Introduction Supervised Machine Learning

MIKE-E, 1^{st} semester, autumn term 2017

November 26, 2019

Daniel S. Hain
Assistant Professor, PhD.
Data Science - Computational Social Science – Innovation Studies
Contact: `dsh@business.aau.dk`
Web: `aau.profile` | `google.scholar` | `github`

Department of Business and Management
IKE, DRUID, GLOBELICS
Aalborg University
Denmark

AALBORG UNIVERSITY DENMARK

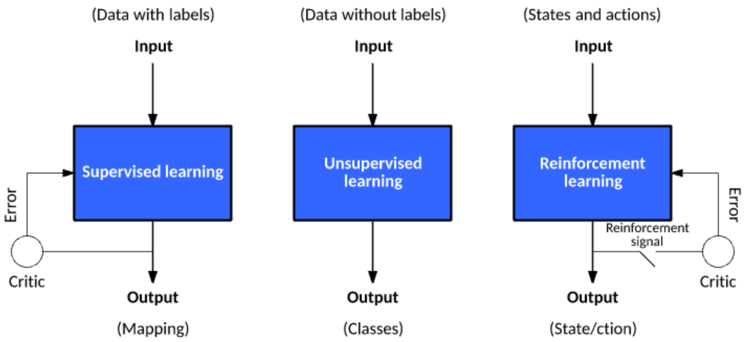
MIKE MSc in Innovation and Entrepreneurship

IKE Innovation, Knowledge and Economic dynamics

Agenda today

Introduction to Machine Learning

Reminderer



Aalborg University Denmark

Reminderer

Table 3: GLM regression results (link=Logit): Investments (Foreign only, Post-PSM)

	Dependent variable: Investor participates in round					
	EU	KINGS	EU	KINGS	EU	KINGS
	(1)	(2)	(3)	(4)	(5)	(6)
			Deal characteristics (controls)			
round.seed	−0.349	−0.660*	−0.292	−0.687*	−0.375	−0.650
	(0.658)	(0.394)	(0.666)	(0.396)	(0.666)	(0.399)
round.venture	0.021	−0.563	0.021	−0.637	0.080	−0.561
	(0.722)	(0.437)	(0.727)	(0.441)	(0.725)	(0.444)
inv.sum	−0.042	0.039**	−0.067	0.041**	−0.069	0.043**
	(0.072)	(0.016)	(0.076)	(0.017)	(0.076)	(0.017)
log(round.usd)	−0.126	0.0002	−0.063	0.008	−0.117	−0.001
			VC characteristic			
inv.exp	−0.010***	−0.0005	−0.009***	−0.006	−0.011***	−0.0005
	(0.003)	(0.001)	(0.003)	(0.004)	(0.003)	(0.001)
inv.exp.dc	0.171***	0.254***	0.182***	0.274***	0.181***	0.210***
	(0.021)	(0.076)	(0.023)	(0.078)	(0.024)	(0.080)
			PC human capital signals			
	(0.104)	(0.059)	(0.106)	(0.060)	(0.105)	(0.060)
comp.press	0.001	−0.0004	0.004*	−0.001	0.003	0.001
	(0.002)	(0.002)	(0.002)	(0.003)	(0.002)	(0.003)
per.entrp	−0.141	0.275	−0.061	0.444**	−0.295	0.392**
	(0.208)	(0.195)	(0.247)	(0.215)	(0.303)	(0.221)
per.edu	0.109	−0.096	0.097	−0.114	0.187	0.110**
	(0.151)	(0.085)	(0.196)	(0.093)	(0.211)	(0.064)
per.mng	−0.022	−0.001	0.034	−0.071	0.012	−0.025
	(0.090)	(0.076)	(0.132)	(0.083)	(0.134)	(0.082)
			*VC * PC interaction terms*			
per.entrp*inv.exp			−0.004	0.015*		
			(0.007)	(0.008)		
comp.press*inv.exp			−0.0001*	0.0001*		
			(0.0001)	(0.0001)		
per.edu*inv.exp			−0.001	−0.0004		
			(0.006)	(0.003)		
per.mng*inv.exp			−0.001	0.006**		
			(0.004)	(0.003)		
per.entrp*inv.exp.dc					0.023	0.108**
					(0.028)	(0.047)
comp.press*invr.exp.dc					−0.0003	−0.009
					(0.0002)	(0.007)
per.edu*inv.exp.dc					0.012	−0.140**
					(0.021)	(0.157)
per.mng*inv.exp.dc					−0.002	0.173**
					(0.012)	(0.081)
comp.country	*Yes*	*Yes*	*Yes*	*Yes*	*Yes*	*Yes*
Observations	1,002	894	1,002	894	1,002	894
Log Likelihood	-229.100	-376.900	-224.700	-342.600	-227.900	-314.100

Note: *p<0.1; **p<0.05; ***p<0.01 , standard errors in parentheses

The Econometric Approach

- Mostly interested in producing good *parameter estimates*
- Aim: Construct models with unbiased estimates for some parameter β, capturing the relationship between a variable of interest x and outcome y.
- Such models are supposed to be "structural", where we not simply aim to reveal correlations between x and y, but rather a causal effect of directionality $x \rightarrow y$, robust across a variety of observed as well as up to now unobserved settings. A
- Therefore, we carefully draw from existing theories and empirical findings, and apply logical reasoning to formulate hypotheses which articulate the expected direction of such causal effects.
- Typically, we do so by studying one or more bivariate relationships under *cetris paribus* conditions (everything else equal).
- Main concern here is to minimize the standard errors ε of our β estimates.
- Not overly concerned with overall predictive power (eg. R^2) of those models, but about various type of endogeneity issues, leading us to develop sophisticated *identification strategies*

The Machine Learning Approach

- To a large extend driven by the needs of the private sector, data analysis is gear towards producing good *predictions* of outcomes.
  - Recommender systems: Amazon, Netflix, Sportify ect.
  - "Risk scores": Eg.g likelihood that a particular person has an accident, turns sick, or defaults on their credit.
  - Image classification: Finding Cats & Dogs online
- Often rely on a lot of data, in terms of the number of observations as well as possible predictors.
- Usually not overly concerned with the properties of parameter estimates, but very rigorous in optimizing the overall prediction accuracy.
- Aims at exploiting degrees of freedom.
- Often uses way more flexibility wrt. the functional form, also often apply non-parametric approaches.
- Yet, since they cannot rely on causality driving the results, and are not to thorough on identification generally, in need of other verification techniques.

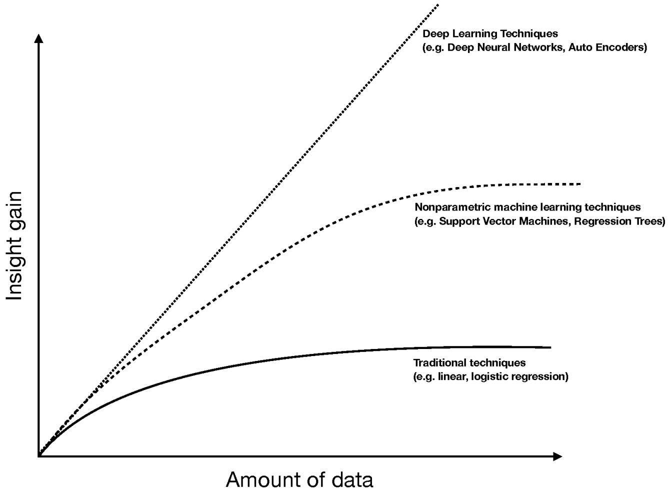
Supervised Machine Learning

- The majority of practical machine learning uses supervised learning.
- You have input variables (x) and an (known, labeled) output variable (Y) and you use an algorithm to learn the mapping function ($Y = f(X)$)from the input to the output.
  - **Classification:** A classification problem is when the output variable is a category, such as *red* or *blue*, or *life* and *die*.
  - **Regression:** A regression problem is when the output variable is a real value, such as *EURO* or *weight*.

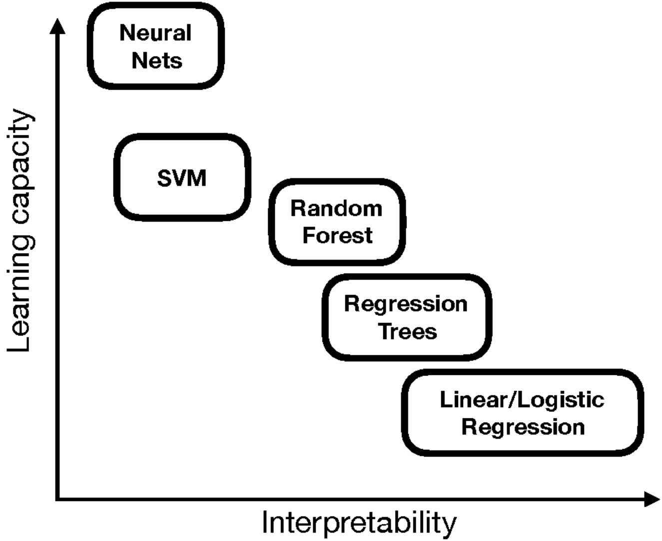
Unsupervised Machine Learning

- Unsupervised learning is where you only have input data (X) and no corresponding (known, labeled) output variables.
- Aims at modeling the underlying structure or distribution in the data in order to learn more about the data.
- There is no correct answers and there is no teacher. Algorithms are left to their own devises to discover and present the interesting structure in the data.
  - **Clustering:** A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior.
  - **Association:** An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.

Complexity vs. Learning Capacity



Performance vs. Interpretability



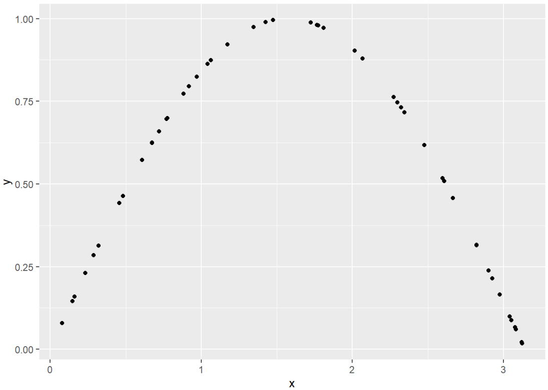
Supervised Machine Learning 101

- Rule-of-thumb: Richer and more complex functional forms and algorithms tend to be better in predicting complex real world pattern.
- This is particularly true for high-dimensional (big) data.
- However, flexible algorithms at one point become so good in mimicking the pattern in our data that they overfit, meaning are to much tuned towards a specific dataset.

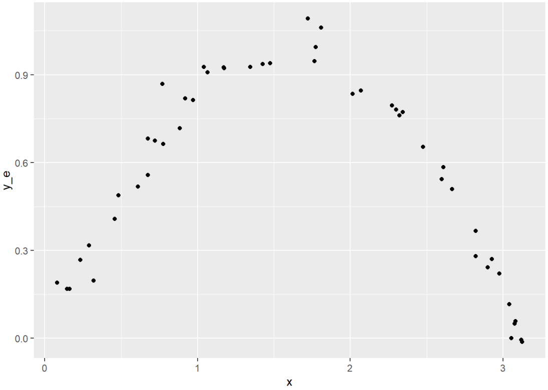
Generally, we call this tension the bias-variance tradeoff, which we can decompose in the two components:

- Bias Error: The simplifying assumptions made by a model to make the target function easier to learn. Generally, simple parametric algorithms have a high bias making them fast to learn and easier to understand but generally less flexible. In turn, they have lower predictive performance on complex problems that fail to meet the simplifying assumptions of the algorithms bias.
- Variance Error: Variance is the amount that the estimate of the target function will change if different data was used. Generally, nonparametric and complex ML algorithms tend to that have a lot of flexibility have a high variance.

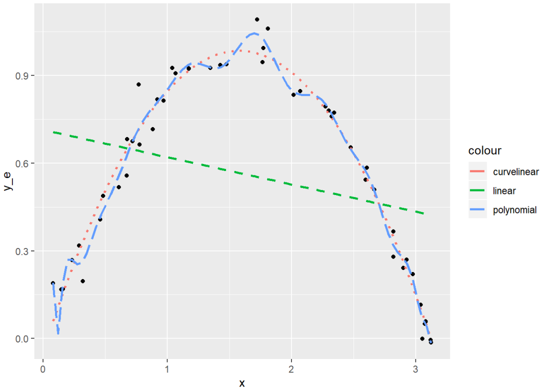
$$\mathrm{E}\left[\left(y-\hat{f}(x)\right)^2\right] = \left(\mathrm{Bias}\left[\hat{f}(x)\right]\right)^2 + \mathrm{Var}\left[\hat{f}(x)\right] + \sigma^2$$



- We create some example data, where x is a uniformly distributed random variable bounded between 0 and 1, and $y = sin(n)$



- we add some random noise, which is normally distributed.
- We see the formerly clearly visible underlying relationship between x and y now to some extent disturbed by this noise.
- However, keep in mind that the process that generated the data is still $y = sinus(x)$, which would also be the best funtional form to identify by any predictive algorithm.

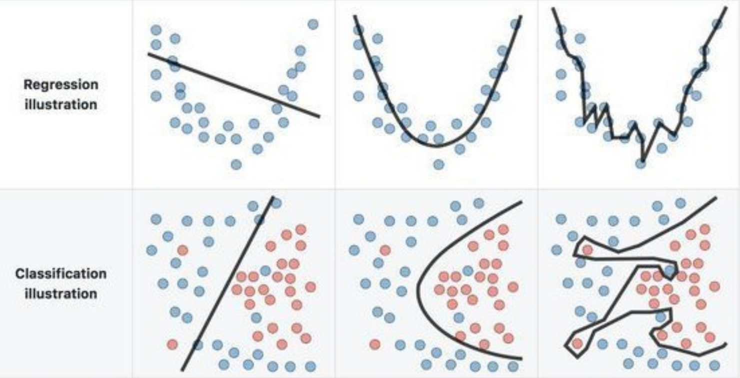


Lets see how models with different levels of complexity would interpret the reælationship between x and y:

1. y is modeled as a linear function of x
2. y is modeled as a curvelinear function of x
3. y is modeled as a compex multinomial function of x

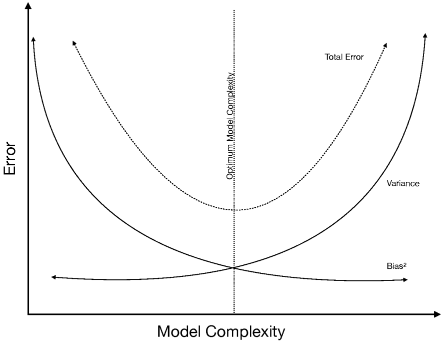
Aalborg University Denmark

	Underfitting	Just right	Overfitting
Symptoms	• High training error • Training error close to test error • High bias	• Training error slightly lower than test error	• Very low training error • Training error much lower than test error • High variance



Aalborg University Denmark

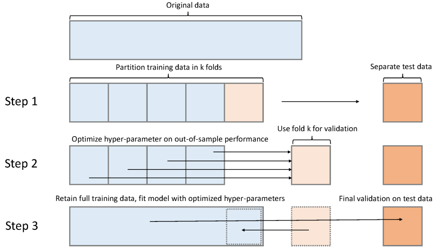
Regularization & Hyperparameter Tuning



The process of minimizing bias and variance errors is called regularization (i.e. hyperparameter-tuning)

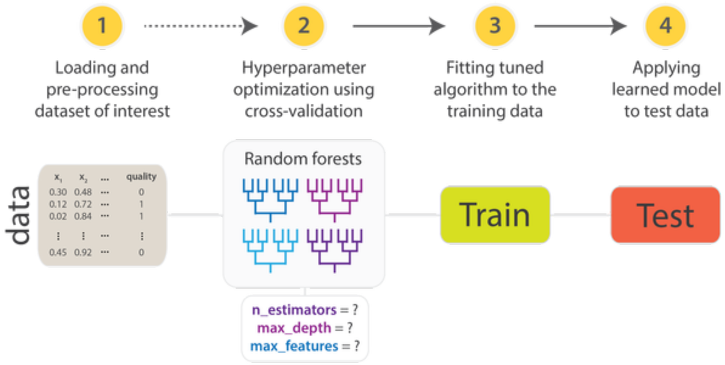
$$\underbrace{minimize \sum_{i=1}^{n} L(f(x_i), y_i),}_{in-sample\ loss} \ over \ \overbrace{f \in F}^{function\ class} \ subject\ to \ \underbrace{R(f) \leq c.}_{complexity\ \ restriction} \quad (1)$$

Partial Solution: Out-Of-Sample Validation

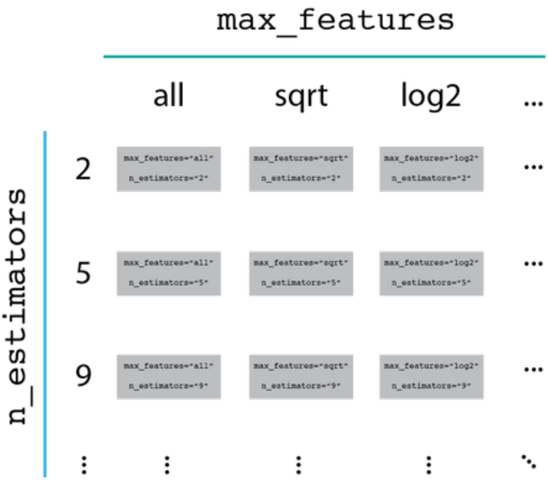


- Split the dataset in a training and a test sample.
- Fit you regression (train your model) on one dataset
- Validate predictive power on test sample, on which model is not fitted.

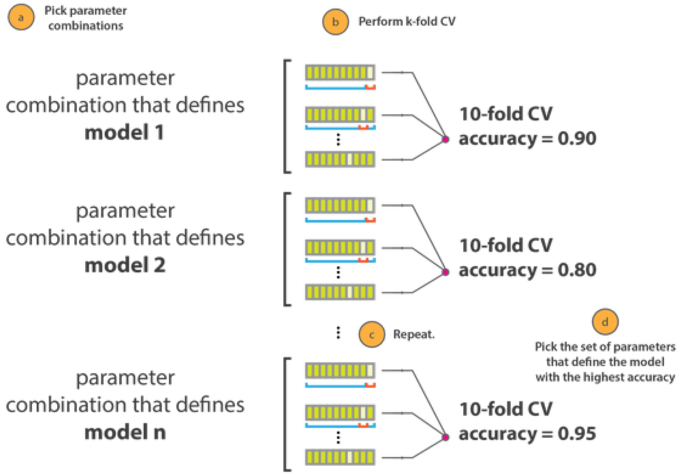
Hyperparameter Tuning Workflows



Search Strategies



Hyperparameter Summary



Supervised ML Models

Lets for a second recap linear regression techniques, foremost the common allrounder and workhorse of statistical research since some 100 years.

Basic Properties

- Outcome: contionous
- Predictors: continous, dichotonomous, categorical
- When to use: Predicting a phenomenon that scales and can be measured continuously

Functional form

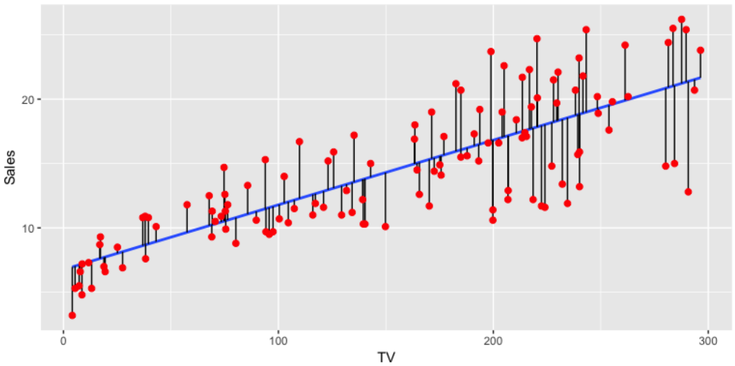
$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n + \varepsilon$$

where:

- y = Outcome, x_i = observed value ID_i
- β_0 = Constant
- β_i = Estimated effect of x_i on y , slope of the linear function
- ε = Error term

Supervised ML Models

OLS = Ordinary Least Squares



Advantages

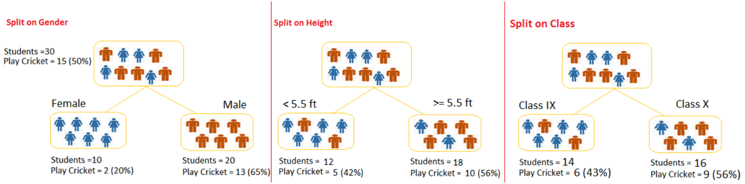
- Easy to Understand: Probably the most intuitive functional form
- Variable importance can directly be inferred
- Functional form can easily tweaked to fit different distributions of outcomes.
- No hyperparameter tuning except of feature selection.

Disadvantages

- Very strong assumption on the functional form (quasi-linear)
- In reality often in need of much feature engineering.

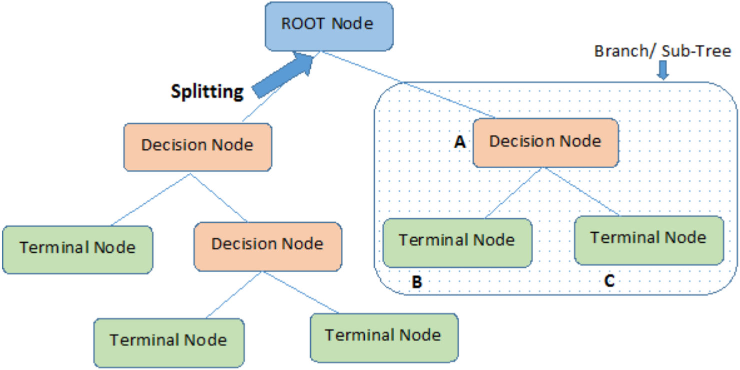
Regression & Classification Trees

- Mostly used in classification problems on continuous or categorical variables.
- Idea: split the population or sample into two or more homogeneous sets (or sub-populations) based on most significant splitter / differentiator in input variables.
- Repeat till stop criterium reachesd. leads to a tree-like structure.



Regression Trees Terminology

- Root Node: Entire population or sample and this further gets divided into two or more homogeneous sets.
- Splitting: It is a process of dividing a node into two or more sub-nodes.
- Decision Node: When a sub-node splits into further sub-nodes, then it is called decision node.
- Leaf/ Terminal Node: Nodes do not split is called Leaf or Terminal node.



Advantages

- Easy to Understand: Decision tree output is very easy to understand even for people from non-analytical background.
- Useful in Data exploration: Decision tree is one of the fastest way to identify most significant variables and relation between two or more variables.
- Data type is not a constraint: It can handle both numerical and categorical variables.
- Non Parametric Method: Decision tree is considered to be a non-parametric method. This means that decision trees have no assumptions about the space distribution and the classifier structure.

Disadvantages

- Overfitting: Being prone to over fitting is one of the most practical difficulty for decision tree models.
- Not fit for continuous variables: While working with continuous numerical variables, decision tree looses information when it categorizes variables in different categories.

How does a tree decide where to split?

The decision of making strategic splits heavily affects a tree's accuracy. The decision criteria is different for classification and regression trees. Common approaches:

- Gini Index
- χ^2
- Reduction in σ^2

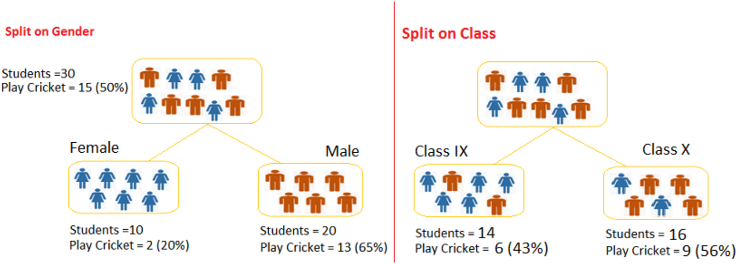
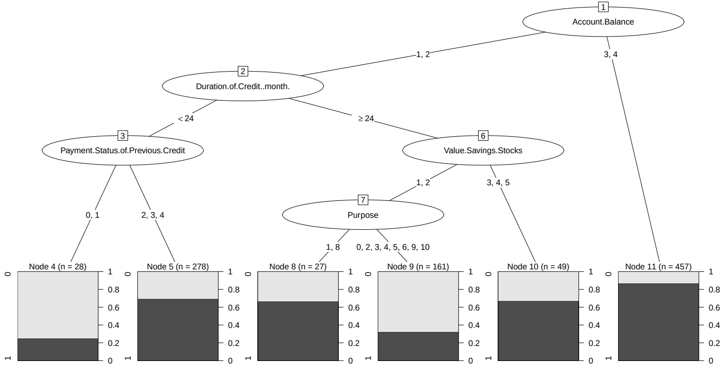


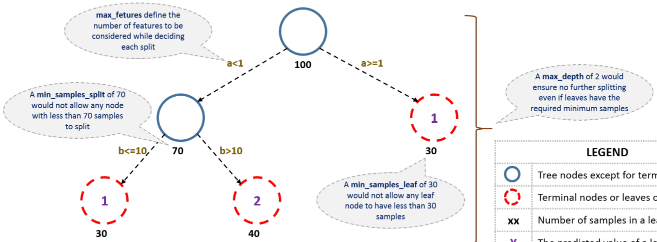
Figure: A regression-tree example on credit defaults



Solution I: Restricting

Common restriction approaches for node-split

- Minimum samples for a node split
- Minimum samples for a terminal node (leaf)
- Maximum depth of tree (vertical depth)
- Maximum number of terminal nodes
- Maximum features to consider for split

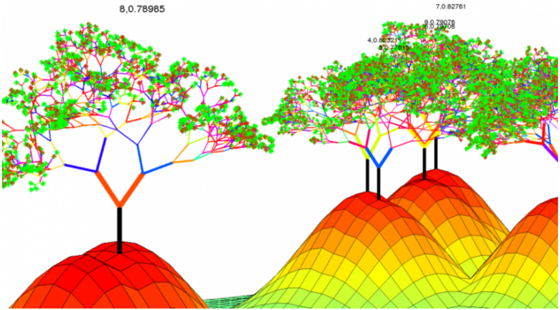


LEGEND	
○	Tree nodes except for terminal node
◌	Terminal nodes or leaves of tree
xx	Number of samples in a leaf
Y	The predicted value of a leaf node
abc	Node splitting criteria

Solution II: Pruning

- Problem: Restriction make lead to global disadvantages.
- Solution: We first make the decision tree to a large depth.
- Minimum samples for a node split
- Then we start at the bottom and start removing leaves which are giving us negative returns when compared from the top.
- Suppose a split is giving us a gain of say -10 (loss of 10) and then the next split on that gives us a gain of 20. A simple decision tree will stop at step 1 but in pruning, we will see that the overall gain is +10 and keep both leaves.

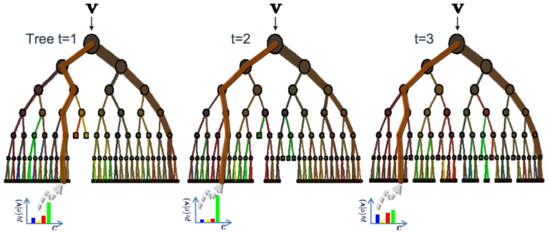
Solution III: Grow a "Random Forest"



- Random Forest is by some considered to be a panacea of all data science problems.
- versatile machine learning method capable of performing both regression and classification tasks.
- Some say: when you can’t think of any algorithm (irrespective of situation), use random forest!
- It is a type of ensemble learning method, where a group of weak models combine to form a powerful model.

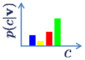
How does it work?

- Idea close to Monte Carlo simulation approaches, tapping in the power of randomness.
- In Random Forest, we grow multiple trees, and do not prune them.
- To classify a new object based on attributes, each tree gives a classification and we say the tree votes for that class.
- The forest chooses the classification having the most votes (over all the trees in the forest) and in case of regression, it takes the average of outputs by different trees.

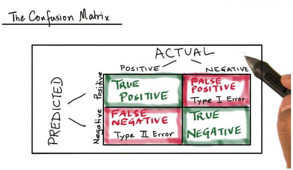


The ensemble model

Forest output probability $p(c|\mathbf{v}) = \frac{1}{T}\sum_{t}^{T} p_t(c|\mathbf{v})$



Model Metrics



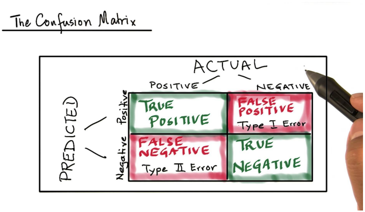
Coefficient of Determination: R^2

$$R^2 \equiv 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}$$

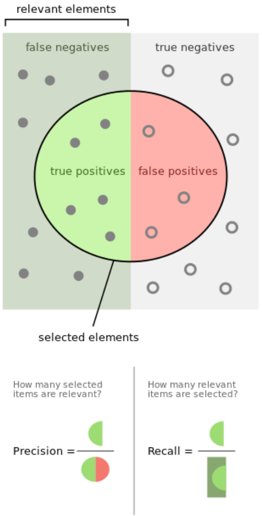
Root-Mean-Squared Error: *RSME*

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

Aalborg University Denmark



- True Positive: (TP)
  - Interpretation: You predicted positive and it's true.
  - You predicted that a woman is pregnant and she actually is.
- True Negative: (TN)
  - Interpretation: You predicted negative and it's true.
  - You predicted that a man is not pregnant and he actually is not.
- False Positive:(FP) - (Type 1 Error)
  - Interpretation: You predicted positive and it's false.
  - You predicted that a man is pregnant but he actually is not.
- False Negative: (FN) - (Type 2 Error)
  - Interpretation: You predicted negative and it's false.
  - You predicted that a woman is not pregnant but she actually is.



relevant elements
false negatives
true negatives
true positives
false positives
selected elements
How many selected items are relevant?
Precision =
How many relevant items are selected?
Recall =

Accuracy (ACC)

$$ACC = \frac{\mathrm{TP}+\mathrm{TN}}{P+N} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}$$

Sensitivity

also called recall, hit rate, or true positive rate (TPR)

$$TPR = \frac{\mathrm{TP}}{P} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} = 1-\mathrm{FNR}$$

Specificity

also called selectivity or true negative rate (TNR)

$$TNR = \frac{\mathrm{TN}}{N} = \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}} = 1-\mathrm{FPR}$$

Precision

also called positive predictive value (PPV)

$$PPV = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$$

Thank you for your attention. Any questions?

AALBORG UNIVERSITY
DENMARK

MIKE
MSc in Innovation and Entrepreneurship

IKE
Innovation, Knowledge and Economic dynamics