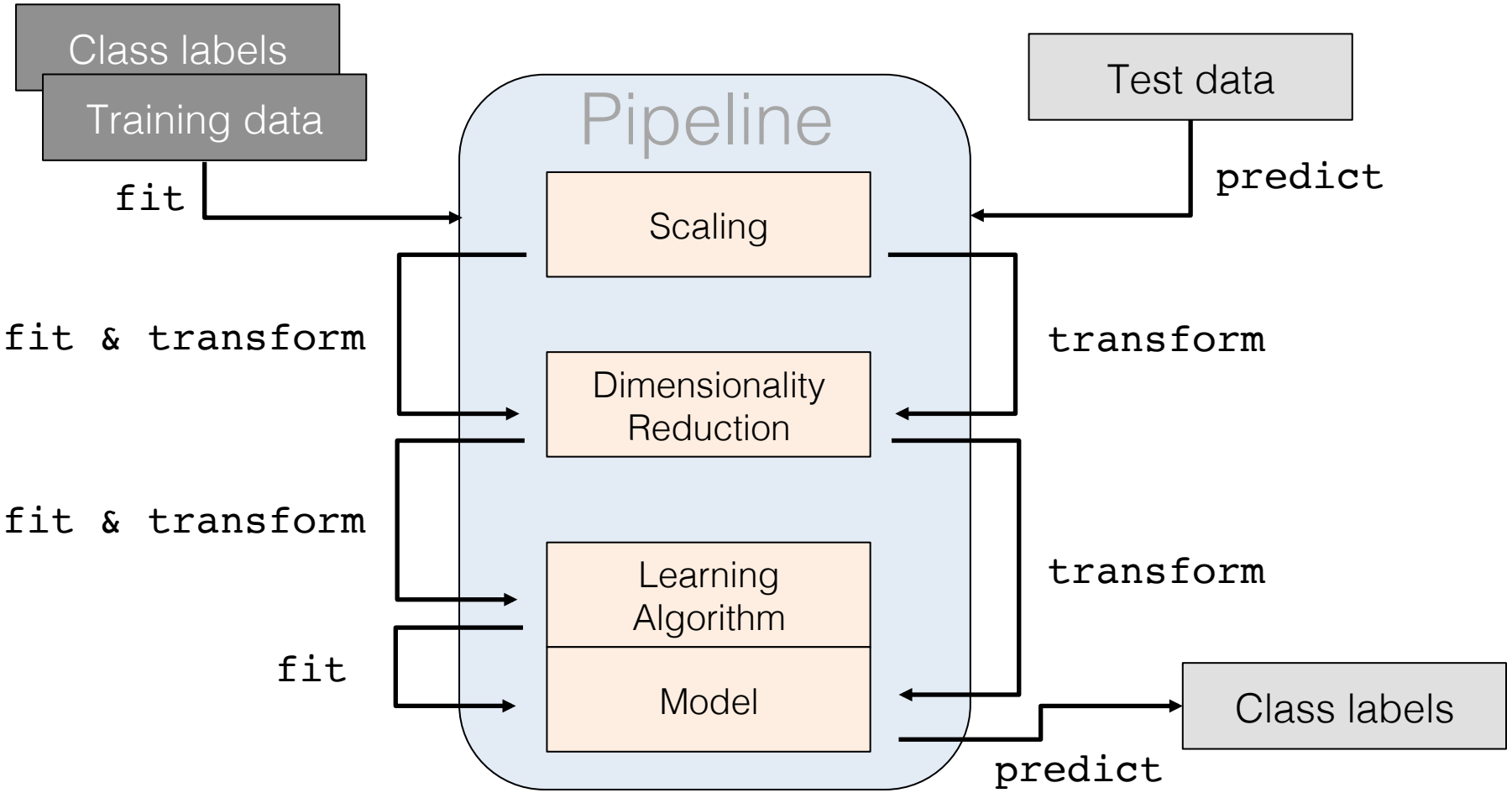
$$z = \frac{x - \mu}{\sigma}$$

feature	minmax	z-score
1.0	0.0	-1.46385
2.0	0.2	-0.87831
3.0	0.4	-0.29277
4.0	0.6	0.29277
5.0	0.8	0.87831
6.0	1.0	1.46385

Scikit-learn API

```
class UnsupervisedEstimator(...):
    def __init__(self, ...):
        ...
    def fit(self, X):
        ...
        return self
    def transform(self, X):
        ...
        return X_transf
 def predict(self, X):
        ...
        return pred
```

Scikit-learn Pipelines



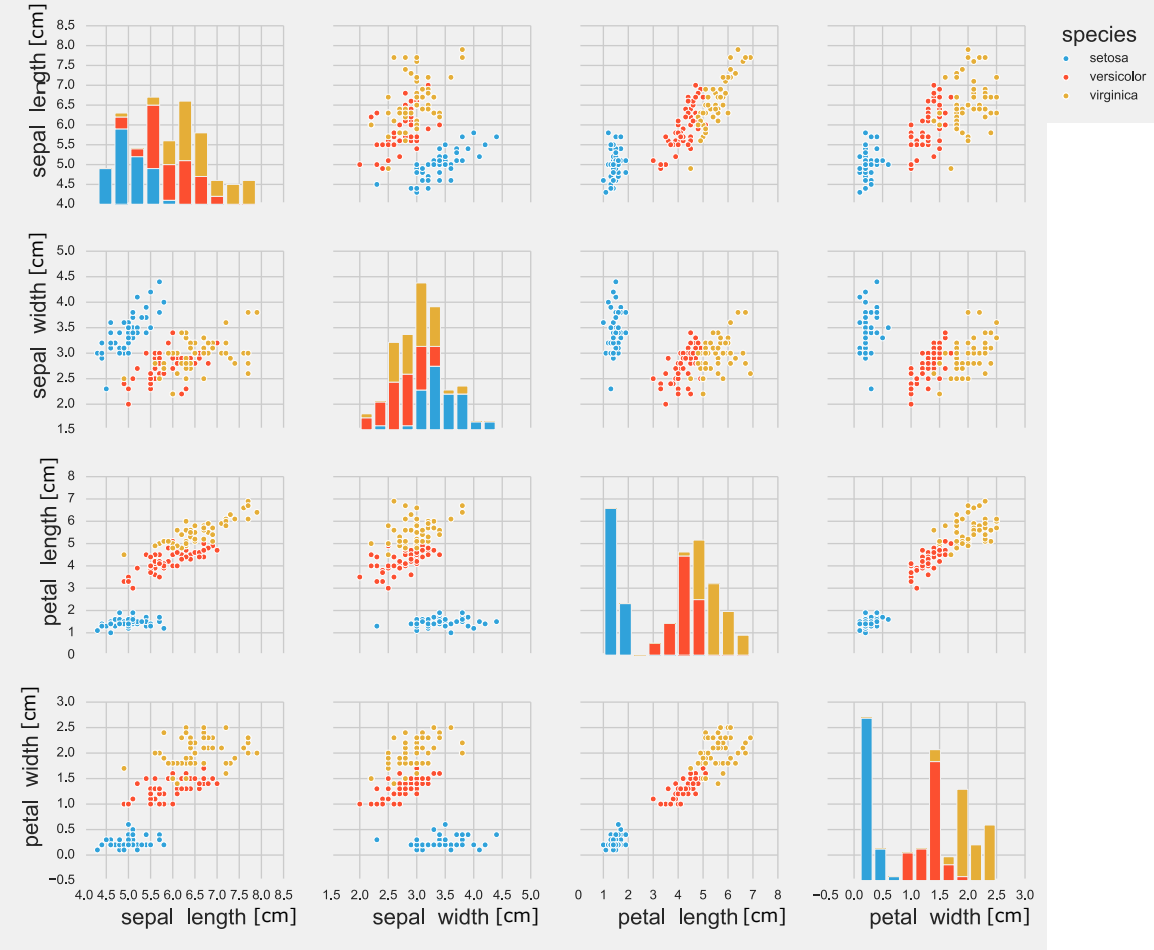
Code Examples

➡ Jupyter Notebook

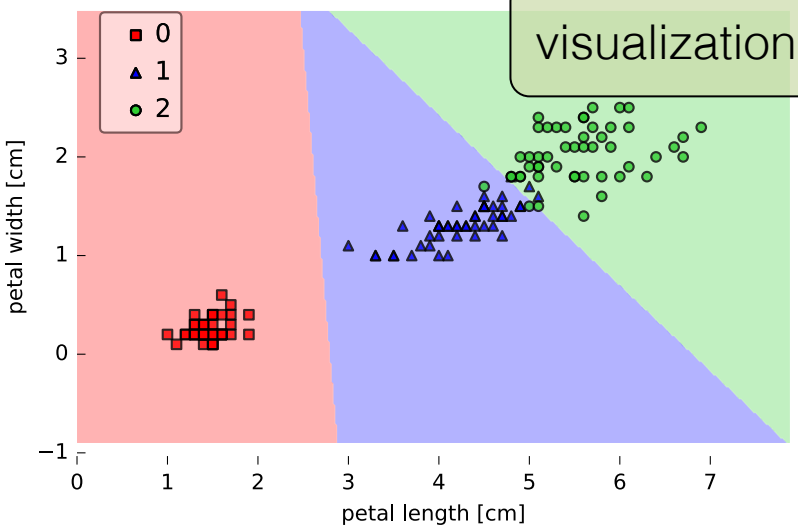
Topics

1. Introduction to Machine Learning
2. Linear Regression
3. Introduction to Classification
4. Feature Preprocessing & scikit-learn Pipelines
5. **Dimensionality Reduction: Feature Selection & Extraction**
6. Model Evaluation & Hyperparameter Tuning

Dimensionality Reduction – why?

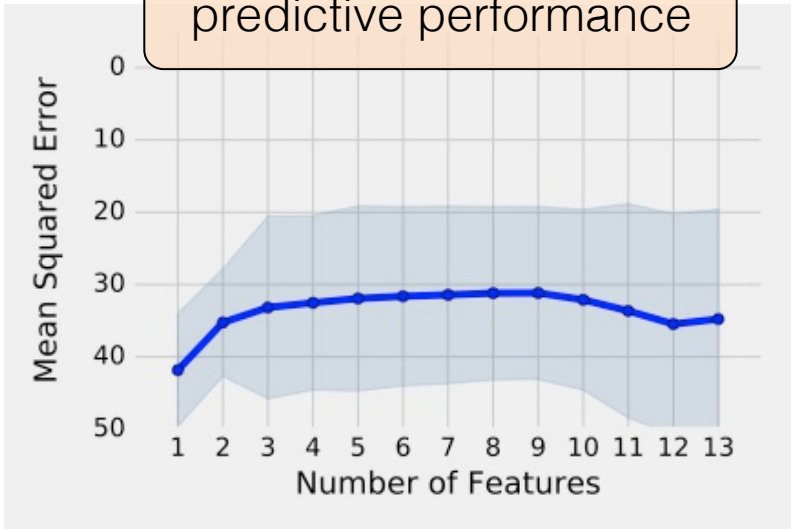


Dimensionality Reduction – why?



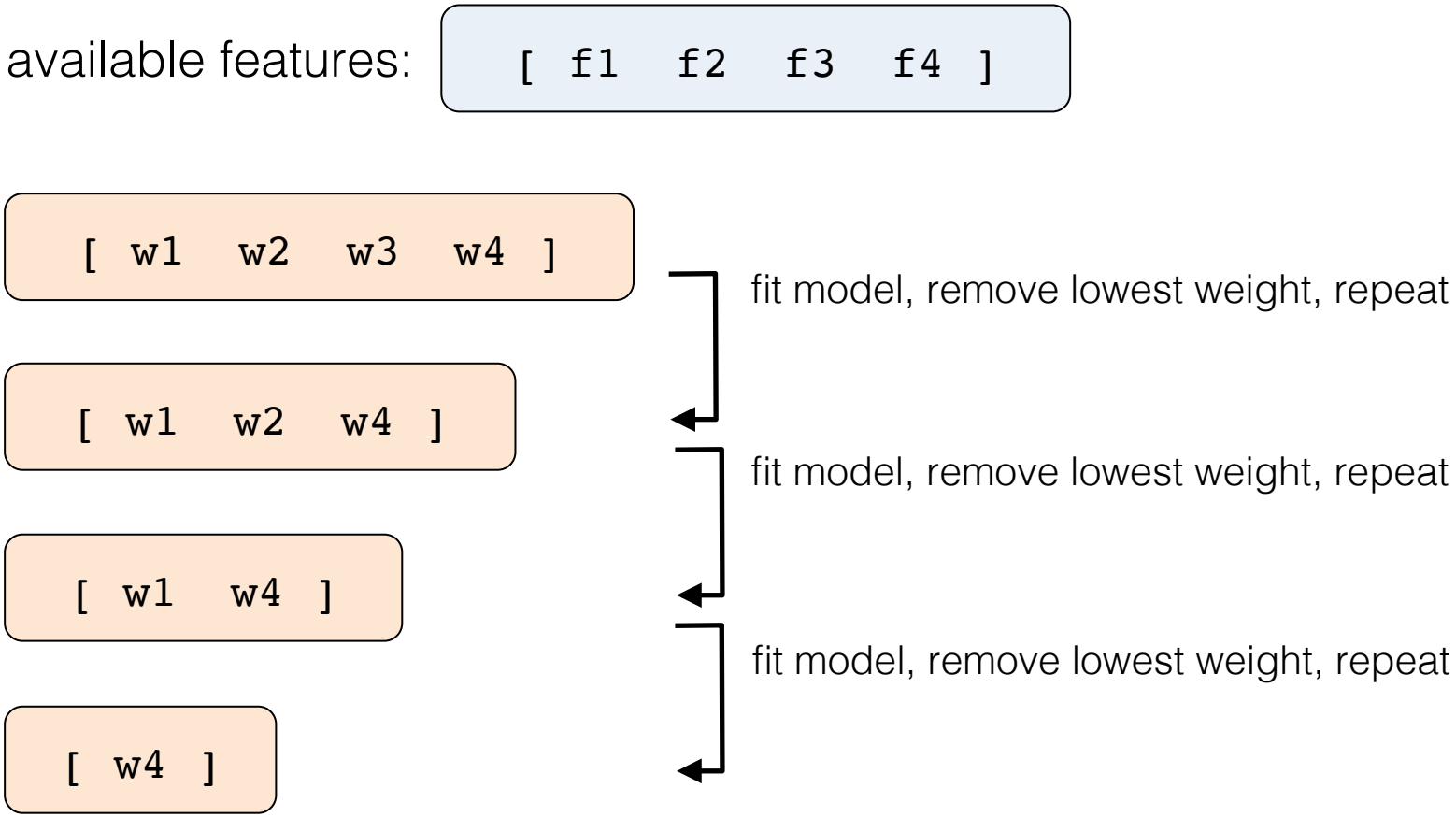
visualization & interpretability

storage & speed

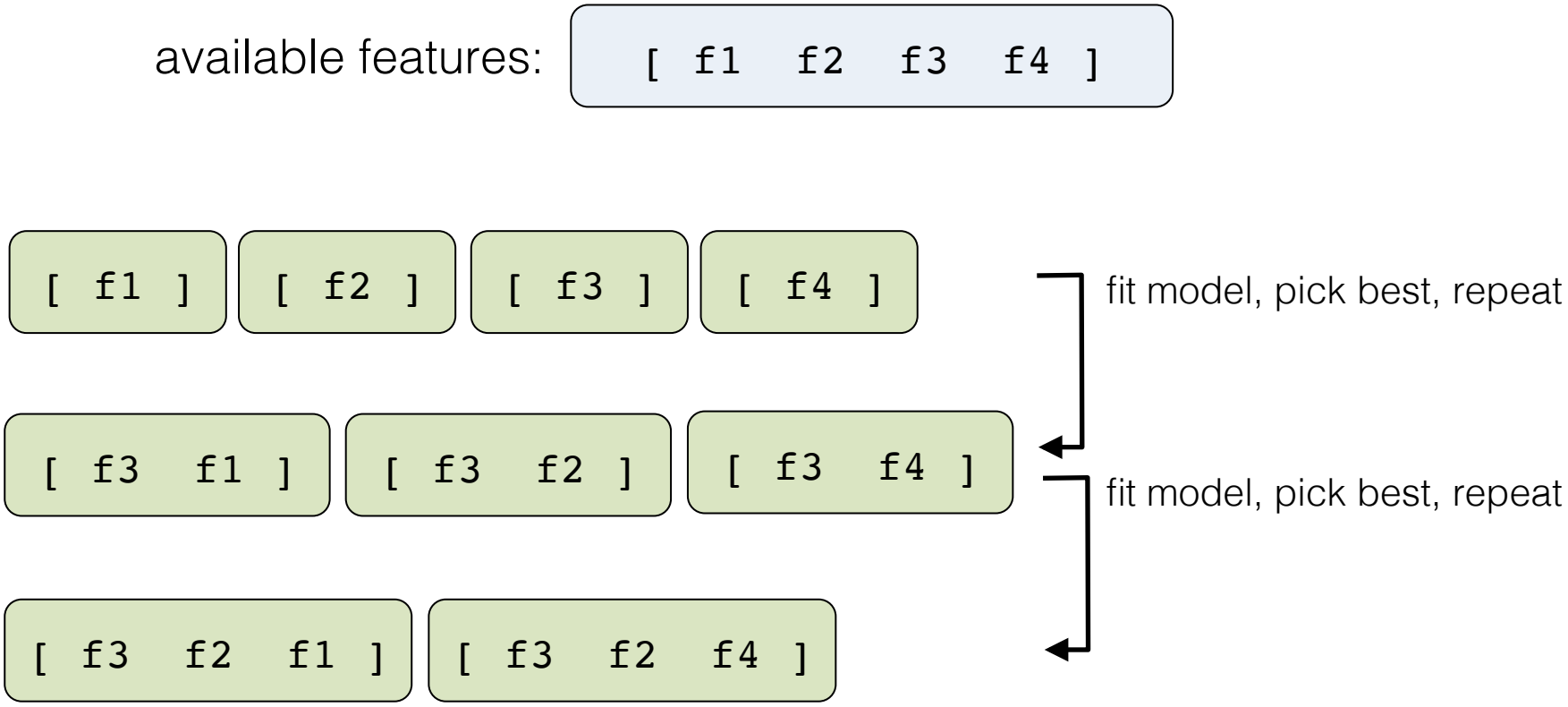


predictive performance

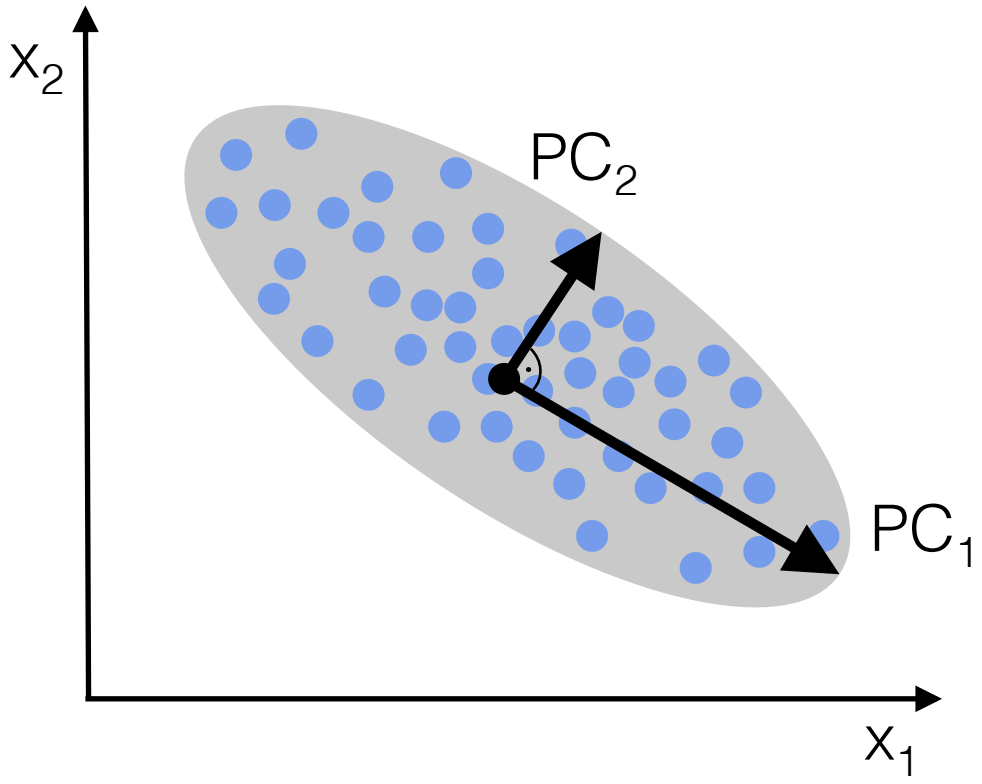
Recursive Feature Elimination



Sequential Feature Selection



Principal Component Analysis



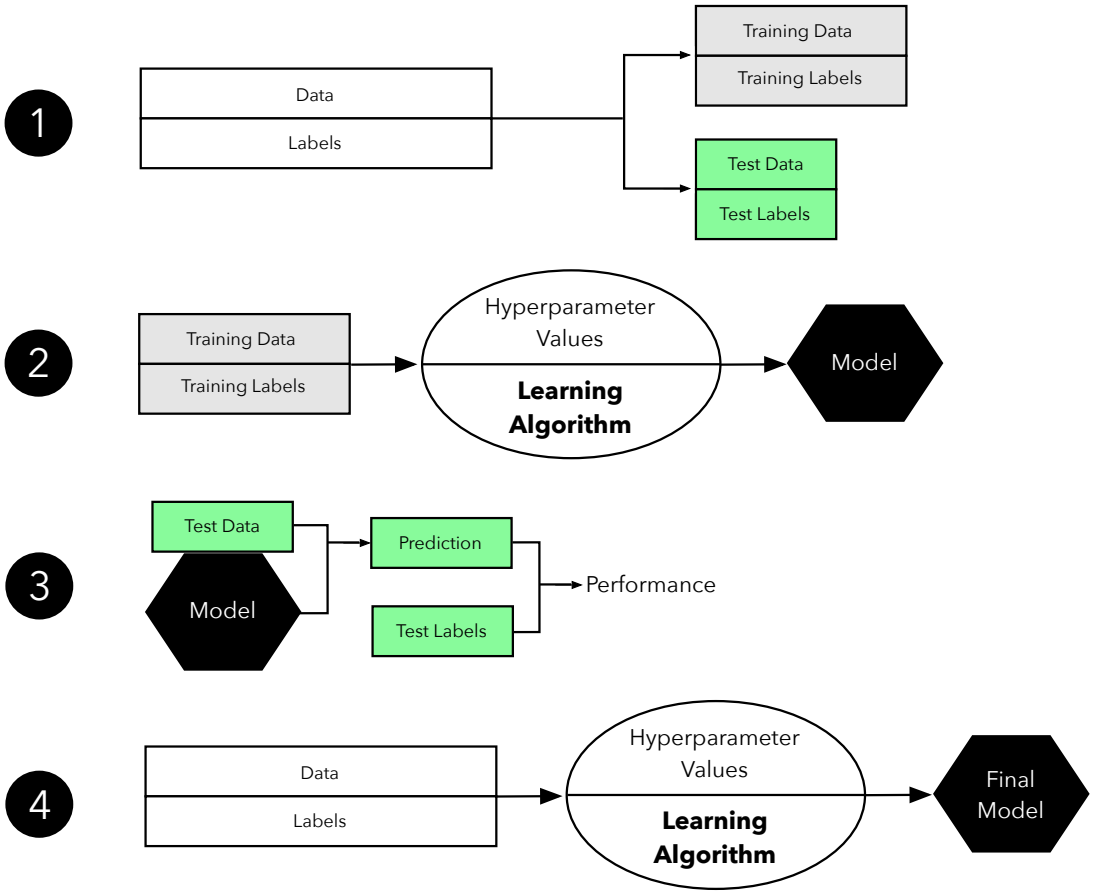
Code Examples

➡ Jupyter Notebook

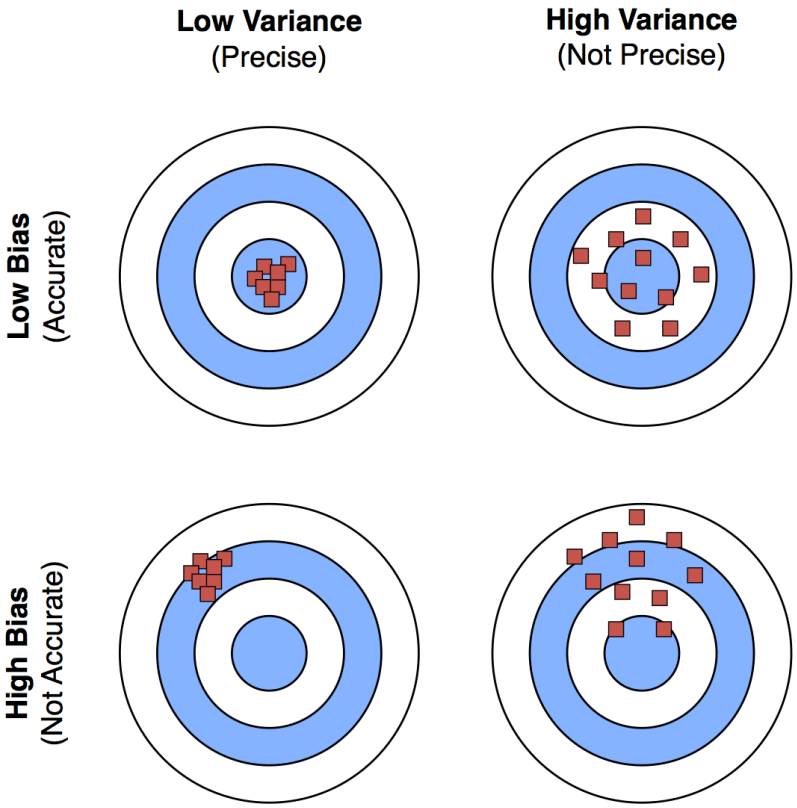
Topics

1. Introduction to Machine Learning
2. Linear Regression
3. Introduction to Classification
4. Feature Preprocessing & scikit-learn Pipelines
5. Dimensionality Reduction: Feature Selection & Extraction
6. **Model Evaluation & Hyperparameter Tuning**

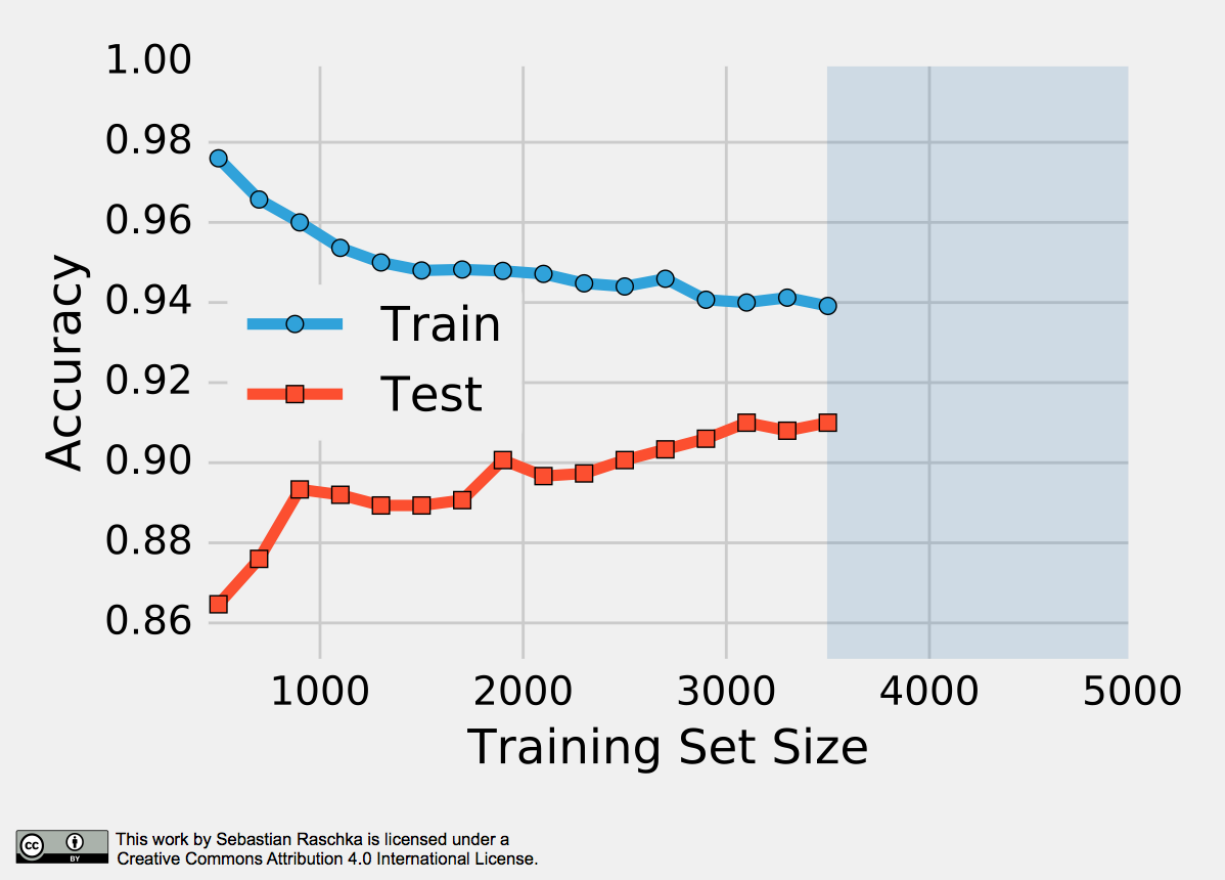
“Basic” Supervised Learning Workflow



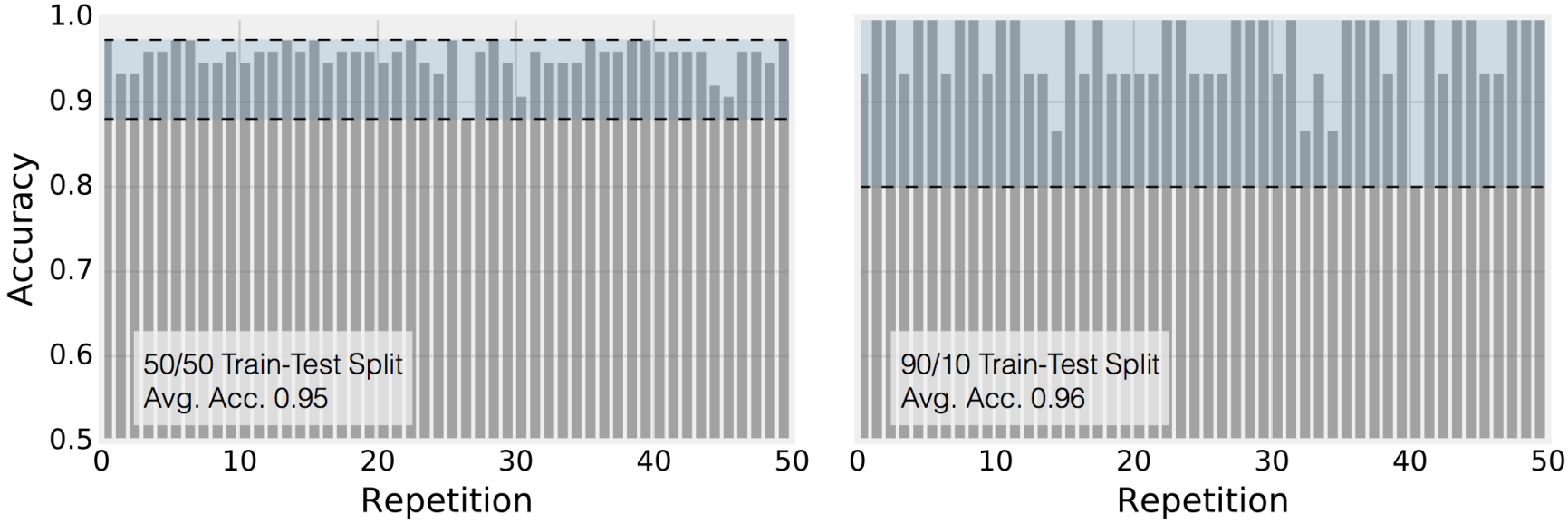
Bias and Variance



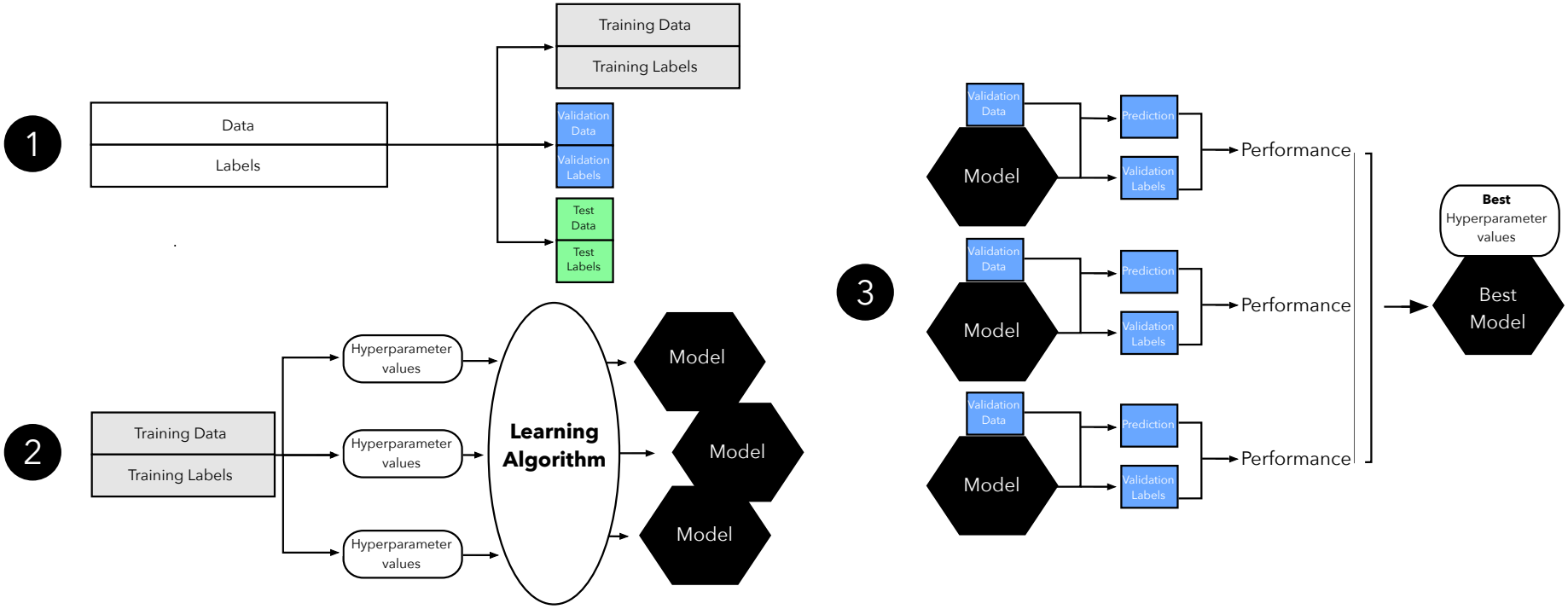
Learning Curves



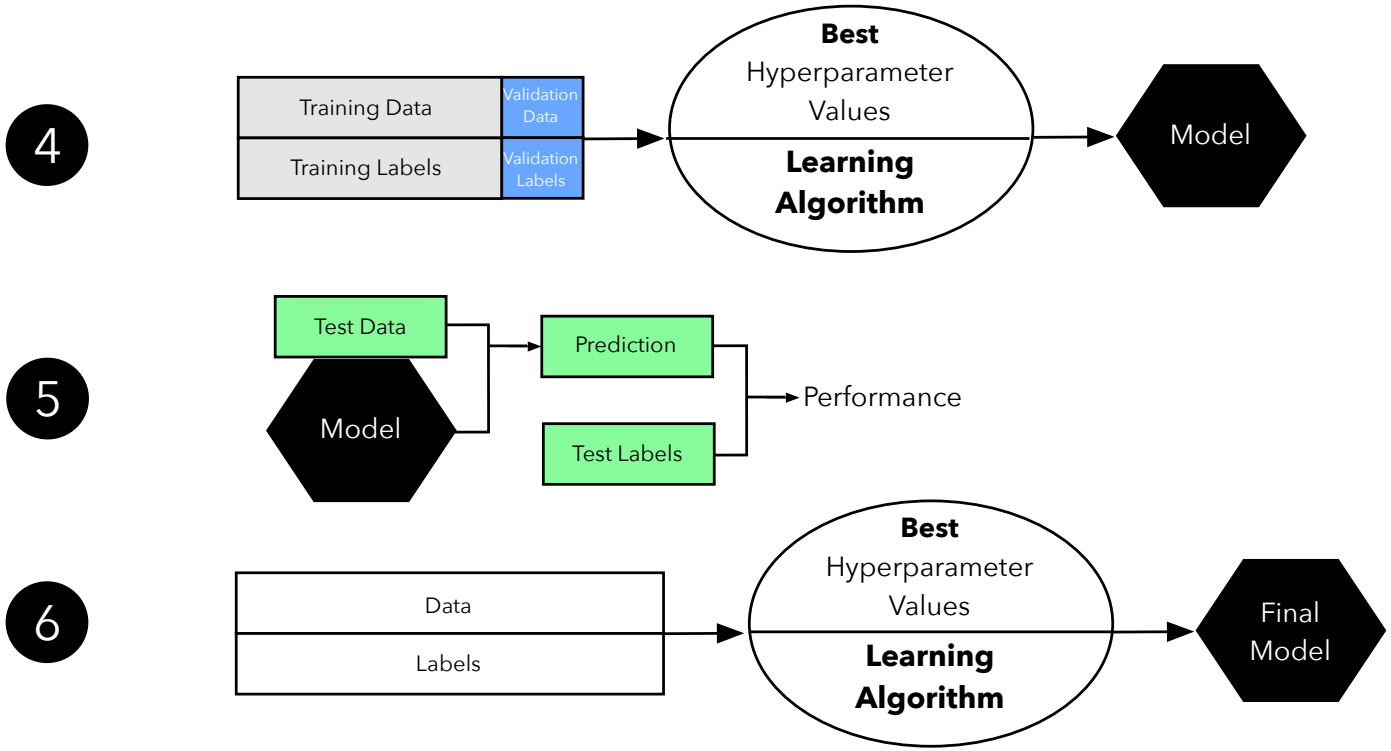
Repeated Holdout



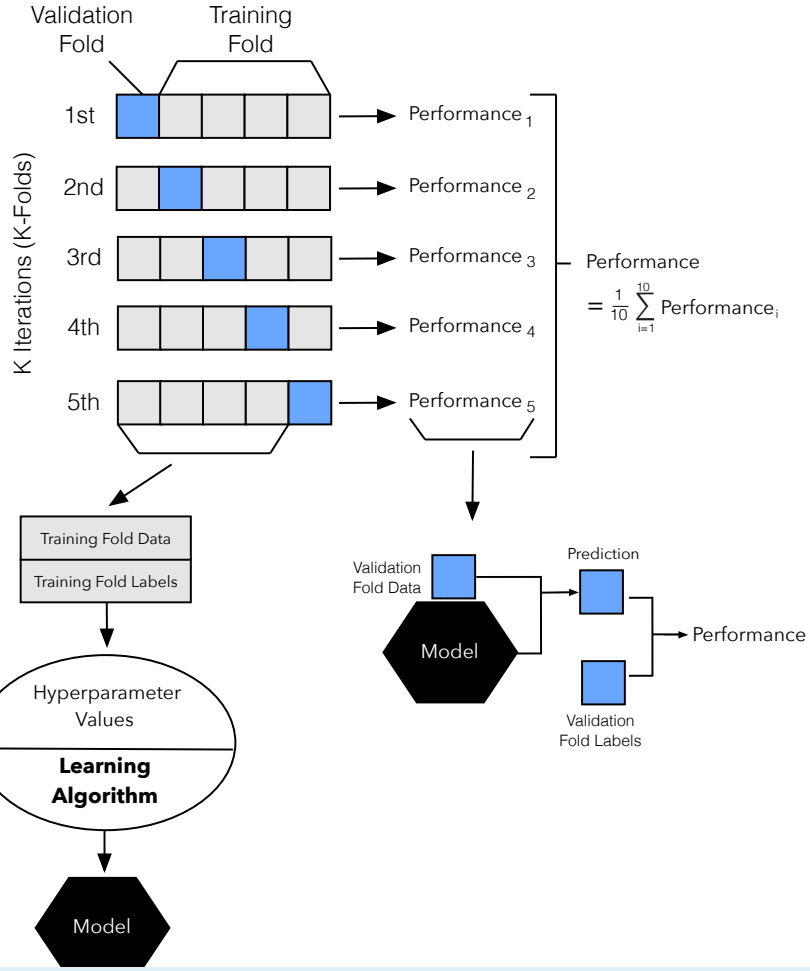
Holdout and Hyperparameter Tuning I



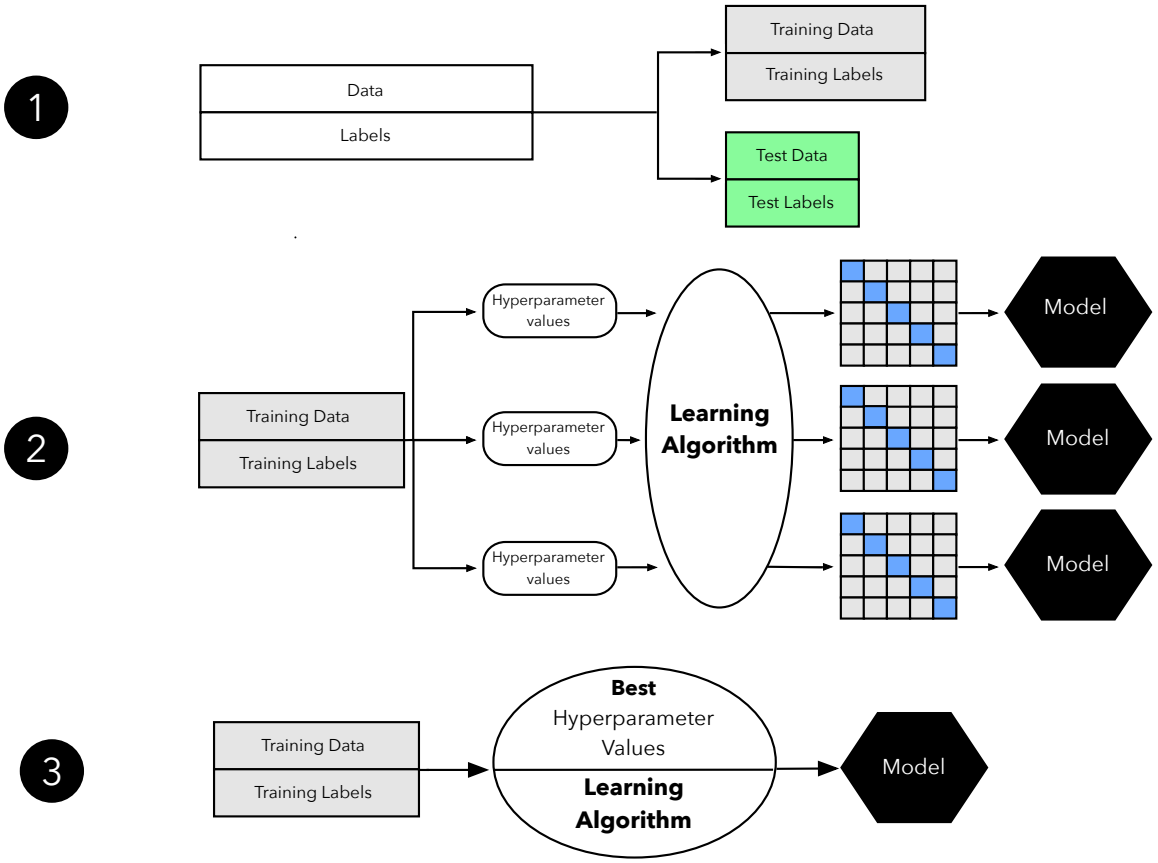
Holdout and Hyperparameter Tuning II



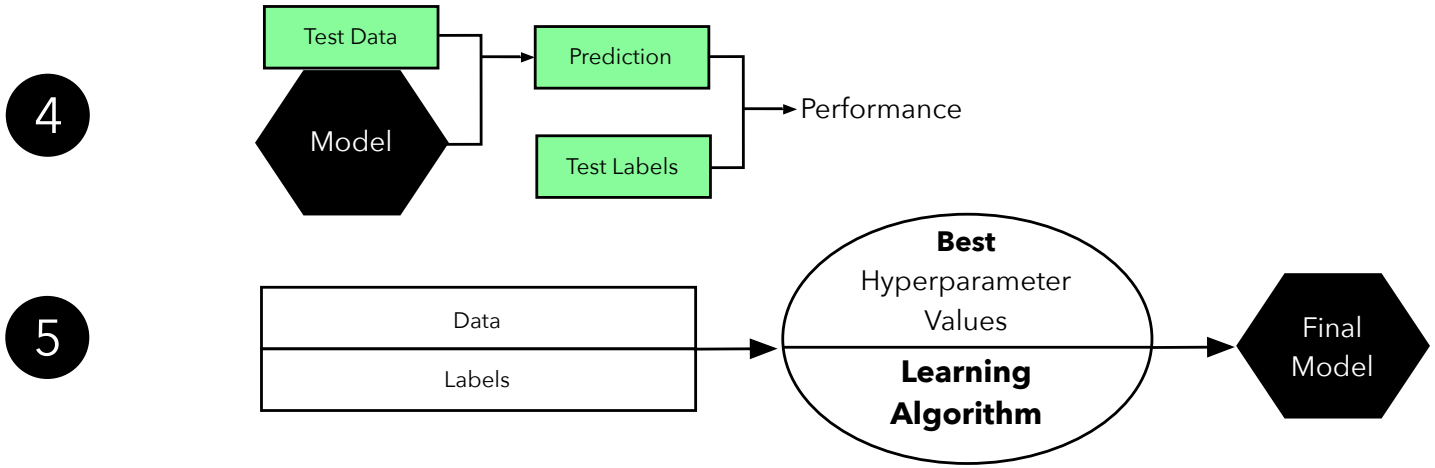
K-fold Cross-Validation



K-fold Cross-Validation Workflow I



K-fold Cross-Validation Workflow II



Code Examples

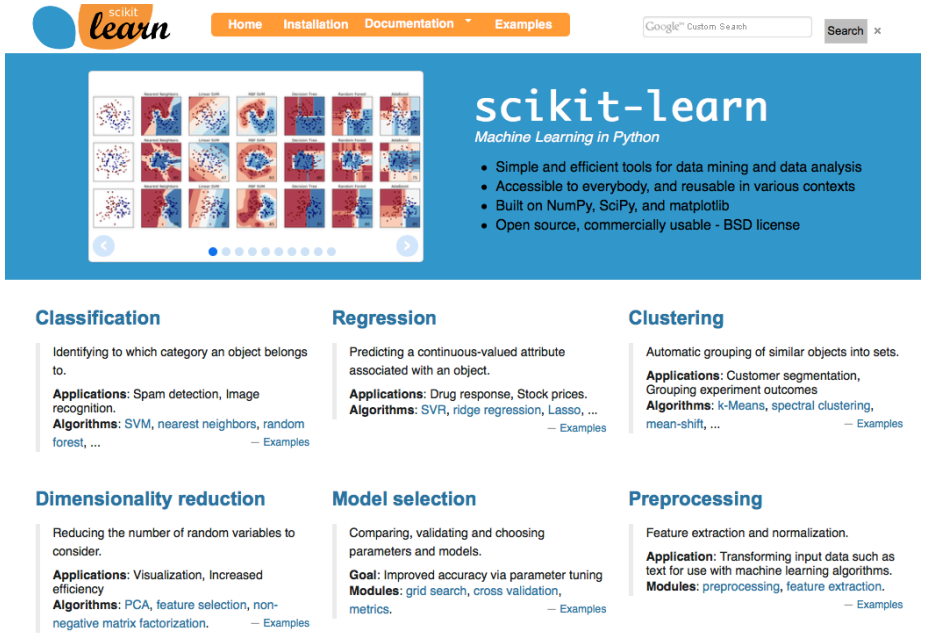
➡ Jupyter Notebook

Performance Metrics

http://scikit-learn.org/stable/modules/model_evaluation.html

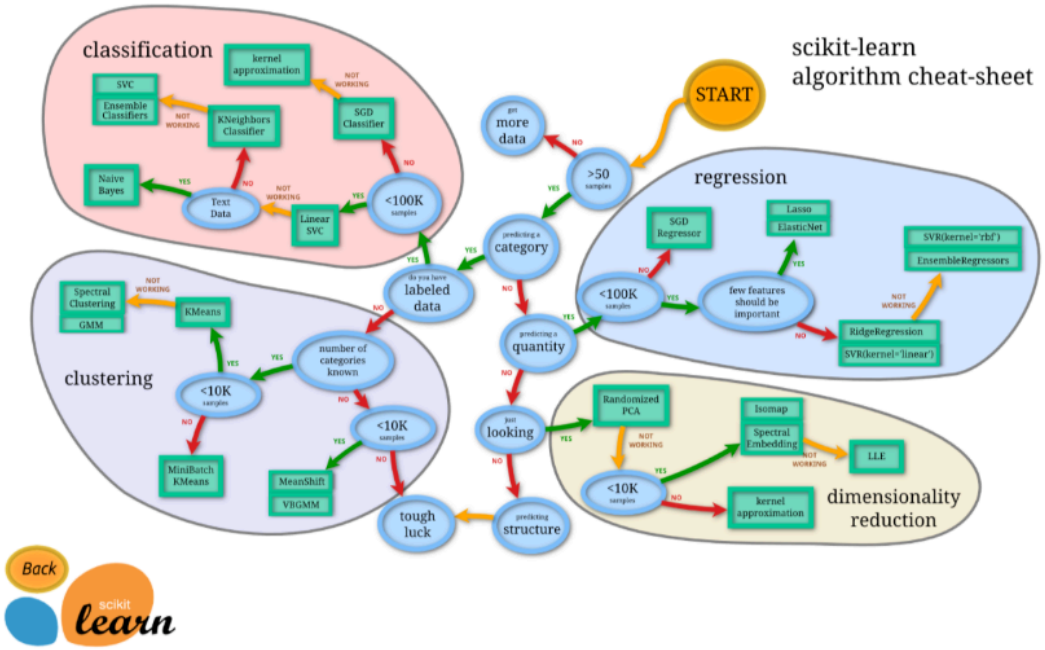
Further Resources

Documentation:
http://scikit-learn.org



Mailing list:
https://mail.python.org/mailman/listinfo/scikit-learn

Further Resources



Andreas' "cheat sheet"

http://scikit-learn.org/stable/tutorial/machine_learning_map/

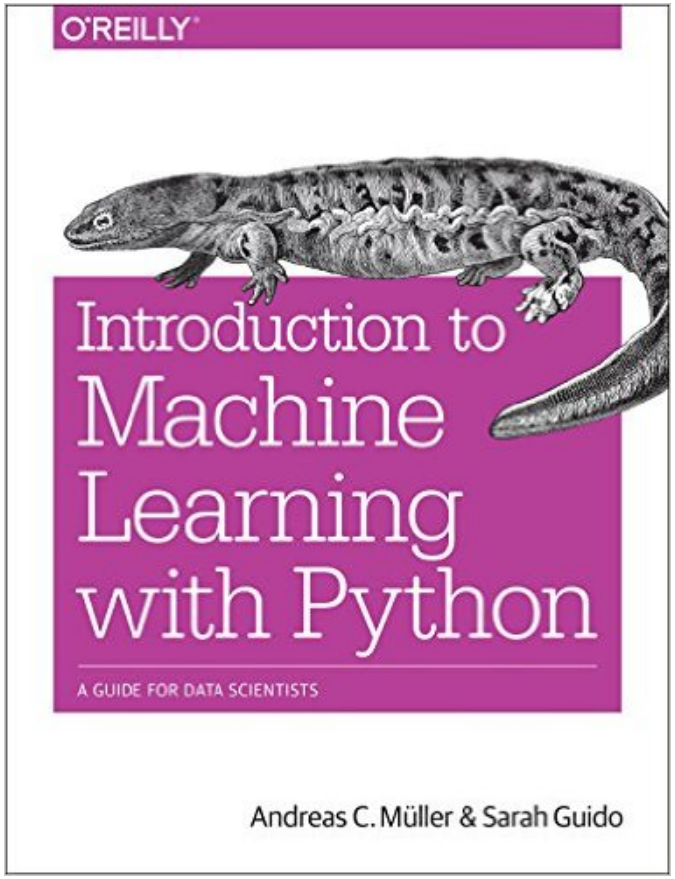
Further Resources

Great "math-free," practical guide to machine learning with scikit-learn

By Andreas Mueller (scikit-learn core developer) and Sarah Guido

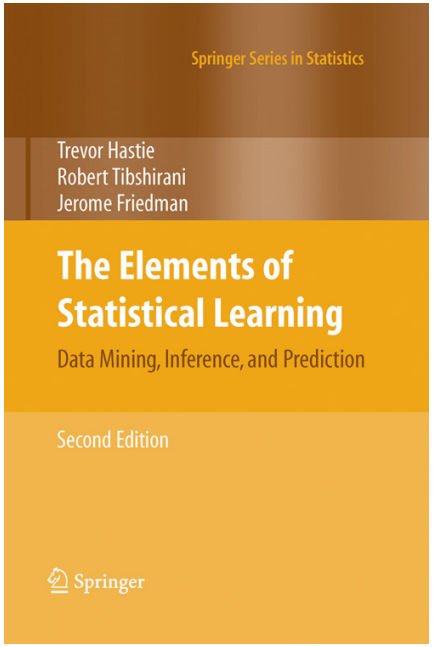
http://shop.oreilly.com/product/0636920030515.do

Estimated release: October 20, 2016

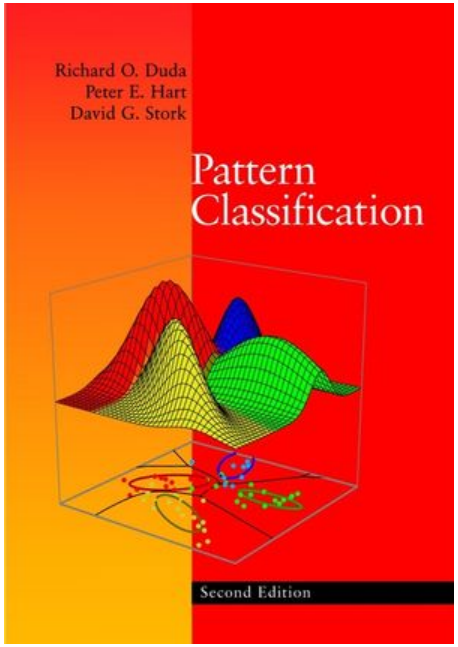


Further Resources

My favorite machine learning "math & theory" books



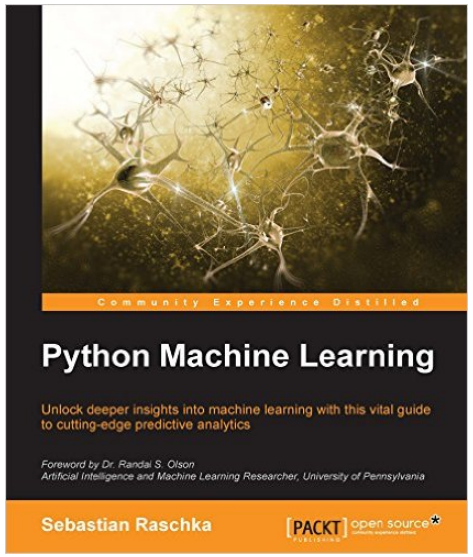
http://statweb.stanford.edu/~tibs/ElemStatLearn/ (free PDF)

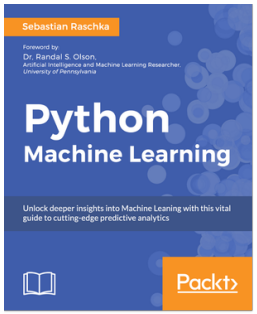


http://www.wiley.com/WileyCDA/WileyTitle/productCd-0471056693.html

Further Resources

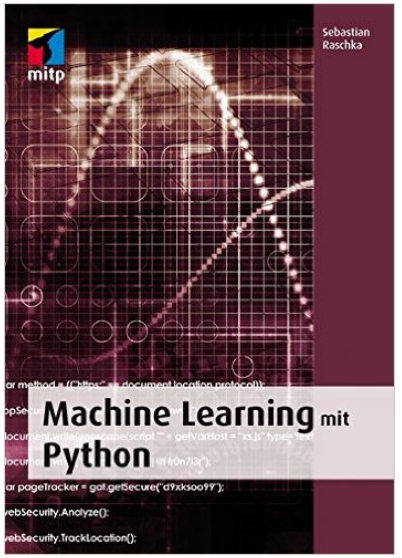
My own book.
Some math,
"from-scratch" code,
and practical scikit-learn examples





GitHub repository:

https://github.com/rasbt/python-machine-learning-book



Amazon link:

https://www.amazon.com/Python-Machine-Learning-Sebastian-Raschka/dp/1783555130/

Thanks for attending!

Link to the material:

https://github.com/rasbt/pydata-chicago2016-ml-tutorial

Contact:

- E-mail: mail@sebastianraschka.com
- Website: http://sebastianraschka.com
- Twitter: @rasbt
- GitHub: rasbt