Modeling Dependencies in Supervised Classification

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

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

Abstract. In this paper we show the advantage of modeling dependencies in supervised classification. The dependencies among variables in a multivariate data set can be linear or non linear. For this reason, it is important to consider flexible tools for modeling such dependencies. Copula functions are able to model different kinds of dependence structures. These copulas were studied and applied in classification of pixels. The results show that the performance of classifiers is improved when using copula functions.

Keywords: Copula function · Graphical model · Likelihood function

1 Introduction

Classification is an important task in Pattern Recognition. The goal in supervised classification is to assign a new object to a category based on its features [1]. Applications in this subject use training data in order to model the distribution of features for each class. In this work we propose the use of bivariate copula functions in order to design a probabilistic model. The copula function allows us to properly model dependencies, not necessarily linear dependencies, among the object features.

By using copula theory, a joint distribution can be built with a copula function and, possibly, several different marginal distributions. Copula theory has been used for modeling multivariate distributions in *unsupervised learning* problems [3,5,9,13] as well as in *supervised classification* [4,6,7,10,12,14,15]. For instance, in [4], a challenging classification problem is solved by means of copula functions and vine graphical models. However, all marginal distributions are modelled with gaussian distributions and the copula parameter is calculated by inverting Kendall's tau. In [10,15], simulated and real data are used to solve classification problems within the framework of copula theory. No graphical models are employed and marginal distributions are based on parametric models. In this paper, we employed flexible marginal distributions such as Gaussian kernels and

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2017, LNCS 10267, pp. 117–126, 2017.
DOI: 10.1007/978-3-319-59226-8_12

the copula parameter is estimated by using the maximum likelihood method. Moreover, the proposed classifier takes into account the most important dependencies by means of a graphical model. The reader interested in applications of copula theory in supervised classification is referred to [6,7,12,14].

The content of the paper is the following: Sect. 2 is a short introduction to copula functions, Sect. 3 presents a copula based probabilistic model for classification. Section 4 presents the experimental setting to classify an image database, and Sect. 5 summarizes the results.

2 Copula Functions

The copula theory was introduced by [11] to separate the effect of dependence from the effect of marginal distributions in a joint distribution. Although copula functions can model linear and nonlinear dependencies, they have rarely been used in supervised classification where nonlinear dependencies are common and need to be represented.

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

$$C(u_1, \ldots, u_d) = Pr[U_1 \leq u_1, \ldots, U_d \leq u_d],$$

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

Due to the Sklar's Theorem, *any* d-dimensional density f can be represented as

$$f(x_1, \ldots, x_d) = c(F_1(x_1), \ldots, F_d(x_d)) \cdot \prod_{i=1}^{d} f_i(x_i), \tag{1}$$

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. This expression separates any joint density function into the product of copula density and marginal densities. This is contrasted with the usual way to model multivariate distributions, which suffers from the restriction that the marginal distributions are usually of the same type. The separation between marginal distributions and a dependence structure explains the modeling flexibility given by copula functions.

In this paper we use two-dimensional parametric copula functions to model the dependence structure of random variables associated by a joint distribution function. The densities of these copula functions are shown in Table 1. We consider the Farlie-Gumbel-Morgenstern (FGM) copula function, elliptical copulas (Gaussian) and archimedean copulas (Independent, Ali-Mikhail-Haq (AMH), Clayton, Frank, Gumbel). These copula functions have been chosen because they cover a wide range of dependencies. For instance, the AMH, Clayton,

Table 1. Bivariate copula densities.

Copula	Description
Independent	$c(u_1, u_2) = 1$
AMH	$c(u_1, u_2; \theta) =$ $\dfrac{1 + \theta(u_1 + u_2 + u_1u_2 - 2) - \theta^2(u_1 + u_2 - u_1u_2 - 1)}{(1 - \theta(1 - u_1)(1 - u_2))^3}$
Clayton	$c(u_1, u_2; \theta) =$ $(1 + \theta)\,(u_1u_2)^{-\theta-1}\left(u_1^{-\theta} + u_2^{-\theta} - 1\right)^{-2-1/\theta}$
FGM	$c(u_1, u_2; \theta) = 1 + \theta(1 - 2u_1)(1 - 2u_2)$
Frank	$c(u_1, u_2; \theta) = \dfrac{-\theta(e^{-\theta} - 1)e^{-\theta(u_1+u_2)}}{((e^{-\theta u_1} - 1)(e^{-\theta u_2} - 1) + (e^{-\theta} - 1))^2}$
Gaussian	$c(u_1, u_2; \theta) = \left(1 - \theta^2\right)^{-1/2} \exp\left(-\dfrac{(x_1^2 + x_2^2 - 2\theta x_1x_2)}{2(1 - \theta^2)} + \dfrac{(x_1^2 + x_2^2)}{2}\right)$ where $x_1 = \Phi^{-1}(u_1)$ and $x_2 = \Phi^{-1}(u_2)$
Gumbel	$c(u_1, u_2; \theta) = \dfrac{C(u_1, u_2)}{u_1u_2} \dfrac{(\tilde{u}_1\tilde{u}_2)^{\theta-1}}{\left(\tilde{u}_1^\theta + \tilde{u}_2^\theta\right)^{2-1/\theta}} \left((\tilde{u}_1^\theta + \tilde{u}_2^\theta)^{1/\theta} + \theta - 1\right)$ where $\tilde{u}_1 = -\ln(u_1)$ and $\tilde{u}_2 = -\ln(u_2)$

FGM, Frank and Gaussian copula functions can model negative and positive dependences between the marginals. One exception is the Gumbel copula, which does not model negative dependence. The AMH and FGM copula functions are adequate for marginals with modest dependence. When dependence is strong between extremes values, the Clayton and Gumbel copula functions can model left and right tail association respectively. The Frank copula is appropriate for data that exhibit weak dependence between extreme values and strong dependence between centered values, while the Gaussian copula is adequate for data that exhibit weak dependence between centered values and strong dependence between extreme values. In general, when the Gaussian copula is used with standard Gaussian marginals, then the joint probabilistic model is equivalent to a multivariate normal distribution.

The dependence parameter θ of a bivariate copula function can be estimated using the maximum likelihood method (ML). To do so, the one-dimensional log-likelihood function

$$\ell\left(\theta; \{(u_{1i}, u_{2i})\}_{i=1}^n\right) = \sum_{i=1}^{n} \log\left(c(u_{1i}, u_{2i}; \theta)\right), \tag{2}$$

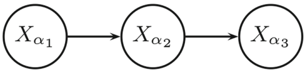
is maximized. Assuming the marginal distributions are known, the pseudo copula observations $\{(u_{1i}, u_{2i})\}_{i=1}^n$ in Eq. (2) are obtained by using the marginal distribution functions of variables X_1 and X_2. Once the maximum likelihood estimator of θ has been found, it is represented by the notation $\hat{\theta}$. It has been shown in [16] that the ML estimator $\hat{\theta}$ has better properties than other estimators.

3 The Probabilistic Model for Classification

The proposed classifier explicitly considers dependencies among variables. The dependence structure for the design of the probabilistic classifier is based on a chain graphical model. Such model, for a d-dimensional continuous random vector $\boldsymbol{X}$, represents a probabilistic model with the following density:

$$f_{\text{chain}}(\mathbf{x}) = f\left(x_{\alpha_1}\right) \prod_{i=2}^{d} f\left(x_{\alpha_i} | x_{\alpha_{(i-1)}}\right), \tag{3}$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d)$ is a permutation of the integers between 1 and d. Figure 1 shows an example of a chain graphical model for a three dimensional vector. Notice that a permutation could not be unique, in the sense that different permutations could yield the same density values in (3).



$$f_{\text{chain}}(\mathbf{x}) = f\left(x_{\alpha_1}\right) f\left(x_{\alpha_2} | x_{\alpha_1}\right) f\left(x_{\alpha_3} | x_{\alpha_2}\right)$$

Fig. 1. Joint distribution over three variables represented by a chain graphical model.

In practice the permutation $\boldsymbol{\alpha}$ is unknown and the chain graphical model must be learnt from data. A way of choosing the permutation $\boldsymbol{\alpha}$ 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 is zero if and only if the distributions are identical. Hence, the Kullback-Leibler divergence can be interpreted as a measure of the dissimilarity between two distributions. Then, the goal is to choose a permutation $\boldsymbol{\alpha}$ that minimizes the Kullback-Leibler divergence between the true distribution $f(\mathbf{x})$ of the data set and the distribution associated to a chain model, $f_{\text{chain}}(\mathbf{x})$. For instance, the Kullback-Leibler divergence between joint densities f and f_{chain} for a continuous random vector $\mathbf{X} = (X_1, X_2, X_3)$ is given by:

$$\begin{aligned} D_{KL}\left(f || f_{\text{chain}}\right) &= E_f\left[\log \frac{f(\mathbf{x})}{f_{\text{chain}}(\mathbf{x})}\right] \\ &= -H(\mathbf{X}) + \int \log\left(f\left(x_{\alpha_1}\right) f\left(x_{\alpha_2} | x_{\alpha_1}\right) f\left(x_{\alpha_3} | x_{\alpha_2}\right)\right) f dx. \end{aligned} \tag{4}$$

The first term in Eq. (4), $H(\mathbf{X})$, is the entropy of the joint distribution $f(\mathbf{x})$ and does not depend on the permutation $\boldsymbol{\alpha}$. By using copula theory and Eq. (1), the second term can be decomposed into the product of marginal distributions and bivariate copula functions.

$$\begin{aligned} D_{KL}\left(f||f_{\text{chain}}\right) = -H(\mathbf{X}) + \sum_{i=1}^{d} H(X_i) \\ - \int \log\left(c\left(u_{\alpha_1}, u_{\alpha_2}; \hat{\theta}_{\alpha_1,\alpha_2}\right)\right) f dx \\ - \int \log\left(c\left(u_{\alpha_2}, u_{\alpha_3}; \hat{\theta}_{\alpha_2,\alpha_3}\right)\right) f dx. \end{aligned} \tag{5}$$

The second term of Eq. (5), the sum of marginal entropies, also does not depend on the permutation $\boldsymbol{\alpha}$. Therefore, minimizing Eq. (5) is equivalent to maximize the sum of the last two terms. Once a sample of size n is obtained from the joint density f, the last two terms can be approximated by a Monte Carlo approach:

$$\int \log\left(c\left(u_{\alpha_1}, u_{\alpha_2}; \hat{\theta}_{\alpha_1,\alpha_2}\right)\right) f dx \approx \frac{1}{n}\sum_{i=1}^{n} \log\left(c\left(u_{1i}, u_{2i}; \hat{\theta}_{\alpha_1,\alpha_2}\right)\right). \tag{6}$$

Through Eq. (6), the D_{KL} is minimized by maximizing the sum of the log-likelihood for the copula parameters. It is worth to noting that the log-likelihood allows us to estimate the copula parameter and to select the appropriate permutation $\boldsymbol{\alpha}$. Finally, by means of copula theory, a chain graphical model for a three dimensional vector has the density

$$f_{\text{chain}}(\mathbf{x}) = f\left(x_{\alpha_1}\right) f\left(x_{\alpha_2}\right) f\left(x_{\alpha_3}\right) c\left(u_{\alpha_1}, u_{\alpha_2}\right) c\left(u_{\alpha_2}, u_{\alpha_3}\right) \tag{7}$$

3.1 The Probabilistic Classifier

Here, we present the incorporation of bivariate copula functions and a chain graphical model in order to design a probabilistic classifier.

The Bayes' theorem states the following:

$$P(K = k|\mathbf{X} = \mathbf{x}) = \frac{P(\mathbf{X} = \mathbf{x}|K = k) \times P(K = k)}{P(\mathbf{X} = \mathbf{x})}, \tag{8}$$

where $P(K = k|\mathbf{X} = \mathbf{x})$ is the posterior probability, $P(\mathbf{X} = \mathbf{x}|K = k)$ is the likelihood function, $P(K = k)$ is the prior probability and $P(\mathbf{X} = \mathbf{x})$ is the data probability.

Equation (8) has been used as a tool in supervised classification. A probabilistic classifier can be designed comparing the posterior probability that an object belongs to the class K given its features $\mathbf{X}$. The object is then assigned to the class with the highest posterior probability. For practical reasons, the data probability $P(\mathbf{X})$ does not need to be evaluated for comparing posterior probabilities. Furthermore, the prior probability $P(K)$ can be substituted by a uniform distribution if the user does not have an informative distribution.

The joint density in Eq. (7) can be used for modeling the likelihood function in Eq. (8). In this case, the Bayes' theorem can be written as:

$$P(K = k|\mathbf{x}) = \frac{\prod_{j=1}^{2} c(F_{\alpha_j}, F_{\alpha_{(j+1)}}|k; \hat{\theta}_{\alpha_j,\alpha_{(j+1)}}) \cdot \prod_{i=1}^{3} f_i(x_i|k) \cdot P(K = k)}{f(x_1, x_2, x_3)} \tag{9}$$

where F_i are the marginal distribution functions and f_i are the marginal densities for each feature. The function c is a bivariate copula density taken from Table 1. As can be seen in Eq. (9), each class determines a likelihood function.

4 Experiments

We use Eq. (9) and copula functions from Table 1 in order to classify pixels of 50 test images. Hence, we prove seven probabilistic classifiers. The image database was used in [2] and is available online [8]. This image database provides information about two classes: the foreground and the background. The training data and the test data are contained in the labelling-lasso files [8], whereas the correct classification is contained in the segmentation files. Figure 2 shows the description of one image from the database. Although the database is used for segmentation purposes, the aim of this work is to model dependencies in supervised classification. Only color features are considered for classifying pixels.

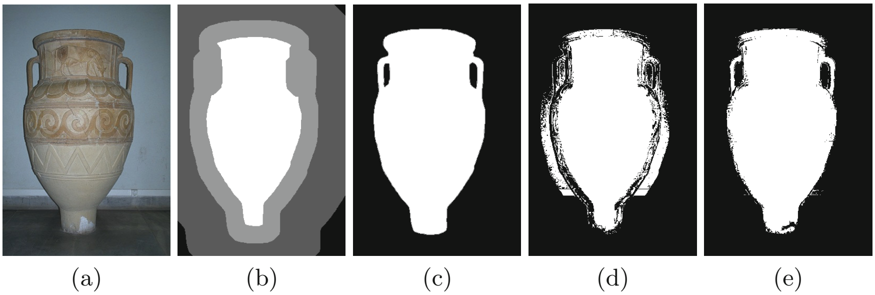


Fig. 2. (a) The color image. (b) The labelling-lasso image with the training data for background (dark gray), for foreground (white) and the test data (gray). (c) The correct classification with foreground (white) and background (black). (d) Classification made by independence. (e) Classification made by Frank Copula.

Three evaluation measures are used in this work: *accuracy*, *sensitivity* and *specificity*. These measures are described in Fig. 3. The sensitivity and specificity measures explain the percentage of well classified pixels for each class, foreground and background, respectively. We define the positive class as the foreground and the negative class as the background.

4.1 Numerical Results

In Table 2 we summarize the measure values reached by the classifiers according to the copula function used to model the dependencies.

To properly compare the performance of the probabilistic classifiers, we conducted an ANOVA test for comparing the accuracy mean among the classifiers. The test reports a statistical difference between Clayton, Frank, Gaussian and

		Truth	
		Positive	Negative
Model	Positive	tp	fp
	Negative	fn	tn

(a)

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

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

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

(b)

Fig. 3. (a) A confusion matrix for binary classification, where tp are true positive, fp false positive, fn false negative, and tn true negative counts. (b) Definitions of accuracy, sensitivity and specificity used in this work.

Table 2. Descriptive results for all evaluation measures. The results are presented in percentages.

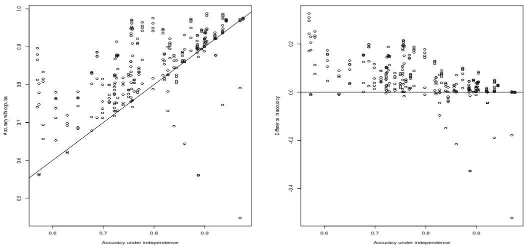
Copula	Accuracy		Sensitivity		Specificity	
Model	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.
Independent	79.4	10.8	77.3	16.6	81.3	13.6
AMH	82.9	9.5	80.7	15.9	84.7	11.9
Clayton	86.0	8.5	81.6	16.4	89.5	9.2
FGM	80.9	9.8	78.9	16.5	82.5	13.2
Frank	87.7	7.1	87.1	12.2	88.1	9.0
Gaussian	86.0	10.6	87.1	11.0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

Gumbel copula functions with respect to the Independent copula (p-value $<$ 0.05). The major difference of accuracy with respect to the independent copula is given by the Frank copula.

4.2 Discussion

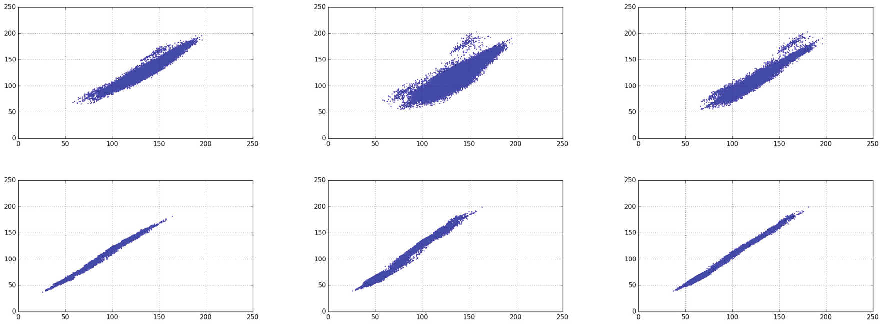
According to Table 2, the classifier based on the Frank copula shows the best behavior for accuracy. For sensitivity, Frank and Gaussian copulas provide the best results. The best mean specificity is reached by the classifier based on the Clayton copula.

As can be seen, the average performance of a classifier is improved by the incorporation of the copula functions. The lowest average performance corresponds to the classifier that uses the independence assumption. Figure 4 shows how the accuracy is increased when dependencies are taken into account by the probabilistic classifier. The line of Fig. 4(a) represents the identity function, so the points above this line correspond to a better accuracy than the



(a) (b)

Fig. 4. (a) Scatterplot of the accuracy values between classifier based on independence assumption (horizontal axis) and classifiers based on copula functions (vertical axis). (b) The gain of accuracy by using copula functions.



(a) (b) (c)

Fig. 5. The first line shows the scatterplot among (a) red and green, (b) red and blue, and (c) green and blue colors for the foreground class. The second line similarly shows the scatterplots for the background class. (Color figure online)

accuracy achieved by the classifier based on the independent copula. To get a better insight, Fig. 4(b) shows the difference in accuracy between using copula functions respect to the naive classifier (independent copula).

Table 2 also shows information about the standard deviations for each evaluation measure. For accuracy, the standard deviation indicates that using a Frank copula in pixel classification is more consistent than the other classifiers.

Figure 2 shows the results of one of the 50 images mentioned before, once we worked on them. In (d), we can see the resultant image when it is classified by independence, (e) shows the same image classified by Frank copula. It is possible to visually perceive the improvement that the use of Frank copula provides to the classifier. For this image, the color data for each class is shown in Fig. 5. In this case, it can be seen that the dependence structure does not correspond to the dependence structure of a bivariate Gaussian distribution. According to the numerical results, the copula Frank is the best model for this kind of dependence.

5 Conclusions

In this paper we have compared the performance of several copula based probabilistic classifiers. The results show that the dependence among features provides important information for supervised classifying. For the images used in this work, the Gumbel copula performs very well in most of the cases. One advantage of using a chain graphical model consists in detecting the most important dependencies among variables. This can be valuable for different applications where associations among variables gives additional knowledge of the problem. Though accuracy is increased by the classifiers based on copula functions, the selection of the copula function has relevant consequences for the performance of the classifier. For instance, in Fig. 4, a few classifiers do not improve the performance achieved by the classifier based on the independent copula. It suggests more experiments are needed in order to select the adequate copula function for a given problem. Moreover, as future work, the classifier based on copula functions must be proved in other datasets and compared with other classifiers in order to achieve a better insight of its benefits and limitations.

Acknowledgments. 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 also acknowledges to the CONACYT for the financial support given through the scholarship number 628293.

References

1. Bishop, C.: Pattern Recognition and Machine Learning. Information Science and Statistics. Springer, New York (2007)
2. Blake, A., Rother, C., Brown, M., Perez, P., Torr, P.: Interactive Image Segmentation Using an Adaptive GMMRF Model. In: Pajdla, T., Matas, J. (eds.) ECCV 2004. LNCS, vol. 3021, pp. 428–441. Springer, Heidelberg (2004). doi:10.1007/978-3-540-24670-1_33
3. Brunel, N., Pieczynski, W., Derrode, S.: Copulas in vectorial hidden Markov chains for multicomponent image segmentation. In: Proceedings of the 2005 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2005), pp. 717–720 (2005). doi:10.1109/ICASSP.2005.1415505
4. Carrera, D., Santana, R., Lozano, J.: Vine copula classifiers for the mind reading problem. Prog. Artif. Intell. (2016). doi:10.1007/s13748-016-0095-z
5. Mercier, G., Bouchemakh, L., Smara, Y.: The use of multidimensional Copulas to describe amplitude distribution of polarimetric SAR Data. In: IGARSS 2007 (2007). doi:10.1109/IGARSS.2007.4423284
6. Ouhbi, N., Voivret, C., Perrin, G., Roux, J.: Real grain shape analysis: characterization and generation of representative virtual grains. application to railway ballast. In: Oñate, E., Bischoff, M., Owen, D., Wriggers, P., Zohdi, T. (eds.) Proceedings of the IV International Conference on Particle-based Methods Fundamentals and Applications (2015)
7. Resti, Y.: Dependence in classification of aluminium waste. J. Phys. Conf. Ser. **622**(012052), 1–6 (2015). doi:10.1088/1742-6596/622/1/012052

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
9. Sakji-Nsibi, S., Benazza-Benyahia, A.: Multivariate indexing of multichannel images based on the copula theory. In: IPTA08 (2008)
10. Sen, S., Diawara, N., Iftekharuddin, K.: Statistical pattern recognition using Gaussian Copula. J. Stat. Theor. Pract. **9**(4), 768–777 (2015). doi:10.1080/15598608.2015.1008607
11. 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)
12. Slechan, L., Górecki, J.: On the accuracy of Copula-based Bayesian classifiers: an experimental comparison with neural networks. In: Núñez, M., Nguyen, N.T., Camacho, D., Trawiński, B. (eds.) ICCCI 2015. LNCS, vol. 9329, pp. 485–493. Springer, Cham (2015). doi:10.1007/978-3-319-24069-5_46
13. Stitou, Y., Lasmar, N., Berthoumieu, Y.: Copulas based multivariate gamma modeling for texture classification. In: Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009), pp. 1045–1048. IEEE Computer Society, Washington, DC (2009). doi:10.1109/ICASSP.2009.4959766
14. Voisin, A., Krylov, V., Moser, G., Serpico, S., Zerubia, J.: Classification of very high resolution SAR images of urban areas using copulas and texture in a hierarchical Markov random field model. IEEE Geosci. Remote Sens. Lett. **10**(1), 96–100 (2013). doi:10.1109/LGRS.2012.2193869
15. Ščavnický, M.: A study of applying copulas in data mining. Master's thesis, Charles University in Prague, Prague (2013)
16. Weiß, G.: Copula parameter estimation by maximum-likelihood and minimum-distance estimators: a simulation study. Comput. Stat. **26**(1), 31–54 (2011). doi:10.1007/s00180-010-0203-7