ISTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

FEATURE SELECTION USING DIFFERENT MUTUAL INFORMATION ESTIMATION METHODS

M.Sc. Thesis By
Ahmet Kenan KULE

Department : Computer Engineering

Programme : Computer Engineering

NOVEMBER 2010

ISTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

FEATURE SELECTION USING DIFFERENT MUTUAL INFORMATION ESTIMATION METHODS

M.Sc. Thesis by
Ahmet Kenan KULE
(504071521)

Date of submission : **13 September 2010**
Date of defence examination : **28 September 2010**

Supervisor(Chairman) : **Assoc. Prof. Dr. Zehra ÇATALTEPE(İTÜ)**
Members of the Examining Committee : **Assis. Prof. Dr. Mustafa E. KAMAŞAK(İTÜ)**
Assis. Prof. Dr. İsmail Arı(ÖÜ)

NOVEMBER 2010

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

FARKLI KARŞILIKLI BİLGİ KESTİRİM YÖNTEMLERİ KULLANARAK ÖZNİTELİK SEÇİMİ

YÜKSEK LİSANS TEZİ

Ahmet Kenan KULE
(504071521)

Tezin Enstitüye Verildiği Tarih : 13 Eylül 2010
Tezin Savunulduğu Tarih : 28 Eylül 2010

Tez Danışmanı : Doç. Dr. Zehra ÇATALTEPE(İTÜ)

Diğer Jüri Üyeleri : Yrd. Doç. Dr. Mustafa E. KAMAŞAK(İTÜ)

Yrd. Doç. Dr. İsmail Arı(ÖÜ)

KASIM 2010

FOREWORD

I would like thank my advisor for being such an inspiring person, my family for being always there supporting me, and my colleagues at ITU Computer Engineering Department for their continuous effort to turn work time into fun.

November 2010

Ahmet Kenan KULE
Computer Engineer

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Page

FEATURE SELECTION USING DIFFERENT MUTUAL INFORMATION ESTIMATION METHODS

SUMMARY

As high dimensional data, such as microarray data become available, fast and accurate feature selection methods have gained more importance. The aim of feature selection is both increasing classification performance and providing ease of understanding of data by keeping its definition simple.

One of the most widely used metrics in feature selection is mutual information. Estimating mutual information accurately contributes to quality of selected features. This study focuses on the role of mutual information estimation in feature selection and aims the following:

1. to give a comparison of mutual information estimation methods based on binning, KNN (K Nearest Neighbor) (Fix & Hodges, 1951) and KDE (Kernel Density Estimation) (Rosenblatt 1956),
2. to measure performance of these mutual information estimation methods on two feature selection methods: relevance based mutual information filter and min-redundancy-max-relevance (mRMR) (Peng 2005) feature selection method
3. to improve the performance of these methods through subset selection or by combination.

The results of this study show that although performance of simple relevance based feature selection improves with more sophisticated mutual information estimation methods such as KNN based and KDE based, mRMR do not benefit from this improvement.

Furthermore, it is shown that neither instance subset selection nor linear combination of these methods yield to improvements in the performance of the classification in microarray data.

FARKLI KARŞILIKLI BİLGİ KESTİRİM YÖNTEMLERİ KULLANARAK ÖZNİTELİK SEÇİMİ

ÖZET

Mikrodizi verisi gibi oldukça fazla öznitelik içeren verinin erişilebilir olması ile birlikte, hızlı ve doğru öznitelik seçim yöntemlerinin önemi artmıştır. Öznitelik seçimi uygulamasındaki amaç, sınıflandırma başarımını arttırmak olduğu kadar, aynı zamanda veriyi daha basit şekilde tanımlayarak anlaşılır kılmaktır.

Öznitelik seçiminde kullanılan ölçü birimlerinin başında karşılıklı bilgi gelmektedir. Karşılıklı bilginin doğru bir şekilde kestirilmesi seçilen özniteliklerin kalitesini arttırmaktadır. Bu çalışma öznitelik seçiminde karşılıklı bilginin kestiriminin etkisi üzerinde yoğunlaşarak, şunları hedefler:

- bölmeleme, KNN (K en yakın komşu) (Fix & Hodges, 1951) ve KDE'ye (çekirdek yoğunluk kestirimi) (Rosenblatt 1956) dayanan karşılıklı bilgi kestirim yöntemlerinin karşılaştırmasını yapmak,
- bu karşılıklı bilgi kestirim yöntemlerinin iki öznitelik seçme yöntemi üzerindeki başarımını ölçmek: ilgi tabanlı karşılıklı bilgi filtresi ve minimum-bolluk-maksimum-ilgi (mRMR) (Peng 2005) öznitelik seçme yöntemi.
- yine bu yöntemlerin başarımını altküme seçimi veya birleştirme ile arttırmak.

Bu çalışmanın sonuçları, KNN tabanlı ve KDE tabanlı yöntemler gibi daha karmaşık karşılıklı bilgi kestirim yöntemlerinin, sadece ilgi tabanlı basit öznitelik seçme işleminin başarımını arttırmasına rağmen, mRMR' ın bu yöntemlerden yararlanamadığını göstermiştir.

Ayrıca, ne altküme seçme yönteminin ne de karşılıklı bilgi kestirim yöntemlerinin lineer olarak birleştirilmesinin mikrodizi verisinin sınıflandırmasında, sınıflandırma başarımını arttırmadığı gösterilmiştir.

1. INTRODUCTION

Amount of data used in computational tasks is growing day by day. Many applications in machine learning domain have to deal with huge amount of data. Notable application areas vary from market basket analysis to Geographic Information Systems, and from Bioinformatics to Web Recommendation Engines;but they all suffer from high computational costs.

One of the recent technologies contributed to that data boom is microarrays. DNA microarrays allow monitoring of thousands of gene expression levels in a single experiment [1]. These gene expressions are used in classification of tumor tissues. Although microarrays enabled examining tissues in great depth through gene expression levels, the sample size is often limited. A side effect of this high dimensionality of microarray gene expression data is the reduction in interpretability.

Feature selection is a common dimensionality reduction approach when the computational costs are infeasibly high. This approach also helps us understand the underlyings of the data (e.g. identifying genes responsible for a certain type of cancer) in bioinformatics.

Feature selection methods are divided into two groups: *filters* and *wrappers*. First determines the usefulness of a feature according to the intrinsic characteristics of data while the second lets a classifier decide which features are better. In classification tasks, filter methods are known to be faster and are easily implemented but wrapper methods perform better due to their strong bonds with a classifier [2, 3].

Filter methods usually need a metric to determine the relation between features. Mutual information is one of the most common among these metrics.

One of the most recently developed filter feature selection methods is minimum-redundancy-maximum-relevance (mRMR) [4] feature selection. This

method relies heavily on mutual information and is explained in detail in Section 3.4.

This study focuses on the following subjects :

1. Comparison of different mutual information estimation methods.
2. Possible improvements on the performance of the mutual information estimators through combination and instance subset selection.
3. Role of mutual information estimation method in mRMR feature selection performance.

This thesis is organized as follows:

- Second chapter provides information about recently developed mutual information estimation methods.
- Third chapter provides information about feature selection methods and especially minimum-redundancy-maximum-relevance (mRMR).
- Fourth chapter contains experimental results for mutual information estimators on artificial data and considers possible improvements.
- Fifth chapter summarizes the previous work on feature selection for microarray data and contains the experiment results for feature selection using different mutual information estimators and mRMR.
- Sixth chapter concludes the findings from this work and discusses future improvements.

2. MUTUAL INFORMATION

Mutual information is a commonly used metric for capturing dependence information between variables. Mutual information [5], [6] for the bivariate random variables (X, Y) is defined as follows:

$$I(X,Y) = \iint p_{XY}(x,y) \log \left(\frac{p_{XY}(x,y)}{p_X(x) p_Y(y)} \right) dxdy \qquad \textbf{(2.1)}$$

In the Equation 2.1, $p_{XY}(x,y)$ is the joint probability density function and $p_X(x)$ and $p_Y(y)$ are the marginal probability distribution functions. The base of the logarithm defines the unit of measurement.

The mutual information is often preferred to other dependence metrics as it captures both linear and nonlinear dependencies and the mutual information between two variables converges to zero if and only if these two variables are independent.

Mutual information has the following properties:

- It is nonnegative: $I(X,Y) \geq 0$.
- It is symmetric: $I(X,Y) = I(Y,X)$.
- It is additive for independent variables: if $P_{XYWZ}(x,y,w,z) = P_{XY}(x,y)P_{WZ}(w,z)$ then $I(X,W:Y,Z) = I(X:Y) + I(W:Z)$.

2.1 Mutual Information Estimation

In real world applications, mutual information cannot be determined exactly since the distributions of the random variables are not known. It can only be estimated from a finite amount of data gathered. Steuer et al. [7] compared different algorithms to estimate mutual information and discussed the effects of finite size data.

In this section, three mutual information estimators namely binning based, KNN based and KDE based, are introduced.

2.1.1 Binning Based Estimator

Since the distributions of the random variables cannot be determined most of the time in real world examples, a common practice is to partition the data into finite size bins and compute mutual information in the discrete domain. In order to compute the probabilities, data points falling into each bin is counted. Equation 2.2 shows the computation of mutual information for discrete variables.

$$I_{binned}(X,Y) = \sum_{ij} p_{xy}(i,j) \log \left(\frac{p_{xy}(i,j)}{p_x(i)p_y(j)} \right) \tag{2.2}$$

This method is known to overestimate the information shared between two uniform random variables [7]. Another drawback of binning based estimator is its sensitivity to the selection of the origin and the bin size [8]. It is improved by changing the bin sizes according to the distribution of data [9]. The adaptive binning method [9] determines the bin sizes so that every bin has equal number of instances.

2.1.2 KNN Based Estimator

Another way to estimate MI is to use the relation between MI and entropy. MI may be estimated by estimating the entropy measures $H(X)$, $H(Y)$ and $H(X,Y)$ separately and then using Equation 2.3.

$$I(X,Y) = H(X) + H(Y) - H(X,Y) \tag{2.3}$$

A common definition for the entropy is done by Shannon:

$$H(X) = -\int p_x(x) \log p_x(x) dx \tag{2.4}$$

While there is extensive literature on the estimators for the Shannon entropy, these estimators have never been used for estimating MI before their work according to Kraskov et al. [10].

For a univariate random variable, entropy may be estimated based on the distances between instances using Equation 2.5 if the instances can be ordered and the difference between the instances vanishes going to infinity. While this is a good approximator, it is not generalized to higher dimensions.

$$H(X) \approx \frac{1}{N-1} \sum_{i=1}^{N-1} \log(x_{i+1} - x_i) + \psi(1) - \psi(N) \tag{2.5}$$

In Equation 2.5, $\psi(x)$ is the digamma function which satisfies the following equations. C is the Euler-Mascheroni constant.

$$\begin{aligned} \psi(x) &= \Gamma(x)^{-1} d\Gamma(x)/dx \\ \psi(x+1) &= \psi(x) + 1/x \\ \psi(1) &= -C \\ C &= 0.5772156\ldots \end{aligned} \tag{2.6}$$

Kraskov et al. [10] generalized this approximation by defining a distance measure in higher dimensional space. In order to rank instances on the spaces X, Y and $Z = (X,Y)$, a previously defined metric $d_{ij} = ||z_i - z_j||$ is redefined as follows:

$$||z - z'|| = \max\{||x - x'||, ||y - y'||\} \tag{2.7}$$

Using this maximum norm, $\varepsilon_x(i)/2$ (or $\varepsilon_y(i)/2$) is defined as the projection of the distance from z_i to its kth neighbour on the x (or y) space. Given this distance, $n_x(i)$ (or $n_y(i)$) is defined as the number of instances who is closer than $\varepsilon_x(i)$ (or $\varepsilon_y(i)$). Equation 2.8 shows the formal definition.

$$n_x(i) = |\{z_{i'} | \|x_i - x_{i'}\| \leq \varepsilon_x(i)\}| \tag{2.8}$$

And the mutual information estimator is defined as follows:

$$I(X,Y) = \psi(k) - \langle \psi(n_x + 1) + \psi(n_y + 1) \rangle + \psi(N) \tag{2.9}$$

KNN (K nearest neighbor) [11] based mutual information estimator is considered the best choice among KDE, KNN and Edgeworth [12] estimators for very short data (50 - 100 data points) with low noise and short data (100 - 1000 data points) in general [13].

One drawback of this estimator is that there seems to be no systematic way of determining optimum k value. Still, this parameter can be optimized by cross validation. Kraskov et al. [10] suggested to set k a value between 2 and 4, and avoid using large values for k as it increases the systematic error.

2.1.3 Kernel Density Estimation (KDE) based estimator

Kernel Density Estimation (KDE) [14] is a nonparametric method for estimating probability densities. The probability density estimator is defined by Equation 2.10.

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \tag{2.10}$$

In Equation (2.10, K is a kernel function that satisfies Equation 2.11, h is the kernel width. One of the most commonly used kernel functions is gaussian kernel.

$$\int_{-\infty}^{\infty} K(x)dx = 1 \tag{2.11}$$

In parametric density estimation, data is assumed to be drawn from a known parametric family of distributions, like normal distribution, and the parameters for that distribution is estimated. For example, if the data is assumed to be drawn from a normal distribution, the parameters to be estimated are mean (μ) and the variance (σ^2). As a nonparametric density estimator, KDE lets the data express itself. The density estimation is constructed with the contribution of bumps at each data point. Kernel function determines the shape of these bumps.

Silverman [8] illustrated that KDE has some advantages over histograms:

- Histograms basically have two parameters: origin and bin width. The choice of origin changes the performance of the estimator. Using KDE, we overcome the problem of selecting an origin.

- Histograms have a fixed shape for bins. With the help of kernel function, shape of bumps may be adjusted.

Mutual information may be estimated using Kernel Density Estimation by estimating the probability densities in Equation 2.1 separately [15].

2.2 Evaluation of a MI Estimator

Performance of MI estimators introduced in this section is measured by different criterias according to the type of data. Since the exact MI value for the artificially generated data is known, systematic error, standard deviation and mean square error for that type of data are reported. On the other hand, the exact distribution for the microarray data is not known. For this reason, the performance of MI estimator is measured by the quality of the features selected by the feature selection method using that MI estimator. The quality of selected features are determined by the classification error on the dataset using these features.

Here are the definitions of systematic error, standard deviation (STD) and mean square error (MSE):

Systematic Error or bias of an estimator is the consistent difference between the estimations and the actual value of the estimated attribute. This type of error has both a direction and a magnitude.

Standard Deviation measures how much the estimated value varies around the actual value.

Mean Square Error measures how much the estimator differs from the actual value. MSE is always positive and has only magnitude.

3. FEATURE SELECTION

Feature selection is the task of finding a subset of features that represents the data most informatively. Once that kind of a subset is found, machine learning applications like classification can be run faster and without accuracy loss.

One can attempt to find an optimum subset of features using a brute force approach by trying every possible subset of features. However, this approach takes exponential time and is not feasible in many real world applications. Two example application areas where feature selection is vital are microarray classification and text categorization. A typical gene expression profile can have a varying number of features from 6000 to 60000. In text categorization domain, feature selection is used to reduce the vocabulary size from hundreds of thousands of words to 15000 [16].

Another benefit of feature selection is to determine what underlies in the data. For example, selecting a small number of relevant genes, apart from reducing computational cost of the classification task, underlines important genes so that results are biologically interpretable.

With so many benefits, many feature selection methods have been developed through the years [4, 17, 18, 19, 20, 21]. Basicallly, these methods are divided into two categories: *filters* and *wrappers*.

In this chapter, basic properties of *filters* and *wrappers* are discussed and then mutual information filter and the mRMR feature selection method [4] is introduced.

3.1 Filter Methods

Filter methods select features based on the intrinsic characteristics of data. For each feature, a score is computed using a predefined metric. This may be a

one-pass process or may consist of several repetitions for some pairs or subsets of features. In the end, low scoring features are removed from feature set [21].

Filter methods are known to be fast and easy to implement. Both univariate [17] methods that deal with feature pairs only and multivariate methods [22, 23, 24], that deal with a subset of features exist. One disadvantage of *filter* methods is that, as they are independent from the classifier, they cannot exploit the unique advantages of classifiers in the feature selection phase.

Widely used filter methods are information gain [17], mutual information [18], Relief-F [19], FCBF [20] and mRMR [4].

3.2 Wrapper Methods

Wrapper methods employ a classifier to decide on the best features. The score for one feature or a group of features is determined by the performance of these features when these features are fed into a specific classifier. Thus, every classifier selects a possibly different subset of features.

Wrapper methods are computationally expensive thus, in most of the studies in the field of DNA microarrays, *filter* methods are used [19].

3.3 Mutual Information Filter

Filter methods need a metric to measure the dependency within the data itself. Mutual information is a commonly used metric to measure both linear and nonlinear dependencies. Most trivial way to employ mutual information as a filter type feature selection method is to measure the MI between each feature and the class label individually, to sort these features according to their MI values and then to take a certain number of top features (features that has most information about the class label). This approach is called mutual information filter throughout this work.

3.4 Minimum-Redundancy-Maximum-Relevance (mRMR)

mRMR is a recently developed filter feature selection method introducing a new criteria, called minimum-redundancy-maximum-relevance [4]. Before introducing the mRMR, terms maximum dependency, maximum relevance and minimum redundancy will be properly defined.

3.4.1 Maximum Dependency

The trivial approach for filter methods of feature selection is to select the best subset according to its similarity(dependency) to class label. This approach is called maximum dependency. In order to compute the dependency among variables, a dependency metric has to be defined. We will use the mutual information metric, which is discussed in Chapter 2. Equation 2.1 may be generalized to a subset of features and the class label as follows:

$$\begin{aligned} I(S_m,c) &= \iint p(S_m,c)\log\left(\frac{p(S_m,c)}{p(S_m)p(c)}\right)dS_m dc \\ &= \iint p(S_{m-1},x_m,c)\log\left(\frac{p(S_{m-1},x_m,c)}{p(S_{m-1},x_m)p(c)}\right)dS_{m-1}dx_m dc \\ &= \int\cdots\int p(x_1,\cdots,x_m,c)\log\left(\frac{p(x_1,\cdots,x_m,c)}{p(x_1,\cdots,x_m)p(c)}\right)dx_1\cdots dx_m dc. \end{aligned} \quad \textbf{(3.1)}$$

In this equation S_m refers to a subset of variables with m variables and c refers to the class label. The idea is to find the most informative subset of features about the class label. Even though the definition is quite simple, computation of mutual information for a particular subset is not easy because of the differenity of making multivariate density estimations in a high dimensional space. There is often a lack of necessary number of samples, especially in bioinformatics.

3.4.2 Maximum Relevance

As an alternative, the maximum relevance approach approximates the dependency among features using a series of bivariate calculations and defined as follows:

$$D(S,c) = \frac{1}{|S|}\sum_{x_i\in S} I(x_i,c) \quad \textbf{(3.2)}$$

By approximating dependence between a subset of variables and the class label to the average dependence value for this subset, maximum relevance approach overcomes the computational cost.

3.4.3 Combining Max-Relevance and Min-Redundancy

mRMR, goes one step further by considering the redundancy among the chosen features. Selected subset by the maximum relevance criteria considers the most informative genes among the full subset. But these features may be highly correlated and therefore classifier may benefit little by using them all together. Therefore, highly similar features should be eliminated from the subset. The same metric used in measuring dependency between features and class label, may be utilized to measure the dependency between features. By this way, features with no or little use together with the previously selected subset may be eliminated at each iteration. Redundancy between two variables are defined in Equation 3.3.

$$R(S) = \frac{1}{|S|^2} \sum_{x_i, x_j \in S} I(x_i, x_j) \tag{3.3}$$

Using the definitions of maximum relevance and minimum redundancy, mRMR defines the term to be optimized in feature selection as follows:

$$\max \Phi_1(D,R), \Phi_1 = D - R \tag{3.4}$$

$$\max \Phi_2(D,R), \Phi_2 = D/R \tag{3.5}$$

These two metrics defined in Equations 3.4 and 3.5, first one optimizing the difference and the second optimizing the ratio, are referred as MID and MIQ throughout this text.

Trying to optimize one of these functions (MID and MIQ), mRMR starts by selecting the most relevant feature as the subset S. Iteratively, most useful (most relevant feature having minimum redundancy among the set S) feature will be added to S. mRMR algorithm is shown in Algorithm 1.

Algorithm 1 mRMR algorithm

$S_{selected} \leftarrow argmax(I(s,c)), s \in S_m$
$S_{left} \leftarrow S_m / S_{selected}$
while $n < 50$ **do**
  if $method = MID$ **then**
    $f \leftarrow argmax(I(s,c) - R(s \cup S_{selected})), s \in S_{left}$
  else
    $f \leftarrow argmax(I(s,c) / R(s \cup S_{selected})), s \in S_{left}$
  end if
  $S_{selected} \leftarrow S_{selected} \cup f$
  $S_{left} \leftarrow S_{left} / f$
end while

4. EVALUATION OF MI ESTIMATORS

In this chapter, performance of binning based, KNN based and KDE based mutual information estimators are evaluated on artificial data and possible improvements for these methods are proposed.

4.1 Performance of MI Estimators on Artificial Data

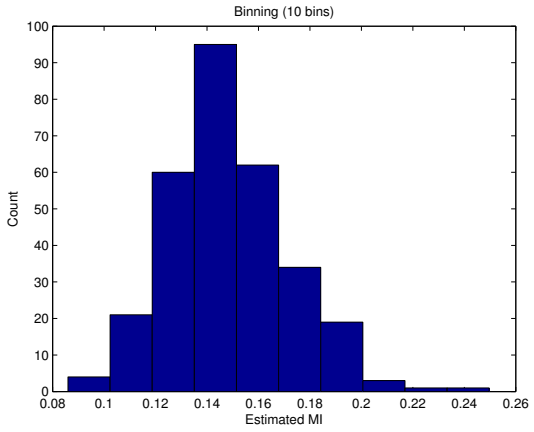
In order to determine the performance of MI estimators, artificial data in the form of uniform and Gaussian distribution are generated. This artificial data is used to determine the optimum values for the method parameters like k for K-nearest-neighbor estimator and bandwidth for KDE.

4.1.1 Uniform Distribution

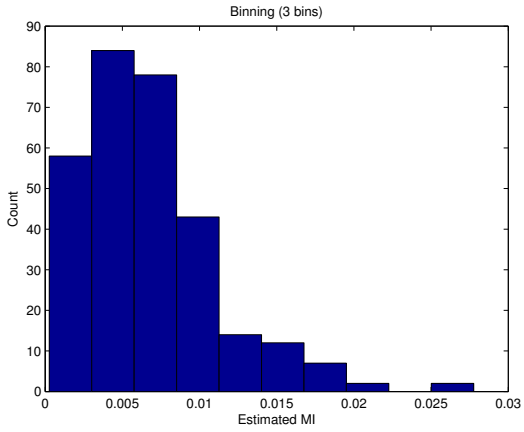
In these experiments, 300 samples are drawn from a uniform distribution. Mutual information is estimated for this artificial dataset using binning estimator with two different discretization methods, KNN based estimator and KDE based estimator. The experiment is repeated 300 times.

For the first binning estimator, data is partitioned into 10 bins. For the second method, data is partitioned into 3 bins using the discretization method in [25]. Equation 4.1 gives the details about the discretization method. In Equation 4.1, μ and σ represents the mean and the standard deviation respectively. KNN parameter K is set to 6 (default in implementation by Kraskov et al. [10]). KDE bandwidth is set to 0.1.

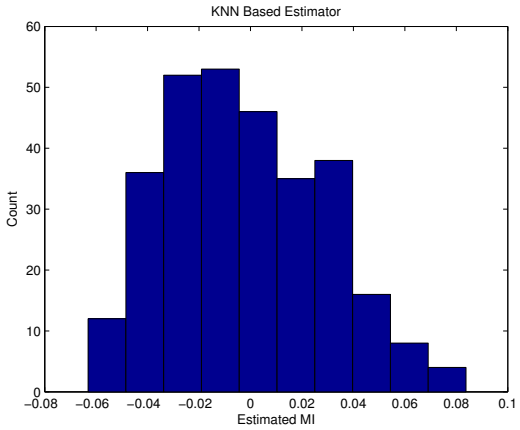
$$\begin{aligned} x \leq \mu - \sigma/2 &\Rightarrow x' = -1 \\ x \geq \mu + \sigma/2 &\Rightarrow x' = 1 \\ \textit{Otherwise} &\Rightarrow x' = 0 \end{aligned} \tag{4.1}$$



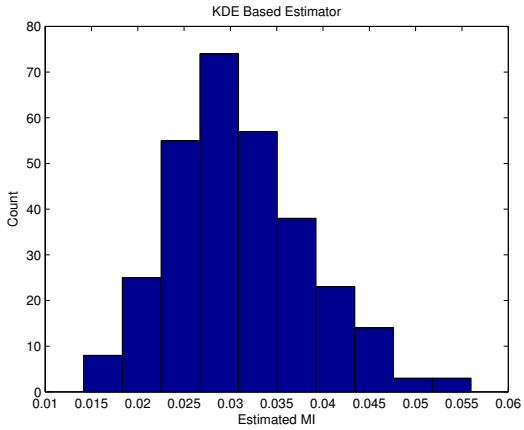
(a) Binning based estimator with 10 bins



(b) Binning based estimator with 3 bins



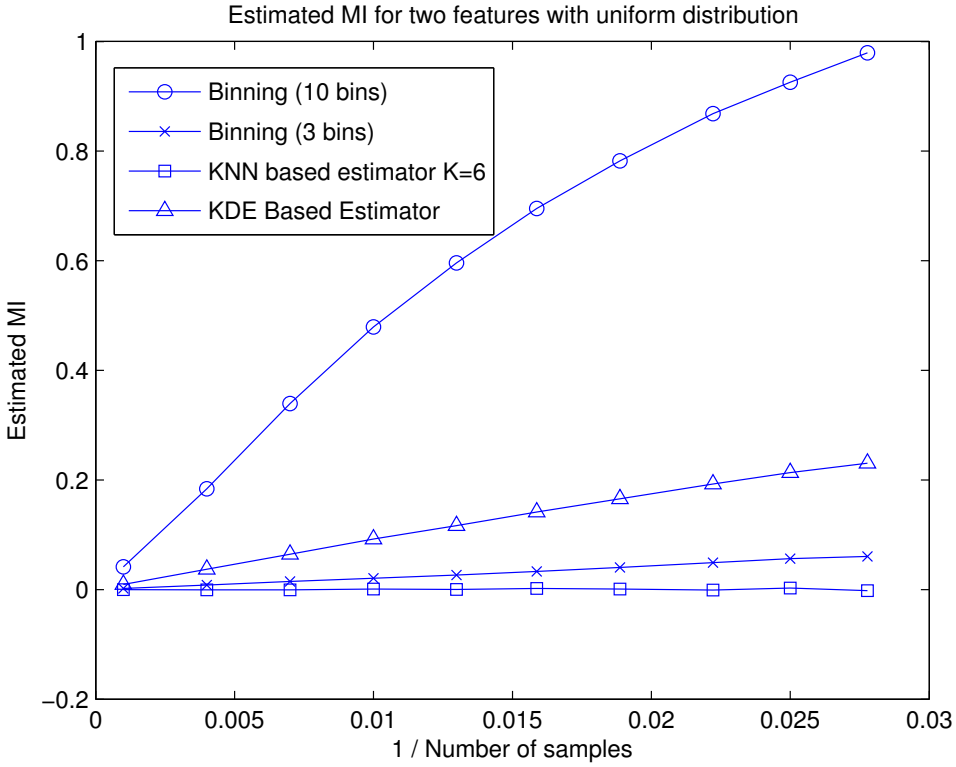
(c) KNN based estimator



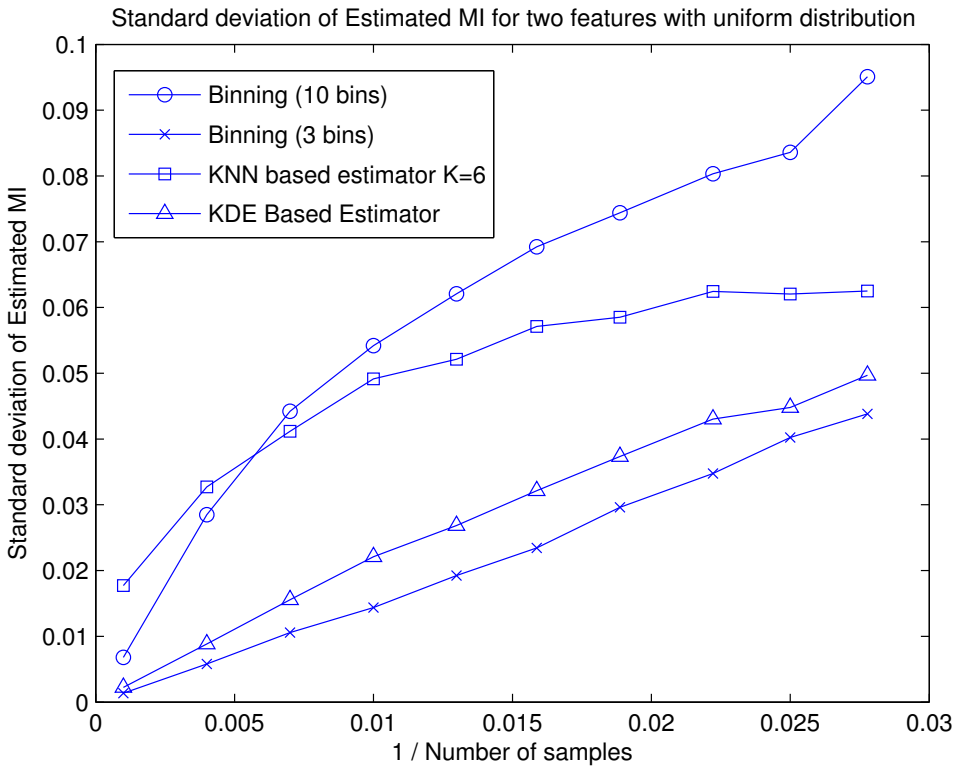
(d) KDE based estimator

Figure 4.1: Histograms of estimated MI for two features with uniform distribution. Since the two features are independent, actual mutual information is zero.

Figure 4.1 shows the results for these experiments. Experiments on uniform artificial data shows that the mean estimated mutual information for 300 runs are 0.15, 0.06, 0, 0.03 for the estimators based on binning with 10 bins, binning with 3 bins, KNN (K=6, default value for reference implementation) and KDE (kernel bandwidth=0.1) respectively. Since the two variables are independent, the actual mutual information is 0. While the mean value for KNN based estimator is very close to the actual MI value it fails to satisfy the rule that mutual information should always be positive. Note that the MI is overestimated by 0.15 using binning based estimator with 10 bins which is in agreement with Steuer et al.'s work [7]. Also note that Figures 4.1a and 4.1b show that discretization effects MI estimations.



(a)



(b)

Figure 4.2: Estimated MI (a) and standard deviations for estimated MI (b) for two features with uniform distribution.

Binning (3 bins) estimator in Figure 4.1b shows very close performance to that of KNN (K=6) estimator and this estimator is also very good in terms of its variance.

4.1.2 Gaussian Distribution

In these experiments N number of samples are drawn from a gaussian distribution with a mean of 0 and a covariance of $r = \{0, 0.3, 0.6, 0.9\}$ and mutual information is estimated for this set using binning based, KNN based ($K = 1..5, 10$) and KDE based estimators. Bandwidth parameter for KDE estimator is calculated using Equation 4.2, optimal gaussian kernel bandwidth from Silverman [8], for 2 dimensions.

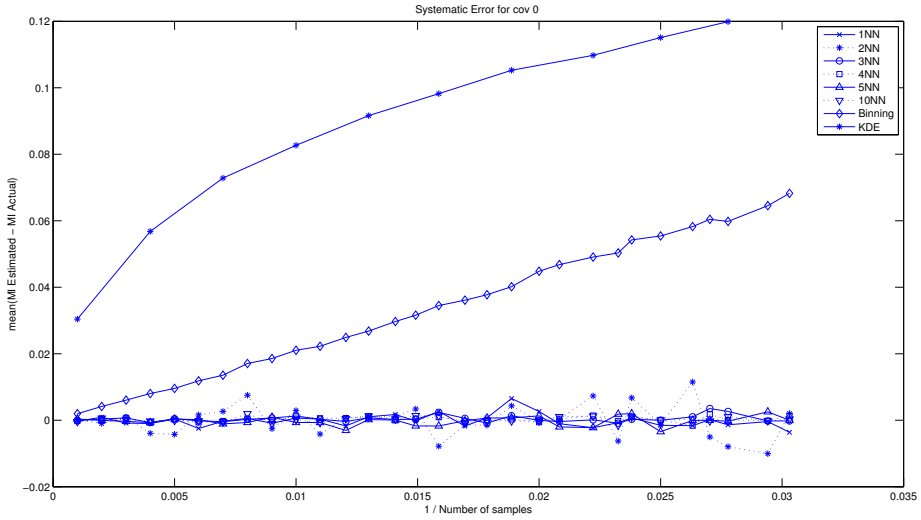
$$h = \left\{ \frac{4}{(d+2)} \right\}^{1/(d+4)} n^{-1/(d+4)} \tag{4.2}$$

Kraskov et al. [10] showed that systematic error (Estimated MI - Actual MI) for KNN based estimator scales with $N^{-1/2}$ for $N \approx 10^3$ and predicted that the true behaviour is probably $\sim 1/N$. Experiment results shown in Figures 4.3 and 4.4, are similar, however, number of samples for a microarray datasets is much less than 10^3. KDE based estimator is superiour to binning for $r = \{0, 0.3\}$ in terms of systematic error and worse for the rest. Another interesting point is that, while KNN based method underestimates the MI most of the time, other methods' behaviour vary with the variance.

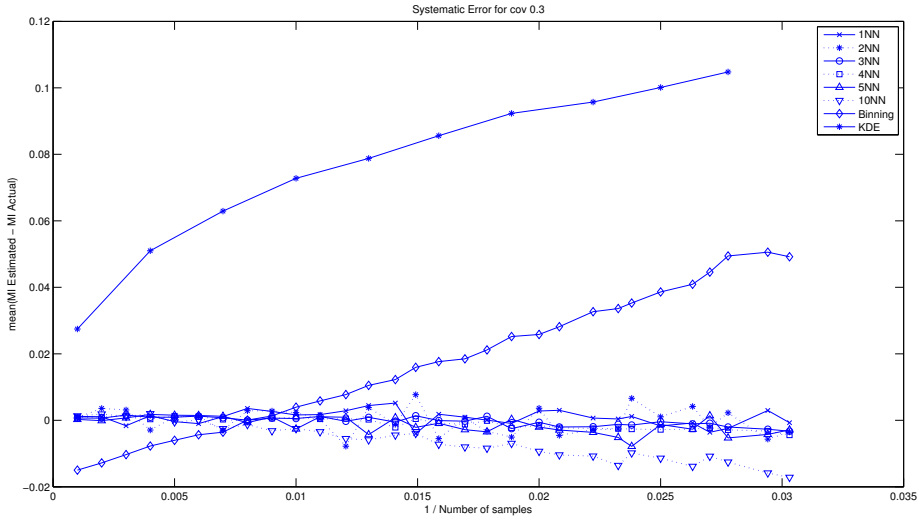
Kraskov et al. [10] also showed that the systematic error tends to zero as $N \to \infty$ for KNN based MI estimator which means that KNN based estimator is unbiased if enough data points are acquired. From Figures 4.3 and 4.4, it is seen that this behaviour is unique to KNN based estimator. While the bias of KDE based estimator decreases with increasing number of samples, it does not vanish. Although increasing the sample size beyond 1000 may help, we are not interested in that scale. Binning based estimator does not even benefit from the increasing sample size.

As seen from Figure B.2a, KDE and binning based estimators are the best in terms of standard deviations and the standard deviation decreases with the increasing

k. Statistical error for KDE and Binning estimators are smaller compared to the KNN based estimator for K $<$ 10. Since standard deviations for different covariance values were almost the same, only results with covariance 0.9 is reported.

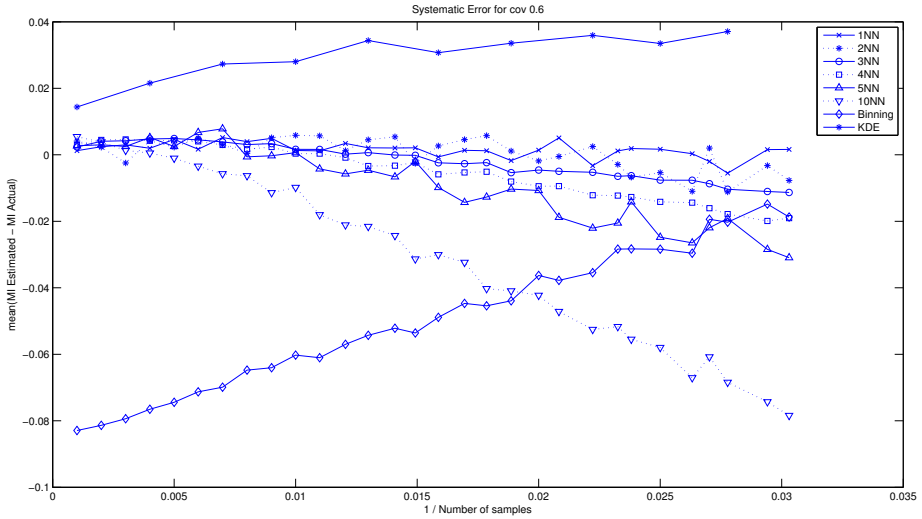


(a)

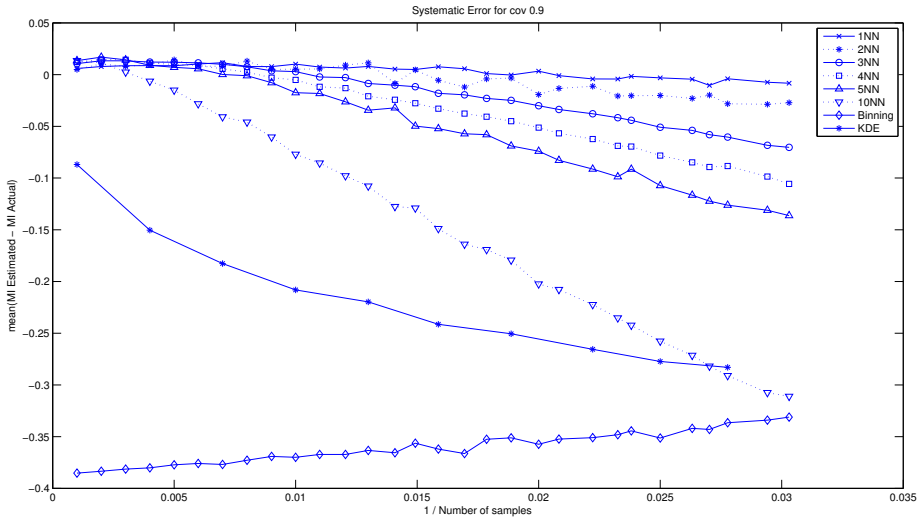


(b)

Figure 4.3: Systematic error of MI estimators for two gaussian random variables with zero mean and covariance 0 (a) and 0.3 (b).



(a)



(b)

Figure 4.4: Systematic error of MI estimators for two gaussian random variables with zero mean and covariance 0.6 (a) and 0.9 (b).

One way to determine the performance of MI estimators on the estimation of relevance between a continuous variable (feature) and a discrete class label is to discretize the second variable so that the second variable is analogous to class labels.

Using this approach, two gaussians with zero mean and $r = 0, 0.3, 0.6, 0.9$ covariance are generated. Second variable is discretized using 0 as a threshold. MI is estimated using KNN based estimator for $K = 1, 2, 3, 4, 5, 10$ and binning based estimator.

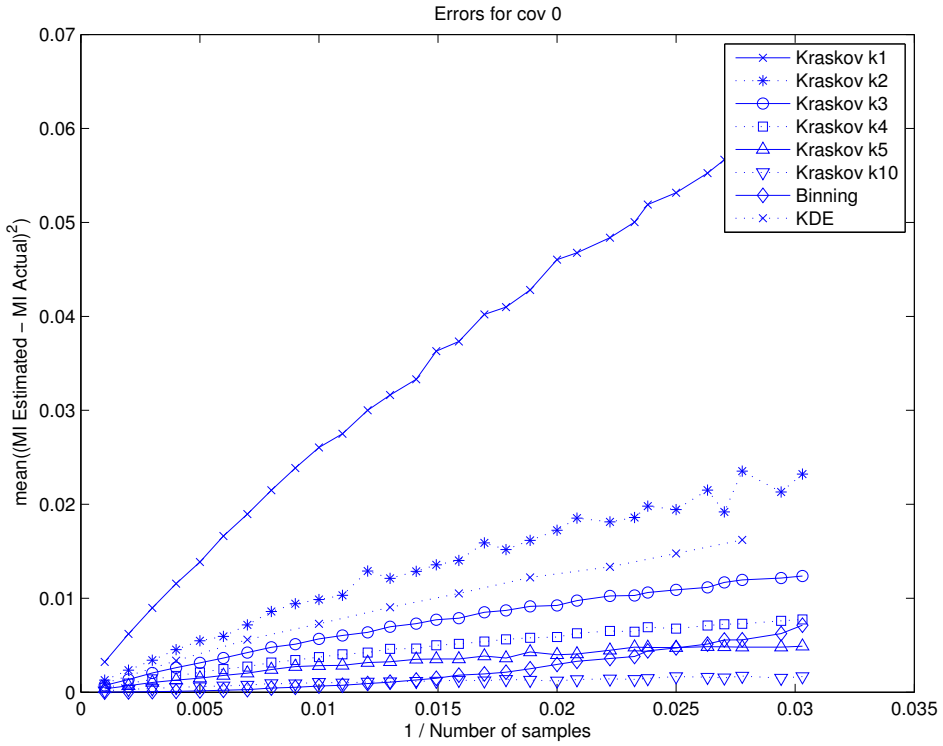
Figure B.1 shows that the performance of KNN based estimator decreases significantly if the second variable is discretized. For this reason, this estimator may not be the suitable for measuring the relevance between a feature and class label. While the experiments are carried out with $r = 0, 0.3, 0.6, 0.9$, only results with $r = 0, 0.9$ are reported for simplicity. With the second variable discretized, KNN based estimator changed its behaviour and is still biased when the number of samples are large enough. Also note that the bias is increasing with increasing covariance.

Figure B.2b shows the statistical error for the MI estimation between a continuous and a discrete variable. Comparing Figures B.2a and B.2b, statistical error decreased when the second variable is discretized. But the systematic error is so large that KNN based estimator should not be considered robust.

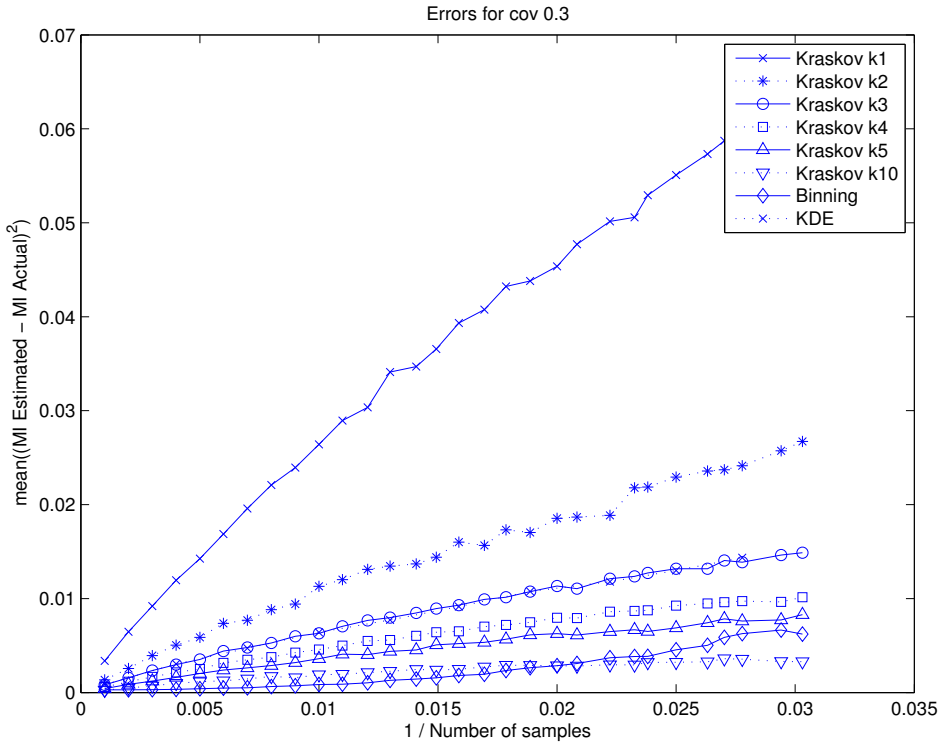
Systematic errors are useful to evaluate the estimators for being biased or unbiased, and gives the direction of the error like the estimator is consistently underestimating or overestimating. To rank estimators based on their performance on gaussian data, a scalar quantity, mean square error may be used. Mean square error determines the quality of the estimation based on the variance and unbiasedness of the estimation.

Figures 4.5 and 4.6 shows the mean square errors for all MI estimators. The performance of the MI estimators are reported as follows according to their mean square errors:

- 1NN gives the worst performance in almost all cases.
- KDE and binning based methods may be preferred for low covariance values. Performance difference between KDE and binning based methods are neglectable for small covariance values.
- Increasing k value for KNN based method decreases performance on high covariance values.
- Although Kraskov et al. [10] suggested using a value between 2-4 for k, slightly higher k values may be preferred.

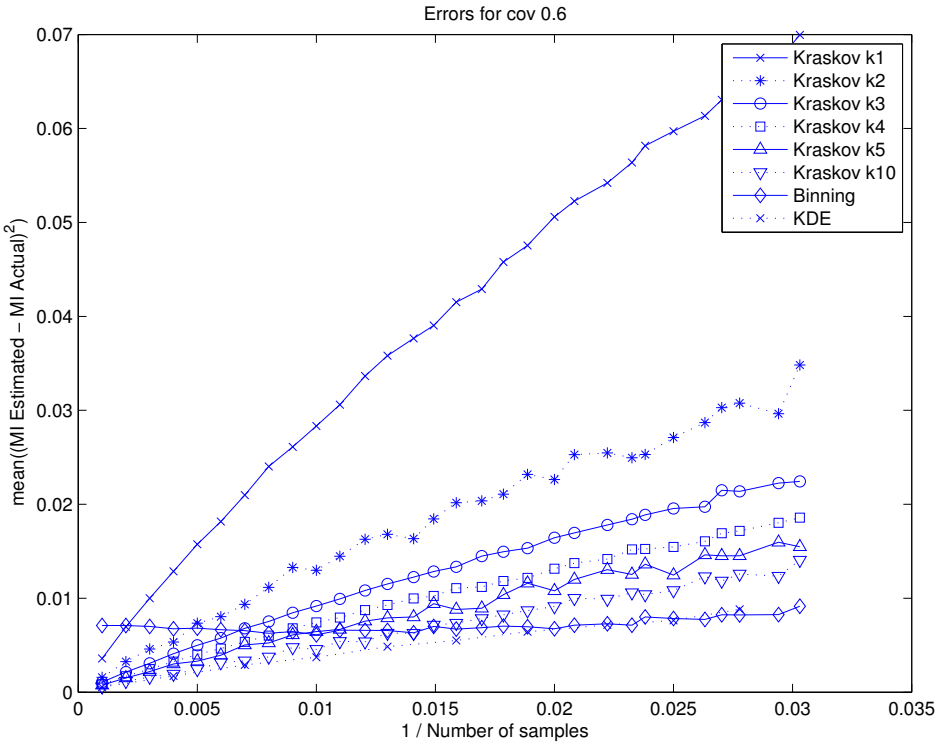


(a)

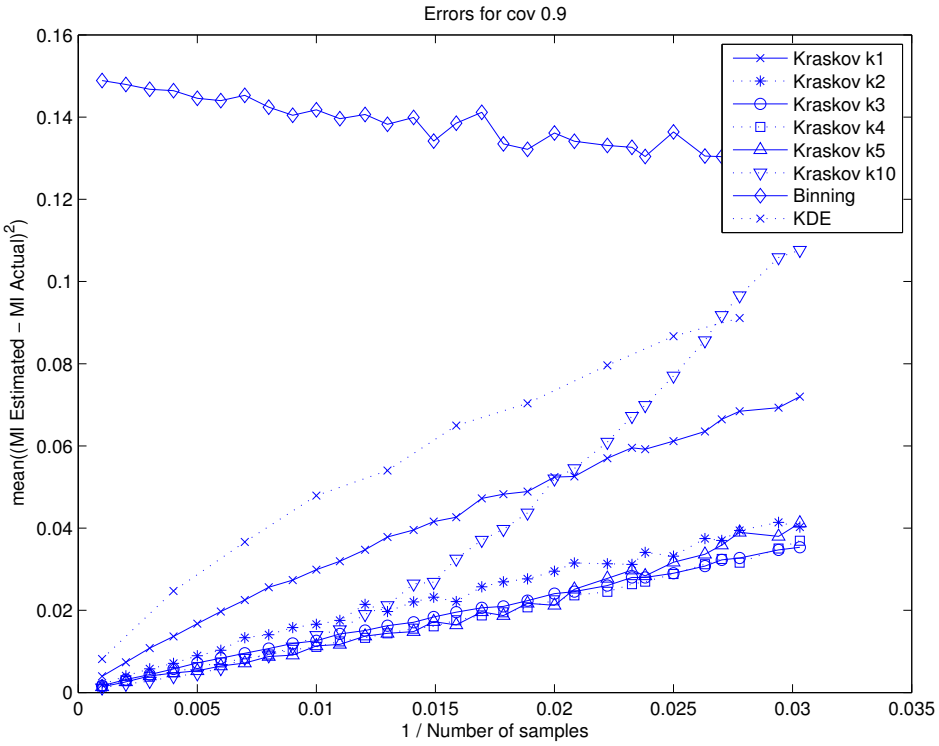


(b)

Figure 4.5: MI estimation mean square errors (MSE) with zero mean and covariance 0 and 0.3.



(a)



(b)

Figure 4.6: MI estimation mean square errors (MSE) with zero mean and covariance 0.6 and 0.9.

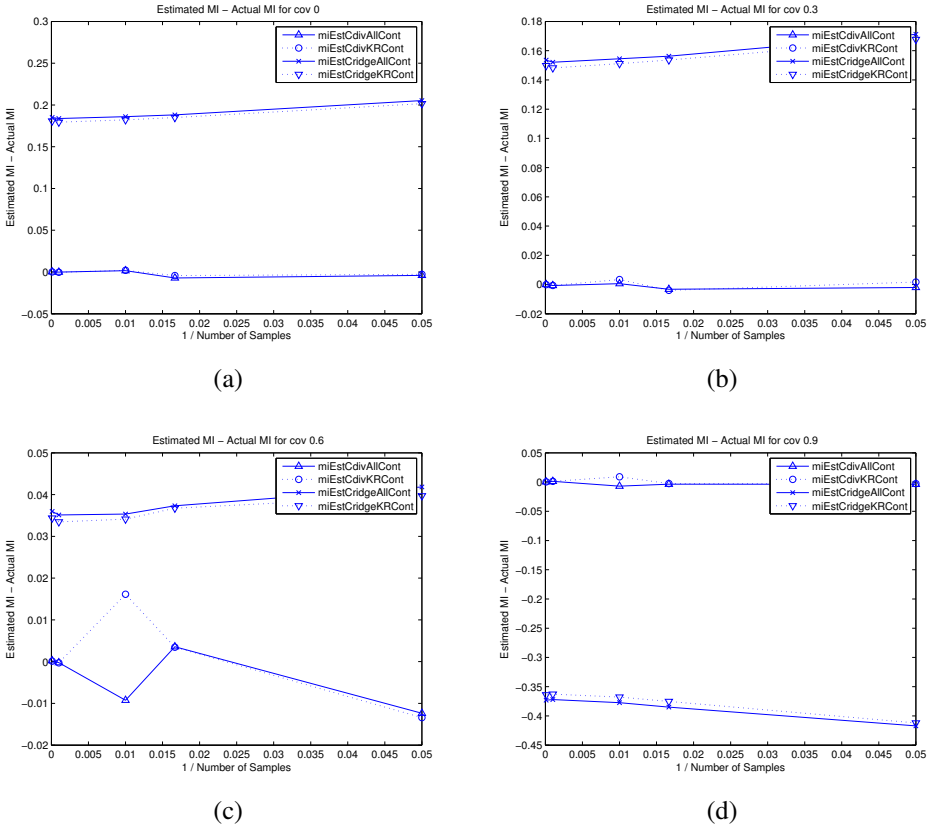


Figure 4.7: Systematic errors for combined MI estimators with zero mean and covariance 0 and 0.3.

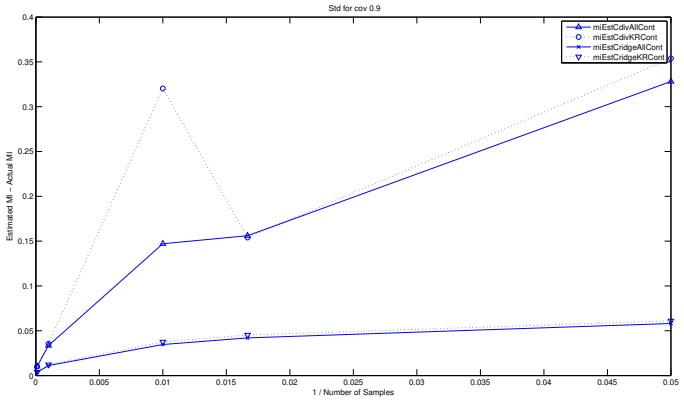
4.2 Possible Improvements

In this section, performance of mutual information estimators are tried to improve through combination and instance subset selection.

4.2.1 Combination of MI Estimators

One of the possible ways to improve estimator performance is to combine estimators.

KNN based MI estimators for K = 1, 2, 3, 4, 5, 10 and binning estimator are linearly combined. Combined estimator is tested on a different set of instances taken from a zero mean gaussian random variable distribution. Combination coefficients are determined using the least square solution for Ax = b and ridge regression separately. Ridge regression parameter lambda is selected with a search in the domain $\lambda = 10^x, x \in [-5,5]$.



(a)

Figure 4.8: Standard deviations for combined MI estimators.

Using the results obtained on gaussian random variables, linear combination of KNN based estimators are tried to be constructed. The following equation is solved using values for different number of samples.

$$\begin{bmatrix} 1 & kr_{1,0} & kr_{2,0} & kr_{3,0} & kr_{4,0} & kr_{5,0} & kr_{10,0} \\ 1 & kr_{1,0.3} & kr_{2,0.3} & kr_{3,0.3} & kr_{4,0.3} & kr_{5,0.3} & kr_{10,0.3} \\ 1 & kr_{1,0.6} & kr_{2,0.6} & kr_{3,0.6} & kr_{4,0.6} & kr_{5,0.6} & kr_{10,0.6} \\ 1 & kr_{1,0.9} & kr_{2,0.9} & kr_{3,0.9} & kr_{4,0.9} & kr_{5,0.9} & kr_{10,0.9} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{bmatrix} = \begin{bmatrix} mi_0 \\ mi_{0.3} \\ mi_{0.6} \\ mi_{0.9} \end{bmatrix} \quad \textbf{(4.3)}$$

In Equation 4.3, mi_r is the actual value of mutual information calculated from the equation for mutual information between two gaussian random variables with zero mean and variance r. $kr_{k,r}$ variables represents the estimated mutual information between two gaussian distributed random variables with zero mean and r covariance. a vector is the calculated coefficients for a certain number of samples.

With this curve fitting approach, coefficients are calculated for number of samples $N = \{20, 60, 100, 1000, 10000\}$. In a second approach, the second variable is discretized to make an analogy to the estimation of mutual information between a feature and the discrete class label. Same experiments are repeated with the addition of binning estimator.

Performance of the combined estimators are tested on microarray data and reported in Chapter 5.

4.2.2 Instance Subset Selection

Another possible way to improve estimator performance is through subset selection. Unlike traditional subset selection, here we select a subset of instances instead of features. This approach reminds the bagging [26] technique, it differs only by selecting the instances without replacement. Breiman et al. [26] showed that bagging increased the performance of decision trees in classification tasks. We believe, the MI estimator constructed by instance subset selection should be more robust to outliers.

Figures 4.9 and 4.10 show the results for two of the five experiments with 100 instances drawn from two gaussian random variables. Errorbars show the standard deviation for 300 subsets selected without replacement. Estimated MI value for the whole dataset is represented with a cross at $N = 100$. Results show that there is not a clear order in the estimated MI values for KNN based estimators, and the instance subset selection is not beneficial as the estimated mi value using the whole dataset is closer to the actual value than average estimated value for the subsets.

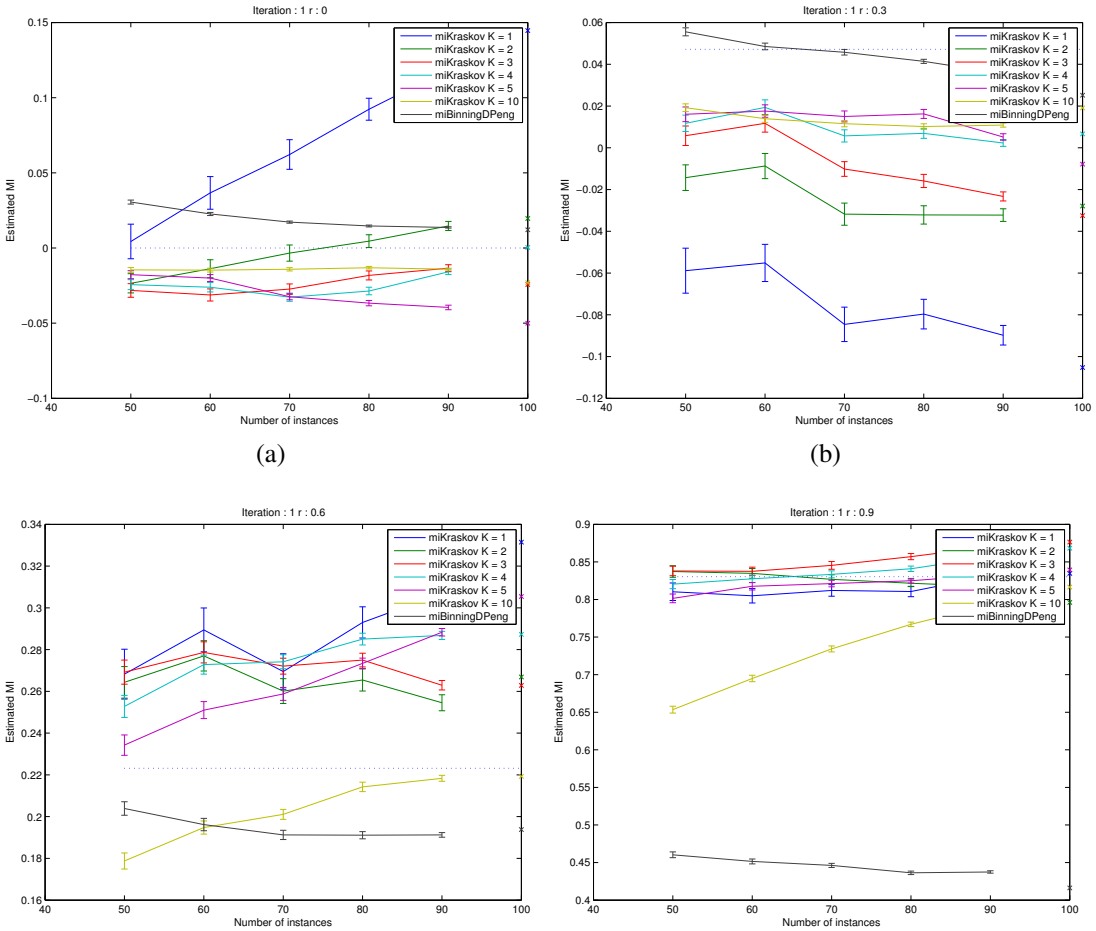


Figure 4.9: Subset selection - Experiment 1

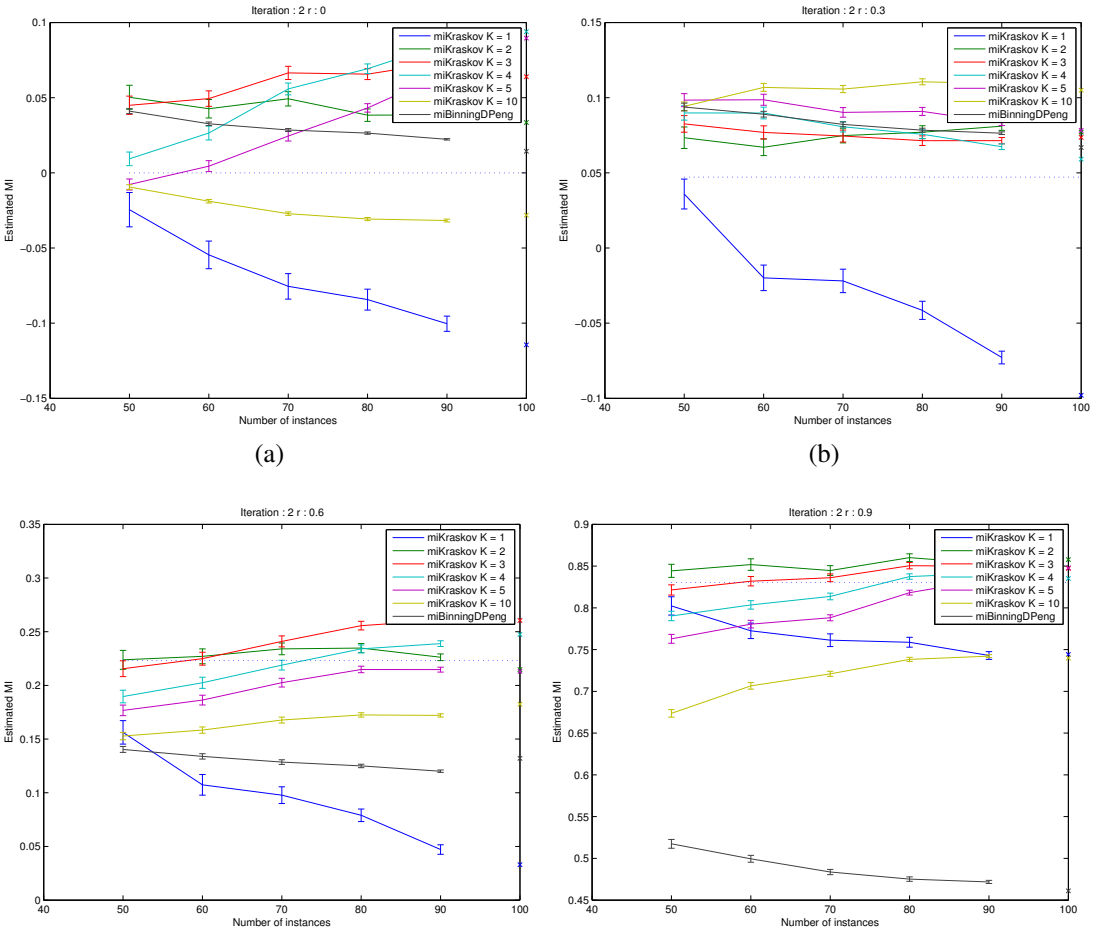


Figure 4.10: Subset selection - Experiment 2

5. FEATURE SELECTION IN MICROARRAY DATA

Feature selection techniques have been used in gene selection before. Ding et al. [27] have used mRMR on most widely used microarray datasets and compared their algorithm with feature selection based solely on mutual information as a baseline. Their work shows that MID and MIQ performs better than their continuous relatives FCD and FCQ. Some other deductions from their work can be summarized as

- In all cases, discretization performed better than the continuous variables.
- In all cases, MIQ method gives more informative genes than mutual information feature selection alone.
- For discrete data, MIQ features outperform MID features with mRMR

5.1 Microarray Data Feature Selection With Different MI Estimators

mRMR feature selection, by default, utilizes binning for mutual information estimation. As stated in Chapter 2 this method is improved by adaptive partitioning. Many other mutual information estimation methods have been developed recently.

In this chapter, experiments for mRMR with mutual information estimators other than binning are reported.

Statistics for datasets used in experiments and their reference works are shown in Table 5.1.

MI estimators' performances are evaluated mostly on Gaussians in this work. Because the evaluation is based on Gaussians, microarray data is checked to see if the features to be worked on are really Gaussians using one of the commonly used normality tests, Kolmogorov-Smirnov. According to the K-S test results displayed in Table 5.2, features that has a normal distribution are low in numbers.

Table 5.1: Dataset statistics and reference works

Name	Reference	# Instances	# Features	# Classes
colon	[28]	62	2000	2
nci	[29]	61	5245	8
prostate	[30]	102	6034	2

Considering MI estimation methods performance depends on the covariance between variables, covariance between features of microarray datasets are calculated to have an idea of which MI estimation method to use. Figure 5.1 shows the histograms for the covariances between features in microarray datasets. Most of the feature pairs have low covariance values. Binning, KDE and KNN based estimator with slightly higher k values are considered as best choices according to the mean square errors shown in Figure 4.5.

Table 5.2: Dataset statistics - number of features passing Kolmogorov-Smirnov normality test

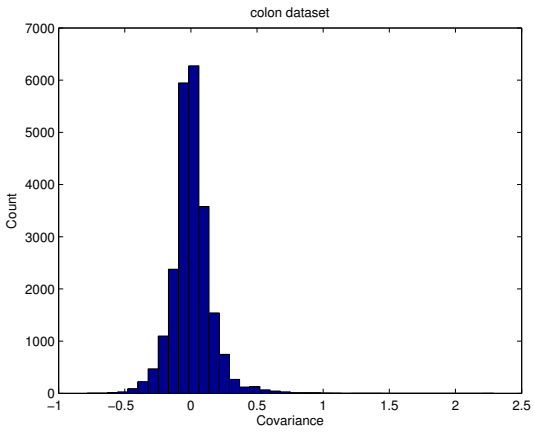
Name	Features	Instances	Normal Features	Normal And Relevant Features
colon	2001	62	8	1
nci	5245	61	78	0
prostate	6034	102	2998	0

5.1.1 Mutual Information Filter

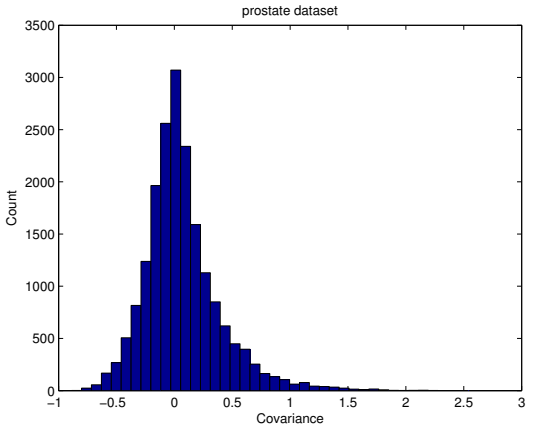
One of the simplest ways for employing mutual information for feature selection is sorting the features by their relevance (similarity to the class label) and using top features for classification.

In these experiments, features are sorted by their relevance values, and top 50 features are collected according to each mutual information estimator. Table 5.3 shows the total number of features selected by 31 different Mutual information estimators (Binning, KNN k = 1:15, KNN with discrete features and k = 1:15). For each selected feature, Leave one out cross-validation error (LOOCV) is calculated. Naive Bayes, KNN with K = 5 and LIBSVM are used as classifiers.

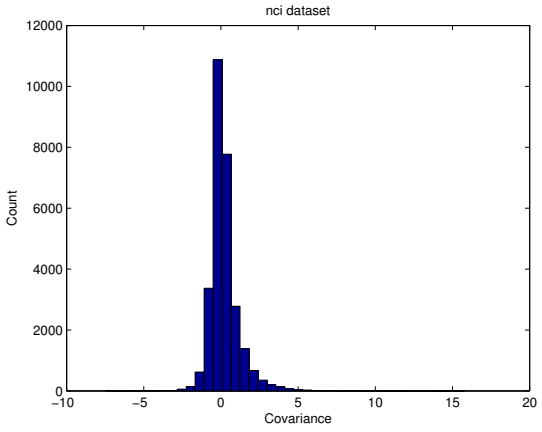
In order to determine the role of discretization in mutual information estimation, feature selection and classification phases are separated. Gene expression levels



(a)



(b)



(c)

Figure 5.1: Histograms of covariance values for features in microarray datasets.

are discretized into three bins as [25] before feature selection and classification. We report both results on the original and discretized data.

Because we use the whole dataset in feature selection phase, these results are known to be biased as [31] reported.

Table 5.3: Number of features selected

Dataset	Features Selected
colon	284
nci	418
prostate	306

Figure A.1 - A.4 shows the estimated MI values and LOOCV errors for the top features of Colon [28] and NCI [29] datasets.

It is seen that performance of the KNN based estimator is sensitive to the (change in) the parameter k. While small k values (2-4) are suggested by [10], some features having high error rates seems to be receiving high relevance values on Colon dataset. As far as we know, there is no systematic way to determine the optimal value for k.

Figure A.5 - A.6 shows the effect of discretization on the feature selection (Mutual information estimation) phase.

Mutual information filter method is a feature selection method based solely on mutual information as a metric. Features are sorted by their relevance (dependency / similarity to the class label) and mutual information is used for all dependency measurements.

While being simple, results in this section show that MI filter is effective. Tables 5.4, 5.5 and 5.6 shows the experiment results with 50 genes.

Table 5.4: MI filter results - Colon dataset

	NB		KNN		SVM	
Method	LOOCV Err	Features	LOOCV Err	Features	LOOCV Err	Features
Binning	5	8	5	31	5	3
3NN	**5**	**4**	5	27	5	17
6NN	5	49	**5**	**6**	4	6
9NN	5	15	6	4	6	4

Table 5.5: MI filter results - NCI dataset

	NB		KNN		SVM	
Method	LOOCV Err	Features	LOOCV Err	Features	LOOCV Err	Features
Binning	19	24	13	27	20	35
3NN	20	25	14	28	22	46
6NN	18	20	17	14	16	28
9NN	22	9	20	29	20	36

Table 5.6: MI filter results - Prostate dataset

	NB		KNN		SVM	
Method	LOOCV Err	Features	LOOCV Err	Features	LOOCV Err	Features
Binning	6	3	5	8	**6**	**3**
3NN	**5**	**3**	**5**	**4**	**6**	**3**
6NN	5	4	**5**	**4**	6	4
9NN	6	7	5	22	6	6

These results show that KNN based estimator performs better than binning in almost all experiments. But the question of how to select the optimum k value still holds.

5.1.2 MI Filter By Combining KNN and Binning Based Estimators

In order to improve the performance of MI Filtering, a combined estimator is designed by calculating the coefficients for KNN and binning based estimators using the approach in Section 4.2.1.

. With this estimator, relevance between features and the class label is estimated and top 50 features are taken for each dataset. For each dataset, coefficients from similar number of samples are used. Lowest LOOCV errors for a the given number of features are reported on Tables 5.7 , 5.8 and 5.9. The results show that linear combination with curve fitting approach does not increase the performance since base estimators are winners for all the experiments.

5.1.3 mRMR

Improvement on the performance of mutual information filter is encouraging. For this reason, these estimation methods are substituted for binning in the mRMR algorithm. Figures 5.2,5.3 and 5.4 shows results on Colon, NCI and Prostate datasets.

Table 5.7: MI filter results with combined MI estimators - Colon dataset

Method	NB		KNN		SVM	
	LOOCV Err	Feat	LOOCV Err	Feat	LOOCV Err	Feat
mrmr_comb_krbase_disc	10	11	14	31	8	13
mrmr_comb_krbinbase_disc	5	5	5	39	5	10
mrmr_comb_krbase_cont	6	5	6	4	6	4
mrmr_comb_krbinbase_cont	5	13	5	41	6	14
Binning	5	8	**6**	**3**	**5**	**3**
3NN	**5**	**4**	**6**	**3**	5	17
6NN	6	7	**5**	**6**	**4**	**6**
9NN	5	15	6	4	6	4

Table 5.8: MI filter results with combined MI estimators - Prostate dataset

Method	NB		KNN		SVM	
	LOOCV Err	Feat	LOOCV Err	Feat	LOOCV Err	Feat
mrmr_comb_krbase_disc	42	20	26	50	26	20
mrmr_comb_krbinbase_disc	16	13	10	17	8	15
mrmr_comb_krbase_cont	8	6	7	9	6	20
mrmr_comb_krbinbase_cont	12	31	8	28	8	31
Binning	6	3	5	8	**6**	**3**
3NN	**5**	**3**	**5**	**4**	**6**	**3**
6NN	5	4	**5**	**4**	6	4
9NN	6	7	6	6	6	6

Table 5.9: MI filter results with combined MI estimators - NCI dataset

Method	NB		KNN		SVM	
	LOOCV Err	Feat	LOOCV Err	Feat	LOOCV Err	Feat
mrmr_comb_krbase_disc	25	48	17	49	23	26
mrmr_comb_krbinbase_disc	20	30	15	36	21	47
mrmr_comb_krbase_cont	21	13	16	33	21	8
mrmr_comb_krbinbase_cont	20	29	14	23	22	6
Binning	20	15	18	15	24	17
3NN	24	19	19	11	23	11
6NN	18	20	17	14	19	13
9NN	22	9	22	7	21	8

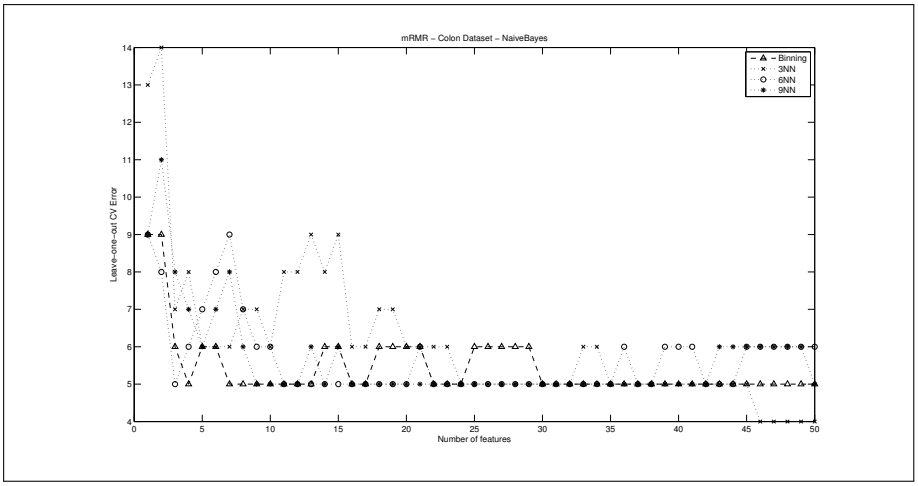


Figure 5.2: KNN based MI estimator is used with K = {3, 6, 9}. Results are in LOOCV errors.

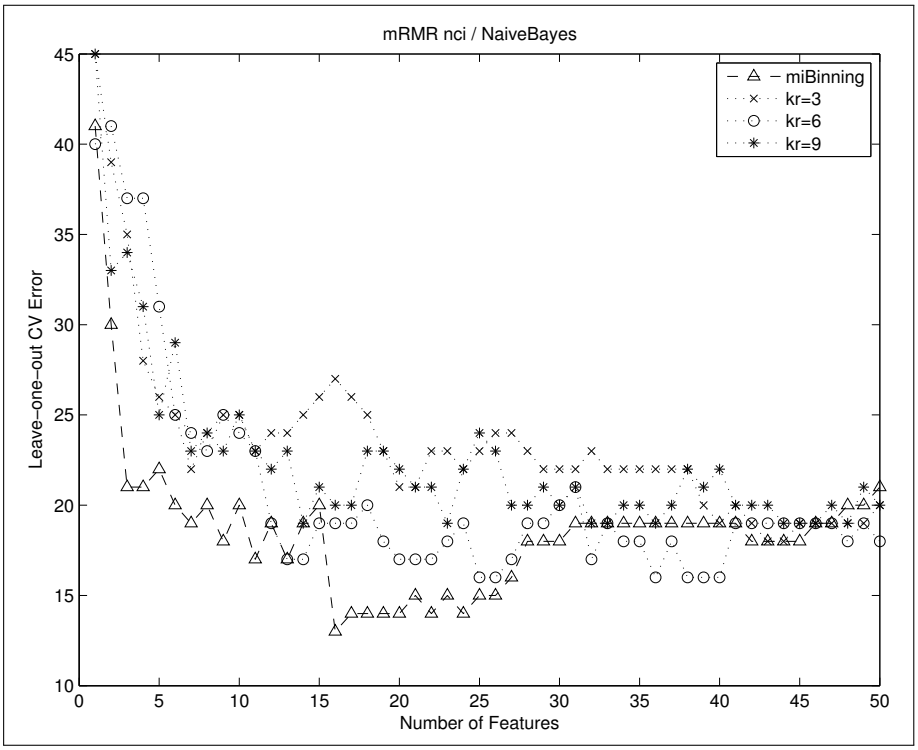


Figure 5.3: KNN based MI estimator is used with K = {3, 6, 9}. Results are in LOOCV errors.

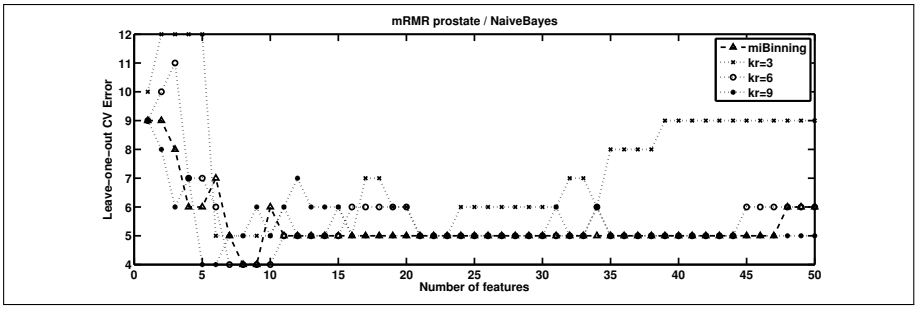


Figure 5.4: KNN based MI estimator is used with K = {3, 6, 9}. Results are in LOOCV errors.

Experiment results show that mRMR do not benefit from the more accurate estimation of mutual information the same way as the mutual information filter does.

Table 5.10: mRMR results - Colon dataset

	NB		KNN		SVM	
Method	LOOCV Err	Feat	LOOCV Err	Feat	LOOCV Err	Feat
Binning – MID	5	4	**4**	**3**	**5**	**13**
Binning – MIQ	4	8	5	6	5	8
Binning – HARM	7	2	6	2	**5**	**5**
3NN - MID	4	46	5	21	5	22
6NN - MID	5	3	5	9	5	22
9NN - MID	5	9	5	12	6	8
KDE - MID	5	7	5	19	5	19

Table 5.11: mRMR results - NCI dataset

	NB		KNN		SVM	
Method	LOOCV Err	Feat	LOOCV Err	Feat	LOOCV Err	Feat
Binning – MID	13	16	13	48	**16**	**15**
Binning – MIQ	8	24	8	27	10	27
Binning – HARM	35	3	33	7	32	3
3NN – MID	18	43	13	47	15	45
6NN – MID	16	25	15	12	**16**	**14**
9NN – MID	19	14	15	28	18	26
KDE – MID	16	12	16	23	24	6

Table 5.12: mRMR results - Prostate dataset

	NB		KNN		SVM	
Method	LOOCV Err	Feat	LOOCV Err	Feat	LOOCV Err	Feat
Binning – MID	4	8	4	12	4	12
Binning – MIQ	4	16	4	9	4	20
Binning – HARM	7	5	8	5	6	5
3NN – MID	5	6	4	16	6	7
6NN – MID	4	7	5	16	4	9
9NN – MID	4	5	4	13	5	5
KDE – MID	**4**	**3**	**4**	**4**	**4**	**3**

6. CONCLUSION AND FUTURE WORK

In this study, performance of recently developed mutual information estimation methods namely KNN based [10] and KDE based [15], when used in feature selection are compared with binning(histogram) based mutual information estimator.

The most basic feature selection method based on mutual information, MI filtering, benefits from the more accurate estimation of MI by these methods but mRMR [4] performance does not increase. This either comes from the fact that mRMR is robust to the mutual information estimator used, or the MI estimation between the class label and features is not completely compatible with our model based on gaussian distributions. Since discretization is shown to reduce the performance of MI estimators, first case gets stronger.

Subset selection and combination techniques are tried to boost the performance of estimators. Both ridge regression and least square curve fitting approaches failed to improve the performance of estimators on artificial data. One possible reason for that behaviour is the correlation between the combined estimators, especially KNN based estimators with different K values.

Taking this work one step further, one may try using other MI estimation methods for feature selection either one by one or in combination. Recent work on MI estimation includes MLMI [32], LSMI [33], Edgeworth [12].

Another possible extension is the change in the combination scheme for mRMR. While selecting features using mRMR, MI is used to estimate the relevance and redundancy values for (feature subset-class label) pairs and feature subsets. After estimating these values, Equations 3.4 and 3.5 are used to rank the features. As an alternative combination scheme to the difference and ratio, results with

harmonic mean are reported. Adaptive or weighted combination schemes [34] may be considered to improve mRMR performance.

REFERENCES

[1] **Xing, E.**, **Jordan, M. and Karp, R.**, 2001. Feature selection for high-dimensional genomic microarray data, MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-, Citeseer, pp. 601–608.

[2] **Kohavi, R. and John, G.**, 1997. Wrappers for feature subset selection, *Artificial intelligence*, **97(1-2)**, 273–324.

[3] **Liu, H. and Motoda, H.**, 1998. Feature selection for knowledge discovery and data mining, Springer.

[4] **Peng, H.**, **Long, F. and Ding, C.**, 2005. Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy, *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, **27(8)**.

[5] **Weaver, W. and Shannon, C.**, 1963. The mathematical theory of communication, University of Illinois Press Urbana.

[6] **Cover, T. and Thomas, J.**, 2006. Elements of information theory, John Wiley and sons.

[7] **Steuer, R.**, **Kurths, J.**, **Daub, C.**, **Weise, J. and Selbig, J.**, 2002. The mutual information: detecting and evaluating dependencies between variables, *BIOINFORMATICS-OXFORD-*, **18**, 231–240.

[8] **Silverman, B.**, 1998. Density estimation for statistics and data analysis, Chapman & Hall/CRC.

[9] **Darbellay, G. and Vajda, I.**, 1999. Estimation of the information by an adaptive partitioning of the observation space, *IEEE Transactions on Information Theory*, **45(4)**, 1315–1321.

[10] **Kraskov, A.**, **Stogbauer, H. and Grassberger, P.**, 2004. Estimating mutual information, *Physical Review E*, **69(6)**, 66138.

[11] **Fix, E. and Hodges, J.** J.(1951). Discriminatory analysis: nonparametricdiscrimination: consistency properties.

[12] **Hulle, M.**, 2005. Edgeworth approximation of multivariate differential entropy, *Neural computation*, **17(9)**, 1903–1910.

[13] **Khan, S.**, **Bandyopadhyay, S.**, **Ganguly, A.**, **Saigal, S.**, **Erickson III, D.**, **Protopopescu, V. and Ostrouchov, G.**, 2007. Relative performance of mutual information estimation methods for

quantifying the dependence among short and noisy data, *Physical Review E*, **76(2)**, 26209.

[14] **Rosenblatt, M.**, 1956. Remarks on some nonparametric estimates of a density function, *The Annals of Mathematical Statistics*, **27(3)**, 832–837.

[15] **Moon, Y.**, **Rajagopalan, B. and Lall, U.**, 1995. Estimation of mutual information using kernel density estimators, *Physical Review E*, **52(3)**, 2318–2321.

[16] **Guyon, I. and Elisseeff, A.**, 2003. An introduction to variable and feature selection, *The Journal of Machine Learning Research*, **3**, 1157–1182.

[17] **Krishnaiah, P.**, editor, 1982. Classification, pattern recognition and reduction of dimensionality, North-Holland, Amsterdam [u.a.], `http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+029343658&sourceid=fbw_bibsonomy`.

[18] **Torkkola, K.**, 2003. Feature extraction by non parametric mutual information maximization, *The Journal of Machine Learning Research*, **3**, 1438.

[19] **Kononenko, I.**, **Hong, S.**, **Kononenko, I.**, **Hong, S.J.**, **Kononenko, I. and Hong, S.J.**, 1997, Attribute Selection for Modeling.

[20] **Yu, L. and Liu, H.**, 2003. Feature selection for high-dimensional data: A fast correlation-based filter solution, **20(2)**, 856.

[21] **Saeys, Y.**, **Inza, I. and Larranaga, P.**, 2007. A review of feature selection techniques in bioinformatics, *Bioinformatics*, **23(19)**, 2507.

[22] **Hall, M.**, 2000. Correlation-based feature selection for discrete and numeric class machine learning, 359–366.

[23] **Koller, D. and Sahami, M.**, 1996. Toward optimal feature selection, MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-, Citeseer, pp. 284–292.

[24] **Yu, L. and Liu, H.**, 2004. Efficient feature selection via analysis of relevance and redundancy, *The Journal of Machine Learning Research*, **5**, 1205–1224.

[25] **Peng, Y.**, **Li, W. and Liu, Y.**, 2006. A hybrid approach for biomarker discovery from microarray gene expression data for cancer classification, *Cancer Informatics*, **2**, 301.

[26] **Breiman, L.**, 1996. Bagging predictors, *Machine learning*, **24(2)**, 123–140.

[27] **Ding, C. and Peng, H.**, 2005. Minimum redundancy feature selection from microarray gene expression data, *Journal of Bioinformatics and Computational Biology*, **3(2)**, 185–206.

[28] **Alon, U.**, **Barkai, N.**, **Notterman, D.**, **Gish, K.**, **Ybarra, S.**, **Mack, D. and Levine, A.**, 1999. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, *Proceedings of the National Academy of Sciences*, **96(12)**, 6745.

[29] **Ross, D.**, **Scherf, U.**, **Eisen, M.**, **Perou, C.**, **Rees, C.**, **Spellman, P.**, **Iyer, V.**, **Jeffrey, S.**, **Van de Rijn, M.**, **Waltham, M.** *et al.*, 2000. Systematic variation in gene expression patterns in human cancer cell lines, *Nature genetics*, **24(3)**, 227–235.

[30] **Singh, D.**, **Febbo, P.**, **Ross, K.**, **Jackson, D.**, **Manola, J.**, **Ladd, C.**, **Tamayo, P.**, **Renshaw, A.**, **D'Amico, A.**, **Richie, J.** *et al.*, 2002. Gene expression correlates of clinical prostate cancer behavior, *Cancer cell*, **1(2)**, 203–209.

[31] **Lai, C.**, **Reinders, M.**, **van't Veer, L.**, **Wessels, L.** *et al.*, 2006. A comparison of univariate and multivariate gene selection techniques for classification of cancer datasets, *BMC bioinformatics*, **7(1)**, 235.

[32] **Suzuki, T.**, **Sugiyama, M.**, **Sese, J. and Kanamori, T.**, 2008. Approximating mutual information by maximum likelihood density ratio estimation, **4**, 5–20.

[33] **Kanamori, T.**, **Hido, S. and Sugiyama, M.**, 2009. A least-squares approach to direct importance estimation, *The Journal of Machine Learning Research*, **10**, 1391–1445.

[34] **Gulgezen, G.**, **Cataltepe, Z. and Yu, L.**, 2009. Stable and Accurate Feature Selection, *Machine Learning and Knowledge Discovery in Databases*, 455–468.

APPENDICES

APPENDIX A: LOOCV Errors for Selected Features
APPENDIX B: Systematic Errors with Second Variable Discretized

APPENDIX A

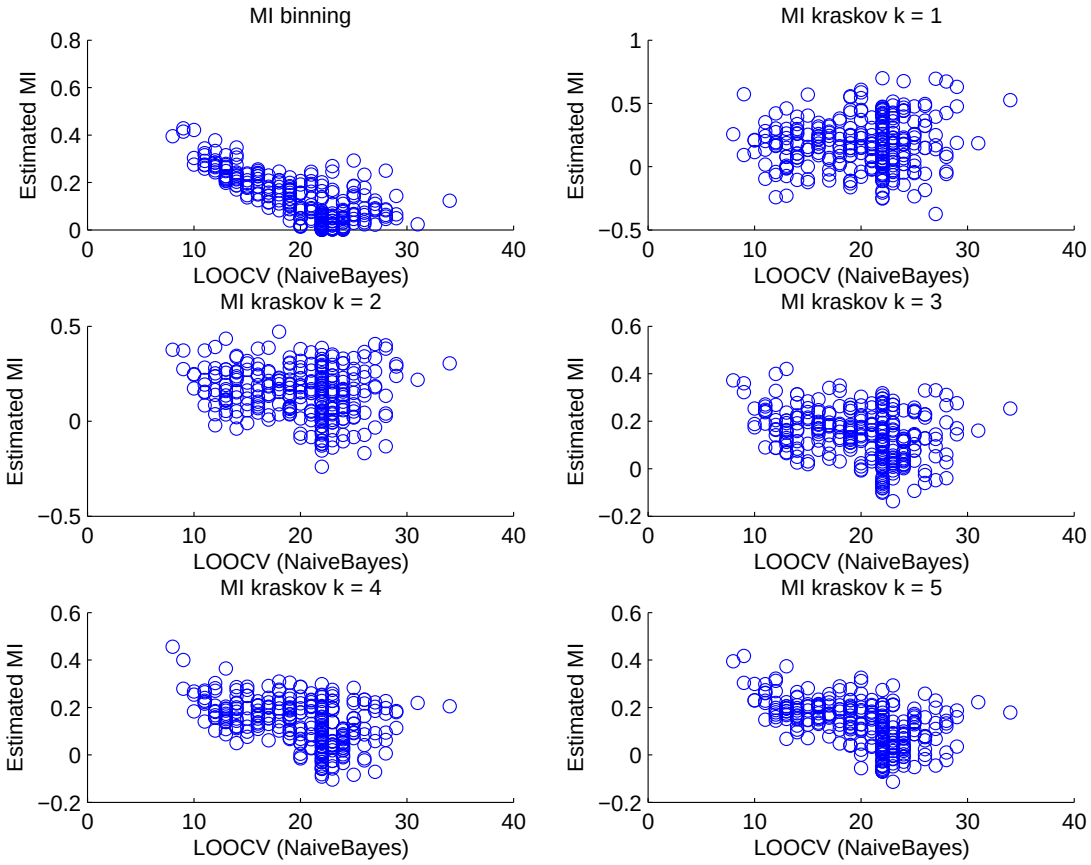


Figure A.1: Binning based estimator vs KNN based estimator (k = 1:5) - Colon Dataset

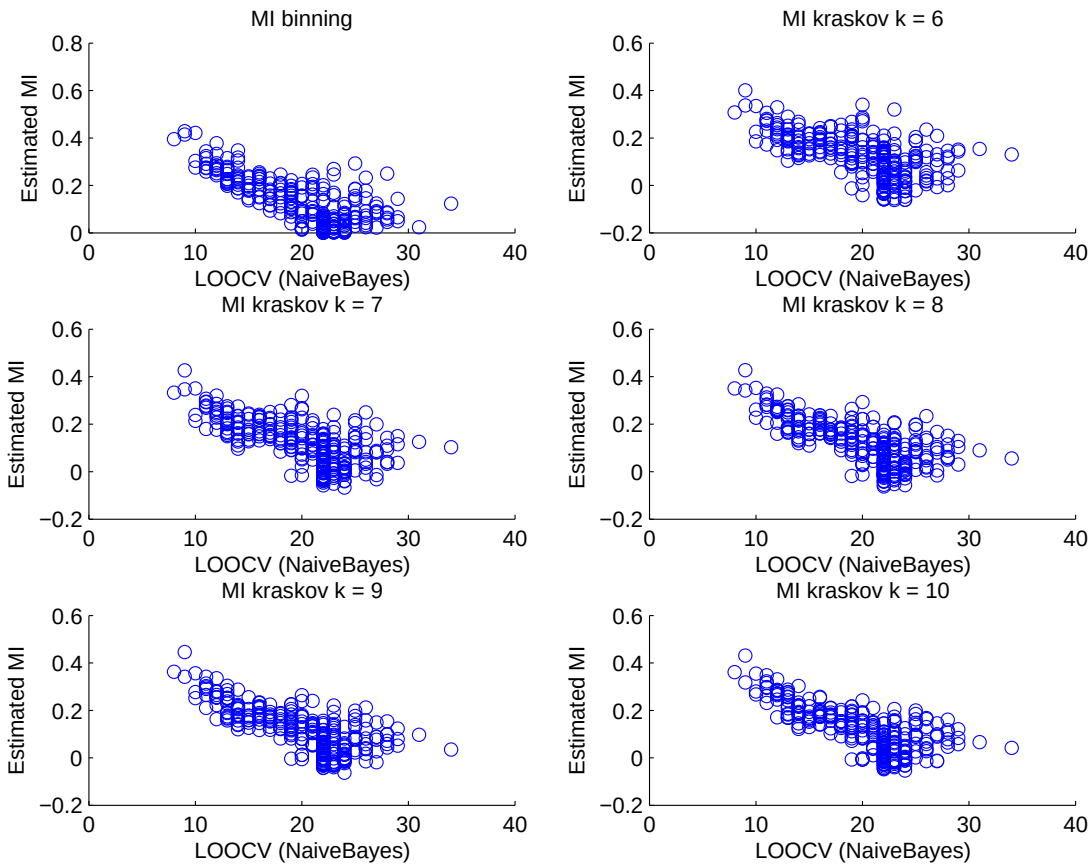


Figure A.2: Binning based estimator vs KNN based estimator (k = 6:10) - Colon Dataset

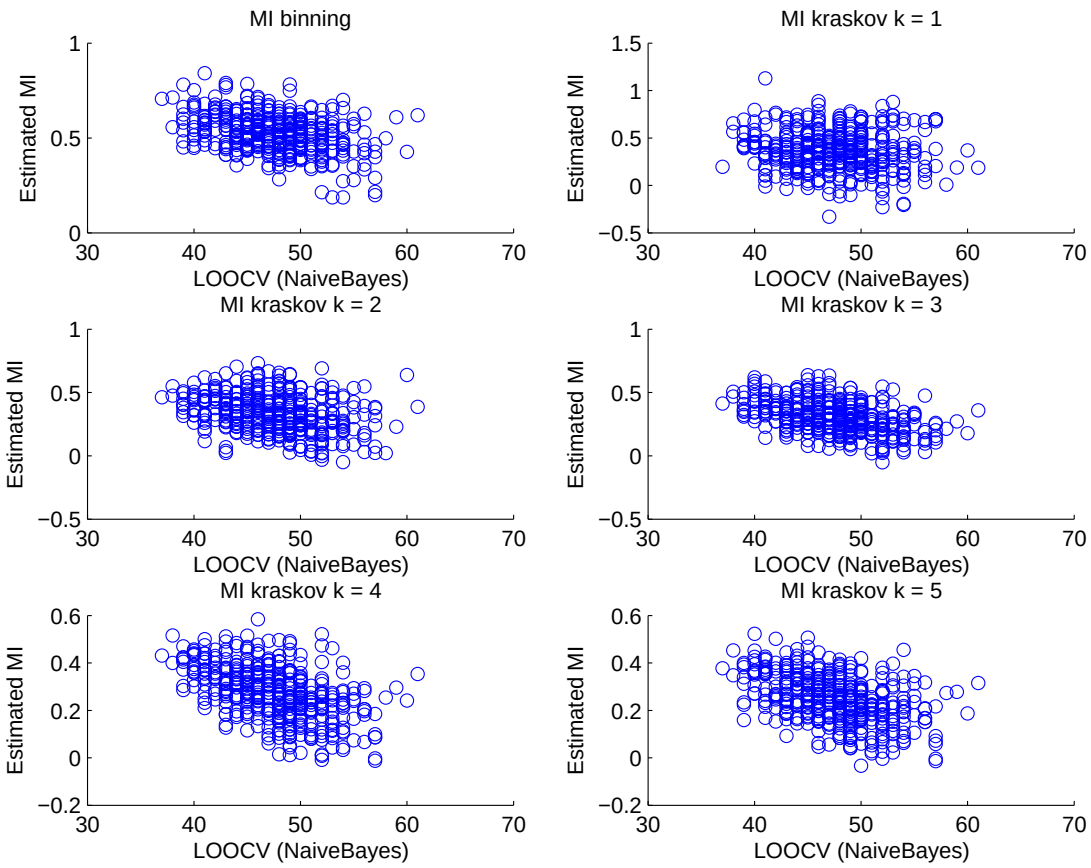


Figure A.3: Binning based estimator vs KNN based estimator (k = 1:5) - NCI Dataset

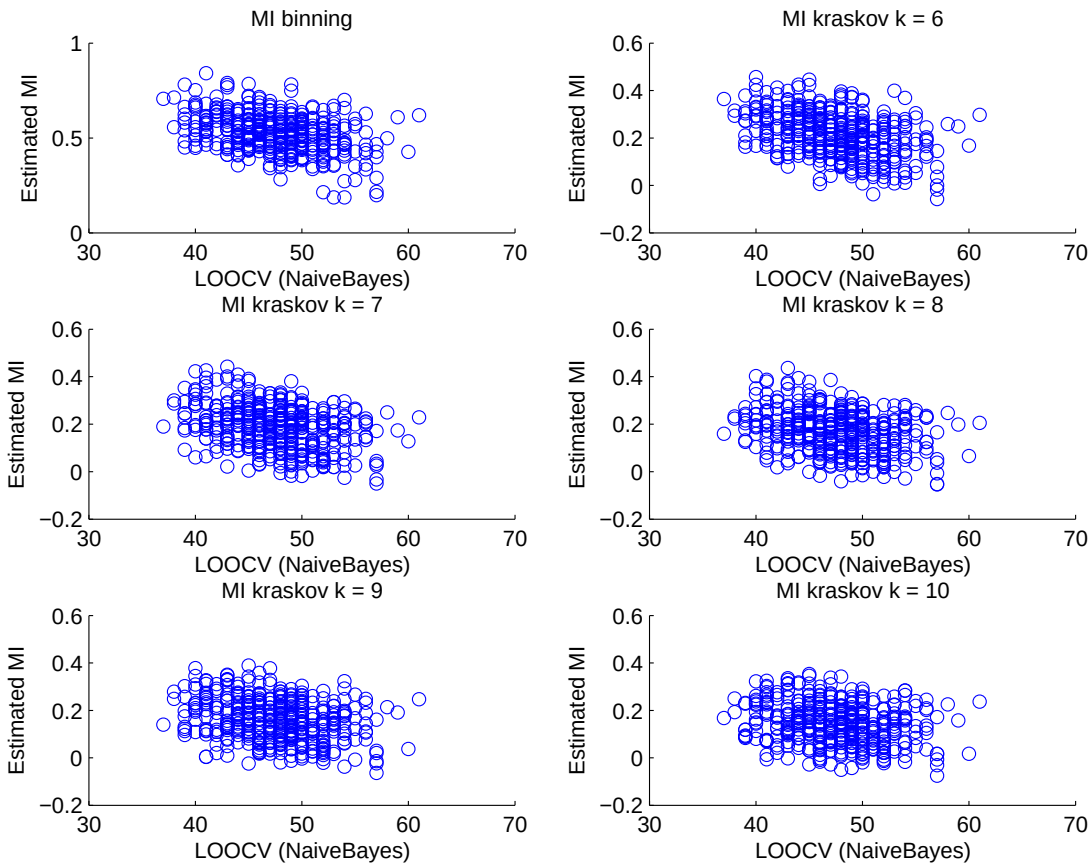


Figure A.4: Binning based estimator vs KNN based estimator (k = 6:10) - NCI Dataset

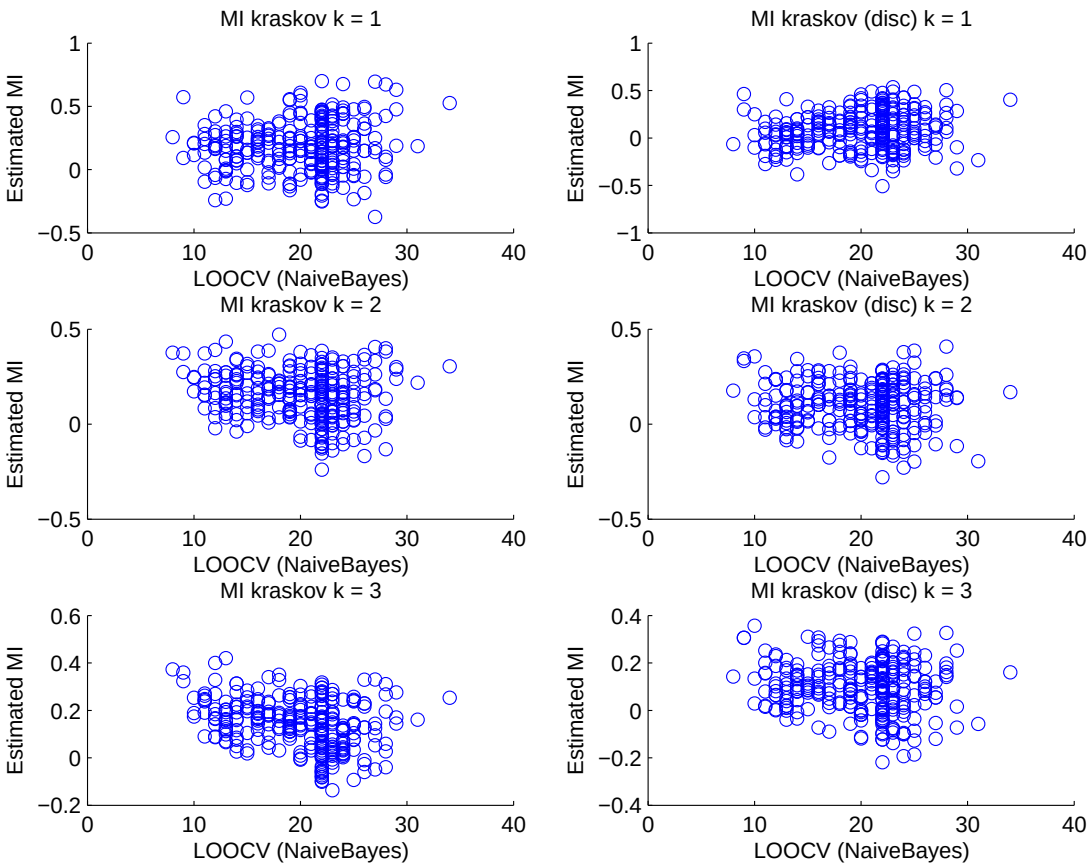


Figure A.5: KNN based estimator with continuous features vs discrete features (k = 1:3) - Colon Dataset

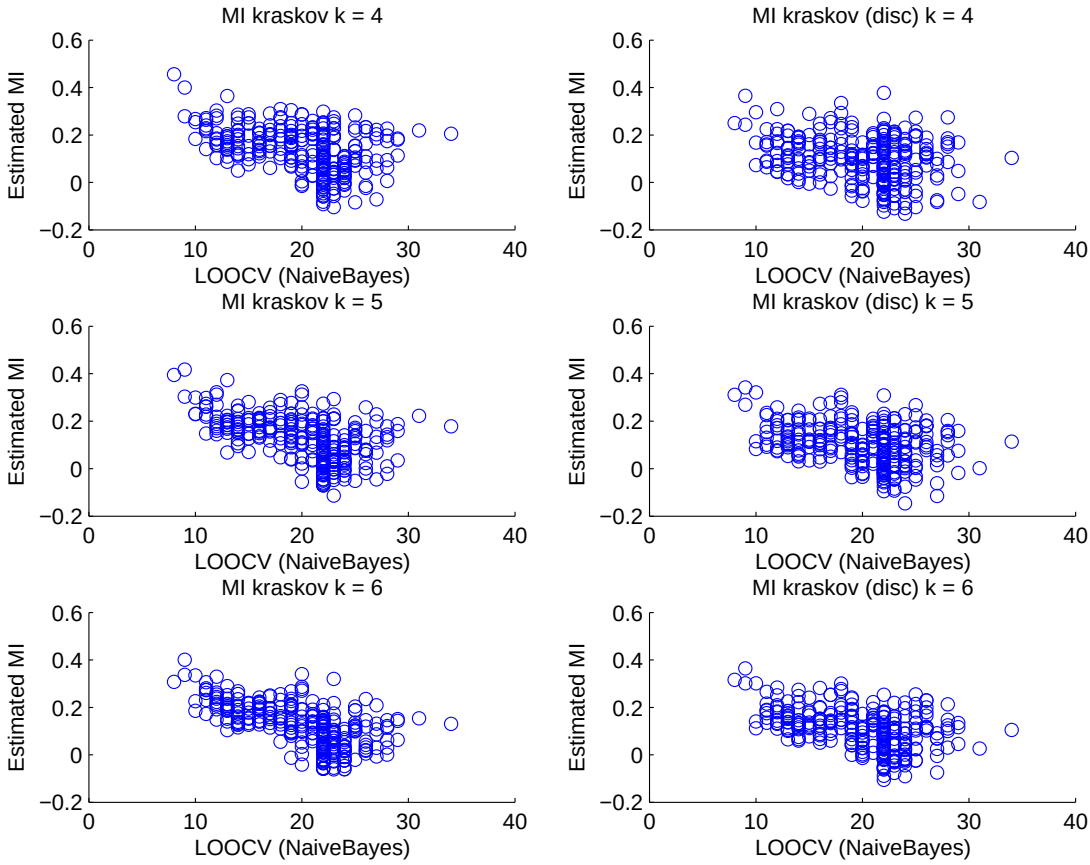
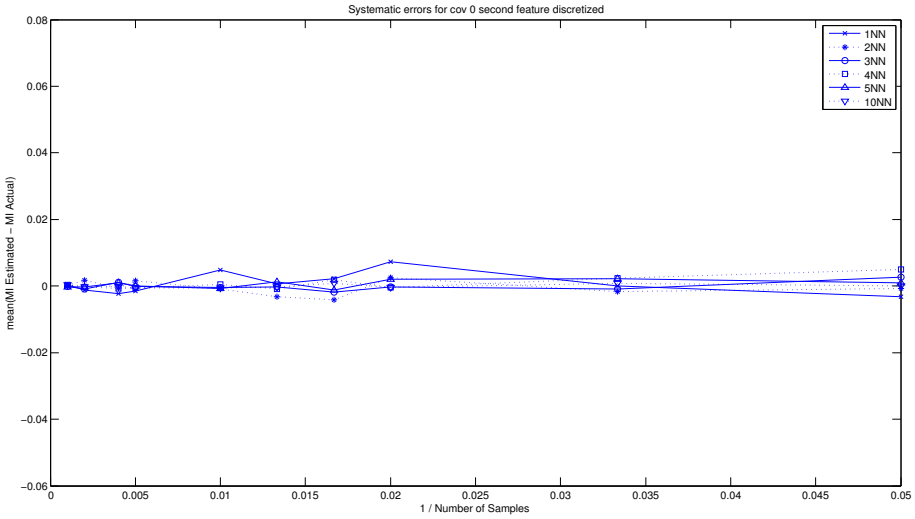
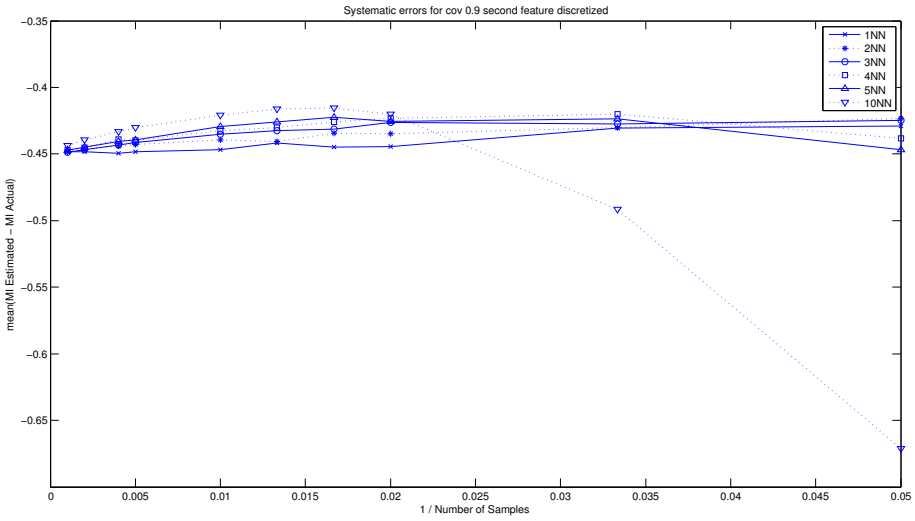


Figure A.6: KNN based estimator with continuous features vs discrete features (k = 4:6) - Colon Dataset

APPENDIX B

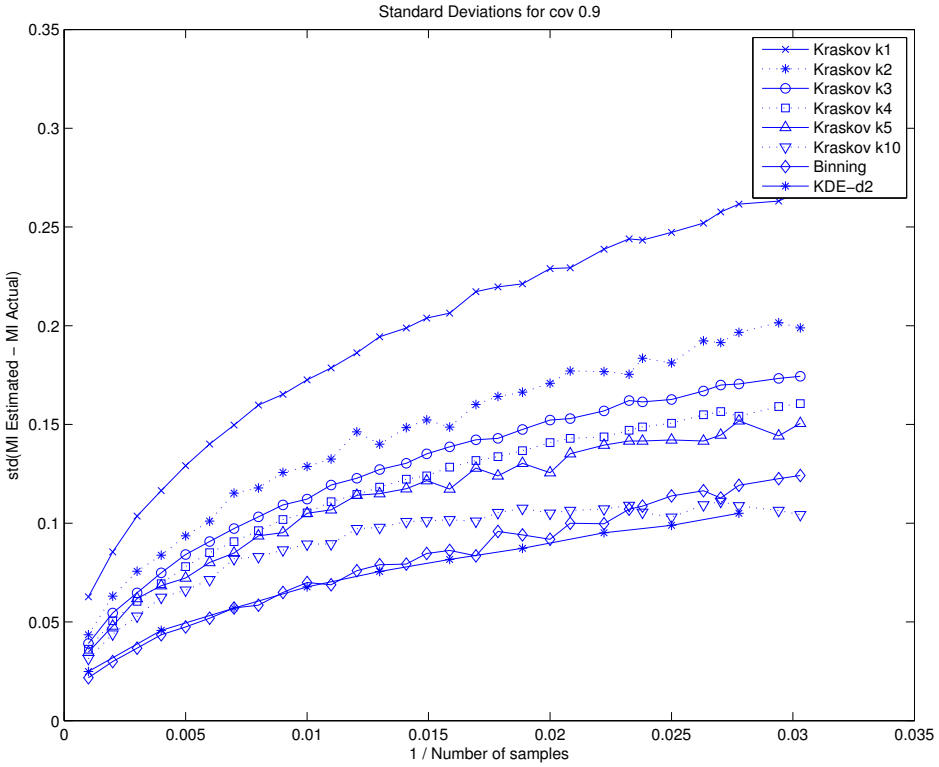


(a)

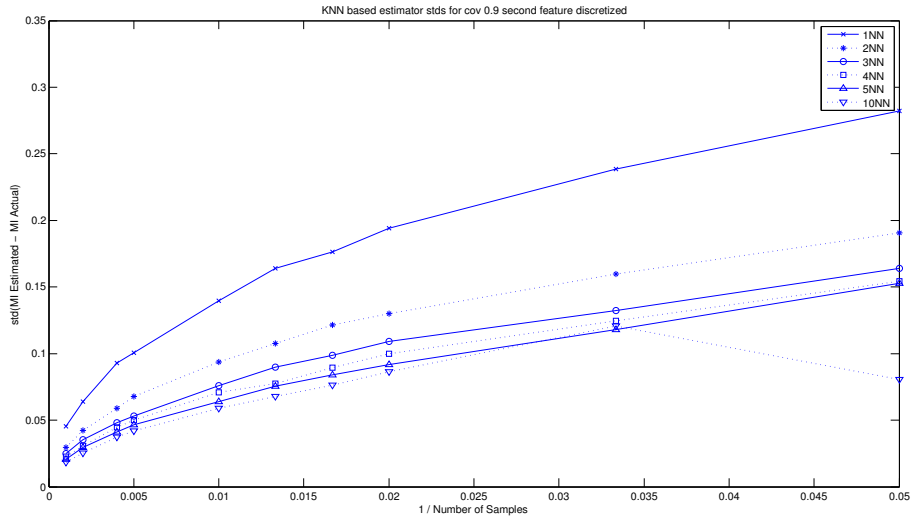


(b)

Figure B.1: Systematic error values for two gaussian (continuous-discretized) random variables with covariance 0 and 0.9.



(a)



(b)

Figure B.2: Standard deviations for two gaussian random variables with zero mean and covariance 0.9 with and without discretization.

CURRICULUM VITAE

Candidate's full name:	Ahmet Kenan KULE
Place and date of birth:	Afyon, 02 October 1985
Universities and Colleges attended	Istanbul Technical University
Publications:	