ISSN 1870-4069

Supervised Classification Based on Copula Functions

Ángela Paulina Pérez-Díaz[1], Rogelio Salinas-Gutiérrez[1], Angélica Hernández-Quintero[1], and Oscar Dalmau-Cedeño[2]

[1]Universidad Autónoma de Aguascalientes, Aguascalientes, México
alegna_287@hotmail.com, rsalinas@correo.uaa.mx,
angelica.hernandez.q@gmail.com
[2]Centro de Investigación en Matemáticas, Guanajuato, México
osdalmau@gmail.com

Abstract. This paper exposes the research being done about the incorporation of copula functions in supervised classification. It is shown, by means of pixel classification, the advantages that modeling dependencies provides to supervised classification and the benefits of doing it through copula functions which are not limited to linear dependencies. The experiments executed so far, show positive results by having improved the performance of the classifiers that do not have copulas incorporated.

Keywords: Pixel classification, Dependence structure, Likelihood function.

1 Introduction

Classification is commonly used nowadays in several sectors like industry and healthcare, among others. There are different kinds of classification and we are working with supervised classification, whose main objective is to group similar objects into different categories based on their features. The categories or classes and the features of the objects that are part of those classes, are known in advance due to some training data that provides the classifier with important information to later, identify the test objects which we want to classify, their category is unknown.

The use of copula functions has increased considerably in classification, thanks to the flexibility that they provide by being able to model different kinds of dependence structures. Copula theory, introduced by [1] to separate the effect of dependence from the effect of the marginal distributions in a joint distribution, allows us to model non-linear dependencies.

This work proposes to use copula functions for solving supervised classification problems. By using gaussian kernels and copula functions whose parameter of dependence is selected with the help of the maximum likelihood method, we intend to observe an improvement in the performance of classifiers.

The paper is organized in the following way: in Section 2, the methodology followed to resolve the research problem is exposed along with some definitions and theorems that help to understand the approach, in Section 3, we describe the main

pp. 9–18; rec. 2017-03-13; acc. 2017-05-25

contribution, Section 4 presents the results obtained so far and Section 5 contains the conclusions.

2 Research Methodology

2.1 Copula Functions

Copula functions' main objective in this research is to model dependencies. We take advantage of the association among features when classifying. The separation between marginal distributions and a dependence structure provides flexibility even when the marginals are not the same type.

Definition 1. *A copula function is a joint distribution function of standard uniform random variables. That is,*

$$C(u_1, u_2, \dots, u_d) = Pr[U_1 \leq u_1, U_2 \leq u_2, \dots, U_d \leq u_d],$$

where, $U_i \sim U(0,1) for\ i = 1,2, \dots d$.

Theorem 1(Sklar's theorem). *Let F be a d-dimensional distribution function with marginals* $F_1, F_2, \dots, F_d$*, then there exists a copula* C *such that for all x in* $\overline{\mathbb{R}}^d$,

$$F(x_1, x_2, \dots, x_d) = C\big(F_1(x_1), F_2(x_2), \dots, F_d(x_d)\big) ,$$

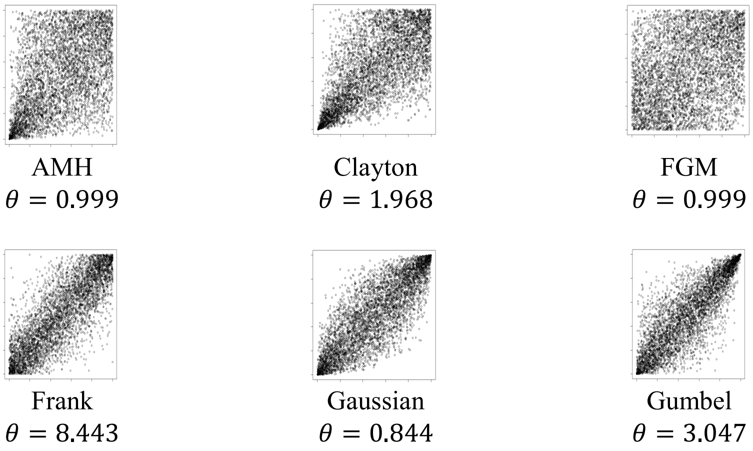
where $\overline{\mathbb{R}}$ *denotes the extended real line* $[-\infty, \infty]$. *If* $F_1(x_1), F_2(x_2), \dots, F_d(x_d)$ *are all continuous, then* C *is unique. Otherwise,* C *is uniquely determined on* $Ran(F_1) \times Ran(F_2) \times \dots \times Ran(F_d)$*, where* Ran *stands for the range.*

Due to Sklar's theorem, any d-dimensional density can be represented as:

$$f(x_1, x_2, \dots, x_d) = c\big(F_1(x_1), F_2(x_2), \dots, F_d(x_d)\big) \times \prod_{i=1}^{d} f_i(x_i) . \qquad (1)$$

where c is the density of the copula C, $F_i(x_i)$ is the marginal distribution function of random variable x_i, and $f_i(x_i)$ is the marginal density of variable x_i. Equation (1) shows that the dependence structure is modeled by the copula function.

In this paper, we work with the following two-dimensional parametric copula functions: Independent, Ali-Mikhail-Haq (AMH), Clayton, Farlie-Gumbel-Morgenstern (FGM), Frank, Gaussian and Gumbel. Fig. 1 shows the dependence structure for each copula and, as can be seen, the dependence structure is different for each copula. Some of these copulas are able to model positive and negative dependencies. The reader interested in copula theory is referred to [2]. The density functions of these copulas are shown in Table 1.



AMH
$\theta = 0.999$

Clayton
$\theta = 1.968$

FGM
$\theta = 0.999$

Frank
$\theta = 8.443$

Gaussian
$\theta = 0.844$

Gumbel
$\theta = 3.047$

Fig. 1. Dependencies structure with different θ values

The parameter θ, the dependence parameter, of a bivariate copula function can be estimated through the maximum likelihood method (ML). The one-dimensional log-likelihood function, see Equation 2, is maximized and we use its optimal value as parameter since it has better properties than other estimators as explained in [3]:

$$\ell(\theta; \{(u_{1i}, u_{2i})\}_{i=1}^{n}) = \sum_{i=1}^{n} \log\big(c(u_{1i}, u_{2i}; \theta)\big). \qquad (2)$$

2.2 Bayes Theorem

There are probabilistic and non probabilistic classifiers, the first ones use probabilistic distributions like bayesian networks, the multivariate normal or even the ones based on copula functions, the non probabilistic classifiers exclude the use of probability on them as neuronal networks or support vector machines.

As we have explained, we study a probabilistic classifier and to do so we have employed Bayes theorem [4], shown in Equation (3), which proposes the estimation of conditional probability of an event “A”, given “B” but we need to know in advance the conditional probability of “B” given “A”:

$$P(A|B) = \frac{P(B|A) \times P(\mathrm{A})}{P(B)}. \qquad (3)$$

That way, for our purposes, it is possible to know the probability that an object belongs to a group (A) given some features (B) because we know in advance the conditional probability of an object that has certain features (B) when it does belong to a class (A).

Table 1. Bivariate copula densities

Copula	**Description**
Independent	$c(u_1,u_2)=1$
AMH	$c(u_1,u_2;\theta)=\frac{1+\theta(u_1+u_2+u_1u_2-2)-\theta^2(u_1+u_2-u_1u_2-1)}{\left(1-\theta(1-u_1)(1-u_2)\right)^3};\theta\epsilon[-1,1)$
Clayton	$c(u_1,u_2;\ \theta)=(1+\theta)(u_1u_2)^{-\theta-1}\left(u_1^{-\theta}+u_2^{-\theta}-1\right)^{-2-\frac{1}{\theta}};\theta\epsilon[-1,\infty)\backslash\{0\}$
FGM	$c(u_1,u_2;\theta)=1+\theta(1-2u_1)(1-2u_2);\ \theta\epsilon[-1,1]$
Frank	$c(u_1,u_2;\theta)=\frac{-\theta\left(e^{-\theta}-1\right)e^{-\theta(u_1+u_2)}}{\left(\left(e^{-\theta u_1}-1\right)\left(e^{-\theta u_2}-1\right)+\left(e^{-\theta}-1\right)\right)^2};\ \theta\epsilon(-\infty,\infty)\backslash\{0\}$
Gaussian	$c(u_1,u_2;\theta)=(1-\theta^2)^{-\frac{1}{2}}\,exp\left(-\frac{\left(x_1^2+x_2^2-2\theta_{x_1x_2}\right)}{2(1-\theta^2)}+\frac{\left(x_1^2+x_2^2\right)}{2}\right);\ \theta\epsilon(-1,1)$ $where\ x_1=\Phi^{-1}(u_1)\ and\ x_2=\Phi^{-1}(u_2)$
Gumbel	$c(u_1,u_2;\theta)=\frac{exp\left(-(\tilde{u}_1^\theta+\tilde{u}_2^\theta)^{\frac{1}{\theta}}\right)}{u_1u_2}\frac{(\tilde{u}_1\tilde{u}_2)^{\theta-1}}{(\tilde{u}_1^\theta+\tilde{u}_2^\theta)^{2-\frac{1}{\theta}}}\left(\left(\tilde{u}_1^\theta+u_2^\theta\right)^{\frac{1}{\theta}}+\theta-1\right);\theta\epsilon[1,\infty)$ $where\ \tilde{u}_1=-\ln(u_1)\ and\ \tilde{u}_2=-\ln(u_2)$

Reasoned on Bayes theorem, there is the naive Bayes classifier [4], which, is based on applying Bayes' theorem, but assuming that each feature is independent of any other feature, meaning, it does not take into account the association that may exist between its features, an example considering three features (b_1, b_2, b_3) can be seen in Equation 4:

$$P\left(A\middle|(b_1,b_2,b_3)\right)=\frac{P(b_1|A)P(b_2|A)P(b_3|A)P(A)}{P(b_1,b_2,b_3)}. \tag{4}$$

However there are also the classifiers by dependency, as shown in Equation (5), that, unlike the previous ones, they consider the association between features of the objects, notice that for Equation 5 we also consider only three features:

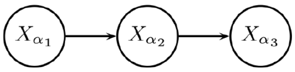
$$P\left(A\middle|(b_1,b_2,b_3)\right)=\frac{c(F_1(b_1),F_2(b_2),F_3(b_3)|A)\times\prod_{i=1}^{3}f_i(b_i|A)\times P(A)}{P(b_1,b_2,b_3)}. \tag{5}$$

3 Main Contribution

As mentioned before, copula functions can model dependencies among variables; the plan in this paper is to use a graphical model as a tool to identify the most important dependencies. The dependence structure is based on a chain model which, for a d-dimensional continuous random vector represents a probabilistic model with density:

$$f_{chain}(x) = f(x_{\alpha 1}) \prod_{i=2}^{d} f\left(x_{\alpha i} | x_{\alpha(i-1)}\right). \tag{6}$$

where $\alpha = (\alpha_1, \dots, \alpha_d)$ is a permutation of the integers between 1 and d. An example of a chain graphical model for a three dimensional vector is shown in Fig. 2.



$$f_{chain}(x) = f(x_{\alpha 1}) f(x_{\alpha 2} | x_{\alpha 1}) f(x_{\alpha 3} | x_{\alpha 2})$$

Fig. 2. Joint distribution over 3 variables represented by a chain graphical model

As presented in [5], the permutation α is unknown and the chain graphical model must be learnt from data. A way of choosing the permutation α is based on the Kullback-Leibler divergence(D_{KL}). This divergence is an information measure between two distributions. It is always non-negative for any two distributions, and zero if and only if the distributions are identical. Hence, the D_{KL} can be interpreted as a measure of the dissimilarity between two distributions. The goal is to choose a permutation α that minimizes the D_{KL} between the true distribution $f(x)$ of the data set and the distribution associated to a chain model, $f_{chain}(x)$, as shown in Equation (6).

The use of copula functions is becoming popular in machine learning as mentioned in [6]; the novel proposal is to employ them along with a graphical model and to not limit the investigation to only one copula function (gaussian copula) as done in previous works [7] and [8].

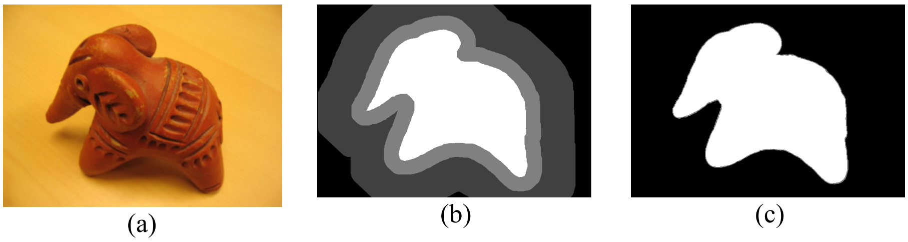
The main contribution in this research is the use of 6 different copulas to select the one that fits the most; this selection is done with a probabilistic model.

4 Achieved Results

During this research, we have been experimenting with pixel classification. From RGB images and having two established groups: background pixels and foreground pixels, we have used the features extracted from training data in order to get a conclusion on test data.

A color image can be represented in a 3-dimension matrix to keep data for red (R), green (G) and blue (B) colors, this is the information that is used as the attributes of each pixel to classify them.

Some classifiers have been computationally implemented; two of them using a normal distribution, from the density, some results have been obtained. We have worked with 50 images from Microsoft repository that can be found online [9]. The database provides 3 different images for each picture: the first one is the color image from where the RGB information is extracted, the second image in gray scale has the training data for both classes and test data, the third image is correctly classified and it is the image that has allowed us to evaluate the performance of the implemented classifiers, in Fig. 3, an example of the images is shown.



(a) (b) (c)

Fig. 3. (a) Color image. (b) Image with the training data for background (dark gray), foreground (white) and test data (gray). (c) Correctly classified image, background (black) and foreground (white)

Three measures, accuracy, sensitivity and specificity, are used to evaluate the performance of the classifiers. We used foreground as positive class and background as negative. As explained in Fig. 4, with the formulas used, the accuracy is calculated to reflect the percentage of correctly classified pixels, sensitivity shows the positive class pixels correctly classified and specificity, the negative class pixels that were correctly classified.

		Truth	
		Positive	Negative
Model	Positive	tp	fp
	Negative	fn	tn

$$accuracy = \frac{tp + tn}{tp + fp + fn + tn}$$

$$sensitivity = \frac{tp}{tp + fn}$$

$$specificity = \frac{tn}{tn + fp}$$

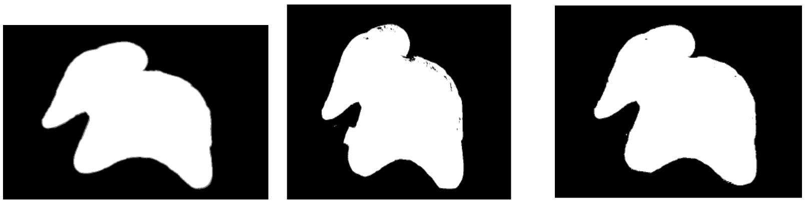
(a) (b)

Fig. 4. (a) Confusion matrix for binary classification, tp stands for true positive, fp is false positive, fn is false negative and tn is true negative. (b) Definition for accuracy, sensitivity and specificity used for this research

At first, we classified some images using normal distribution; the classification was made in two cases: without taking into account the association among the features and taking into account the dependencies or association among them.

It is shown in Fig. 5, the results obtained in the first experiment with a normal distribution and independence between features (b).

The evaluation measures of the classification with normal distribution and independence are: Accuracy - 86.67%, Sensitivity - 97.50% and Specificity - 77.80%.



(a) (b) (c)

Fig. 5. (a) Correctly classified image. (b) Image classified with normal density and independence between features. (c) Image classified with normal density and dependence between features

The same image was classified taking into account the dependency among the features, as can be seen in Fig. 5 (c) and the results were: Accuracy - 90.24%, Sensitivity - 98.85%, Specificity - 83.19%.

From Fig. 5, we have seen that the association among the attributes of an object can provide an improvement in supervised classification, to further, we experimented with 30 images from [9], the same classification that we used in the images above, with normal distribution. We noticed a trend and the next step was to try gaussian kernels instead of normal distribution and classify 50 images instead of 30.

One of the advantages of using gaussian kernel is the flexibility that they provide, we employed this flexible marginal distribution with independence at first, the results are shown in Fig. 6 (b).

The evaluation measures for the image shown in Fig. 6 (b) which was classified with gaussian kernel density and independence among features are: Accuracy - 86.01%, Sensitivity - 99.44%, Specificity - 75.02%.

The next step was to incorporate copula functions in classifiers with gaussian kernel distribution; the main objective is to model dependency among the attributes. In order to cover a considerable amount of models, we worked with six different copulas, the ones mentioned before (Table 1).

Through the extraction of the copula parameter using maximum likelihood, the classification was done for 50 images with all six copulas; in Fig. 6 (c) is the classification of the figure we have been showing, with Clayton copula. Another image from database that has been classified using copula functions is shown in Fig. 7.

The images shown as example here, have been classified using AMH, Clayton, Frank, FGM, Gaussian and Gumbel copulas, however we only included their classification using one copula, Clayton in the first example and Frank in the second one to exemplify the results. The use of images is helpful to notice the differences and improvements between one classification and another.

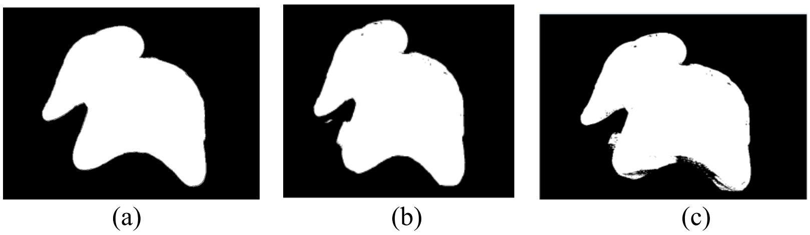


Fig. 6. (a) Correctly classified image. (b) Image classified with kernel density and by independence. (c) Image classified by Clayton copula

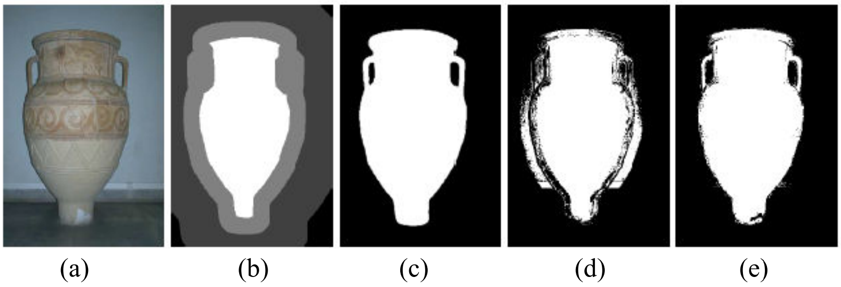


Fig. 7. (a) The color image. (b) Image with training and test data. (c) Correctly classified image. (d) Image classified with kernel density and by independence. (e) Image classified by Frank copula

Table 2. Evaluation measures represented in percentages

Copula	**Accuracy**		**Sensitivity**		**Specificity**	
Model	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
Independent	79.4	10.8	77.3	16.6	81.3	13.6
AMH	82.9	9.5	80.7	15.9	84.7	11.9
Clayton	86.0	8.5	81.6	16.4	89.5	9.2
FGM	80.9	9.8	78.9	16.5	82.5	13.2
Frank	87.7	7.1	87.1	12.2	88.1	9.0
Gaussian	86.0	10.6	87.1	11.0	85.0	18.6
Gumbel	86.7	8.2	87.0	10.9	86.5	13.2

However, from 50 classified images we summarized the measure values obtained by the classifiers when copulas were incorporated, in Table 2, we can visualize these results and observe the improvements. All copulas had a better behavior than the independent copula which represents no association among features; an ANOVA test for comparing the accuracy mean among the classifiers was performed in [5]. The test reports a statistical difference between Clayton, Frank, Gaussian and Gumbel copula functions with respect to the Independent copula(p-value < 0.05). The major difference in accuracy with respect to the independent copula is given by the Frank copula.

Accuracy, as can be seen in Fig. 4, shows the amount of pixels that were classified correctly.

5 Conclusions

In this paper, we showed some of the advantages of incorporating copula functions in supervised classification. By using a chain graphical model and modeling dependencies through copula functions we have shown the improvements that classification can have. Thanks to the graphical model we are able to identify the most important dependencies between the attributes of an object.

The results in pixel classification were satisfactory in accuracy, sensitivity and specificity having two classes and 3 attributes. Since the experiments performed so far have been with images, the classification is visually observable and is possible to easily notice the improvements.

From evaluation measures, we can notice that some copulas have had a better performance than others because they modeled the images from database in a better way. We proposed the use of 6 copulas from which we have obtained different results but all of them, compared with the independent copula, have improved the classification results.

As future work it has been planned to select copulas based on the maximum likelihood, meaning that, instead of using only one copula when classifying, use a combination of the “best” dependencies of all 6 copulas. We are also interested in experimenting with non parametric copulas and, from a statistics perspective, with a more random set of training and test data. The classifier based on copula functions must be proved in other datasets, compared with other classifiers and it is necessary to perform more experiments in order to have a better understanding on its advantages and limitations.

Acknowledgements. The authors acknowledge the financial support from the National Council of Science and Technology of México (CONACyT, grant number 258033) and from the Universidad Autónoma de Aguascalientes (project number PIM17-3). The student Ángela Paulina Pérez Díaz also acknowledges to the CONACyT for the financial support given through the scholarship number 628293.

References

1. Sklar, A.: Fonctions de répartition à *n* dimensions et leurs marges. Publications de l’Institut de Statistique de l’Université de Paris 8, 229–231 (1959)

2. Nelsen, R.B.: An Introduction to Copulas. Springer, Heidelberg (2006)Weiß, G.: Copula parameter estimation by maximum-likelihood and minimum distance estimators: a simulation study. Computational Statistics 26(1), 31–54 (2011)
3. Dougherty, G.: Pattern Recognition and Classification. Springer New York (2013)
4. Salinas-Gutiérrez, R., Hernández-Quintero, A., Dalmau-Cedeño, O., Pérez-Díaz, A.: Modeling Dependencies in Supervised Classification. Accepted for the MCPR (2017)
5. Carrera, D., Santana, R., Lozano, J.: Vine copula classifiers for the mind reading problem. Progress in Artificial Intelligence (2016)
6. Salinas-Gutiérrez, R., Hernández-Aguirre, A., Rivera-Meraz, M. J., Villa-Diharce, E. R.: Using Gaussian Copulas in Supervised Probabilistic Classification. In Soft Computing for Intelligent Control and Mobile Robotics. pp. 355–372. Springer-Verlag (2011)
7. Sen, S., Diawara, N., Iftekharuddin, K.: Statistical Pattern Recognition Using Gaussian Copula. Journal of Statistical Theory and Practice 9(4), 768–777 (2015)
8. Rother, C., Kolmogorov, V., Blake, A., Brown, M.: Image and video editing. http://research.microsoft.com/en-us/um/cambridge/projects/visionimagevideoediting/segmentation/grabcut.htm