Machine Learning

Saumya Bhatnagar

March 24, 2020

Table of contents I

Table of contents II

Box cox transformation

convert data to a normal distribution

CRISP-DM

Cross-industry standard process for data mining

Dimensionality Reduction

- PCA: is a linear method
- Random Forest

Cross-Validation Techniques

why cross-validation?

1. LOOCV (leave one out cv): $\downarrow$ bias, $\uparrow$variability
   n MSEs
2. k-fold: $\uparrow$k $\Longrightarrow$ $\downarrow$Variance; $\downarrow$bias; (generally k=10 with 10 MSEs)
3. stratified CV: $\downarrow$Variance & $\downarrow$bias
   uses stratified sampling for each fold
4. forward chaining or rolling origin

Under and Over Sampling

Univariate and Multivariate Analysis

Class Imbalance

Data Visualization I

Tableau and Qlikview

- compare distributions
  - pie-chart and donut (gives space to tell the total number)
  - Nightingale rose's chart
  - tree (tree-boxes)
  - scatter plot
  - Lollipop chart (trends)
  - bubble chart (3d of scatter)
- trends:
  - comparison 2-vars: bar-chart
  - radial bar chart
  - comp multiple vars: line chart
  - bridge charts

Data Visualization II

- stack char
- area or density plot (as a pctg of whole)

- statistical trends:
  - box and whisker plots
  - violin plots
- correlations or classifications
  - scatter plot
  - correlation plots
  - Likert scale
  - population pyt=ramids
- floe-map
  - hierarchy-tree
  - pyramid and funnel

Data Visualization III

- heat maps
- network diagrams
- timeline

- sun-burst and radial column
- Text Analytics - word cloud
- Sankey charts - flows, intensity, relationships bw vars
- waffle chart
- radar chart
- pictograms
- Maps

what is regression?

Regression analysis is a statistical technique to assess the relationship between an predictor variable and one or more response factors.
http://www.statisticshowto.com/probability-and-statistics/regression-analysis/ (go to definitions)

Linear vs Logistic

Basis	Linear Regression	Logistic Regression
Core Concept	The data is modelled using a straight line	The probability of some obtained event is represented as a linear function of a combination of predictor variables.
Used with	Continuous Variable	Categorical Variable
Output/Prediction	Value of the variable	Probability of occurrence of event
Accuracy and Goodness of fit	measured by loss, R squared, Adjusted R squared etc.	Accuracy, Precision, Recall, F1 score, ROC curve, Confusion Matrix, etc

Data is modelled using a straight line vs using a sigmoid function maps continuous x to cont y, vs maps cont x to binary y

Outcome Variable	**GLM Family**	**Link**	**Mean to Variance**
Continuous, unbounded	Normal or Standard Gaussian	Identity	
Continuous, non-negative	Gamma or inverse Gamma		
Discrete/ counts/ rate	Poisson Quassi-poisson or negative binomial	Log	Identity If not Identity
Count	Gamma		Over dispersion
Counts with multiple zero	Zero inflated poisson may be checked for fitting		
Binary	Binomial or Logistic regression		
Nominal	Multinomial regression		

Regression Model Selection Criteria

bias vs variance

Linear model: $\theta_0 + x\theta_1 \rightarrow$ increased bias, underfit
Polynomial model: $\theta_0 + \sum x\theta_n \rightarrow$ increased variance, overfit
So, **optimization on training data**, using

1. OLS
2. gradient descent,
3. max likelihood estimation (mle)

Types of Bias:

1. selection bias
2. survivorship bias (air plane)
3. under coverage bias

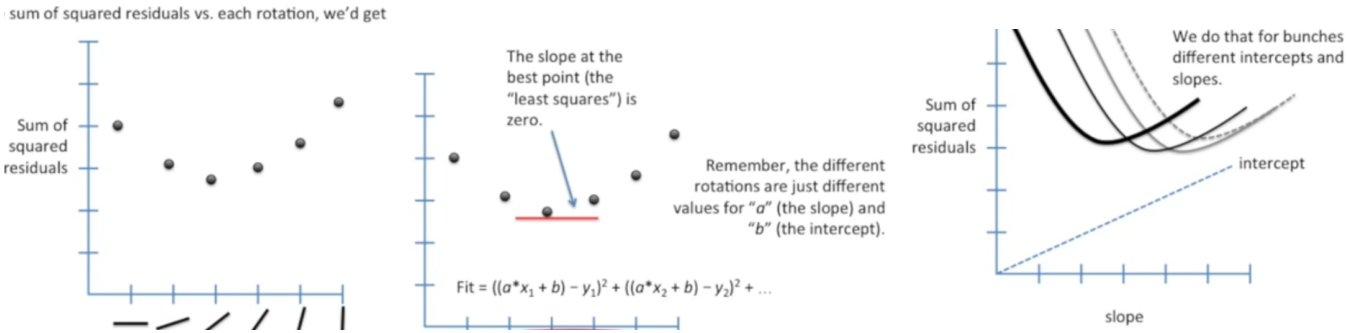
When overfit use Regularization Techniques

restrict freedom $\Rightarrow$ regularization
model selection algo
overfit $\Rightarrow$ fail to generalize to new examples
num datapoints ¿ num features or num parameters
reduce magnitude of features by penalization

L1 Regularization lasso regression	**L2 Regularization** Ridge regression
Sum of sq residual + λ $\lvert$slope$\rvert$	Sum of likelihoods + λ slope^2
can shrink slope to exact 0	asymptotically 0
LAR = Least absolute regression multiple solution	

Linear Regression using Least Squares

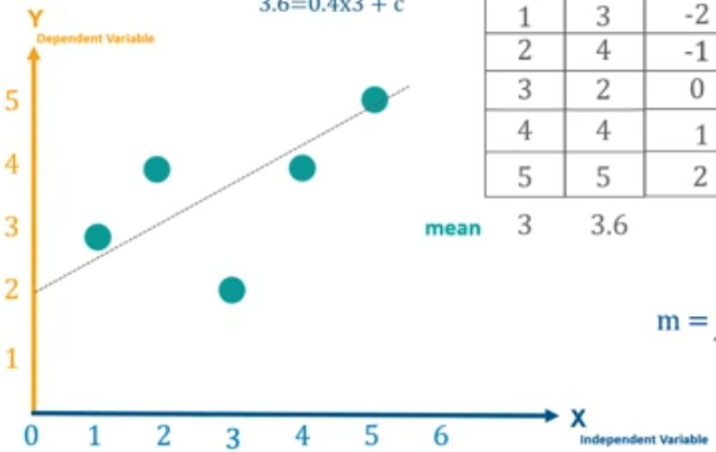
1. **fitting a line to the data**: as shown below
2. Find best fit using **Least Squares**
   sum of squared residuals $= \sum(y - \hat{y})^2$
3. Find goodness of fit using **R squared** method
   R squared (aka coefficient of determination) is a statistical measure of how well the data fits line
   low R squared doesn't always mean bad



Linear regression

$$y = mx+c$$

$$3.6 = 0.4 \times 3 + c$$



x	y	$x-\bar{x}$	$y-\bar{y}$	$(x-\bar{x})^2$	$(x-\bar{x})(y-\bar{y})$
1	3	-2	-0.6	4	1.2
2	4	-1	0.4	1	-0.4
3	2	0	-1.6	0	0
4	4	1	0.4	1	0.4
5	5	2	1.4	4	2.8
mean	3	3.6		$\Sigma = 10$	$\Sigma = 4$

$$m = \sum \frac{(x-\bar{x})(y-\bar{y})}{(x-\bar{x})^2}$$

x	y	$y-\bar{y}$	$(y-\bar{y})^2$	y_p	$(y_p-\bar{y})$	$(y_p-\bar{y})^2$
1	3	-0.6	0.36	2.8	-0.8	0.64
2	4	0.4	0.16	3.2	-0.4	0.16
3	2	-1.6	2.56	3.6	0	0
4	4	0.4	0.16	4.0	0.4	0.16
5	5	1.4	1.96	4.4	0.8	0.64
mean y	3.6		Σ 5.2			Σ 1.6

$$R^2 = \frac{1.6}{5.2} = \frac{\Sigma\,(y_p-\bar{y})^2}{\Sigma\,(y-\bar{y})^2}$$

characteristics of Linear Regression

outliers have big bad impact
Computational complexity O(n)
comprehensible and transparent

Linear vs Multiple Regression

Linear

fit line

Calculate R^2

$$R^2 = \frac{SS(mean_y) - SS(fit)}{SS(mean_y)}$$

cal F-score and p-val

$$F = \frac{\frac{SS(mean) - SS(fit)}{p_{fit} - p_{mean}}}{\frac{SS(fit)}{n - p_{fit}}}$$

Multiple

fit plane or higher dimensional

Cal R^2

Adjust R^2 to compensate for additional parameters

Multiple Regression

Fit plane of higher dimensional obj to the data

Logistic Regression

content...

Three methods to classifier

1. model a classification rule - knn, decision tree, perceptron, svm
2. model the probability of class membership given input data - perceptron with cross-entropy cost
3. make a probabilistic model of data within each class - naive bayes

1 & 2 are discriminative classifications
3 is generative classification
2 & 3 probabilistic classification

Accuracy: most used metric
Precision: out of those predicted positive, how many of them are actual positive
when the costs of False Positive is high
F1 Score: harmonic mean of precision and recall; is needed when you want to seek a balance between Precision and Recall
Precision is how sure you are of your true positives whilst recall is how sure you are that you are not missing any positives.
Choose Recall if the idea of false positives is far better than false negatives, in other words, if the occurrence of false negatives is unaccepted/intolerable, that you'd rather get some extra false positives(false alarms) over saving some false negatives, like in our diabetes example.

You'd rather get some healthy people labeled diabetic over leaving a diabetic person labeled healthy. Choose precision if you want to be more confident of your true positives. for example, Spam emails. You'd rather have some spam emails in your inbox rather than some regular emails in your spam box. So, the email company wants to be extra sure that email Y is spam before they put it in the spam box and you never get to see it.
Choose Specificity if you want to cover all true negatives, meaning you don't want any false alarms, you don't want any false positives. for example, you're running a drug test in which all people who test positive will immediately go to jail, you don't want anyone drug-free going to jail. False positives here are intolerable.

- Accuracy value of 90% means that 1 of every 10 labels is incorrect, and 9 is correct.

- Precision value of 80% means that on average, 2 of every 10 diabetic labeled student by our program is healthy, and 8 is diabetic.
- Recall value is 70% means that 3 of every 10 diabetic people in reality are missed by our program and 7 labeled as diabetic.
- Specificity value is 60% means that 4 of every 10 healthy people in reality are miss-labeled as diabetic and 6 are correctly labeled as healthy.

Accuracy, precision, recall, f1 score

Accuracy, precision, recall, f1 score

Accuracy, precision, recall, f1 score

Decision Tree

content...

Random Forest

content...

"Help me understand our customers better so that we can market our products to them in a better manner!

Monothetic: Cluster members have some common property Expectation–Maximization (EM) Clustering using Gaussian Mixture Models (GMM)

Polythetic: Cluster members are similar to each other. Distance between elements define relationship

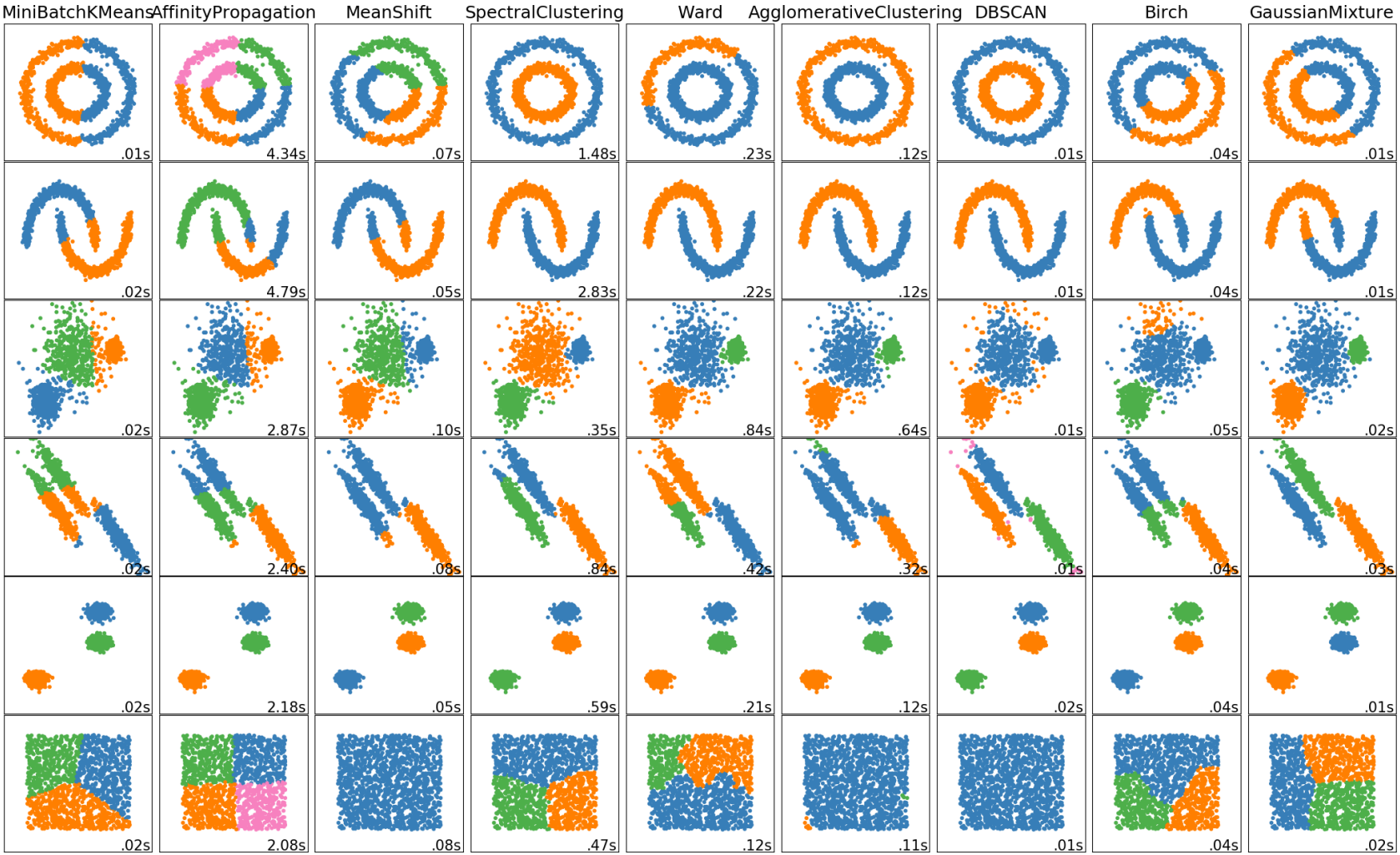
Hard Clustering: each data point either belongs to a cluster completely or not

Soft Clustering: a probability or likelihood of that data point to be in those clusters is assigned.

Clustering Models

Connectivity models	Distribution models	Centroid models	Density models
data points closer in data space exhibit more similarity to each other than the data points lying farther away	how probable is it that all data points in the cluster belong to the same distribution (e.g: Normal, Gaussian)	iterative clustering algorithms in which the notion of similarity is derived by the closeness of a data point to the centroid of the clusters	isolates various different density regions and assign the data points within these regions in the same cluster
hierarchical clustering	Expectation-maximization	K-Means, k-median	mean-shift, DBSCAN and OPTICS
Approaches: 1) Top-bottom, 2) bottom-up	EM uses multivariate normal distributions	DZA	DBSCAN uses radius ϵ and Center c
lacks scalability for handling big datasets, Time complexity: $O(n^2)$	These models often suffer from over-fitting. Prior knowledge to define num clusters	important to have prior knowledge of the dataset. results change in every trial	DBSCAN doesn't perform as well when the clusters are of varying density
Results are reproducible	more flexibility in terms of cluster covariance due to $\mu and \sigma$ (additional σ)	can handle big data , Time complexity: O(n)	DBSCAN identifies outliers as noises
chk1	elliptical shape (since we have a standard deviation in both the x and y directions	work well when the shape of the clusters is hyper spherical (like circle in 2D, sphere in 3D)	DBSCAN: can find arbitrarily sized and arbitrarily shaped clusters
Angola	GMMs support mixed membership since is probability based	AGO	DBSCAN: drawback in high-dimensional data since the distance threshold ϵ becomes challenging to estimate

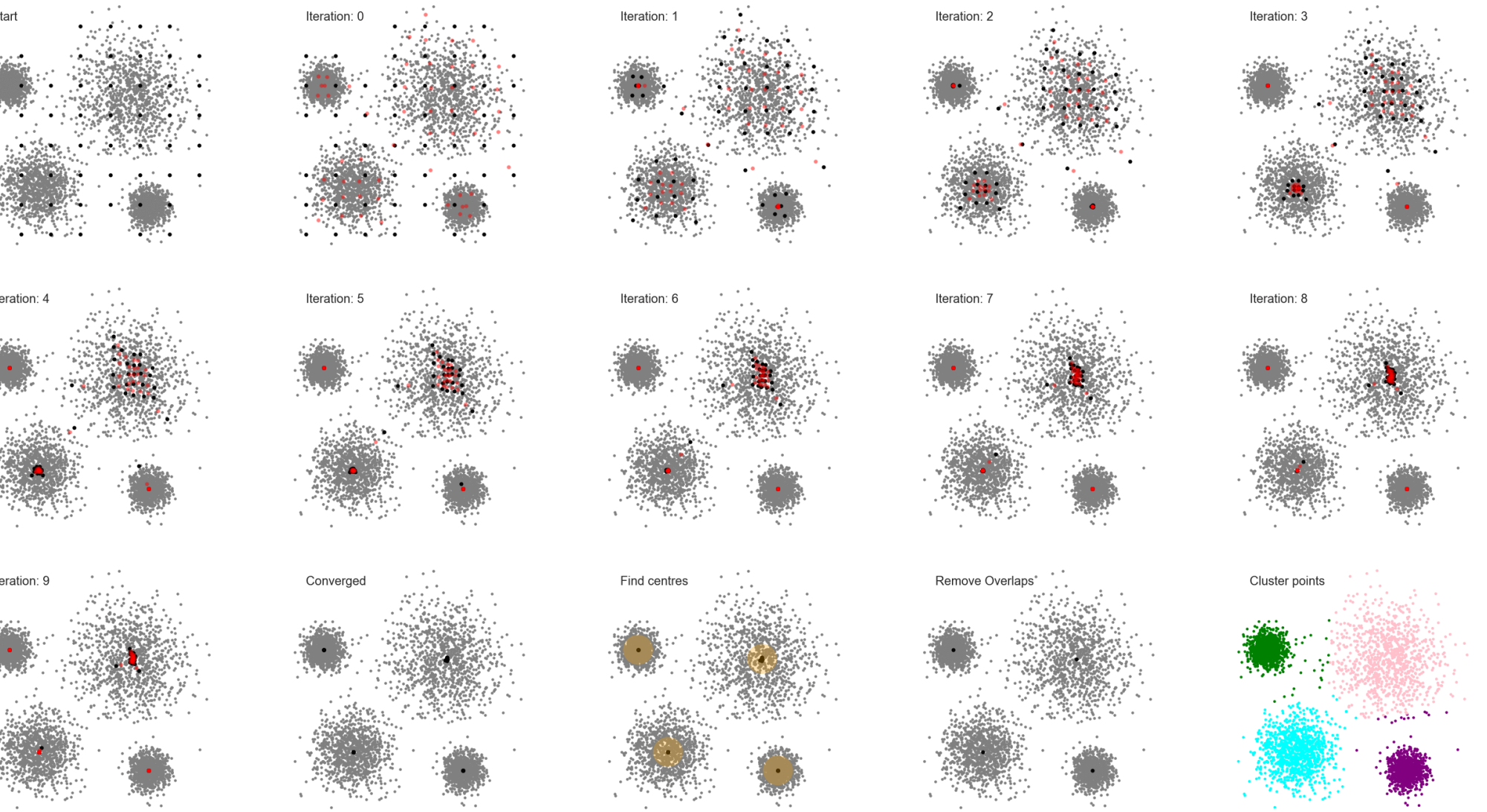
Clustering Types



mean-shift clustering

consider a set of points in two-dimensional space
a circular sliding window C centered and radius r as the kernel
hill-climbing algorithm that involves shifting this kernel iteratively to a higher density ($\propto$ number of points) region until convergence
At every iteration,
- shift the center point to the mean of the points within the window (hence the name)
-gradually move towards areas of higher point density
- until no longer increase in the density
- When multiple sliding windows overlap the window containing the most points is preserved. The data points are then clustered according to the sliding window in which they reside.

Density Models



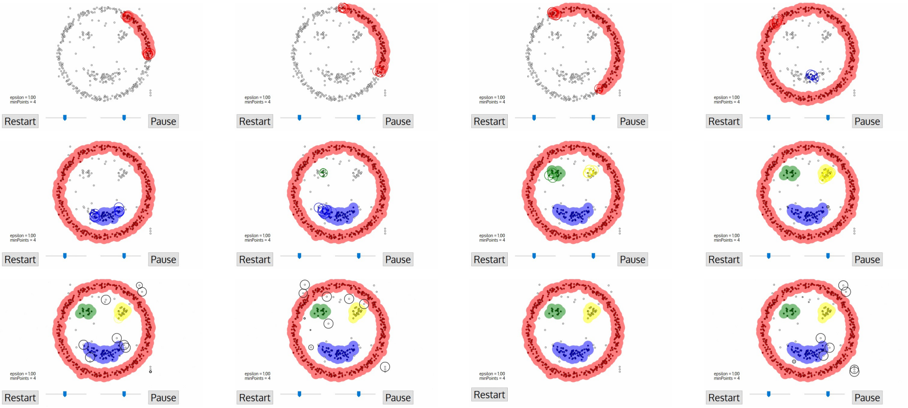
Density-Based Spatial Clustering of Applications with Noise-DBSCAN

-label all data point to be unvisited. For all unvisited points:

1. All points which are within the ϵ distance are neighborhood points (part of the same cluster)
2. If neighborhood points ¿= minPoints, then the clustering process starts and the current data point becomes the first point in the new cluster - Otherwise, mark the point as noise - In both cases that point is marked as "visited"
3. repeated for all of the new points in the cluster group
4. next an new unvisited point is retrieved and processed

Since at the end of this all points have been visited, each point will have been marked as either belonging to a cluster or being noise.

Density Models



Hierarchical DBSCAN - HDBSCAN

content...

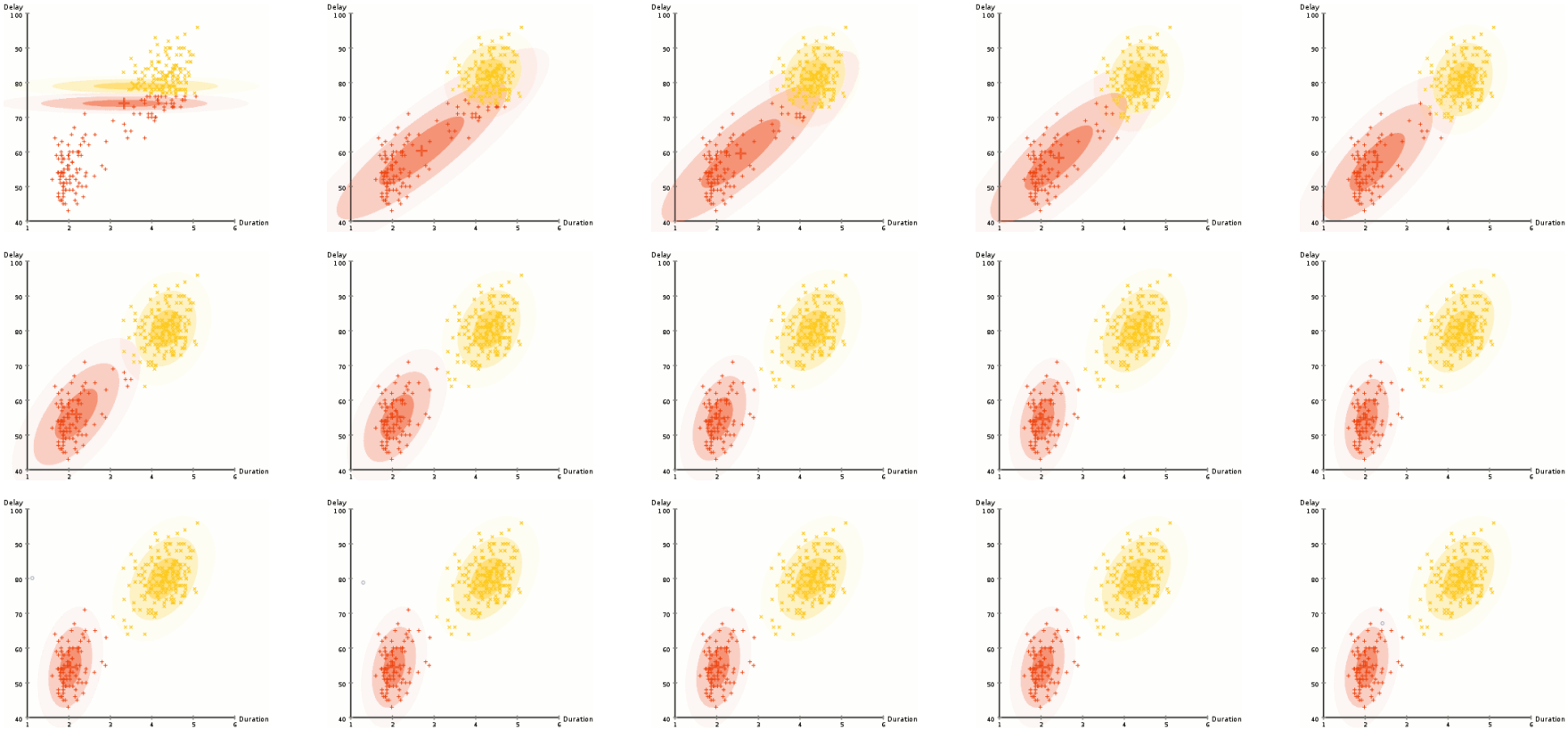
Gaussian Mixture Models (GMMs)

Assumption: the data points are Gaussian distributed (parameters: the mean and the standard deviation)! Each Gaussian distribution is assigned to a single cluster. To find the parameters of the Gaussian for each cluster, use an optimization algorithm called Expectation–Maximization (EM).

Expectation–Maximization (EM) using GMM

choose num of clusters
compute the probability that each data point belongs to a particular cluster. With a Gaussian distribution we are assuming that most of the data lies closer to the center of the cluster.
From probabilities $\rightarrow$ recompute set of parameters such that we maximize the probabilities of data points within the clusters
We compute these new parameters using a weighted sum of the data point positions, where the weights are the probabilities of the data point belonging in that particular cluster.
Repeat till convergence

Distribution Models



K means

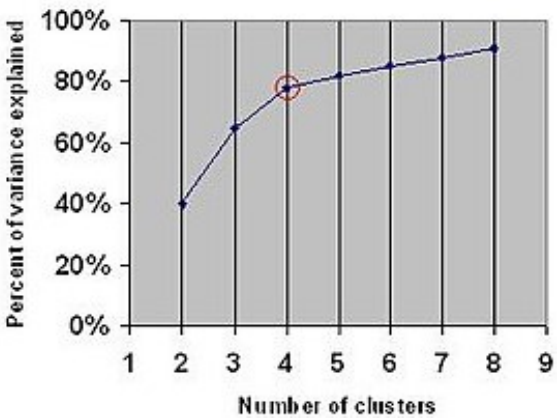
iterative clustering algorithm that aims to find local maxima in each iteration

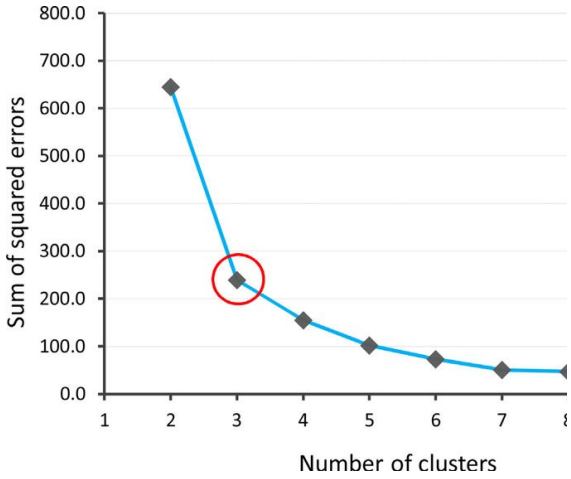
take a quick look at the data and choose k (num clusters)

assign data points to the cluster $\leftrightarrow$ compute cluster centroid

repeat to reduce variation error

elbow method



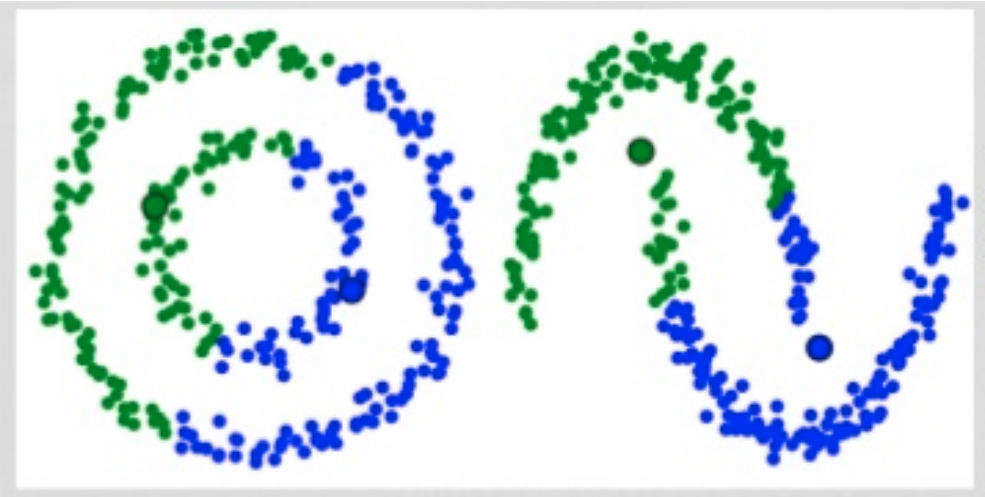


k-median

K median vs kmeans: instead of recomputing the group center points using the mean (like in K-Means) we use the median vector of the group. This method is **less sensitive to outliers** (because of using the Median) but is **much slower for larger datasets as sorting is required** on each iteration when computing the Median vector

mean-shift vs kmeans: Instead of selecting the number of clusters as mean-shift automatically discovers this (advantage), the selection of the window size/radius "r" can be non-trivial.

kmeans fail



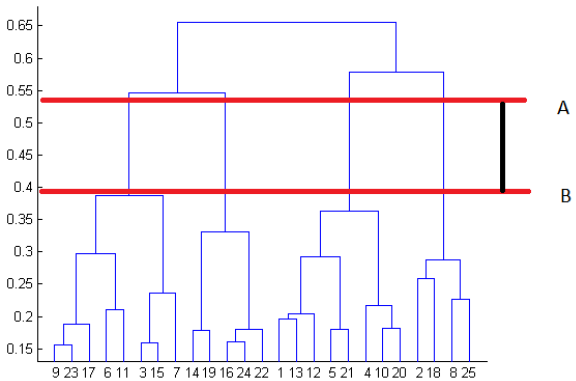
K-Means is actually a special case of GMM in which each cluster's covariance along all dimensions approaches 0

Agglomerative Hierarchical Clustering

The decision of dividing into or merging **two** clusters is taken on the basis of closeness of these clusters. Metrics for deciding the closeness of two clusters:
Euclidean distance: $||a-b||_2 = \sqrt{\sum(a_i - b_i)}$
Squared Euclidean distance: $||a-b||_2^2 = \sum(a_i - b_i)^2$
Manhattan distance: $||a-b||_1 = \sum |a_i - b_i|$
Maximum distance: $||a-b||_{INFINITY} = \max_i |a_i - b_i|$
Mahalanobis distance: $\sqrt{(a-b)^T S^{-1}(-b)}$
Maybe, use **average linkage** which defines the distance between two clusters to be the average distance between data points in the first cluster and data points in the second cluster.

hierarchical agglomerative clustering (HAC) or bottom-up

1. Each data point as a single cluster
2. select a distance metric
3. Iterate till convergence
   - combine two clusters with the smallest average linkage



The height in the dendrogram at which two clusters are merged represents the distance between two clusters in the data space. take 4 clusters as the red horizontal line in the dendrogram covers maximum vertical distance AB.

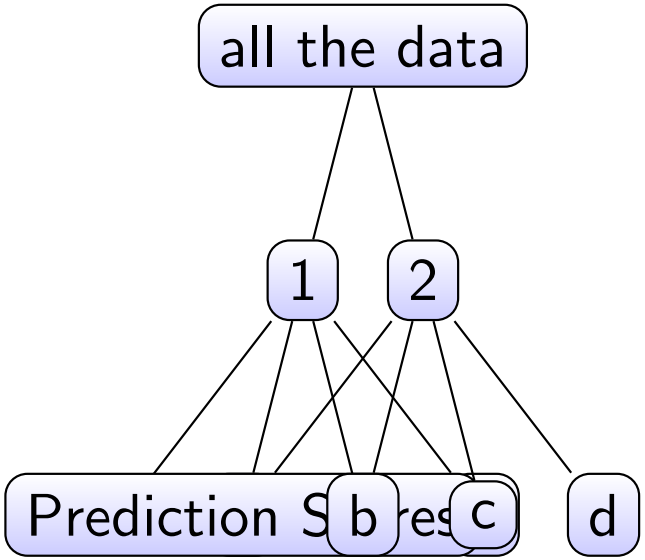
Support Vector Machines

what are support vectors?
which kernel (linear, radial, polynomial,sigmoid/logistic, gaussian/RBF - radial basis func)
can handle categorical/numerical data?
In a N-dimensional Euclidean space, plot a hyper-plane $\Rightarrow$ (n-1) dim subset of n-dimensional Euclidean space dividing it into 2

sklearn code

```
clf = sklearn.svm.SVC(kernel='linear')
clf.fit(X,y)
clf.predict([])
```

C: smooth decision boundary (C$\downarrow$) & classified training points (C$\uparrow$)
Gamma: $\uparrow \gamma \Rightarrow$ points closer to hyperplance will have more wt



DT = Low bias high variance

Uses:

- D^3 algo
- Entropy $(-p \log_2 p - q \log_2 q)$
- Information Gain

Gini impurity or entropy? The truth is, most of the time it does not make a big difference: they lead to similar trees. Gini impurity is slightly faster to compute, so it is a good default. However, when they differ, Gini impurity tends to isolate the most frequent class in its own branch of the tree, while entropy tends to produce slightly more balanced trees.

Linear Model vs decision trees:

linear models make assumptions... dec trees may over-fit LM $\Rightarrow$ parametric (predetermined number of parameters $\Rightarrow$ limited degree of freedom $\Rightarrow$ reducing the risk of overfitting (but increasing the risk of underfitting) DT $\Rightarrow$ non-parametric (number of parameters is not determined prior to training) $\Rightarrow$ over-fitting

Standard statistical tests, such as the χ^2 test, are used to estimate the probability that the improvement is purely the result of chance

(H_0). If this probability, called the pvalue, is higher than a given threshold (typically 5%, controlled by a hyperparameter), then the node is considered unnecessary and its children are deleted. The **pruning** continues until all unnecessary nodes have been pruned
cons of decision tree: orthogonal boundaries, not a good model; sensitive to small variations;
Voting classifiers: hard voting (majority-vote classifier) & soft voting
even if each classifier is a weak learner (meaning it does only slightly better than random guessing), the ensemble can still be a strong learner (achieving high accuracy), provided there are a sufficient number of weak learners and they are sufficiently diverse.
this is only true if all classifiers are perfectly independent, making uncorrelated errors, which is clearly not the case since they are

trained on the same data. They are likely to make the same types of errors, so there will be many majority votes for the wrong class, reducing the ensemble's accuracy.

Ensemble methods work best when the predictors are as independent from one another as possible. One way to get diverse classifiers is to train them using very different algorithms. This increases the chance that they will make very different types of errors, improving the ensemble's accuracy.

Sequential - Boosting

- adaptive Boosting:
  start with equal wt to decision stump, then misclassified gets higher weights for the next decision stump
- Gradient Boosting:
  optimize loss funct of prev learner
  additive model regularize the loss function
- XG Boosting: extreme gradient
  computational speed and model efficiency
  distributed ML: create decision tree ||-ly
  out of core computing
  cache optimization
  $\uparrow$ bias, $\downarrow$ variance
  uses DT upto some depth

Parallel - Bagging - Random Forest

DT = $\downarrow$ bias, $\uparrow$ variance
RF = DT1 + DT2 + ...
= $\downarrow$ bias, $\downarrow$ variance
classification = votes
regression = average

Stacking

content...

title

methods for putting machine learning models into production, and to determine which method is best for which use case

Training-stochastic

training algorithm used by Scikit-Learn : stochastic; may get very different models even on the same training data (unless you set the random_state hyperparameter).

numpy vs pandas vs scipy

content...

Trend Analysis

content...

ARIMA Modeling I

Auto-regressive Integrated Moving Average

1. AR Model (p)
2. MA Model (q)
3. ARMA Model (p,q)
4. **ARIMA(p,d,q)**
5. **SARIMA(p,d,q)(P,D,Q)$_m$**

ARIMA Modeling II

5.1 S [$(P,D,Q)_m$]: Seasonality; m = Seasonality; (P,D,Q) are analogous of (p,d,q) except for seasonal components
what is Seasonality? Repeating patterns within a year (/weekly/monthly). Remove Seasonality:

Math alert

$Z_t = S_{t-365} - S_t$

What is Cycle?:

- repeating patterns over years
- not as predictable, I cycle might take 2 years, II might take 3 years, III might take 2.5 years

5.2 AR [p]: predict for today based on previous value

ARIMA Modeling III

5.3 I [d]: TS has forwards or backwards trends, so use differencing to make it stationary

Math alert

$Z_t = S_{t-1} - S_t = \phi Z_{t-1} + \theta \epsilon_{t-1} + \epsilon_t$

To Recover, $S_k = Z_{k-1} + S_{k-1} = \sum_{i=1}^{l} Z_{k-i} + S_l$

ARIMA Modeling IV

5.4 MA [q]: error from prev period t make prediction about today

Math alert

Actual, $X_t = \mu + \phi_1 \epsilon_{t-1} + \phi_2 \epsilon_{t-2} + ... + \phi_q \epsilon_{t-q} + \epsilon_t$

Pred, $\hat{X}_t = \mu + \phi_1 \epsilon_{t-1} + \phi_2 \epsilon_{t-2} + ... + \phi_q \epsilon_{t-q}$ (won't account for error)

where,

ϵ_{t-q} and ϵ_t : error from q time periods ago and current time period

S_t, S_{t-1}, S_{t-2}: avg value @t, t-1, t-2

ACF: Auto-correlation factor = Corr(S_{t-2}, S_t)

Pearson's r correlation: measures the strength relation btw 2 variables

ARIMA Modeling V

AR $\Rightarrow$ regression $\Rightarrow$ auto regression (predict something based on past values of same "something") (PACF)
MA $\Rightarrow$ (ACF)

Trend Analysis

content...

Trend Analysis

content...

Trend Analysis

content...

Thank You!