COMPUTER SCIENCE & ENGINEERING
TEXAS A&M UNIVERSITY

CARDWATCH

Dr. Martin “Doc” Carlisle

Catching Credit Card Fraud

271,823 reports in US in 2019

(This paper, 1997) cites

- $700M/year US
- $10B worldwide

Two types

- Card stolen
- Card number stolen

Catching Fraud

- People behave fairly consistently
  - Look for anomalies!
  - (Except when you don’t)

Neural Network Topology

- Visual Basic GUI
  - # input units, hidden units, output units, weight initial value, activation functions
  - Three layer (not configurable)

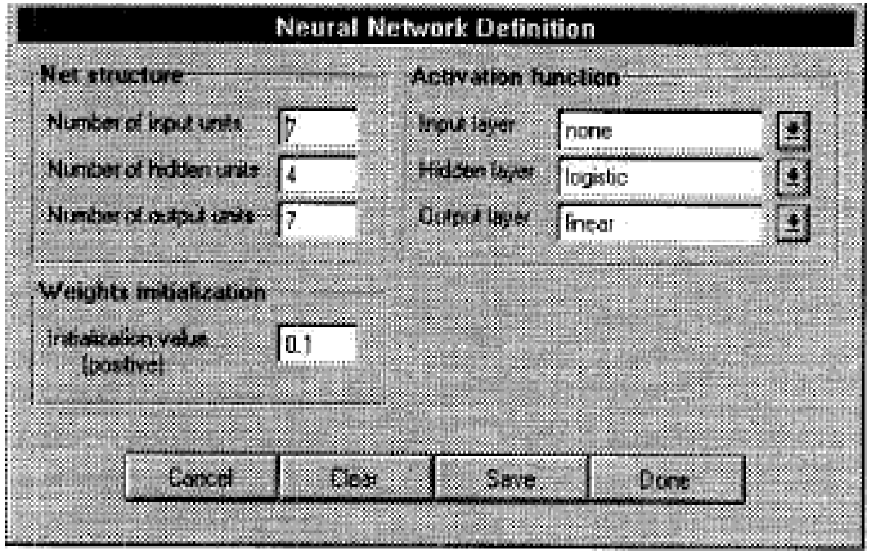


Fig. 2. Topology definition in CARDWATCH

Optimization

- Min/Max epochs, learning rate, momentum, tolerances
- Momentum moves weights in direction of last correction

$$\Delta w_{ij}(n+1) = \eta o_j(n+1)\delta_i(n+1) + \alpha \Delta w_{ij}(n)$$

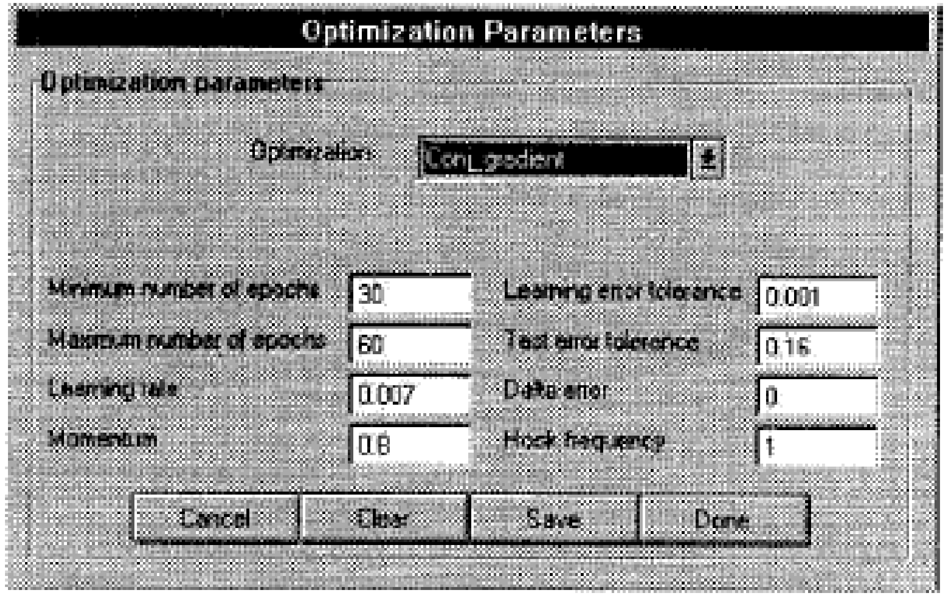


Fig. 3. Parameter definition in CARDWATCH

Synthetic data

TABLE I
EXAMPLE OF THE DATA SYNTHESIS

	category	amount of money	time passed since last purchase of the same category
generator input	integer code	distribution (type param1 param2)	distribution (type param1 param2)
	3	(0 10 2)	(0 48 5)
examples of resulting transactions	lexical	U$	Hours
	Grocery	10.60	46
	Grocery	11.80	50
	Grocery	13.00	44
	Grocery	10.10	53
corresponding neural network input	binary	real value	real value
	0 0 1 0 0	10.60	46
	0 0 1 0 0	11.80	50
	0 0 1 0 0	13.00	44
	0 0 1 0 0	10.10	53

Neural Network

- Auto-associator
  - Reproduce input pattern on output layer
  - Network produces “legal” patterns, but not “fraudulent” ones
- N-2 binary values categories
  - Amount of $ spent
  - Time elapsed since last purchase
  - 7-4-7 architecture (5 categories)

Autoassociative Neural Nets

“Autoassociative neural networks are feedforward nets trained to produce an approximation of the identity mapping between network inputs and outputs using backpropagation or similar learning procedures. The key feature of an autoassociative network is a dimensional bottleneck between input and output. Compression of information by the bottleneck results in the acquisition of a correlation model of the input data, useful for performing a variety of data screening tasks.”

M.A.Kramer – Computers and Chem Eng, April 1992

Neural Net Architecture

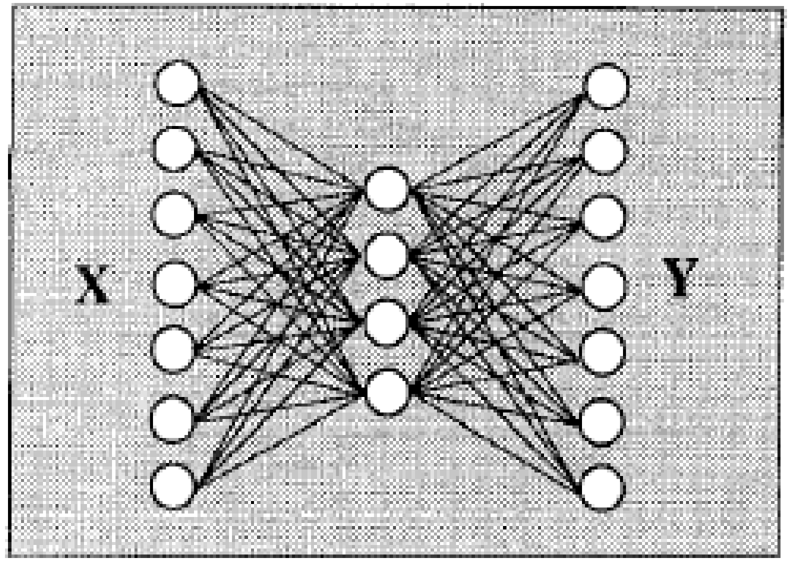


Fig. 5. Neural network architecture

Test data

- Created 323 transactions
  - Used first 264 for training (3 categories)
  - Last 20% reserved for testing along with generated “fraudulent transactions”

TABLE II
TRANSACTIONS USED FOR TRAINING

Category	#Transactions, total	#Transactions, fraudulent
Grocery	142	0
Air Tickets	4	0
Restaurants	118	0

TABLE III
TRANSACTIONS USED FOR TESTING

Category	#Transactions, total	#Transactions, fraudulent
Grocery	32	0
Air Tickets	4	2
Restaurants	60	35
Car Repair	16	16

Metric for fraud

- RMSE >= 0.16 means fraud
  - Test data valid was < 0.05 and fraudulent was > 0.18

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(t_n - o_n)^2}, \qquad (9)$$

Results

TABLE IV

DETECTION RATES

Category	detected legal %	detected fraudulent %
Grocery	100	-
Air Tickets	100	100
Restaurants	100	77
Car Repair	-	100
Total	100	85

Their Conclusions

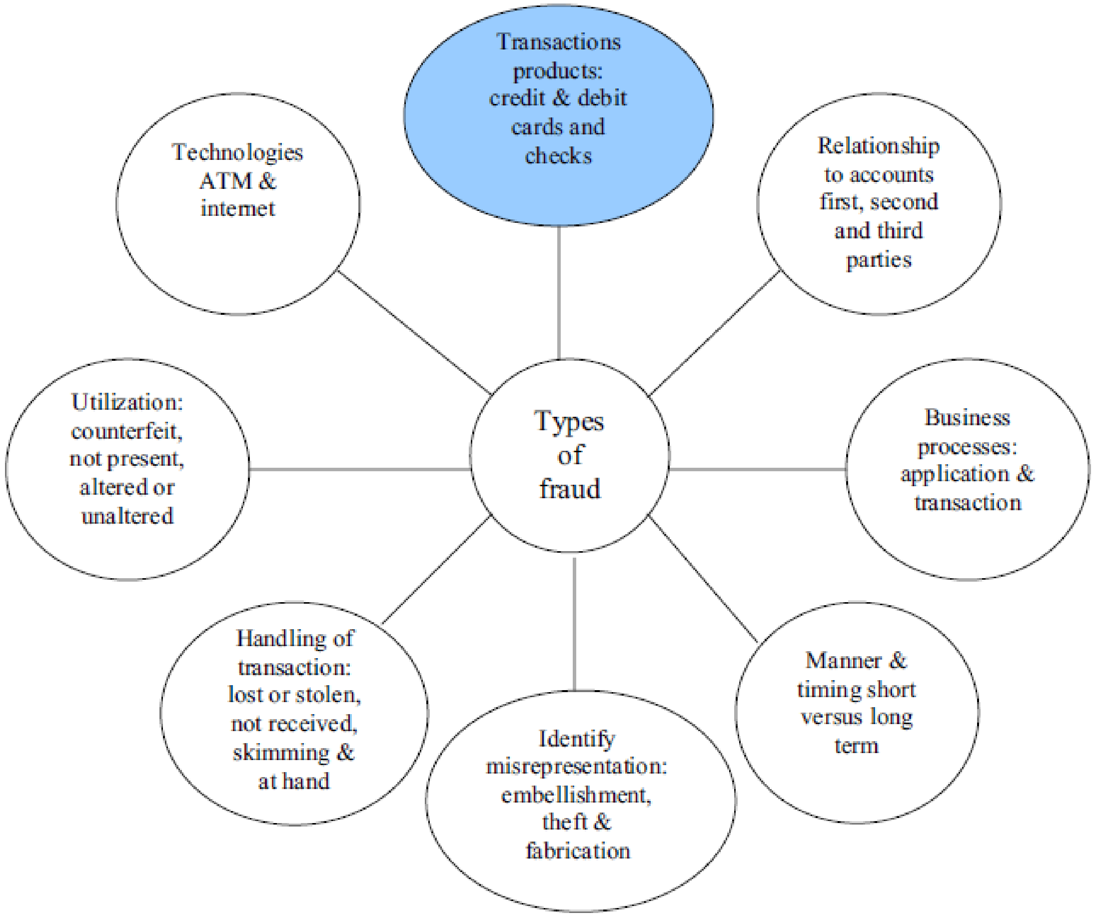
- Downside- one network per customer
- Make into general-purpose anomaly detection system

COMPUTER SCIENCE
& ENGINEERING
TEXAS A&M UNIVERSITY

Credit card fraud and detection review

Dr. Martin "Doc" Carlisle

Types of fraud



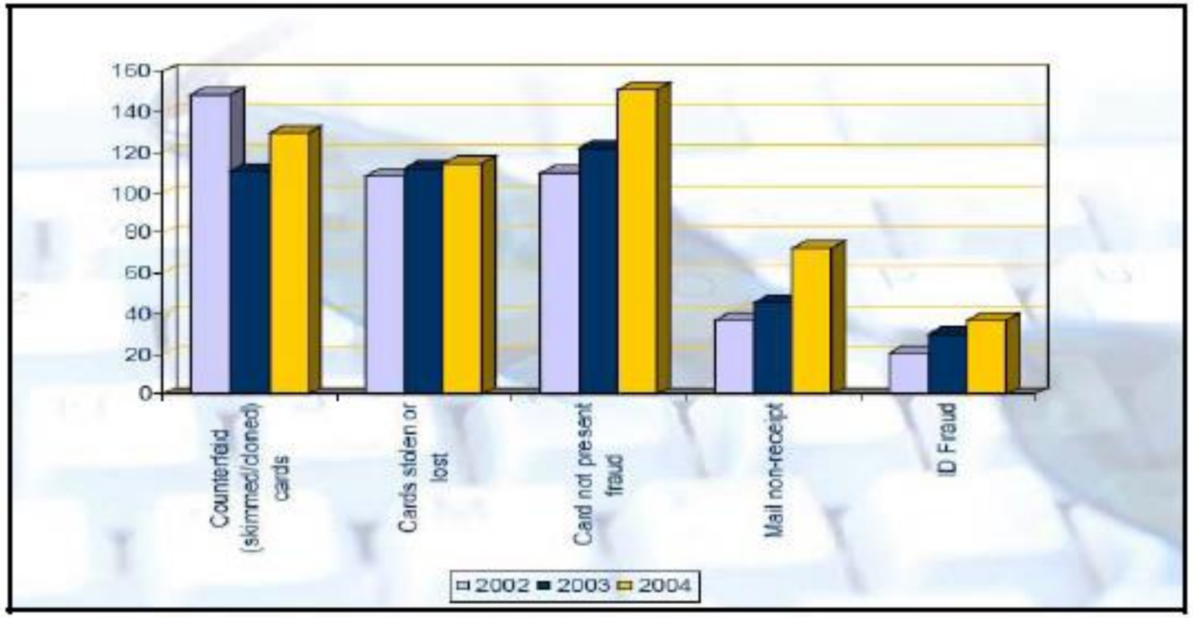
llowing Anderson's classification (2007).

Fig. 1. Types of fraud

Scope of transactions

- 120M new cards in 2004 in Germany
- €375B in 2004
- $7T in US in 2019 https://www.federalreserve.gov/paymentsystems/2019-December-The-Federal-Reserve-Payments-Study.htm
- UK £423M losses in 2006
- US "Card not present" $4.57B in 2016 https://www.washingtonpost.com/business/think-your-credit-card-is-safe-in-your-wallet-think-again/2019/09/11/05e316e4-be0e-11e9-b873-63ace636af08_story.html

Scope of transactions



Source: DRF EU Speech, Amsterdam, April 19th 2005 (Pago e-Transaction Services GmbH, 2005)

Fig. 3. Fraud distribution in Europe

Bankruptcy Fraud

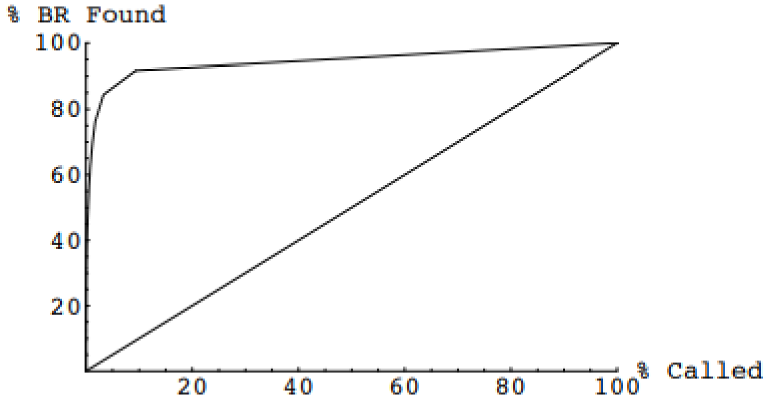
- Using a credit card while insolvent
  - Purchaser knows they won't be able to pay
- Foster & Stine (2004)
  - Regression models

Foster and Stine

- 67,160 variables
- Built step-wise regression including all pairwise interactions
  - Pairs are key (including only 2-3 outperforms 100 best linear predictors)

Results

Figure 1: *Lift chart for the regression model that uses 39 predictors to predict the onset of personal bankruptcy. The chart shows the percentage of bankrupt customers in the validation data found when the validation observations are sorted by predicted scores. For example, the largest 1% of the predictions holds 60% of the bankruptcies. The diagonal line is the expected performance under a random sorting.*



Results

Table 3: *Interactions that appear in 3 or more of the 5 stepwise regression models obtained in the five-fold cross-validation analysis. The shown prevalence indicates the number of interactions with that predictor among the 159 interactions in the 5 regression models. (Table 2 summarizes the fits of these models.)*

Common Interactions		Prevalence		Appears in
X_1	X_2	X_1	X_2	k models
Number of credit cards	Prior cards past due 60 days	35	36	5
Number of credit cards	Number of credit cards	35	35	5
Number of credit cards	Prior cards closed	35	20	4
Number of credit cards	Late charge in prior month	35	31	3
Number of credit cards	External flag unavailable	35	20	3
Number of credit cards	External credit flag-2	35	16	3
Number of credit cards	External credit flag-1	35	9	3
Prior cards past 60 days	Late charge in prior month	36	31	5
Prior cards past 60 days	Prior cards closed	36	20	5
Prior cards past 60 days	External flag unavailable	36	20	5
Prior cards past 60 days	Internal bank status code-2	36	8	3
Prior cards past 60 days	External credit flag-2	36	16	3
Prior cards past 60 days	External flag-1, prior quarter	36	5	3
Late charge prior month	Prior cards closed	31	20	5
Late charge prior month	Missing FICO score	31	7	3
External flag unavailable	External credit flag-3	36	4	3

Theft fraud/counterfeit

- Using a card that's not yours, or a fake card (e.g. card not present)

Application fraud

- Apply for card with fake info

Skewed Data Problem

- Dealing with skewed data (far more legitimate than fraudulent entries)
- This means you could always predict legit and be “successful”!
- Solutions
  - Meta-learning (apply different algorithms)
  - Manipulate class distribution (use fraudulent entries more often)

Phua's Minority Report

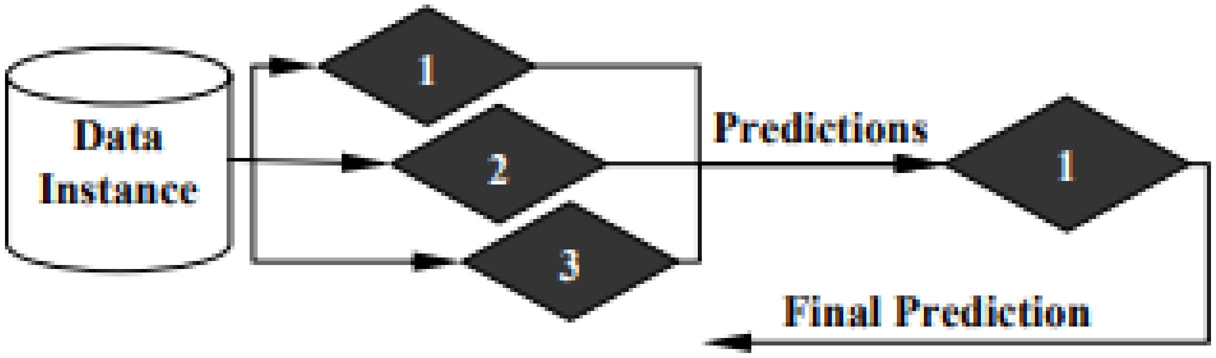
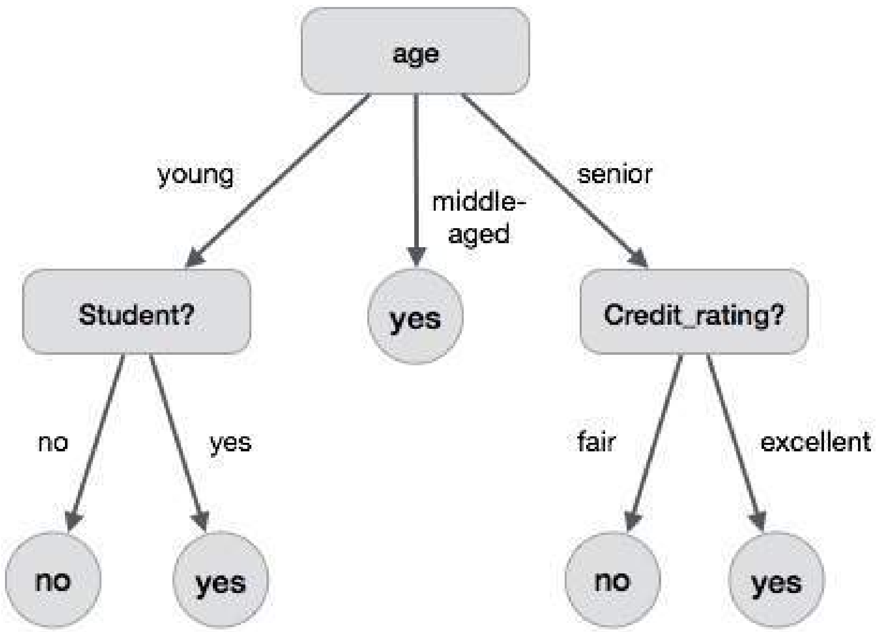


Figure 1: Predictions on a single data instance using precogs

Three “Precogs”

- Naïve Bayesian
- C4.5
  - Decision tree rule induction
- Backpropagation Neural Network

Decision Tree



The benefits of having a decision tree are as follows –

- It does not require any domain knowledge.
- It is easy to comprehend.
- The learning and classification steps of a decision tree are simple and fast.

C4.5

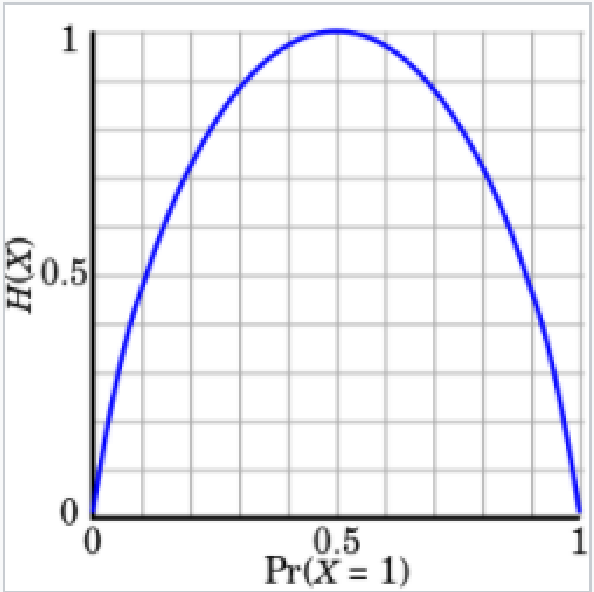
- This algorithm has a few base cases.
  - All the samples in the list belong to the same class. When this happens, it simply creates a leaf node for the decision tree saying to choose that class.
  - None of the features provide any information gain. In this case, C4.5 creates a decision node higher up the tree using the expected value of the class.
  - Instance of previously-unseen class encountered. Again, C4.5 creates a decision node higher up the tree using the expected value.

C4.5 Recursive case

- For each attribute a, find the normalized information gain ratio from splitting on a.
- Let a_best be the attribute with the highest normalized information gain.
- Create a decision node that splits on a_best.
- Recurse on the sublists obtained by splitting on a_best, and add those nodes as children of node.

Entropy of coin flip

The expected value of the information gain is the mutual information I(X;A) of X and A – i.e. the reduction in the entropy of X achieved by learning the state of the random variable A.



Entropy $\mathrm{H}(X)$ (i.e. the expected surprisal) of a coin flip, measured in bits, graphed versus the bias of the coin $\Pr(X=1)$, where $X=1$ represents a result of heads.[9]:14–15

Consider the Cost

Table 2: Cost model for insurance fraud detection

Outcome	Cost
Hits	Number of Hits * Average Cost Per Investigation
False Alarms	Number of False Alarms * (Average Cost Per Investigation + Average Cost Per Claim)
Misses	Number of Misses * Average Cost Per Claim
Normals	Number of Normal Claims * Average Cost Per Claim

Model Cost Savings = No Action – [Misses Cost + False Alarms Cost + Normals Cost + Hits Cost]

Behavioral Fraud

- Details of legitimate cards obtained fraudulently (phone and e-commerce)

Detection Techniques

- Decision Tree
- Genetic Algorithms
- Clustering Techniques
- Neural nets

Genetic Algorithms

- Metaheuristic inspired by natural selection
- Need
  - Genetic representation of solution
  - Fitness function for solution
- Process
  - Initialize with random solutions
  - Select “best” to breed new generation
    - Might also add “elitism” (keep best of previous generation)
  - Apply crossover (mixing two solutions) and mutation (changing parts of a solution)

Bentley et al. Tree

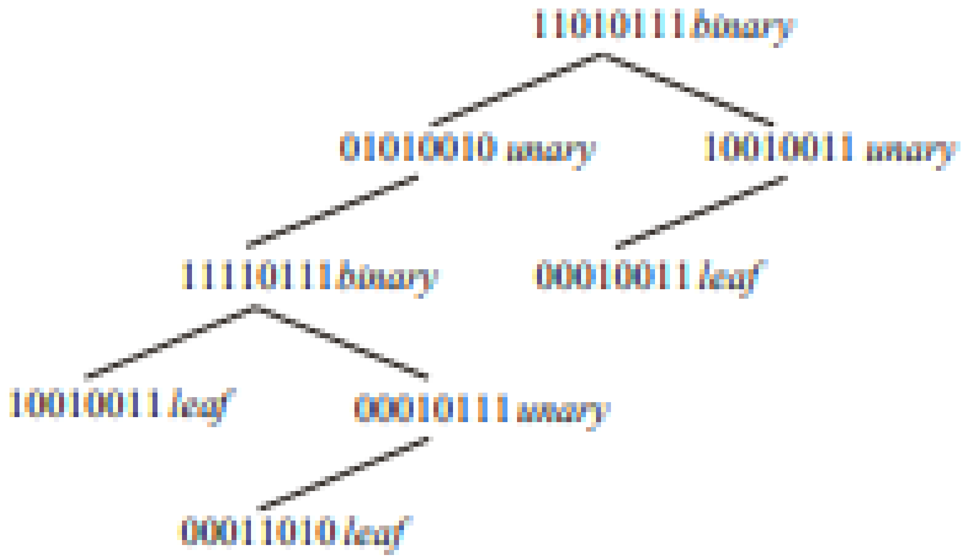


Figure 2: An example genotype used by the system.

Bentley et al Genetic Alg

relatively low. In addition, the most accurate and intelligible rule sets that are generated by [B] contain just three rules. Overall, the best rule set as reported by the committee decision maker is for experiment 2:

(IS_LOW field57 OR field50)
IS_MEDIUM field56
(field56 OR field56)

and for the experiment 3:

(Filed49 OR Field56)
(IS_LOW Field26 OR field15)
IS_MEDIUM field56

Bentley et al Results

	[A] Fuzzy Logic with non-overlappingMFs					[B] Fuzzy Logic with overlapping MFs					[C] MP-Fuzzy Logic with overlapping MFs					[D] MP-Fuzzy Logic with smooth MFs				
	R	Training		Test		R	Training		Test		R	Training		Test		R	Training		Test	
		TP%	FN%	TP%	FN%		TP%	FN%	TP%	FN%		TP%	FN%	TP%	FN%		TP%	FN%	TP%	FN%
1	3	6.09	3.81	10.4	3.35	2	100	0	100	85.1	16	10.9	5.79	100	100	5	48.6	5.79	42.5	10.3
2	2	44.1	5.79	47.8	9.45	3	100	1.67	99.7	6.38	3	1.37	5.64	99.7	100	10	41.6	5.79	47.6	12.5
3	3	46.8	5.18	46.9	6.09	3	100	5.78	100	5.79	4	1.67	5.64	86.9	100	16	42.7	5.94	42.9	6.40

Table 2 Intelligibility (number of rules) and accuracy (number of correct classifications of "suspicious" items) of rule sets for test and training data. **R** shows the number of rules in the generated rule set and **TP** and **FN** is represented in %.

COMPUTER SCIENCE & ENGINEERING
TEXAS A&M UNIVERSITY

Behavior-cluster … Credit Card Fraud Detection

Dr. Martin “Doc” Carlisle

Big Idea

- We need to cluster data first before doing sampling to address class imbalance

Class Imbalance

- Far more genuine than fraud
  - Hurts traditional machine learning
- Data-level
  - Sampling and cost-sensitive methods
- Model-level
  - Ensemble classifiers divide majority into subsets and train with minority class

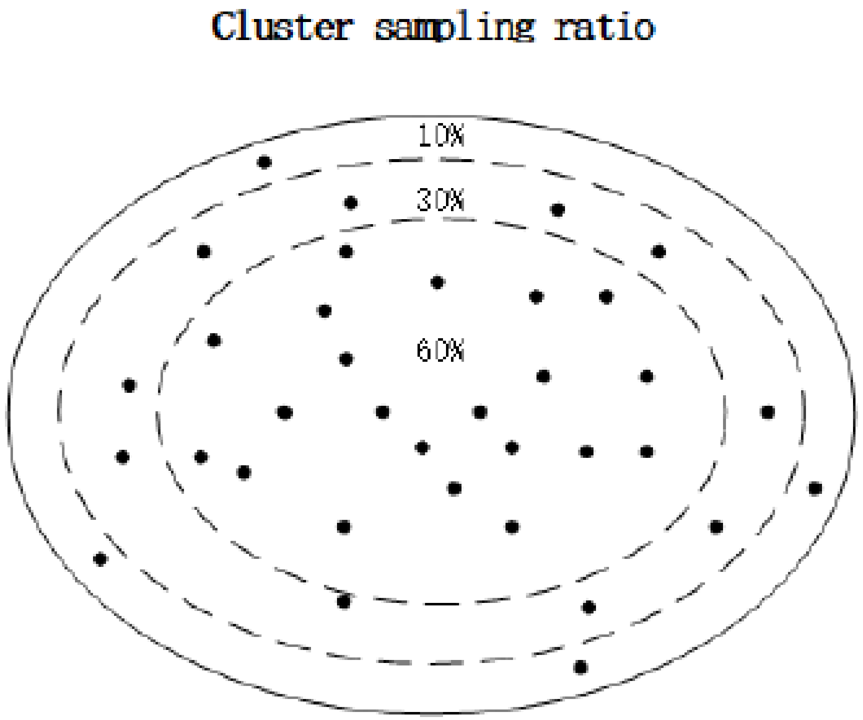
What's the Imbalance

- Volume of data
  - Authors posit complexity of data is ignored (i.e. some users look like fraud)
    - Example – multiple large transactions in short time frame
    - Authors define as "behavior noise"

Proposed Solution

- Behavior-cluster based under-sampling
  - Divide two classes into multiple subsets (clusters)
  - Reduce noise in each cluster
  - Hierarchically under-sample each cluster w/o noise

Cluster Sampling Ratio



Dataset

- 5M transactions from financial institution
- 18 UCI data sets
  - https://archive.ics.uci.edu/ml/datasets.php?task=cla&area=bus&type=&view=list

Behavior Noise (I)

- Points of opposite label in feature space or outliers

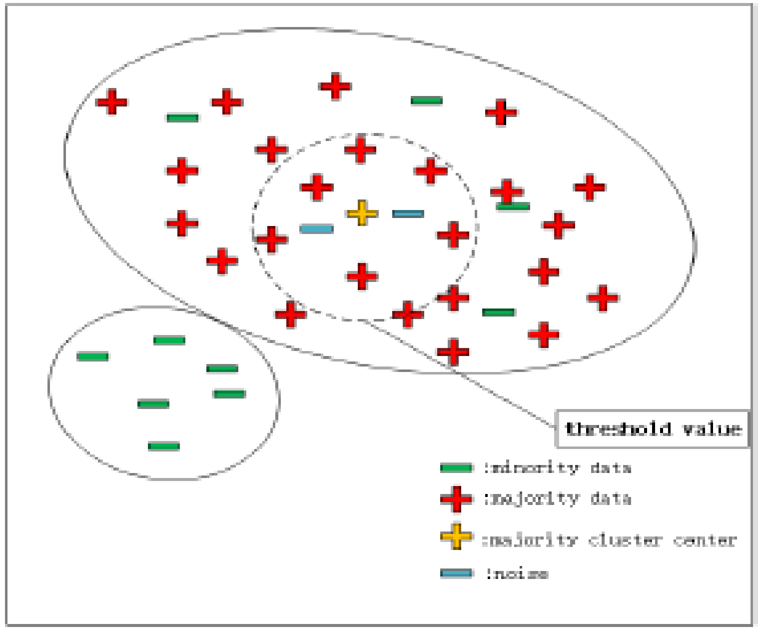


Figure 1: Behavior Noise in Majority Class

Behavior Noise (II)

- Points of opposite label in feature space or outliers

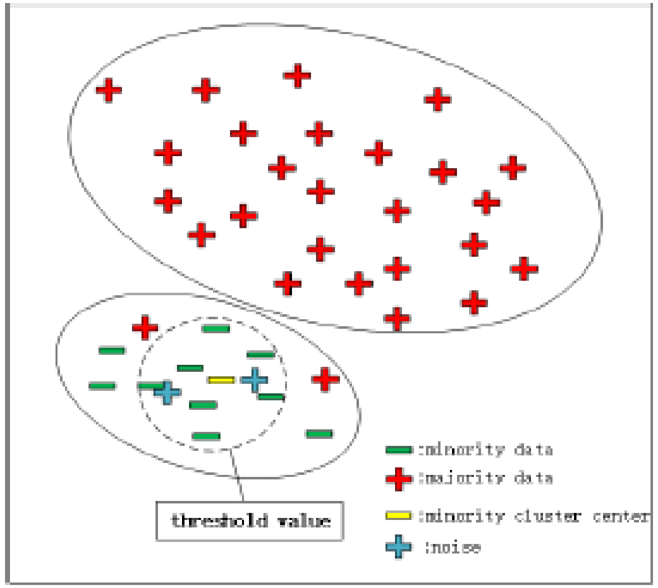


Figure 2: Behavior Noise in Minority Class

Data Flow

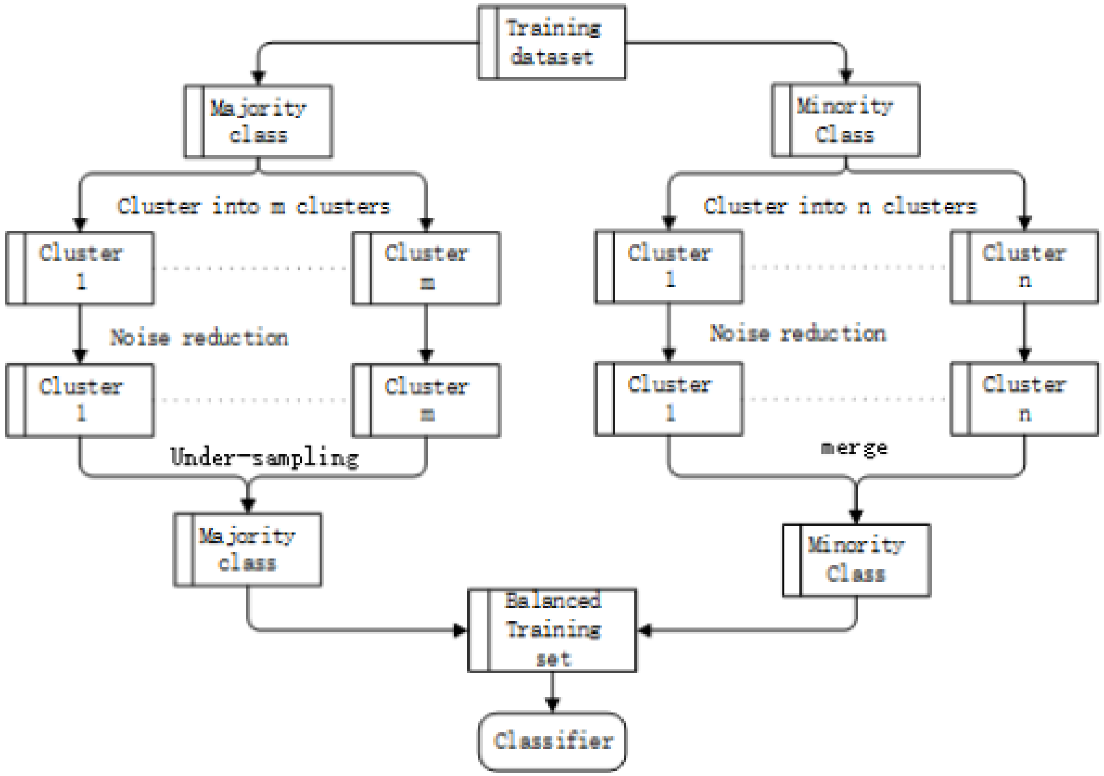


Figure 3: Flow Diagram

KMeans clustering (review)

1. Choose k cluster centers randomly
   – From k random points or k random patterns
2. Assign each pattern to closest cluster center
3. Recompute centers
4. If haven't converged,

   repeat from step 2

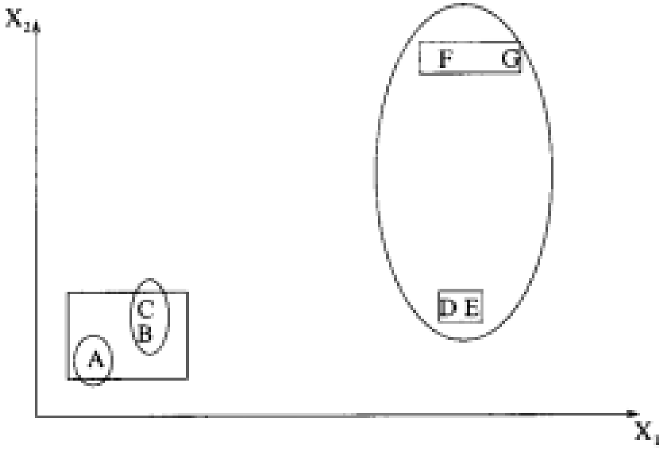


Figure 14. The k-means algorithm is sensitive to the initial partition.

Noise Reduction

For each cluster, compute farthest distance from center

μ is threshold value

Delete other class items <= μ*max distance

Algorithm 1 Noise Reduction

Require: $X_{N_{maj}}$: The majority set
$X_{N_{min}}$: The minority set
m : The number of majority set clustering
n : The number of minority set clustering
N_{maj_i} : The number of majority set
N_{min_r} : The number of minority set
X_{maj_i} : The majority cluster
X_{min_r} : The minority cluster
C_{maj_i} : The center of majority cluster
C_{min_r} : The center of minority cluster
μ : Threshold value

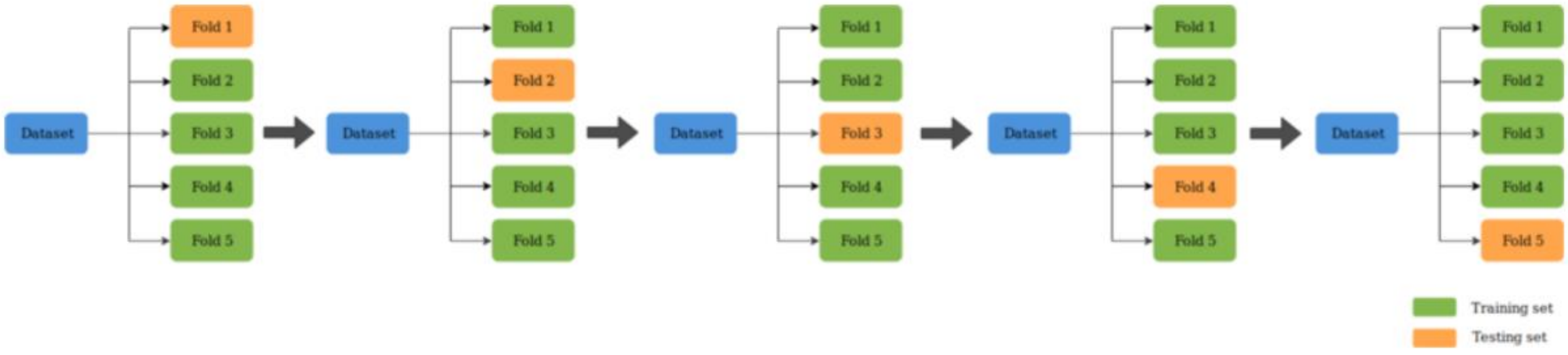
Noise Reduction:
- **for** $i = 1$ to m **do**
  - $d_{i_{max}} = max(EuclideanDistance(X_{maj_i}, C_{maj_i}))$
  - **for** $j = 1$ to N_{min_r} **do**
    - $d_{ij} = EuclideanDistance(X_{N_{min}}[j], C_{maj_i})$
    - **if** $d_{i_{max}} {}^{*} \mu \geq d_{ij}$ **then**
      - $DeleteX_{N_{min}}[j]$
      - $returnX'_{N_{min}}$;
    - **end if**
  - **end for**
- **end for**
- **for** $r = 1$ to n **do**
  - $d_{r_{max}} = max(EuclideanDistance(X_{min_r}, C_{min_r}))$
  - **for** $l = 1$ to N_{maj_i} **do**
    - $d_{rl} = EuclideanDistance(X_{N_{maj}}[l], C_{min_r})$
    - **if** $d_{r_{max}} {}^{*} \mu \geq d_{rl}$ **then**
      - $DeleteX_{N_{maj}}[l]$
      - $returnX'_{N_{maj}}$;
    - **end if**
  - **end for**
- **end for**

Output: $X'_{N_{min}}$: The minority set after noise reduction;
$X'_{N_{maj}}$: The majority set after noise reduction;

5-fold Cross Validation

Split into 5 folds, each fold is used as testing set at some point

1. Shuffle the dataset randomly.
2. Split the dataset into k groups
3. For each unique group:
    1. Take the group as a hold out or test data set
    2. Take the remaining groups as a training data set
    3. Fit a model on the training set and evaluate it on the test set
    4. Retain the evaluation score and discard the model
4. Summarize the skill of the model using the sample of model evaluation scores



5-Fold Cross Validation

More on k-fold cross validation

- It is also important that any preparation of the data prior to fitting the model occur on the CV-assigned training dataset within the loop rather than on the broader data set. This also applies to any tuning of hyperparameters. A failure to perform these operations within the loop may result in data leakage and an optimistic estimate of the model skill.

- *Despite the best efforts of statistical methodologists, users frequently invalidate their results by inadvertently peeking at the test data.*

— Page 708, Artificial Intelligence: A Modern Approach (3rd Edition), 2009.

Under-sampling

- Select a small number from each majority class
  - More samples from near center of cluster
  - Number of samples from each cluster is related to proportion of positive and negative transactions
- All negative samples aggregated

Data Attributes

Table 1: Description of the Attributes in Credit Card Transaction Data

Attributes name	Description
Common_phone	Customer ' s usual mobile phone number
Pay_bind_phone	Customer ' s number bound on the electronic payment platform
Pre_trade_result	Customer ' s verification results of the last
Is common_ip	Whether this transaction is a common IP
Trade_amount	Amount of a transaction
Pay_single_limit	Limit on the amount of a single transaction
Pay_accumulate_limit	Total daily transaction amount limit
Account_number	Credit card number
Client_mac	MAC address of a transaction
Trade_date	Date of transaction
Trade_time	Exact time of transaction
White_list_mark	Whether the account is in the trusted list
Card_balance	Account balance before payment
Transaction_object	Is the receiver a person or a business
Receiver_number	Receiver number
Last_trade_time	Account ' s last transaction time

AUC/ROC

- "AUC" = "area under curve", specifically receiver operating characteristics graph (ROC)
- ROC
  - Used to depict trade-off between hit rate and false alarms
  - Especially useful with skewed class distribution!

Refresher on F1

True class

Hypothesized class	p	n
Y	True Positives	False Positives
N	False Negatives	True Negatives
Column totals:	P	N

$$\text{fp rate} = \frac{FP}{N} \qquad \text{tp rate} = \frac{TP}{P}$$

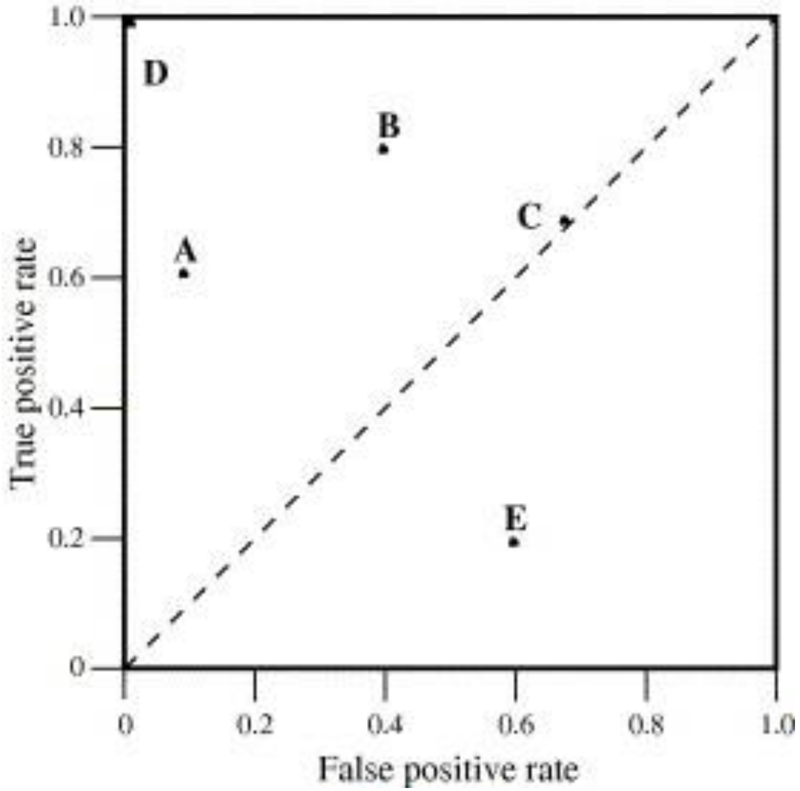
$$\text{precision} = \frac{TP}{TP+FP} \qquad \text{recall} = \frac{TP}{P}$$

$$\text{accuracy} = \frac{TP+TN}{P+N}$$

$$\text{F-measure} = \frac{2}{1/\text{precision}+1/\text{recall}}$$

Receiver Operating Characteristic Graph (ROC) (I)

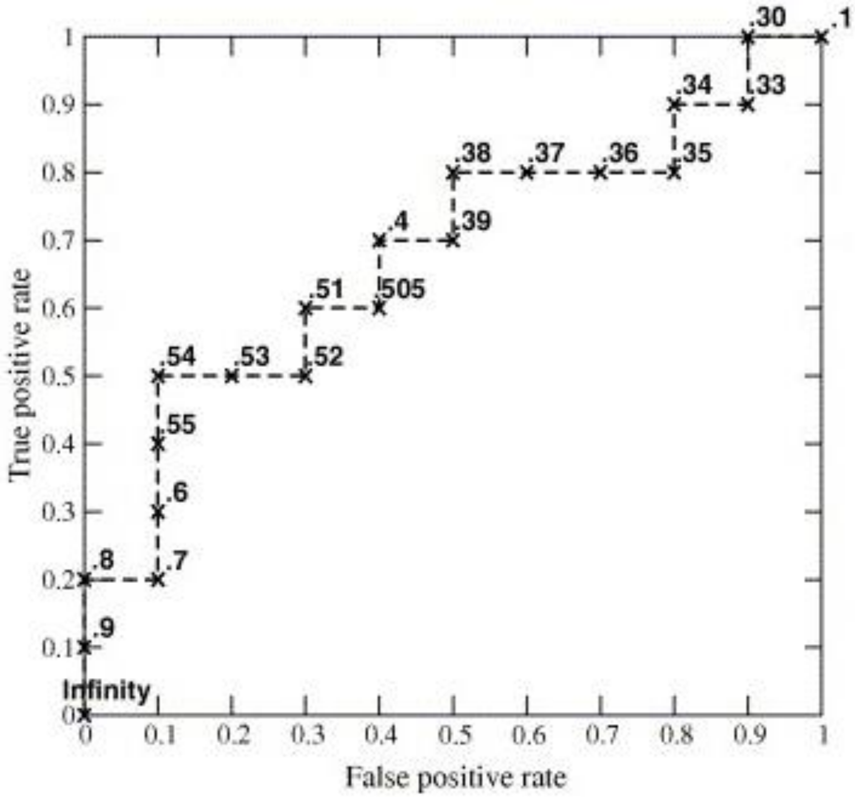
- True positive on Y, false positive on X
- Perfect is top left
- (0,0) – just say no
- (1,1) – always say yes
- Diagonal is random
  - (x,x) guess yes x%
- Bottom right worse



Receiver Operating Characteristic Graph (ROC) (II)

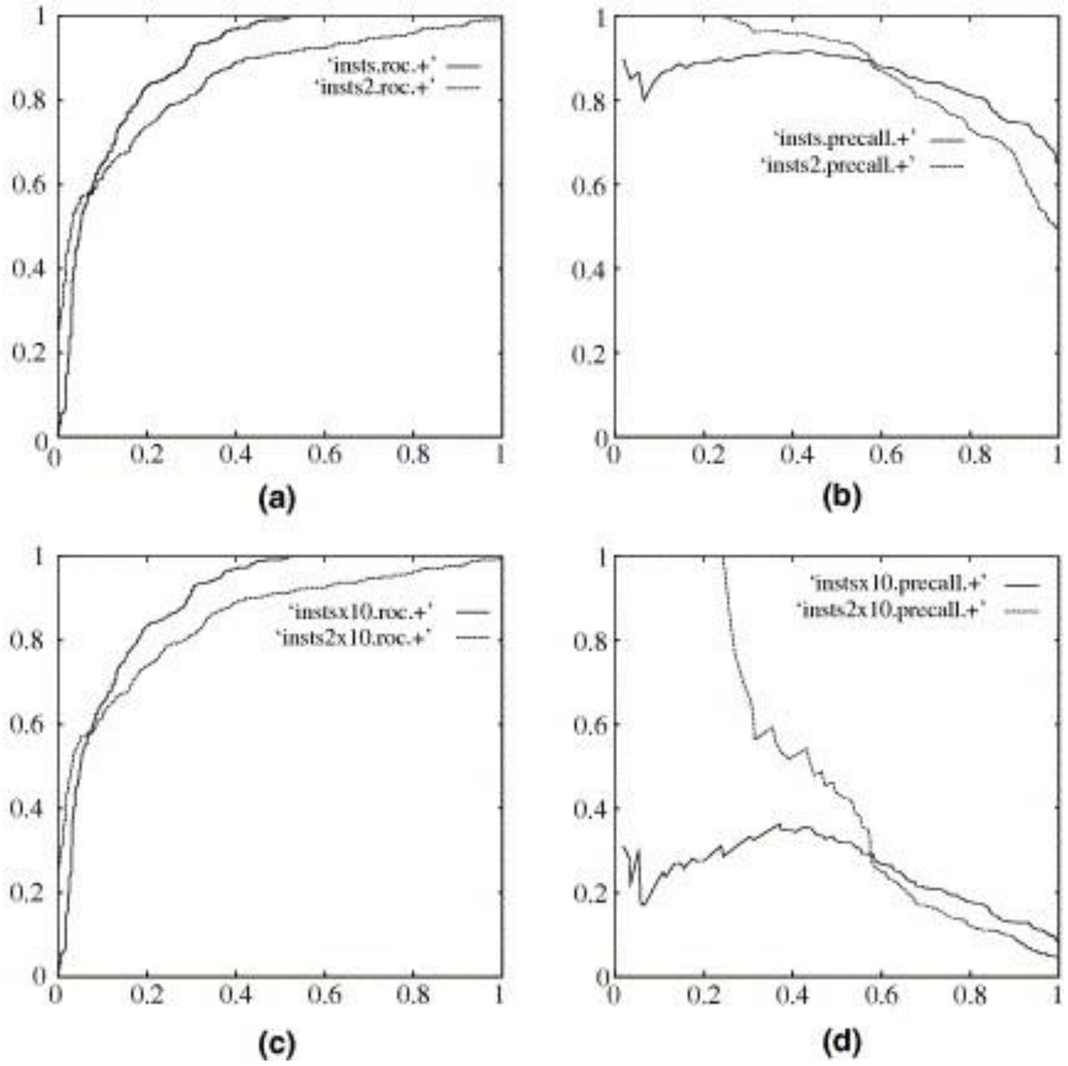
- Obtain curve from changing threshold

Inst#	Class	Score	Inst#	Class	Score
1	p	.9	11	p	.4
2	p	.8	12	n	.39
3	n	.7	13	p	.38
4	p	.6	14	n	.37
5	p	.55	15	n	.36
6	p	.54	16	n	.35
7	n	.53	17	p	.34
8	n	.52	18	n	.33
9	p	.51	19	p	.30
10	n	.505	20	n	.1



ROC vs precision-recall

A,B – balanced 1:1
C,D – increased no by 10x



COMPUTER SCIENCE
& ENGINEERING
TEXAS A&M UNIVERSITY

Example ML ROC curves

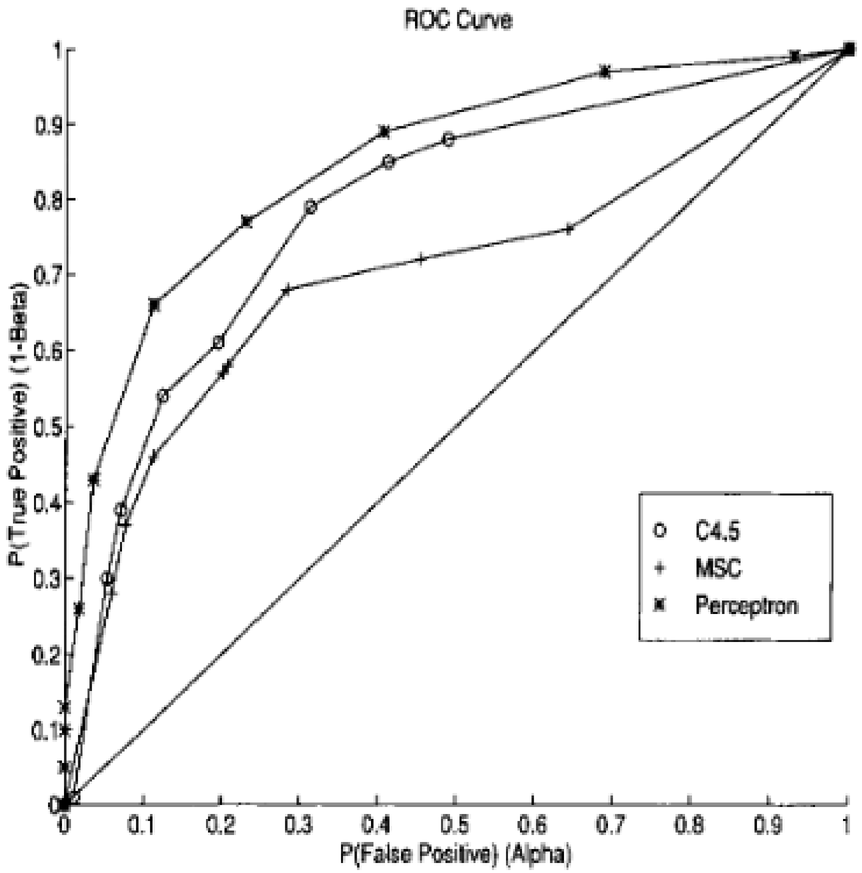


Fig. 12. ROC curve for C4.5, MSC, and Perceptron on the Hungarian heart disease data.

CC experiments

Table 4: Experimental Results of Credit Card Transaction Data

model	accuracy	recall	precision	f1	auc
RF CNMP	0.985	0.979	0.655	0.785	0.994
RF RUS	0.984	0.979	0.653	0.783	0.993
RF EE	0.985	0.980	0.654	0.784	0.993
RF ROS	0.995	0.919	0.919	0.919	0.986
RF AD	0.995	0.905	0.927	0.916	0.987
RF SM	0.995	0.921	0.908	0.915	0.987

Note lower accuracy of CNMP, but higher AUC
Authors claim recall is more important – ability to detect fraud, but low precision means a lot of false positives….

18 UCI Datasets

Table 5: Auc of 18 UCI Data Sets

Datasets	C4.5	RUS	ROS	SMOTE	Chan	EasyEnsemble	Asym	IRUS	CNPM
Abalone	0.711	0.736	0.800	0.794	0.856	0.860	0.853	0.855	**0.882(0.000)**
Arrhythmia	0.900	0.885	0.940	0.907	0.973	0.972	0.974	**0.977**	**0.977(0.000)**
Balance-scale	0.500	0.523	0.627	0.540	0.544	0.612	0.565	0.588	**0.636(0.000)**
Cmc	0.681	0.667	0.673	0.699	0.709	0.706	0.716	**0.736**	0.732(-0.004)
Flag	0.719	0.778	0.749	0.695	**0.807**	0.751	0.795	0.804	0.736(-0.071)
German	0.704	0.697	0.705	0.714	0.728	0.782	0.728	**0.766**	0.735(-0.031)
Glass	0.645	0.718	0.776	0.791	0.796	0.780	0.805	0.803	**0.812(0.000)**
Haberman	0.619	0.620	0.650	0.683	0.668	0.681	0.664	0.673	**0.722(0.000)**
Heart-stalog	0.852	0.841	0.850	0.852	0.853	0.884	0.840	0.888	**0.892(0.000)**
Hepatitis	0.795	0.789	0.782	0.781	0.828	0.848	0.836	0.838	**0.875(0.000)**
Housing	0.748	0.742	0.759	0.767	0.800	**0.817**	0.789	0.811	**0.817(0.000)**
Ionosphere	0.926	0.938	0.940	0.935	0.943	**0.974**	0.931	0.954	0.955(-0.019)
Nursery	**1.000**	0.982	0.998	**1.000**	0.999	0.999	0.999	0.999	0.994(-0.006)
Phoneme	0.920	0.900	0.926	0.918	0.924	**0.956**	0.927	0.923	0.943(-0.013)
Pima	0.778	0.765	0.777	0.777	0.801	0.809	0.769	**0.812**	0.806(-0.006)
Satimage	0.918	0.915	0.920	0.925	0.947	**0.956**	0.949	0.951	**0.956(0.000)**
Vehicle	0.825	0.785	0.824	0.820	0.839	**0.860**	0.833	0.853	0.793(-0.067)
Wpdc	0.642	0.663	0.696	0.700	0.698	0.699	0.712	0.732	**0.767(0.000)**
Average	0.771	0.775	0.800	0.794	0.817	0.830	0.816	0.831	**0.835**