Learning Feature Dependencies for Noise Correction in Biomedical Prediction

By : Peyman Navidy


ARBABA
BanehPedia.com
LanK

Introduction

* Noise or Errors in The Biomedical Data
* Bayesian Network-based Noise Correction
* Experiment Setup
* Results and Discussions
* Conclusion

Noise or Errors in The Biomedical Data

* Biomedical Instances
* Detect Association between **Feature**

  Naive Bayes - Decision Tree - SVM
* Detect Association and Dependencies for Feature

  Bayesian Network
* Bayesian Network with Estimates Errors

Bayesian Network-based Noise Correction

* Step 1 - Select Useful Features
* Step 2 - Capture Feature Dependencies
* Step 3 - Estimate Error Rates of Features
* Step 4 - Predict with the Noisy Features

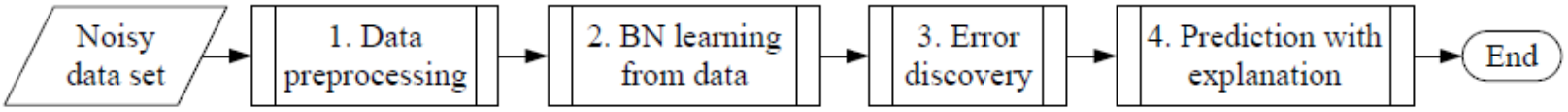


Figure 1: The Bayesian Network with Noise Correction (BN-NC) framework.

Step 1 - Select Useful Features

* Preprocessing and discretizing feature for BN model learning
* Freature selection from **Fayyad and Irani** technique
* Combine entropy data **decision tree** and **MDL**
* Converting the continuous features into discrete features to enable learning of discrete BN models

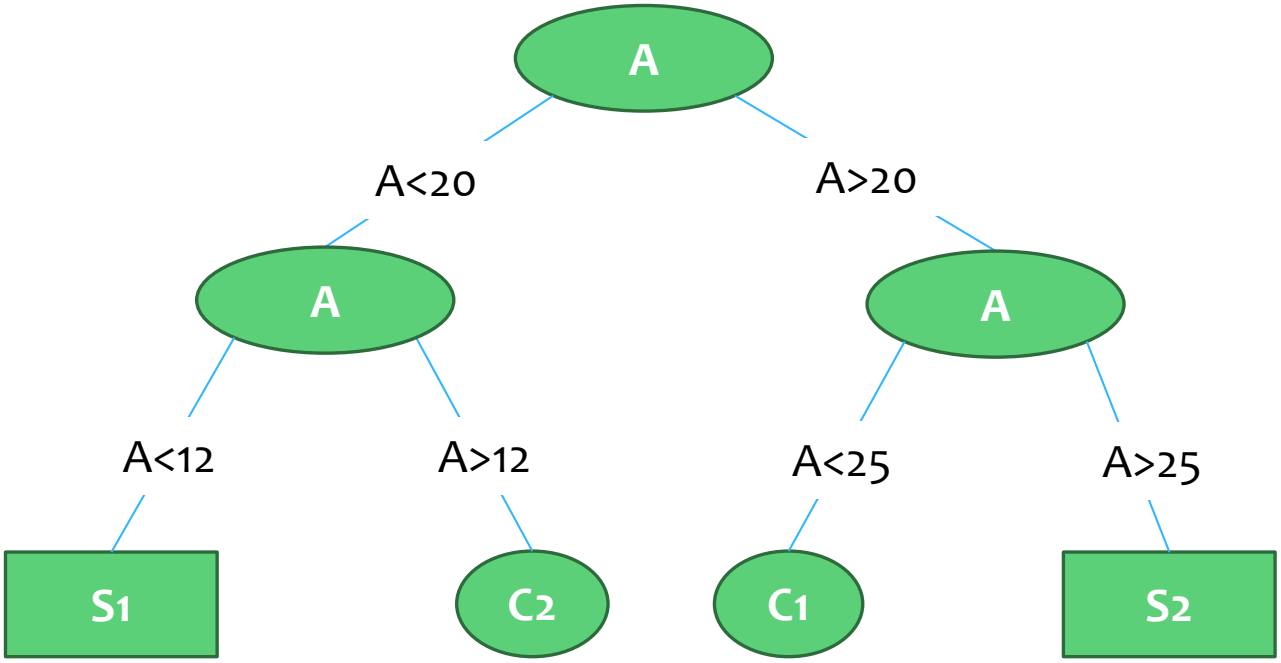
InputLet ***A*** be the set of attributes in train.

for each attribute **a** ∈ ***A*** **do**

if a does not separate the target classes **then**

$A = A - a$.

Decision Tree Example



Step 2 - Capture Feature Dependencies

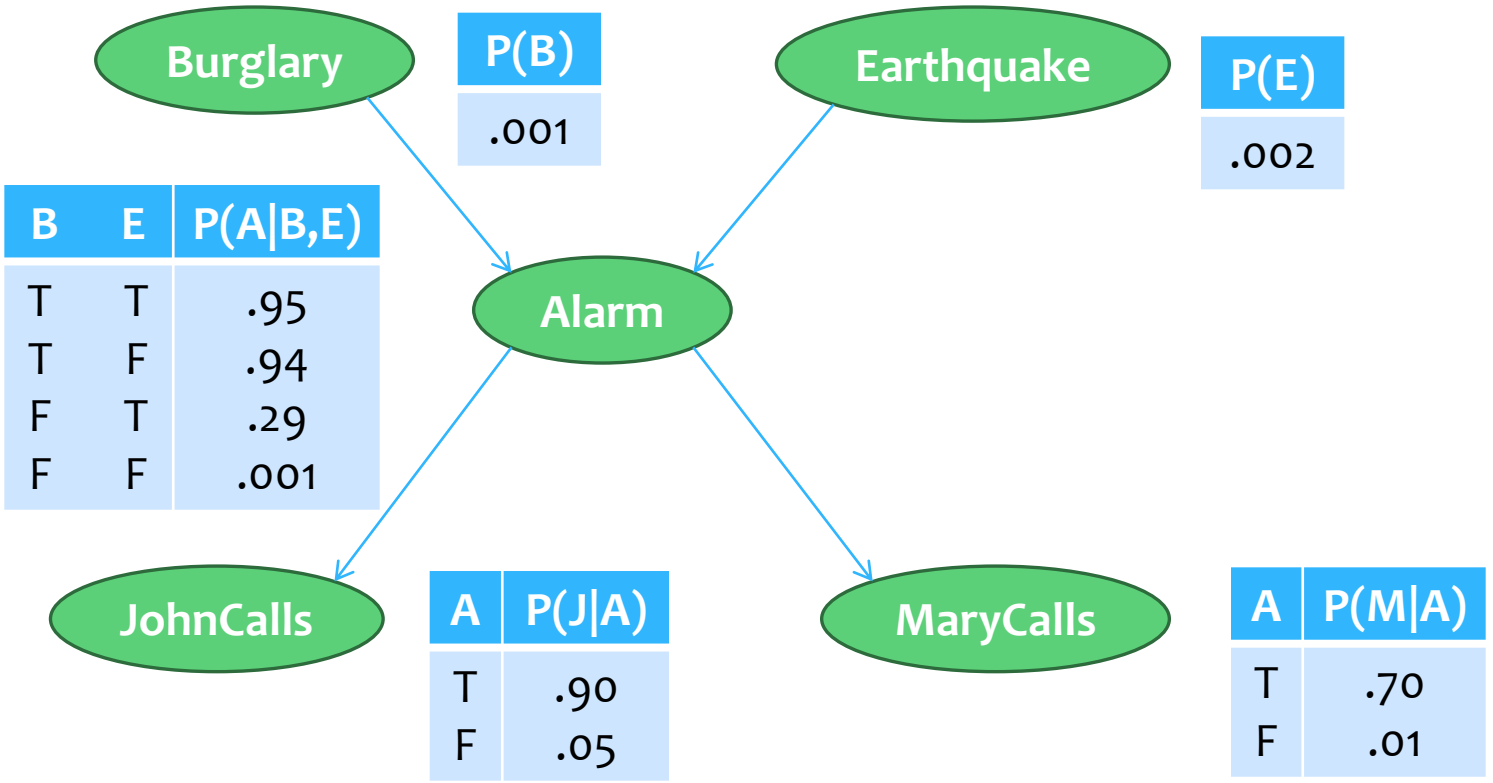
* Generate probabilistic rule in the class variable's

  << **Markov Blanket** >>

* Learn a BN model based on the top-k most-informative features
* Sort the discretized features in decreasing order
* The CaMML BN learning program

Select the top-k most-informative features (**top A**).
Learn BN from train based on top **A1**

Bayesian Network Example



Step 3: Estimate Error Rates of Features

* Error rate or probability of **e** for a feature **f**
* Predicting with the BN and using the proportion of misclassified training cases
* Enters that feature's value for each training case as a **likelihood finding,** or a **soft evidence**

Identify F and estimate R values using Algorithm 2

Algorithm 2 : Error Discovery in BN-NC

Input: Training data (train), BN from train, and error threshold (t).
Output: Erroneous features (F), and their corresponding error rates (R).

Step 1: Identify the top erroneous feature ftop.

for each feature f_i **do**
  for each record ∈ train **do**
    Cover-up f_i and predict its value using BN.
  $P_{err}(f_i)$ = fraction of train that fi is misclassified.
ftop = argmax(Perr(fi)), max err = P_{err}(ftop).
if max err < t **then return** F and R as empty sets.
else F = F + ftop, R = R + max err.

Step 2: Identify the rest of sets F and R.

min err = Perr(ftop).
while 9 feature /2 F ^ min err t **do**
  for each feature fi 62 F **do**
    for each record 2 train **do**
      Estimate likelihoods L for feature values in F.
      Cover-up fi; predict its value with BN and L.
    Perr(fi) = fraction of train with fi misclassified.
  fnext = argmax(Perr(fi)), max err = Perr(fnext).
  F = F + fnext; R = ;.
  for each feature fi 2 F **do**
    R = R + Perr(fi).
  fleast = argmin(Perr(fi)), min err = Perr(fleast).
if min err < t **then** F=F-fleast, R=R-Perr(fleast).
return the non-empty sets F and R.

Step 4 - Predict with the Noisy Features

* Likelihoods Estimation of **Test data**
* Explaining the BN Inferences (**EBI**)

for each test case $c_i \in$ test **do**
    Estimate likelihoods of F, based on R and case ci.
    Enter ci into BN to obtain inference i; I = I + i.
    Generate explanation e for inference i; E = E + e.

Experiment Setup

* HIV-1 Drug Resistance Prediction
* Acute Leukemia Subtype Classification
* Data source

Initial Noise in The Seven Mutations

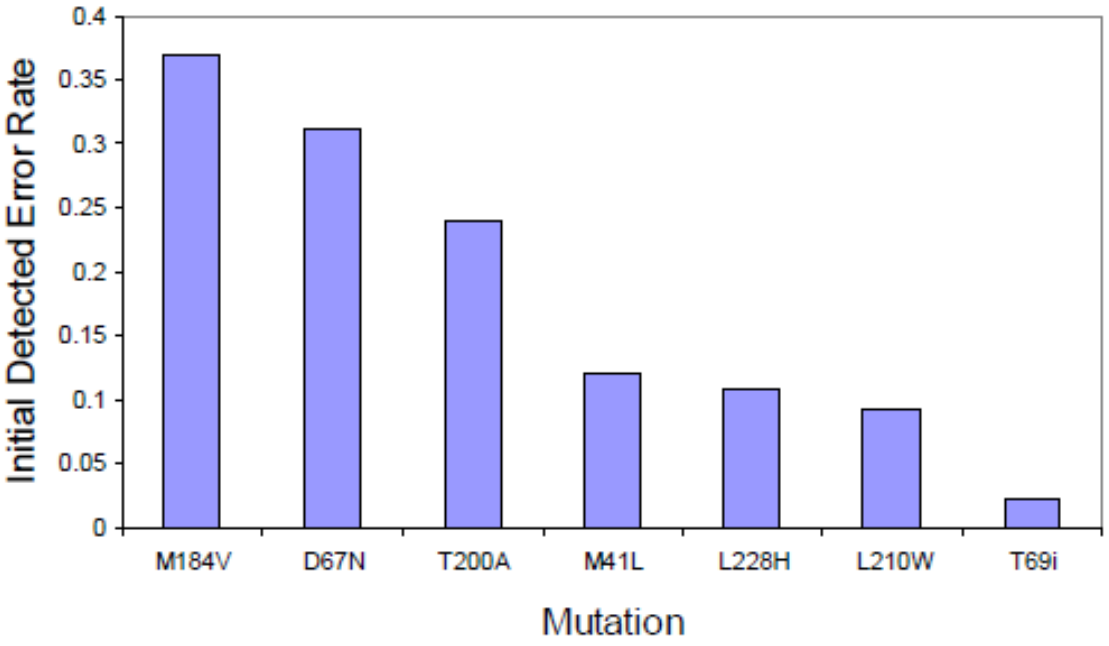
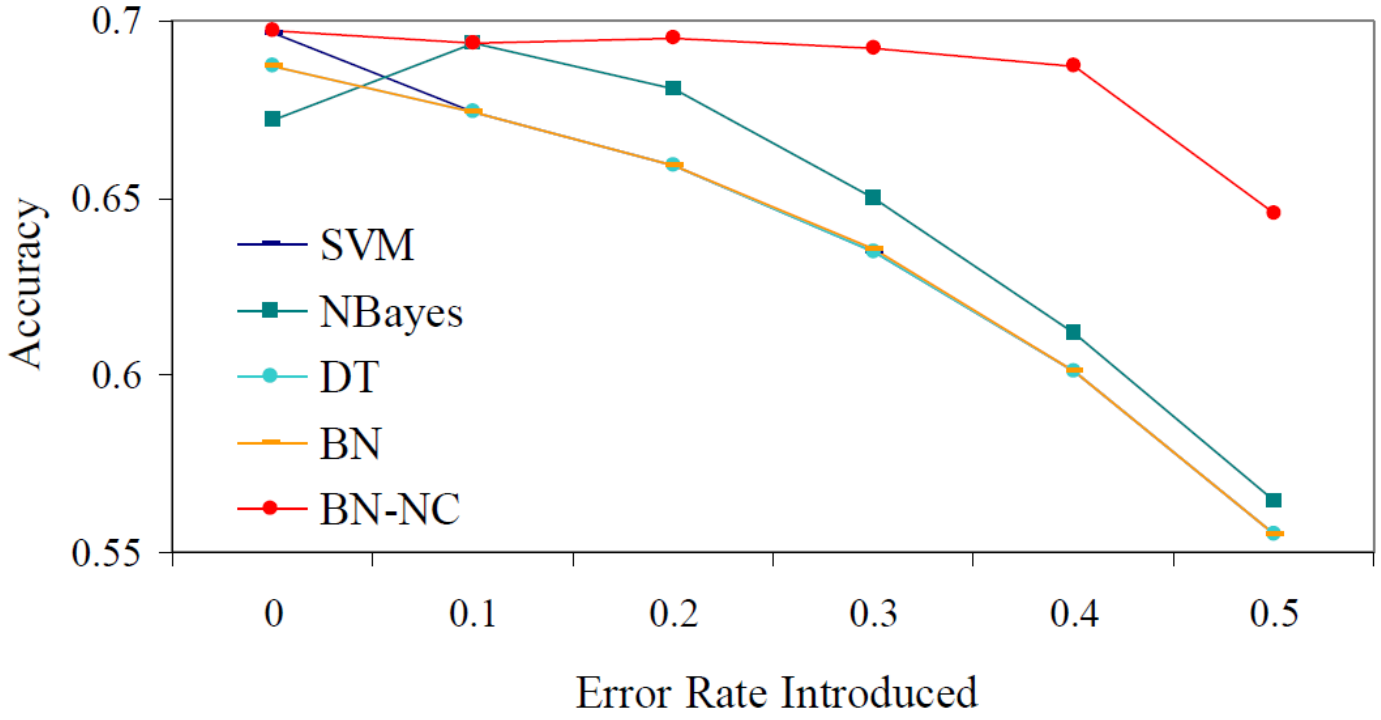


Figure 2: The initial error rates detected for TDF.

Accuracies Obtained from Each Method

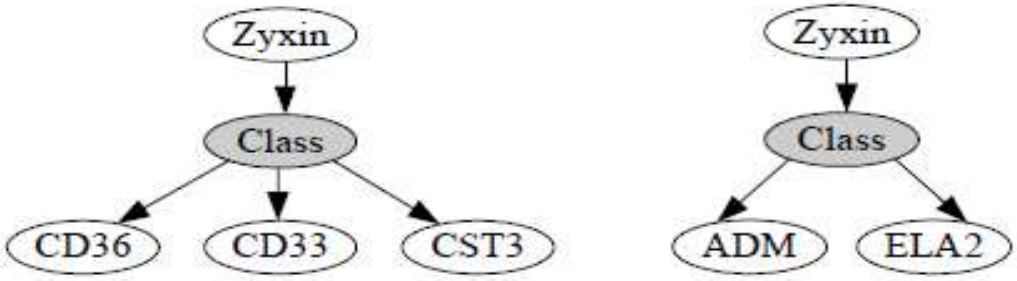


Accuracy on The ALL/AML Test Set

Table 2: Accuracy on the ALL/AML test set.

Method	Prediction Accuracy
Weighted Voting [2]	0.85
SVM [29]	0.88
EP [31]	0.91
ARAM [32]	0.94
SVM	0.88
NBayes	0.88
DT	0.91
BN	0.97
BN-NC	0.97

Shows The Markov Blanket of The Learned BN



(a) Using 24 genes (b) Using 18 genes

Figure 5: Markov blanket in learned BN for ALL/AML.

Table 3: Info. gains and error ests. for genes in Fig 5(a).

Gene-ID	Full Description	Info. Gain	Error Estimate
X95735	Zyxin	0.8680	0.0000
M27891	CST3 Cystatin C	0.7043	0.0000
M23197	CD33 antigen	0.5917	0.0526
M98399	CD36 antigen	0.5917	0.0789

Conclusion

* The proposed BN-NC framework is readily applicable to noisy biomedical classification tasks, and it also extends to other domains suffering from noisy features.