Psychoinformatics & Neuroinformatics

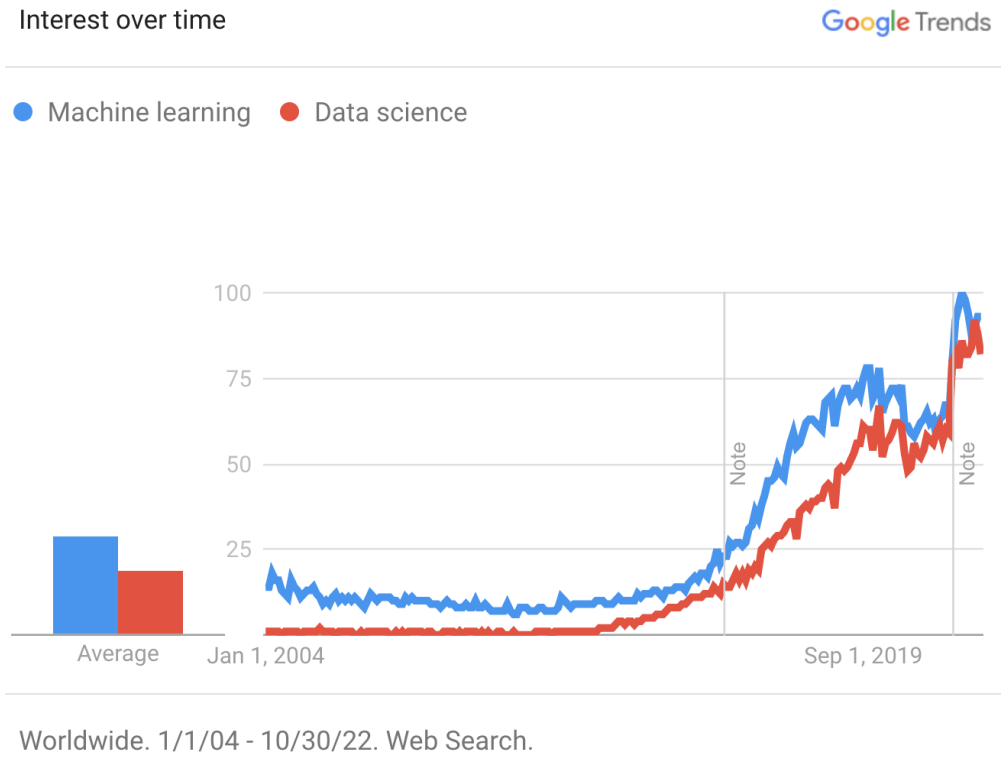
Week 9

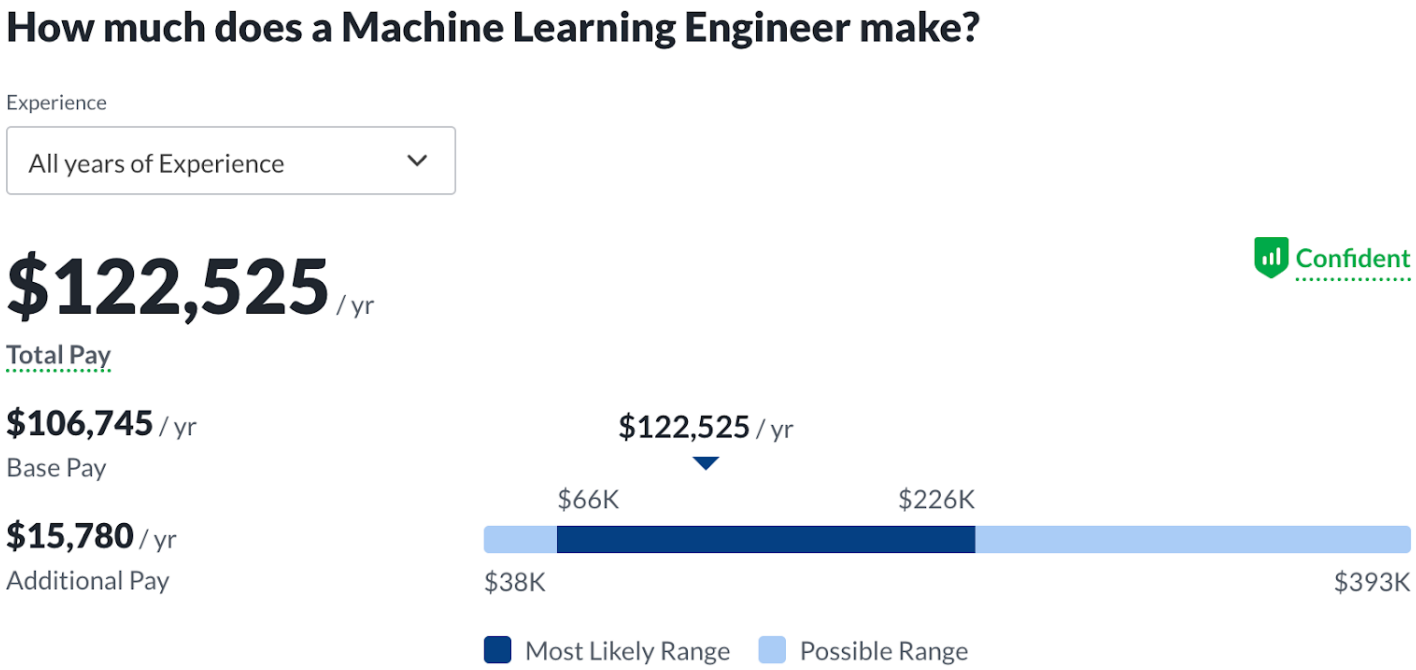
Machine Learning (1/3)

by Tsung-Ren (Tren) Huang 黃從仁

Global interests & Job demands

ML becomes an integral part of data science





Attention in Academia

https://www.nature.com › articles · 翻譯這個網頁

Physics-informed machine learning | Nature Reviews Physics

由 GE Karniadakis 著作 · 2021 · 被引用 793 次 — Physics-informed machine learning integrates seamlessly data and mathematical physics models, even in partially understood, uncertain…

https://www.annualreviews.org › doi › abs · 翻譯這個網頁

Machine Learning for Social Science: An Agnostic Approach

由 J Grimmer 著作 · 2021 · 被引用 75 次 — Machine Learning for Social Science: An Agnostic Approach. Annual Review of Political Science. Vol. 24:395-419 (Volume publication date May…

https://www.annualreviews.org › doi › abs · 翻譯這個網頁

Machine Learning for Sociology - Annual Reviews

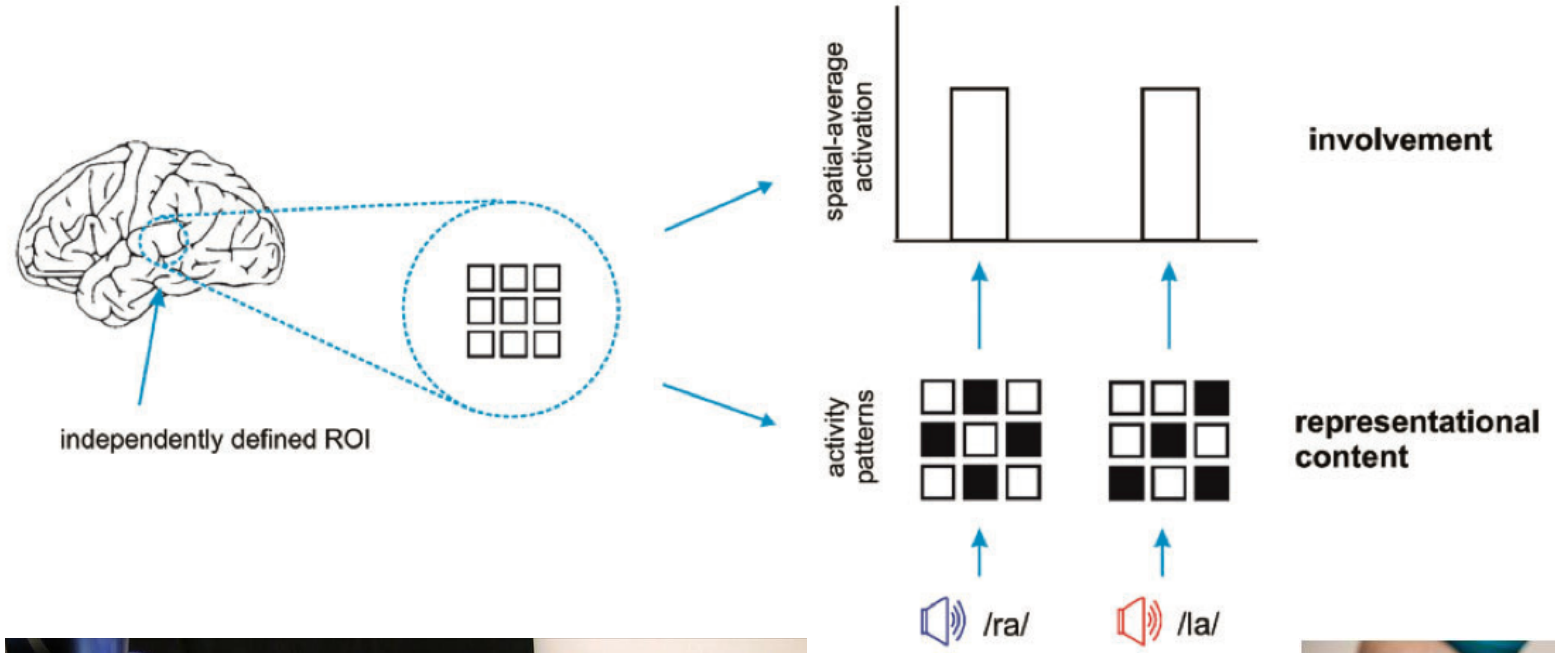
由 M Molina 著作 · 2019 · 被引用 153 次 — Machine learning is a field at the intersection of statistics and computer science that uses algorithms to extract information and knowledge fro…

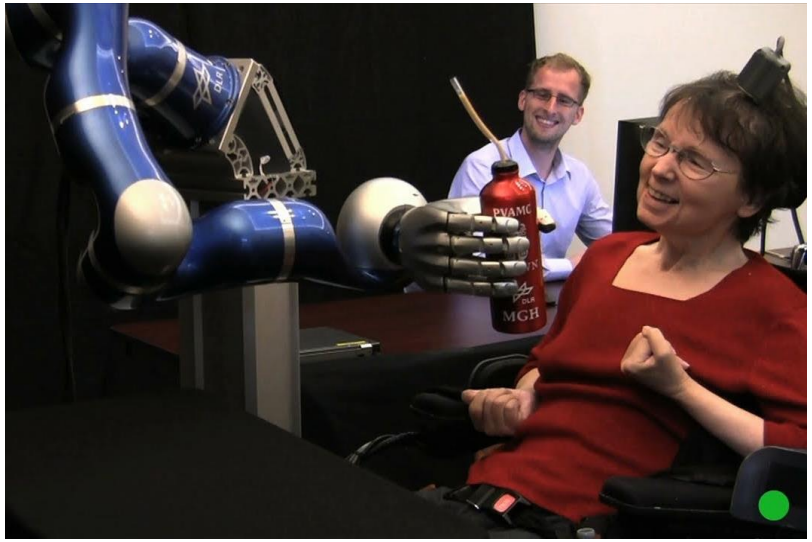
https://www.annualreviews.org › doi › abs · 翻譯這個網頁

Machine Learning Methods That Economists Should Know ...

由 S Athey 著作 · 2019 · 被引用 536 次 — These include supervised learning methods for regression and classification, unsupervised learning methods, and matrix completion methods.…

Numerous Applications





Goals for today

Features of Machine Learning

explanatory vs. predictive modeling

Implementations of Machine Learning

supervised & unsupervised learning

Inferences from Machine Learning

common pitfalls

Goals for today

Features of Machine Learning

explanatory vs. predictive modeling

Implementations of Machine Learning

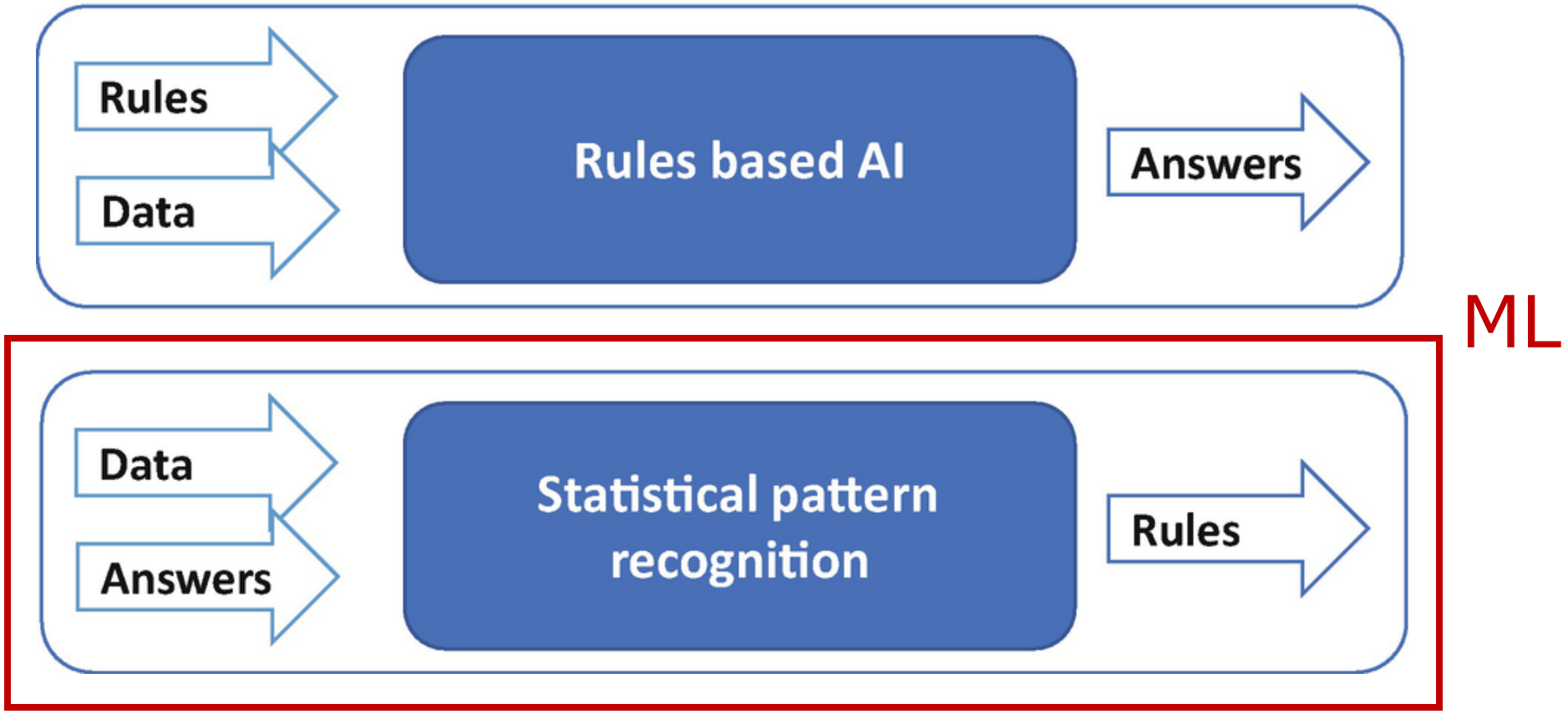
supervised & unsupervised learning

Inferences from Machine Learning

common pitfalls

Two Traditions of AI

AI powered by memory/deduction or learning/induction:

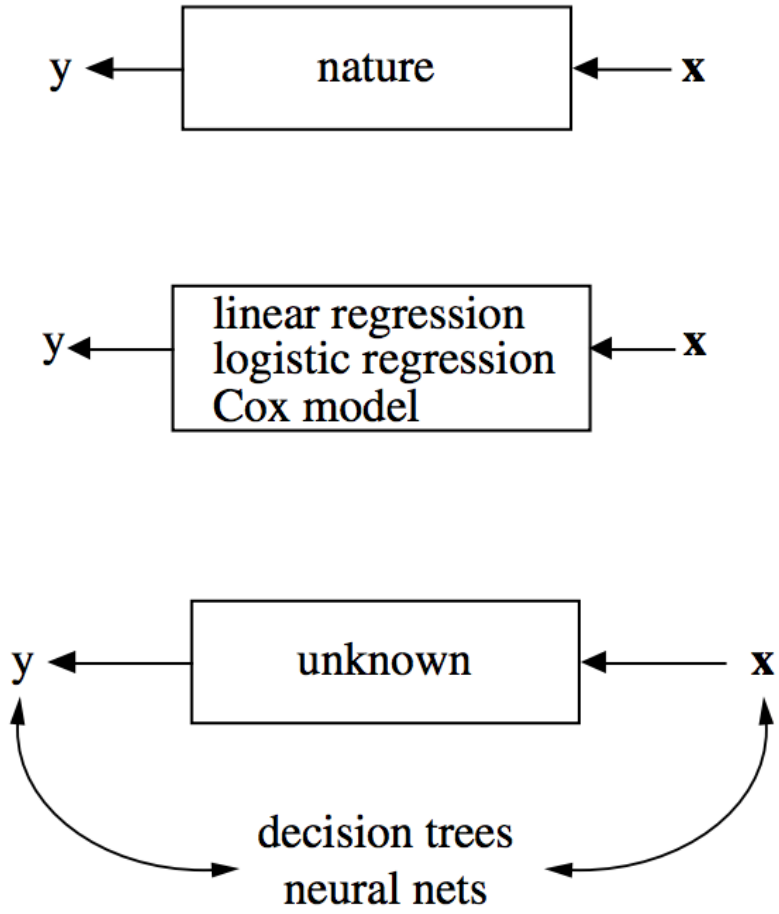


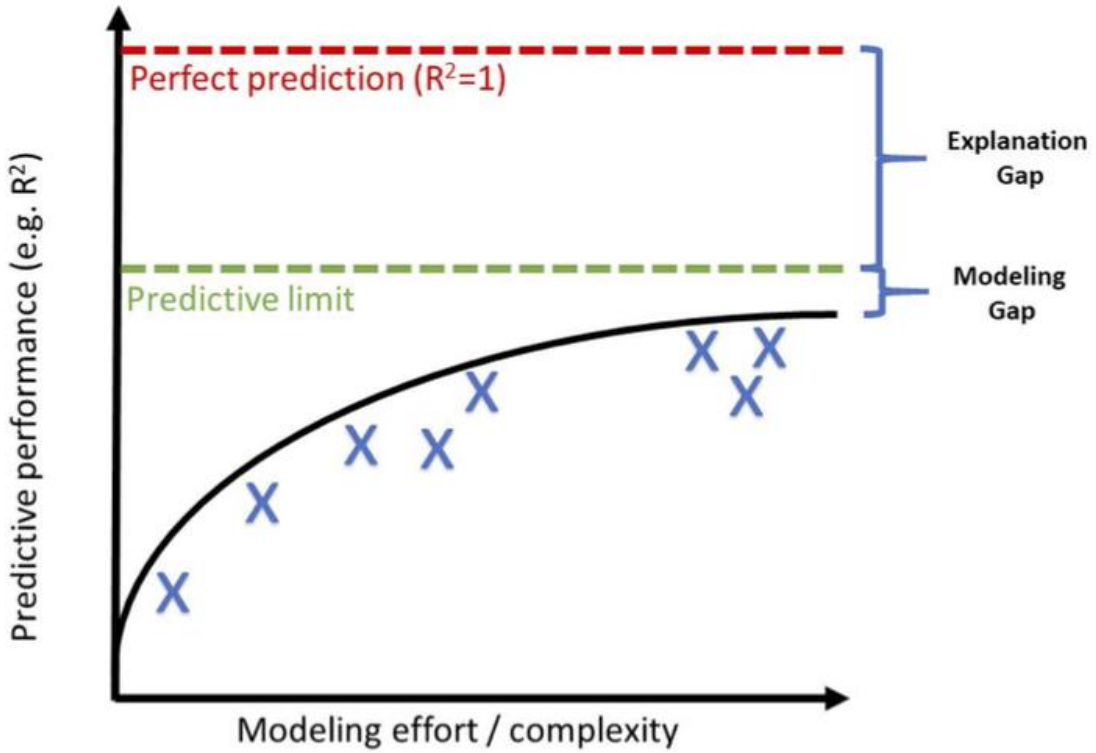
Explanatory vs. Predictive Modeling

Statistical Science
2001, Vol. 16, No. 3, 199–231

Statistical Modeling: The Two Cultures

Leo Breiman

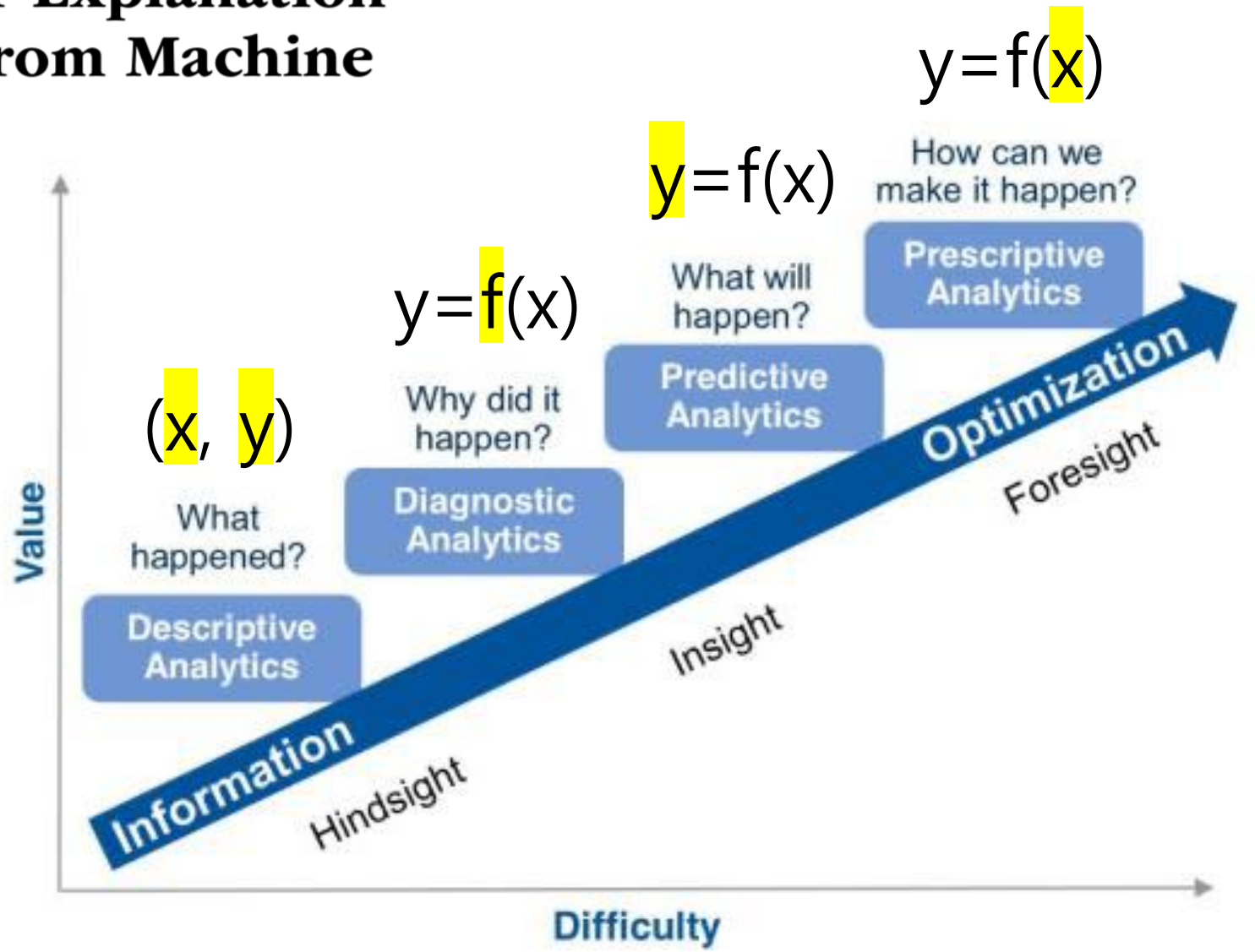




Explanatory vs. Predictive Modeling

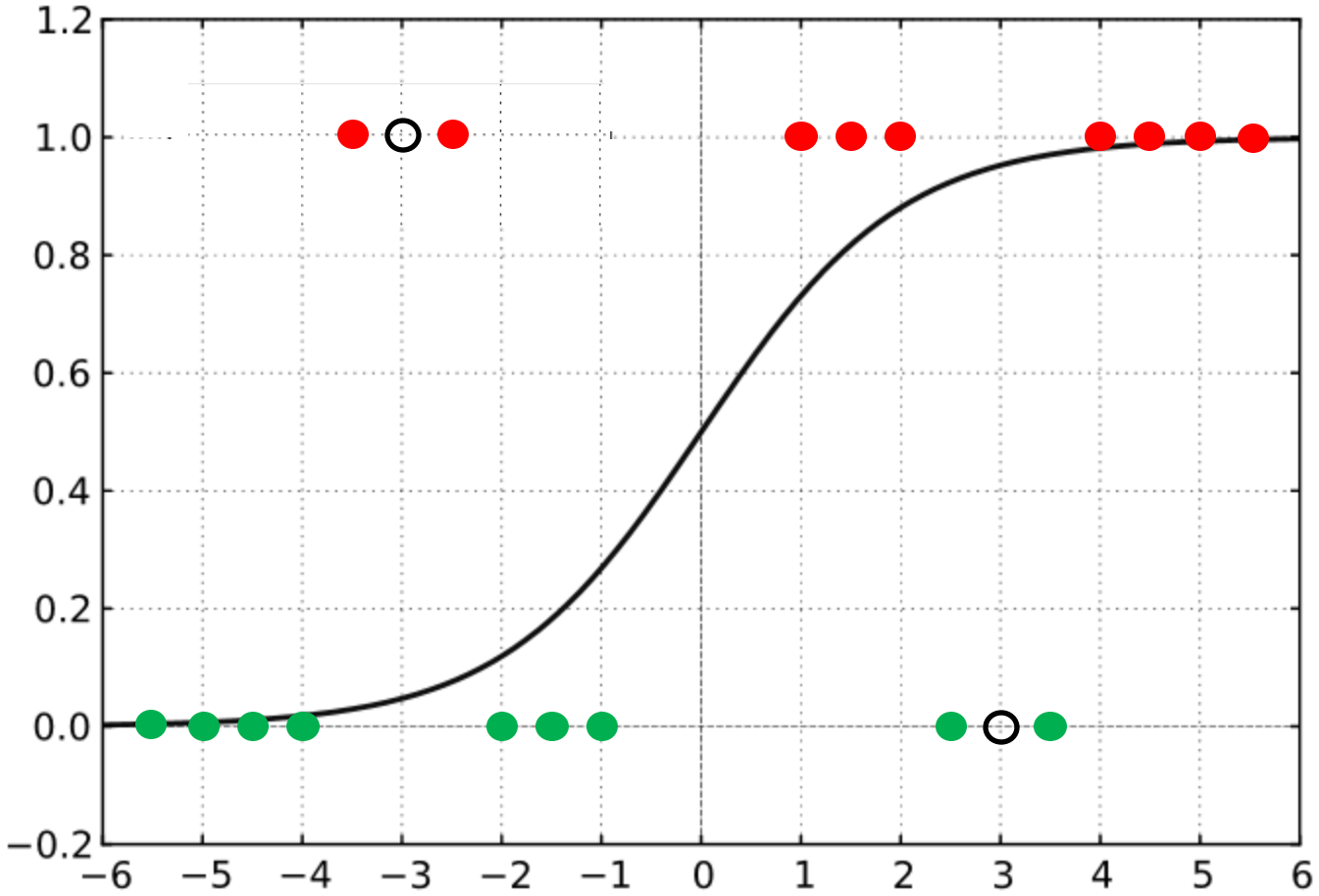
Choosing Prediction Over Explanation in Psychology: Lessons From Machine Learning

Tal Yarkoni and Jacob Westfall
University of Texas at Austin



Why are Predictive Models Better?

Compare the predictions of y(x=-3) & y(x=3)
by a logistic regression & 2-nearest neighbor:



Goals for today

Features of Machine Learning

explanatory vs. predictive modeling

Implementations of Machine Learning

supervised & unsupervised learning

Inferences from Machine Learning

common pitfalls

Scikit-learn: The de facto standard

Classification

Identifying to which set of categories a new observation belong to.

Applications: Spam detection, Image recognition.
Algorithms: *SVM*, *nearest neighbors*, *random forest*, ... — *Examples*

Regression

Predicting a continuous value for a new example.

Applications: Drug response, Stock prices.
Algorithms: *SVR*, *ridge regression*, *Lasso*, ... — *Examples*

Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: *k-Means*, *spectral clustering*, *mean-shift*, ... — *Examples*

Dimensionality reduction

Reducing the number of random variables to consider.

Applications: Visualization, Increased efficiency
Algorithms: *PCA*, *Isomap*, *non-negative matrix factorization*. — *Examples*

Model selection

Comparing, validating and choosing parameters and models.

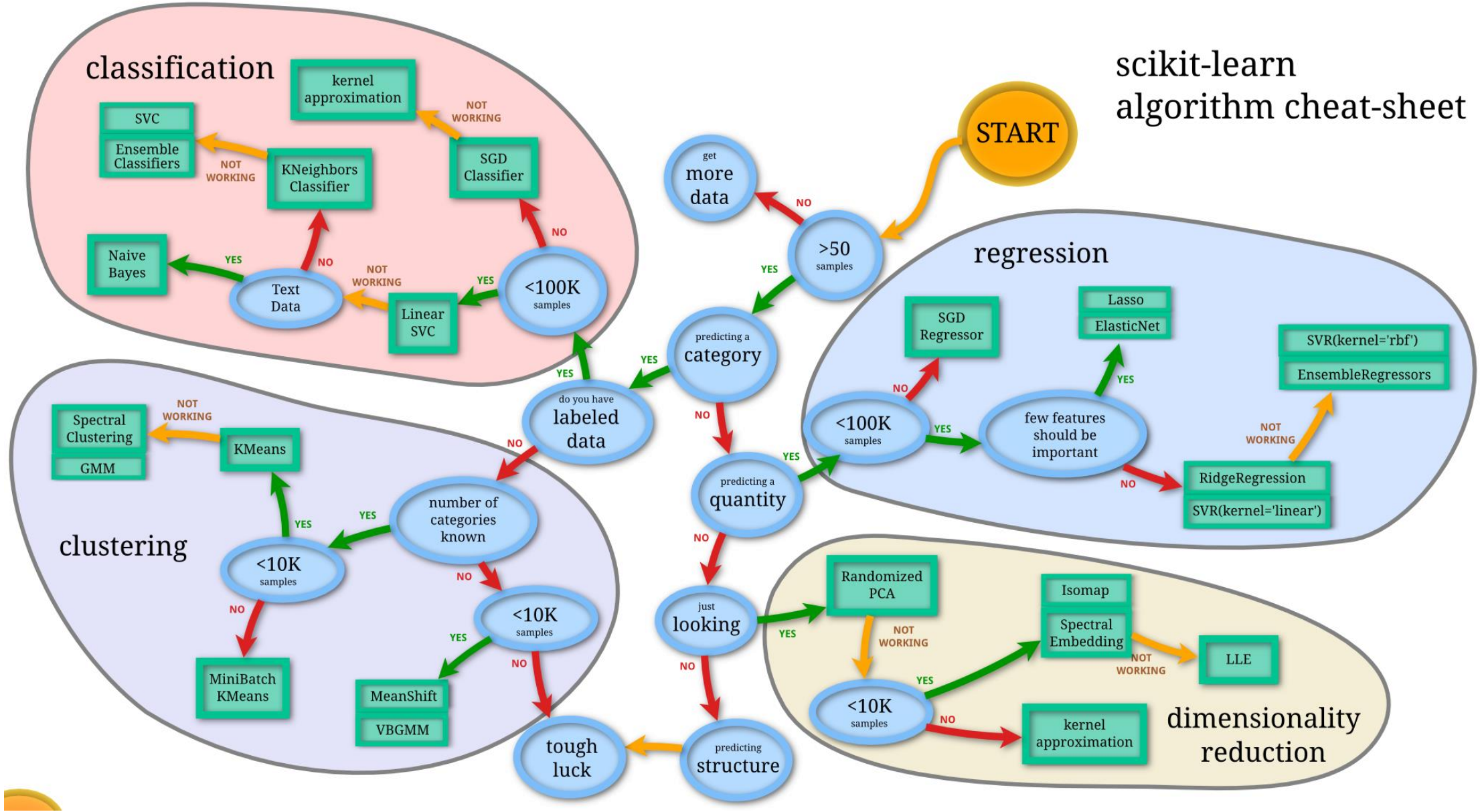
Goal: Improved accuracy via parameter tuning
Modules: *grid search*, *cross validation*, *metrics*. — *Examples*

Preprocessing

Feature extraction and normalization.

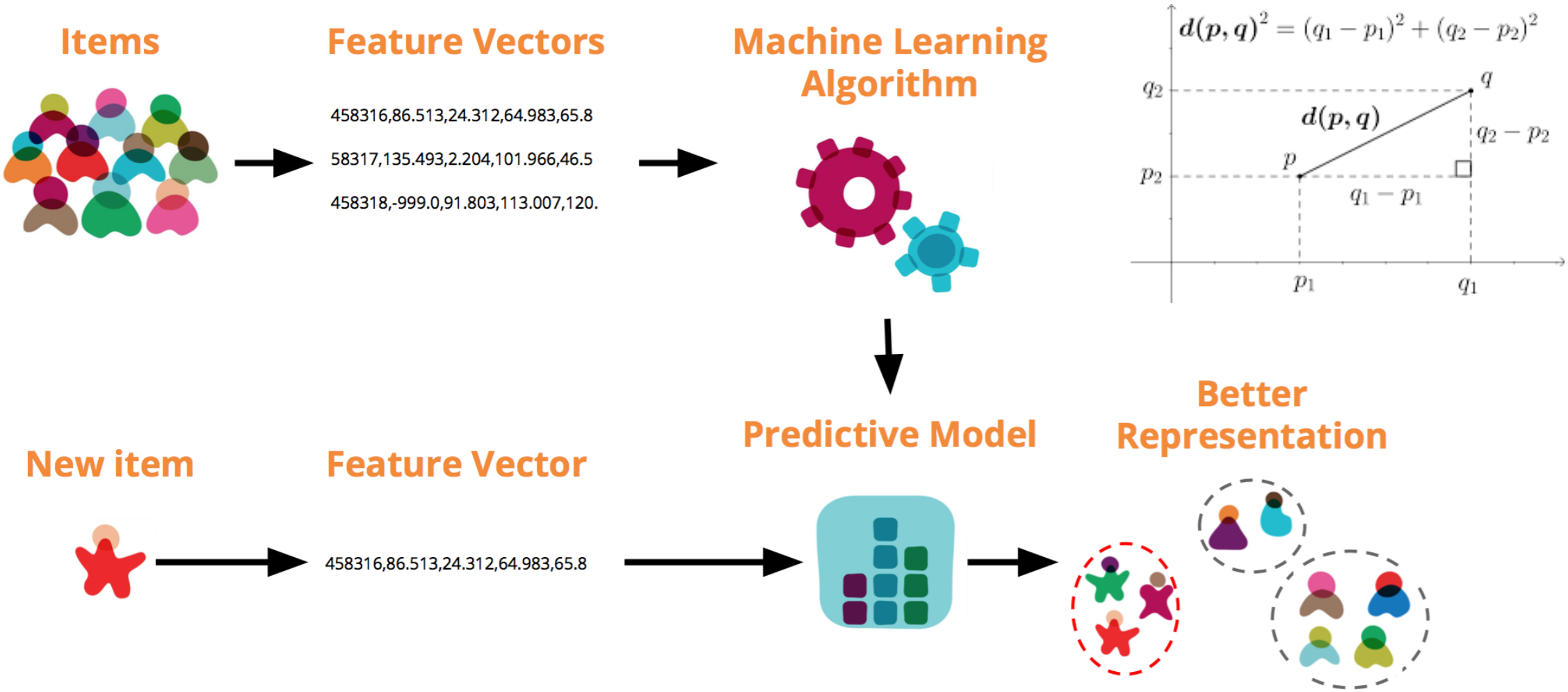
Application: Transforming input data such as text for use with machine learning algorithms.
Modules: *preprocessing*, *feature extraction*. — *Examples*

The scikit-learn cheat sheet



Features & Normalization

Normalizing features to avoid feature-biased distance measures

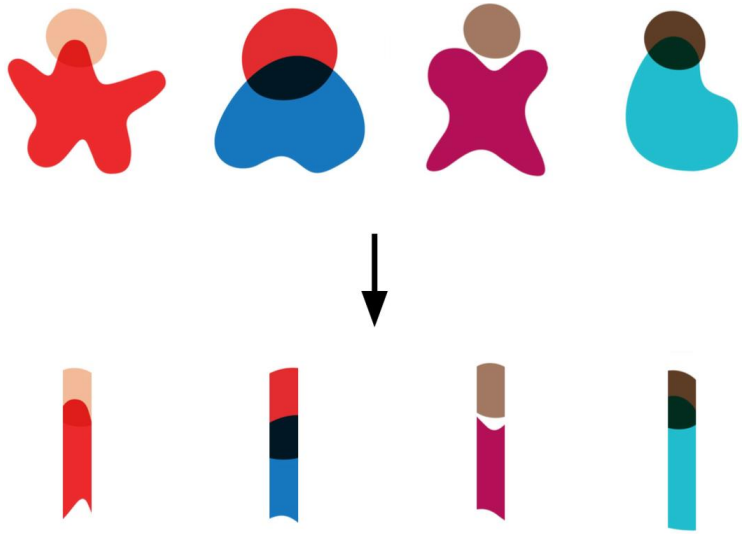


Dimensionality Reduction

Reducing feature dimensions can reduce search space:

For learning $Y=f([X_1, X_2, X_3, X_4])=X_1*X_3$:

The search space has $2^4*2=32$ mappings:
$[X_1=0/1, X_2=0/1, X_3=0/1, X_4=0/1] \rightarrow Y=0/1$



For learning $Y=f([X_1, X_3])=X_1*X_3$:

The search space only has $2^2*2=8$ mappings:
$[X_1=0/1, X_3=0/1] \rightarrow Y=0/1$

Supervised vs. Unsupervised Learning

Normalizing features to avoid feature-biased distance measures

Supervised

X_1	X_2	X_p	Y

Target

Un-Supervised

X_1	X_2	X_p	Y

No Target

Generating Predictions

Improving Theories

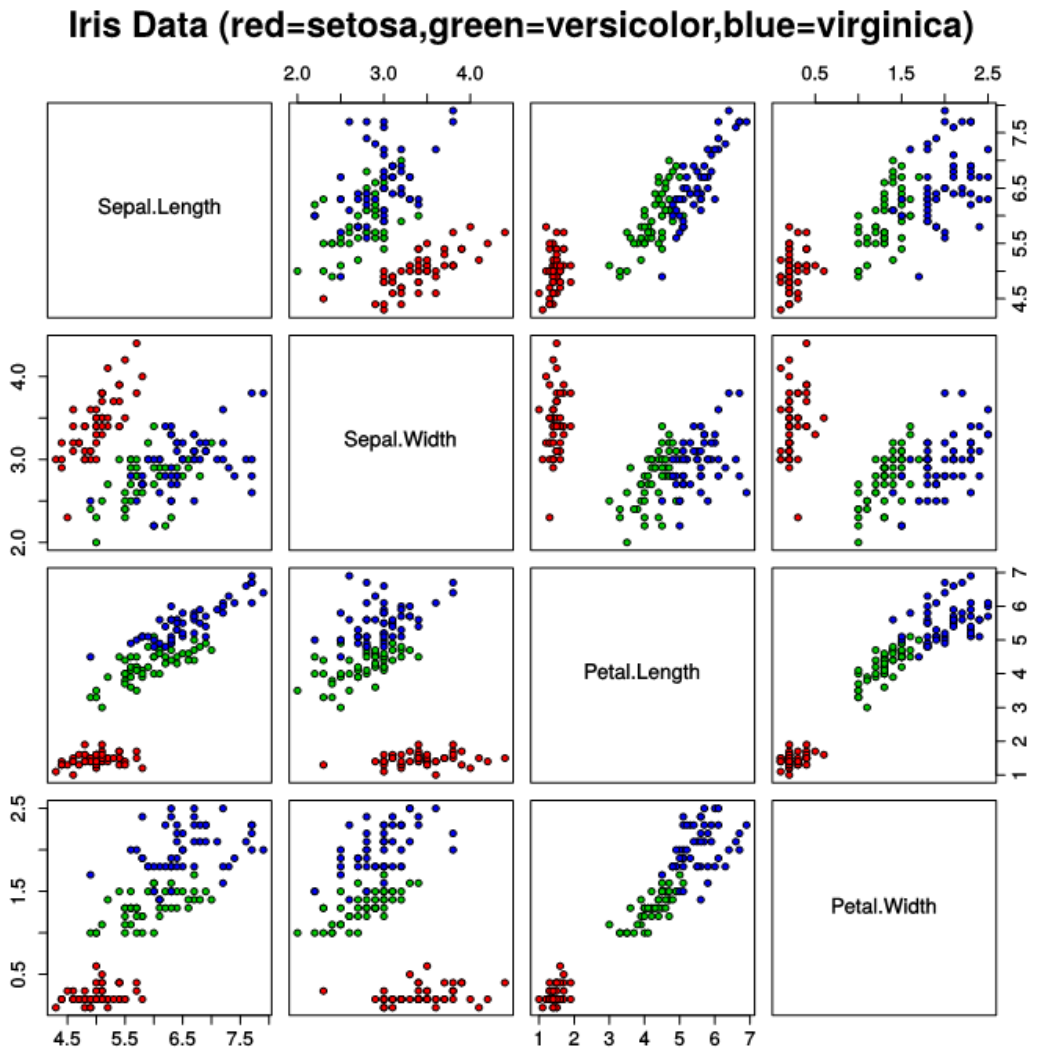
Exploring heterogeneity

Uncovering Latent Vars

	Unsupervised	Supervised
Continuous	Clustering & Dimensionality Reduction SVD PCA K-Means	Regression Linear Polynomial Decision Trees Random Forests Nerual Networks
Categorical	Association Analysis Apriori FP-Growth Hidden Markov Model	Classification KNN Trees Logistic Regression Naive-Bayes SVN Nerual Networks

The Iris Dataset

It’s a common dataset for testing new algorithms



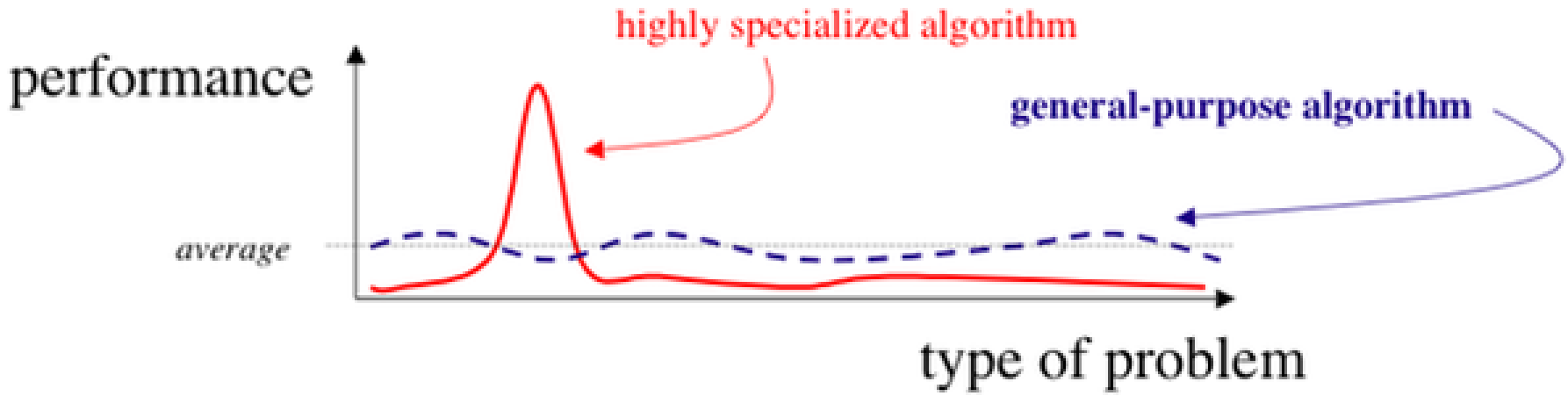
Comparisons of Different Algorithms

Need to have some conceptual understanding of them

1. Supervised learning
- 1.1. Linear Models
- 1.2. Linear and Quadratic Discriminant Analysis
- 1.3. Kernel ridge regression
- 1.4. Support Vector Machines
- 1.5. Stochastic Gradient Descent
- 1.6. Nearest Neighbors
- 1.7. Gaussian Processes
- 1.8. Cross decomposition
- 1.9. Naive Bayes
- 1.10. Decision Trees
- 1.11. Ensemble methods
- 1.12. Multiclass and multioutput algorithms
- 1.13. Feature selection
- 1.14. Semi-supervised learning
- 1.15. Isotonic regression
- 1.16. Probability calibration
- 1.17. Neural network models (supervised)

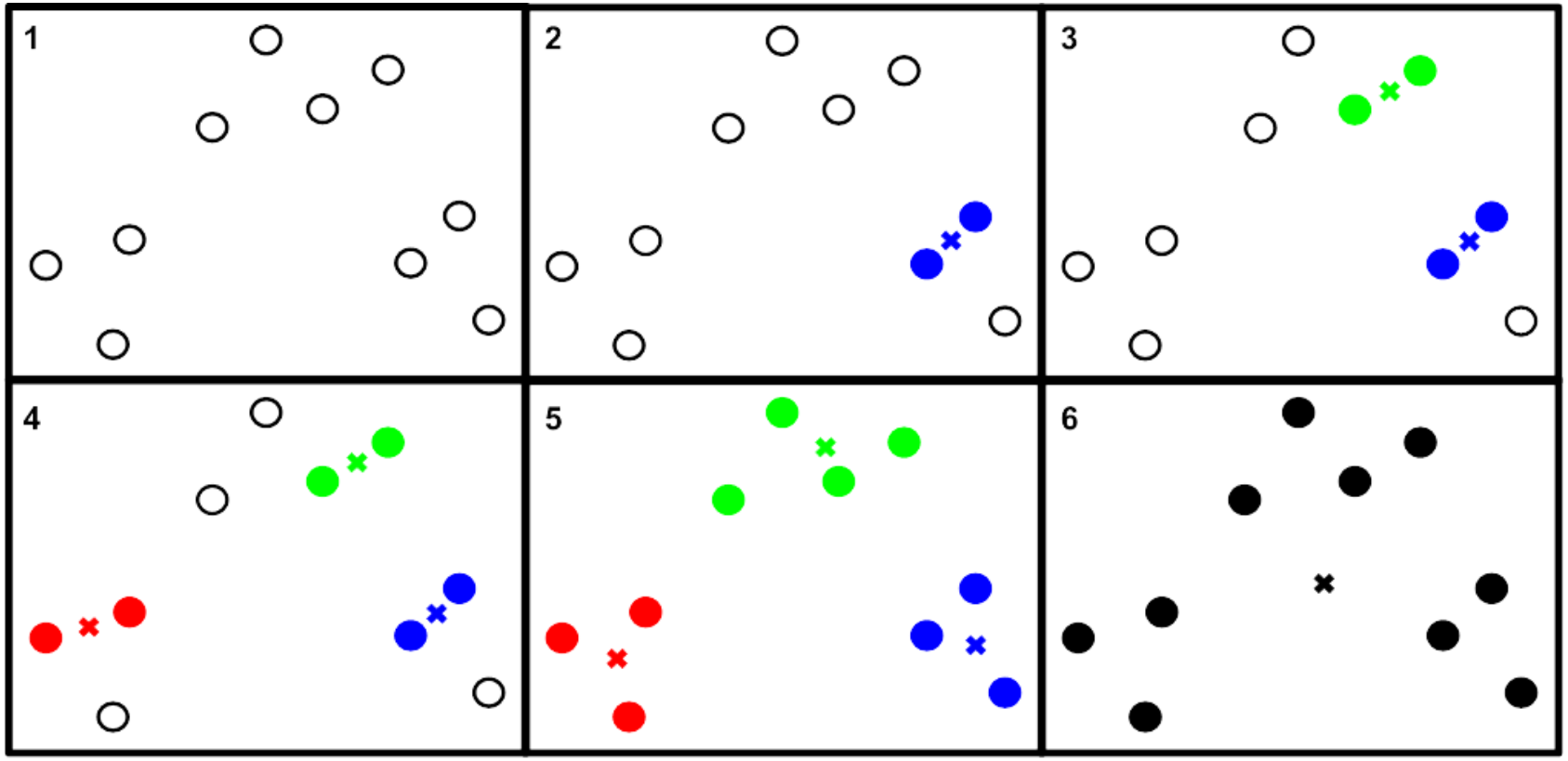
2. Unsupervised learning
- 2.1. Gaussian mixture models
- 2.2. Manifold learning
- 2.3. Clustering
- 2.4. Biclustering
- 2.5. Decomposing signals in components (matrix factorization problems)
- 2.6. Covariance estimation
- 2.7. Novelty and Outlier Detection
- 2.8. Density Estimation
- 2.9. Neural network models (unsupe

No Free Lunch Theorem:



Hierarchical Clustering

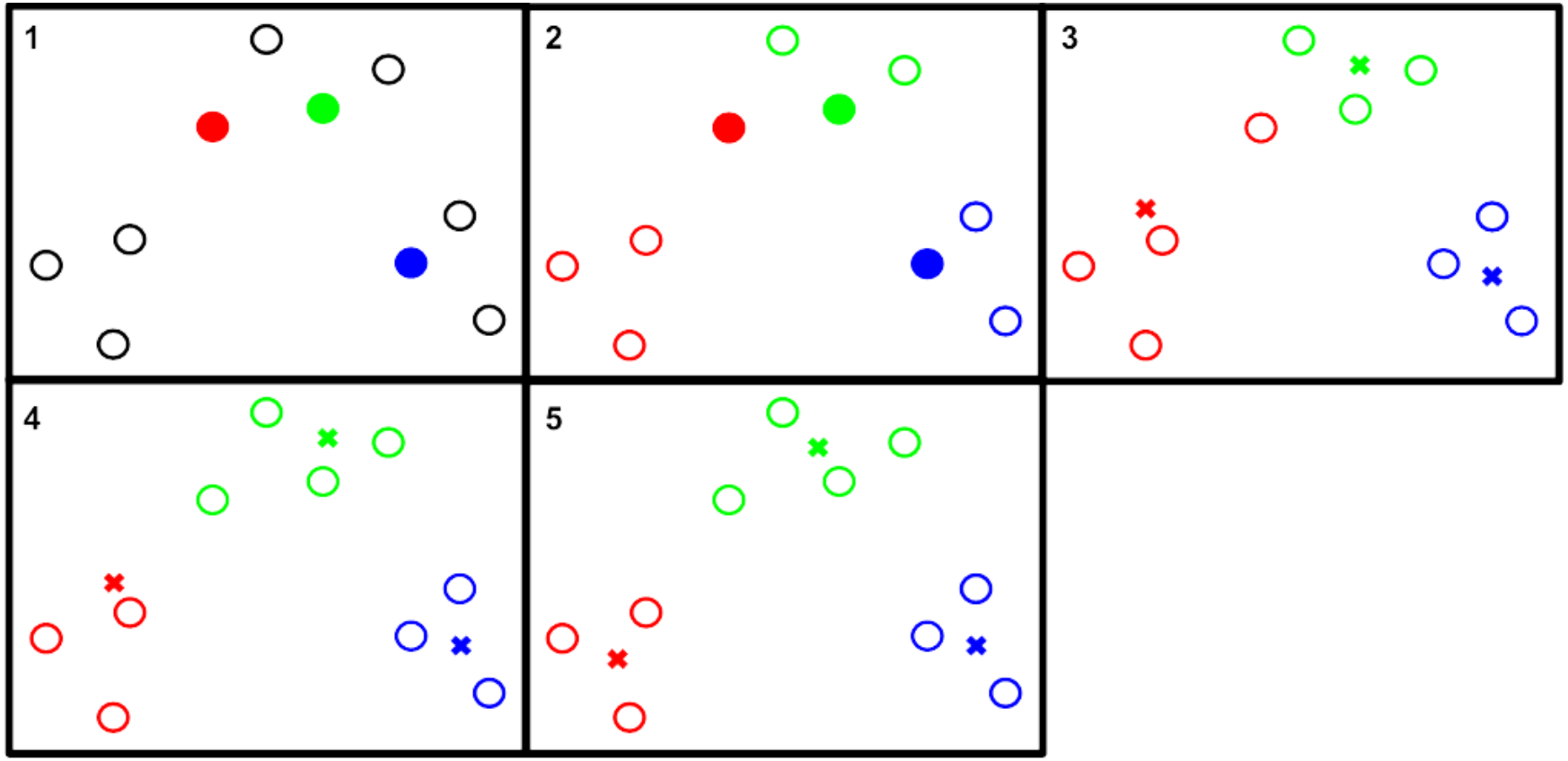
In each iteration, grouping the two nearest clusters:




```
model=AgglomerativeClustering(n_clusters=3)
model.fit(X); print(model.labels_)
```

K-Means Clustering

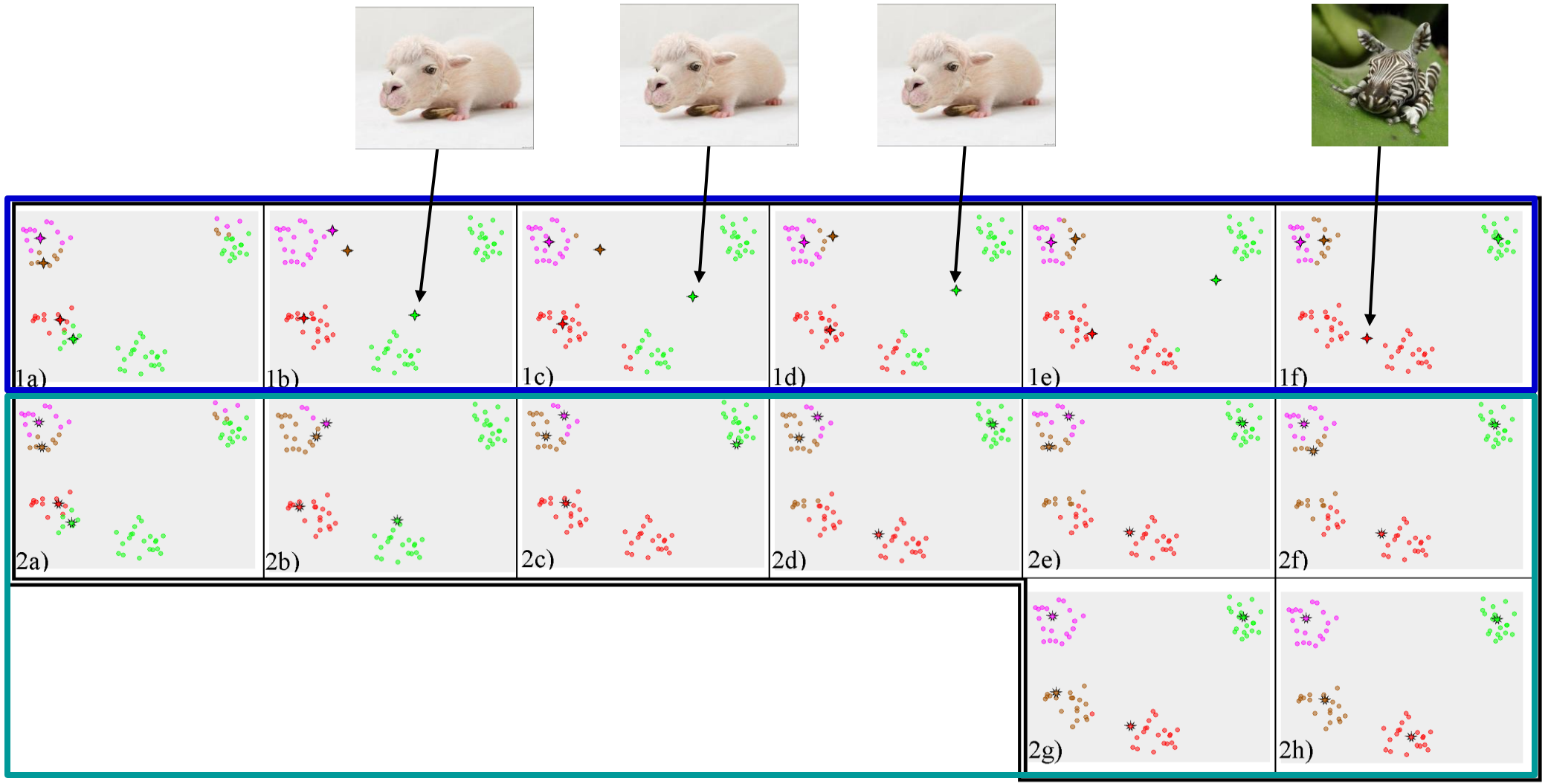
Assigning X_i to the nearest C_j & recompute C_j at the end:




```
model=KMeans(n_clusters=3)
model.fit(X); print(model.labels_)
```

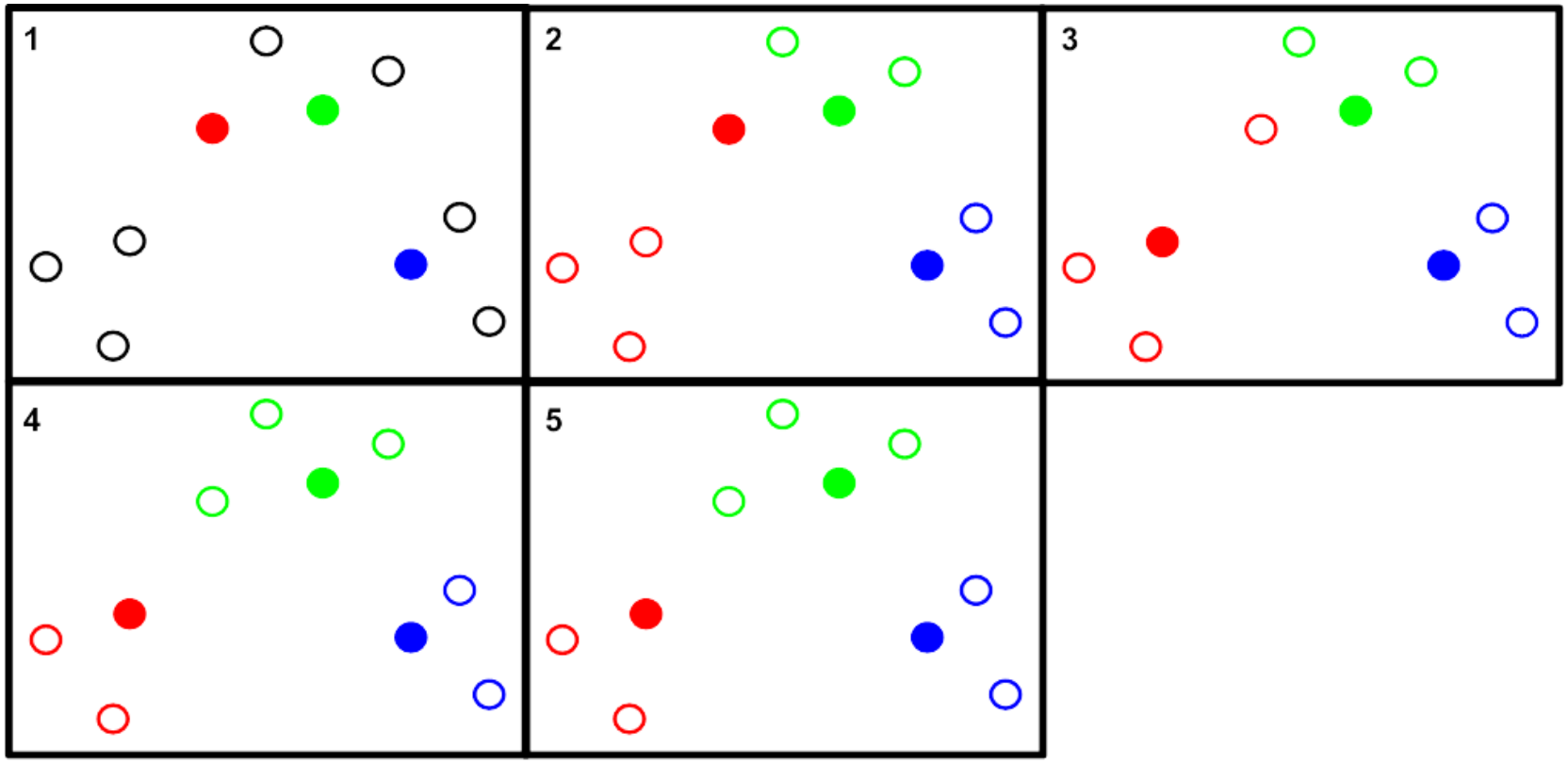
The problem of K-means

The cluster centers C_j are not representative of X_i:



K-Medoids Clustering

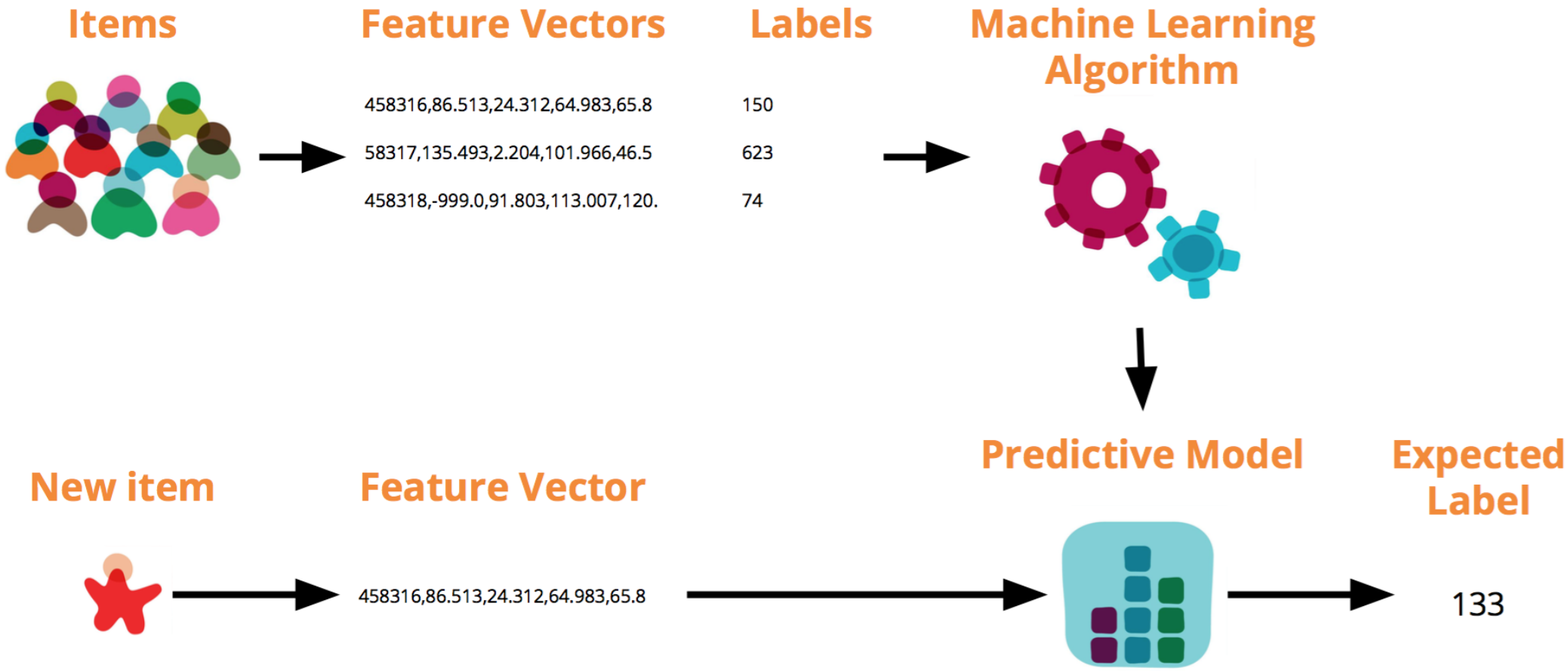
Unlike K-means, C_j is the most typical X_i of C_j:




```
model=KMedoids (n_clusters=3)
model.fit(X); print(model.labels_)
```

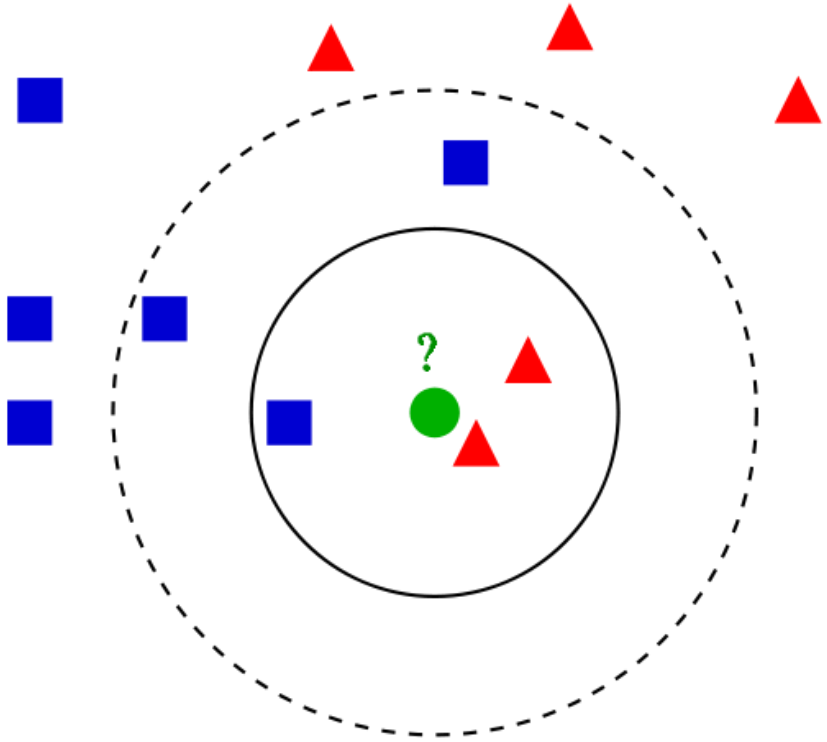
Supervised Learning

Unlike unsupervised learning, there are labels here:



K-Nearest Neighbors (kNN)

Prediction=The majority/average of Y_i of the nearest X_i



k=3 ●= ▲

k=5 ●= ■

k=11 ●= ?

```
clf=neighbors.KNeighborsClassifier(3) #try 1
clf.fit(X,Y) #training
print(np.mean(clf.predict(X)==Y)) #testing
```

Distance-Weighted kNN

Prediction=The weighted average of Y_i of the nearest X_i

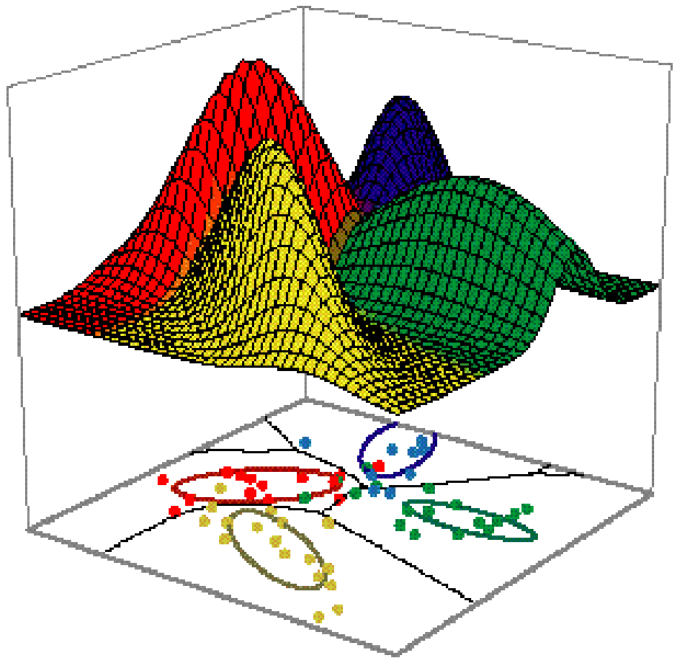
$$\hat{f}(x_q) = \frac{\sum_{i=1}^{k} w_i f(x_i)}{\sum_{i=1}^{k} w_i} \quad \text{with } w_i = \frac{1}{d(x_q, x_i)^2}$$

distance

```
clf=neighbors.KNeighborsClassifier(3,'distance')
clf.fit(X,Y) #training
print(np.mean(clf.predict(X)==Y)) #testing
```

Naive Bayes Classifier

Estimate the normal distributions of X_i for generalization:

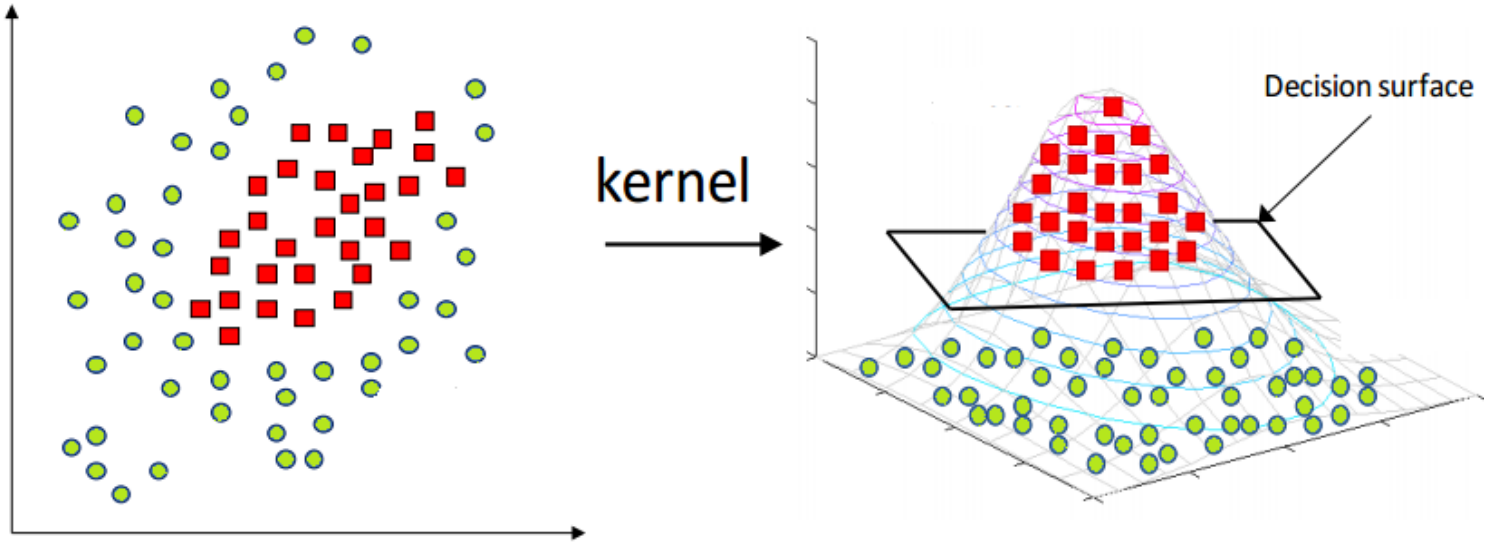


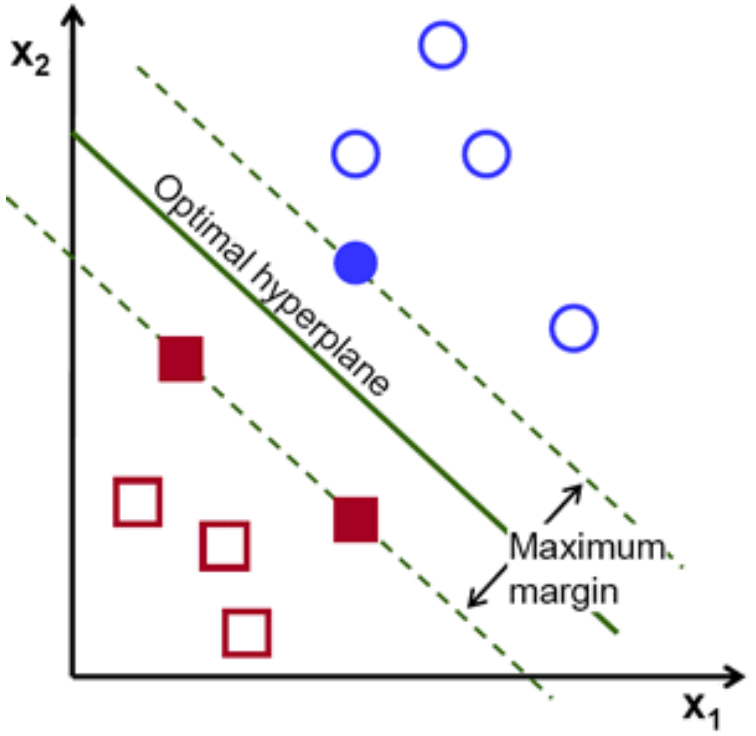
$$P(C_i|\vec{F}) = \frac{P(\vec{F}|C_i)P(C_i)}{P(\vec{F})}$$

```
clf=naive_bayes.GaussianNB()
clf.fit(X,Y) #training
print(np.mean(clf.predict(X)==Y)) #testing
```

Support Vector Machine

Using kernel trick + maximum margin to set boundary:






```
clf=svm.SVC()
clf.fit(X,Y) #training
print(np.mean(clf.predict(X)==Y)) #testing
```

Goals for today

Features of Machine Learning

explanatory vs. predictive modeling

Implementations of Machine Learning

supervised & unsupervised learning

Inferences from Machine Learning

common pitfalls

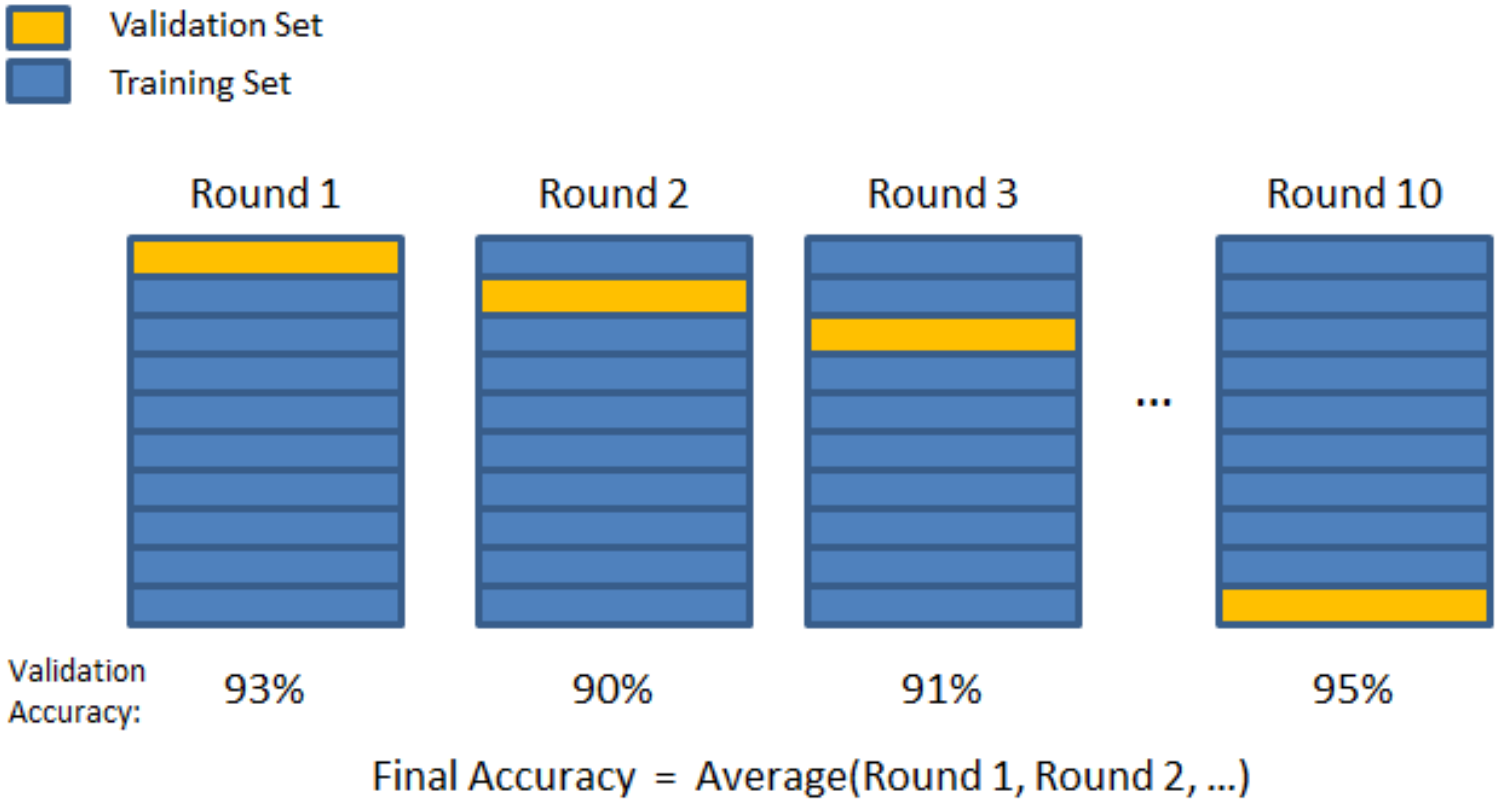
Common Pitfalls (1/3)

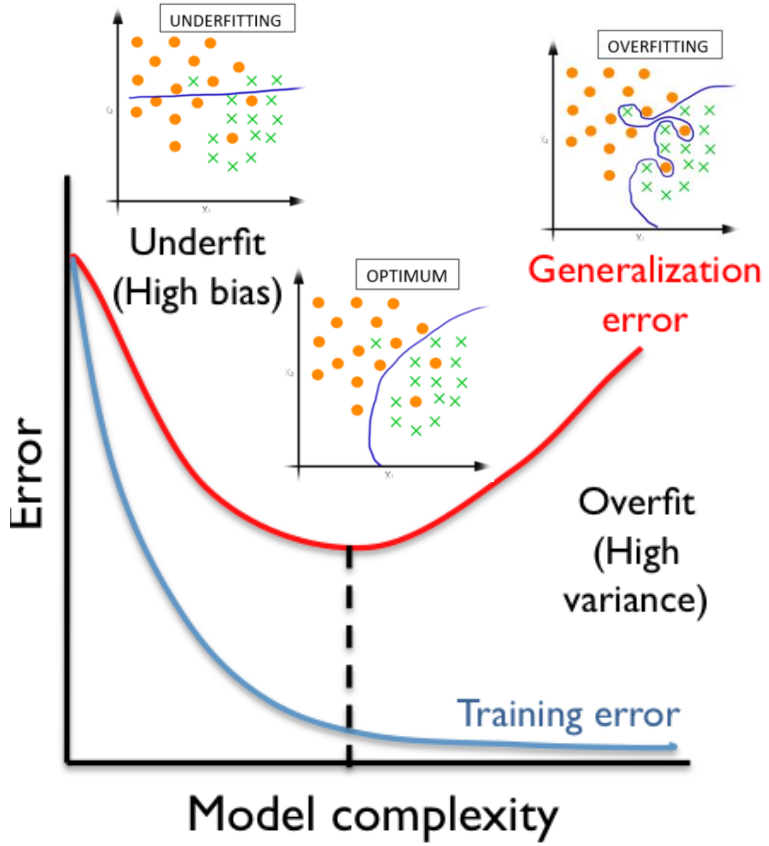
Why getting a **better**-than-chance accuracy if there is no relationship between X_i and Y_i?


```
X2=np.random.rand(150,4)
Y=iris.target
clf=neighbors.KNeighborsClassifier(3)
#clf=svm.SVC()
clf.fit(X2,Y);
pred=clf.predict(X2)
print(np.mean(pred==Y)) # 0.64
print(metrics.confusion_matrix(Y,pred))
```

K-Fold Cross-Validation

Unlike explanatory models, ML models are often cross-validated on unseen data to assess generalizability:





Common Pitfalls (2/3)

Why getting a **worse**-than-chance accuracy if there is no relationship between X_i and Y_i?

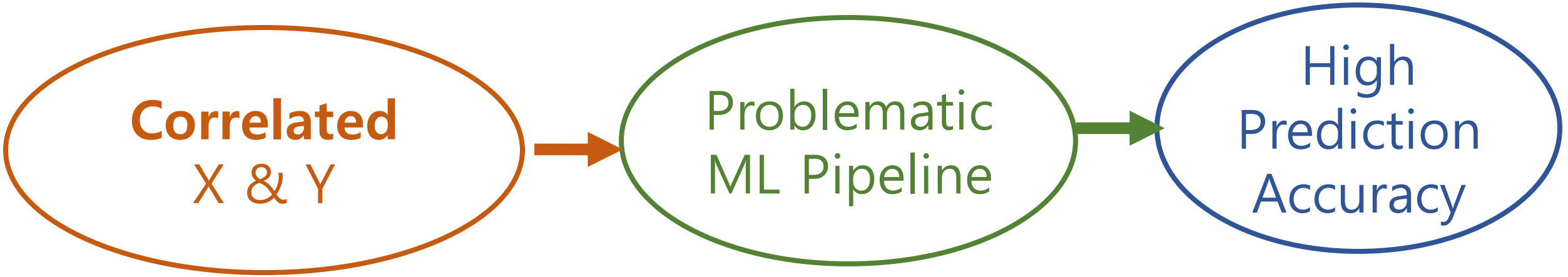

```
from sklearn import model_selection
clf=svm.SVC() # try other supervised classifiers
kf=model_selection.KFold(5)
s1=model_selection.cross_val_score(clf,X,Y,cv=kf)
s2=model_selection.cross_val_score(clf,X2,Y,cv=kf)
print(s1.mean(),s2.mean()) #0.89, 0.02
```

Common Pitfalls (3/3)

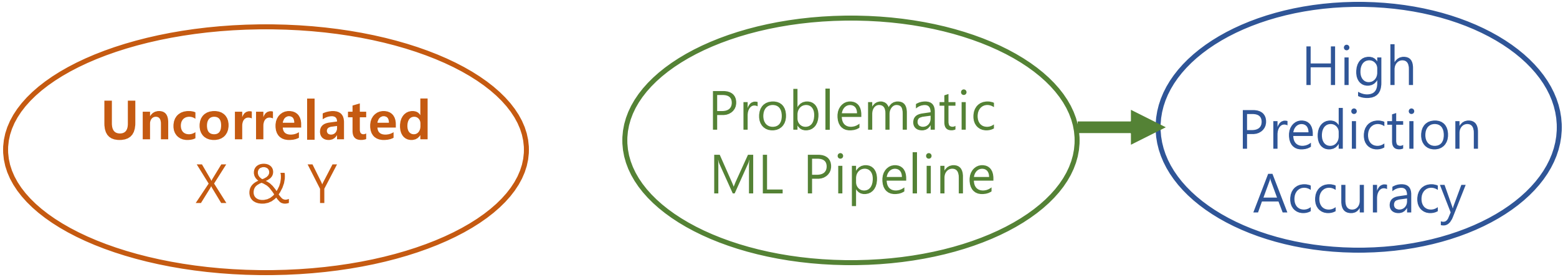
Why getting a **better**-than-chance accuracy if there is no relationship between X_i and Y_i?

```
from sklearn.model.selection import *
from sklearn.metrics import *
x=random.rand(100,3) # 3-d random features
y=random.permutation([0]*90+[1]*10) # 2 categories
clf=svm.SVC(); cv=KFold(100)
yp=cross_val_predict(clf,x,y,cv=cv)# leave-1-out
print('Accuracy:',mean(y==yp)) # mean accuracy =0.9
print('C. Matrix:\n',confusion_matrix(y,yp)) # c. matrix
```

Tren's advice for doing ML



Always test your ML pipelines with random X or Y:



Summary for today

Features of Machine Learning

explanatory vs. predictive modeling

Implementations of Machine Learning

supervised & unsupervised learning

Inferences from Machine Learning

common pitfalls

Game Over