Predicting positions using Independence Graphs

Volker Strobel

June 19, 2016

Abstract

In this report, the dependence structure between image features and x, y-coordinates is analyzed using linear regression, graphical Gaussian models, and vine copulas. The dependencies between average color per channel and image location are used to build a predictive model for given image features. The predictive power of the used models are compared using the mean absolute error of the predictions on a hold-out test set. The associated uncertainty in the predictions are studied and indications are given, how the proposed model can be used in a real-world scenario.

1 Introduction & Problem Statement

Computer vision tasks—such as object detection, image restoration, or *localization*—often need a high-dimensional reprentation of a feature domain. This task can be fulfilled using probabilistic graphical models and has been used in a variety of applications.

Combining copula modelling with machine learning techniques was longtime neglected—and to date, many problems are studied in the bivariate case only. However, recently, more and more research is focusing on combining machine learning and regression techniques to tackle high-dimensional regression problems (Cooke, Joe, and Chang 2015; Elidan 2013).

In this report, we consider the following computer vision localization problem:

> Based on a given patch of an image, we would like to predict, where the patch was taken in a larger image—the map image.

While existing approaches extract keypoints of the current patch and the map image, followed by finding a homography between both keypoint sets, these approaches are usually computationally complex and do not allow for in-depth analyses of the problem (e.g. how to change the map image to improve the quality of the algorithm).

This report addresses this multivariate problem consisting of five random variables: average red, green, and blue value of an image patch and corresponding x, y-position of the patch in a larger map image. The dataset has been obtained by generating random image patches from a given map image. These image patches could simulate camera images obtained during a flight with a micro aerial vehicle (MAV) over the map image.

The goal of this report is to compare models for predicting the x, y-coordinate of unseen image patches. To this end, dependency models are infered that are able to capture the multivariate distribution of the sample data. We compare approaches using linear regression, graphical Gaussian models, and vine copulas.

The use of *copulas* allows to model marginal distributions and dependence structure independently, allowing for convenient and powerful representation of joint probability distributions. Vine copulas leverage these advantages and bring the advantages to higher dimensional distributions.

The vine copula approach is based on Sklar's theorem that states that multivariate distribution can be described by marginal distributions and the dependence structure—the copulas. Using bivariate copulas as building blocks, more complex interaction structures can be achieved by building *vine copulas*—nested sets of connected trees.

The remainder of this report is structured as follows. Section 2 introduces the method for generating the dataset, and for inferring the linear least square predictor, graphical Gaussian models, and vine copulas. Section 3 shows the obtained results with respect to root mean squared error (RMSE). In Section 4, the results are discussed and models are compared. Finally, in Section 5, conclusions and future research directions are given.

2 Methods

2.1 Dataset Generation

For generating a dataset, a suitable map image was chosen. Since the color distribution should be related to the position of the image patch—and not randomly spread over the image—the search term 'rainbow art' was used in Google's image search. Rainbow art features an image gradient, without having a too strong or too weak correlation between mean color and x, y position. The selected image can be seen in Figure 1.

In order to obtain image patches from a given map image, $N = 1000$ different views of the map image are generated using the tool *draug*[1]. The camera positions are sampled from a normal distribution with the following parameters:

$$(x, y) \sim \mathcal{N}(\mu, \Sigma) \tag{1}$$

$$\mu = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix} \tag{2}$$

$$\Sigma = \begin{pmatrix} 107 & 0 \\ 0 & 80 \end{pmatrix} \tag{3}$$

[1]draug is a tool that I developed during my graduation project. It can be found on: `https://github.com/Pold87/draug`

(a) Map image

(b) Random image patch

Figure 1: Figure 1a Shows the map image that is used for generating the image patches. Figure 1b shows one sample of the $N = 1000$ image patches that were generated for the generation of the data set.

Therefore, x and y positions are sampled independently. The standard deviations were chosen, such that $99\,\%$ of the data points are expected to be in the ranges $[0, 640]$ and $[0, 480]$, respectively for x, and y values. The resulting dataset can be found in the appendix. 500 of the images are used as training images and 500 as test images.

Three features per image patch are used: the average red, green, and blue color. The data set based on $N = 1000$ image patches can be found in the appendix. By construction of the dataset, it is known that the samples are identically and independently distributed (i.i.d).

2.2 Test for Normality

For motivating the use of graphical Gaussian models, the Shapiro-Wilk test is used (significance level $\alpha = 0.05$), which tests if the univariate marginal distributions are normally distributed.

2.3 Linear Least Squares Predictor

The used random vectors are $Y = (x, y)$ and $X = (R, G, B)$. Given a measurement of the average color values $X = \mathbf{x} = (r, g, b)$, the linear least squares predictor is calculated as follows:

$$\hat{Y}(X) = EY + \operatorname{cov}(Y, X)\operatorname{var}(X)^{-1}(\mathbf{x} - EX)$$

The explained variation R^2 and residual variance are calculated.

2.4 Gaussian Graphical Model

A Gaussian graphical model from the training set is constructed. Two steps are involved in this procedure: model selection, that is, the construction of the independence graph and likelihood inference for the variance matrix V. Log-likelihood and deviance compared to the saturated model are calculated.

2.4.1 Model Selection

For selecting the model, sequential backward elimination from a saturated graphical model is used. In each step, the edge with the highest p-value based on the χ^2 test of independence is deleted. If no p-value is significant (significance level $\alpha = 0.05$), the iteration is stopped.

2.4.2 Fitting

Based on the graphical model achieved in the model selection step, the maximum likelihood estimate of the variance matrix V is calculated using the the IPF algorithm. The IPF algorithm updates the sample variance matrix and fits it to the given graph.

2.4.3 Predictions

The expectation $E_{b|a}(X_b)$ is used to calculate the point estimate. Using Proposition 6.3.1 (Whittaker 2009), the conditional distribution of X_b given $X_a = a$ is normally distributed with mean

$$E_{b|a}(X_b) = \mu_b + V_{ba}V_{aa}^{-1}(x_a - \mu_a) \tag{4}$$

and variance

$$\mathrm{var}_{b|a}(X_b) = V_{bb|a} = V_{bb} - V_{ba}V_{aa}^{-1}V_{ab} \tag{5}$$

R, G, B are given, while x, y are desired. Therefore, $X_a = (R, G, B)$ and $X_b = (x, y)$. The equation for $E_{b|a}(X_b)$ directly related to the linear least squares predictor using the fitted matrix V, where V_{ba} relates to $\mathrm{cov}(Y, X)$ and V_{aa}^{-1} to $\mathrm{var}(X)^{-1}$.

2.5 Vine Copula

Finally, the dependence structure between the variables is described using vine copulas. The samples are converted to pseudo observations and ties are broken at random. The tree structure of the vine is chosen using forward selection. The selection is performed among all possible families in the R package *Vine Copula*[2]. To this end, the available copulas are fitted using maximum likelihood estimation. Then, bivariate copula families for the edges of the vine are chosen using log-likelihood as selection criterion.

[2] An overview of the possible families can be found at: `https://cran.r-project.org/web/packages/VineCopula/VineCopula.pdf`

3 Results

3.1 Test for Normality

Figure 2 shows the pairwise correlation plot of the training dataset. It can be seen that the variables R, G, B seem to be non-normally distributed. This is underlined by Shapiro-Wilk's Normality Test (Table 1). Still, the analysis is

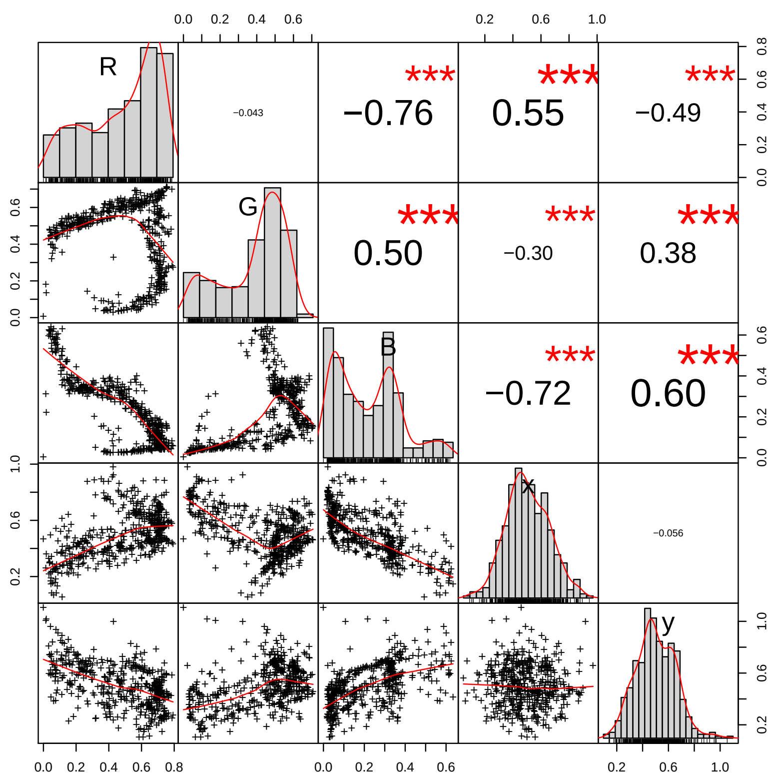


Figure 2: This figure shows a scattermatrix of the training dataset. The upper triangle displays the Spearman correlation coefficients (asterisks indicate significance level), the diagonal displays the histograms and kernel density estimates, and the lower diagonal shows the bivariate sample distributions.

continued assuming a multivariate normal distribution.

3.2 Linear least squares predictor

Table 2 shows the empirical variance matrix and the mean values of the training dataset. Given a measurement of the average color values $X = \mathbf{x} = (r, g, b)$, the

Variable	Statistic	p-value	Normality
R	0.8941	0.0000	no
G	0.8843	0.0000	no
B	0.9183	0.0000	no
x	0.9979	0.2373	yes
y	0.9983	0.4175	yes

Table 1: Shapiro-Wilk's normality test

R	0.05				
G	−0.01	0.04			
B	−0.03	0.02	0.03		
x	0.02	−0.01	−0.02	0.03	
y	−0.02	0.01	0.01	0.00	0.03
means	0.49	0.43	0.22	0.50	0.49
	R	G	B	x	y

Table 2: The sample variance matrix of the data set. The variances are calculated using the operator $\mathrm{var}_{500}(X, Y)$.

linear least squares predictor becomes:

$$
\begin{aligned}
\hat{Y}(X) &= EY + \mathrm{cov}(Y, X)\,\mathrm{var}(X)^{-1}(\mathbf{x} - EX) \\
&= \begin{pmatrix} 0.50 \\ 0.49 \end{pmatrix} + \begin{pmatrix} 0.02 & -0.01 & -0.02 \\ -0.02 & 0.01 & 0.01 \end{pmatrix} \begin{pmatrix} 0.05 & -0.01 & -0.03 \\ -0.01 & 0.04 & 0.02 \\ -0.03 & 0.02 & 0.03 \end{pmatrix}^{-1} \begin{pmatrix} r - 0.49 \\ g - 0.43 \\ b - 0.22 \end{pmatrix} \\
&= \begin{pmatrix} 0.68 + 0.01r - 0.11g - 0.64b \\ 0.60 - 0.45r + 0.36g - 0.15b \end{pmatrix}
\end{aligned}
$$

In Table 3, the inverse correlation matrix is shown. From the table, the pro-

R	7.02				
G	−3.46	3.30			
B	5.47	−2.55	8.95		
x	−1.57	1.59	2.08	4.00	
y	1.97	−1.68	−1.02	−2.49	3.20
	R	G	B	x	y

Table 3: Inverse correlation matrix

portion of explained variation for x and y can be calulated:

$$R^2(x; rest) = (4.00 - 1)/4.00 = 75\,\%$$
$$R^2(y; rest) = (3.20 - 1)/3.20 = 68.75\,\%$$

The determined coefficients from Section 2 are used to obtain x, y-predictions for the $N = 500$ test images. A root mean squared error (RMSE) of 0.13 and 0.15 has been obtained for the x and y coordinate, respectively.

3.3 Gaussian Graphical Model

In the first iteration of the backward elimination method, the edge $\{x, y\}$ is deleted ($\chi^2 = 0.0013, p = 0.972$). In the second iteration, the edge $\{G, x\}$ is deleted ($\chi^2 = 1.165, p = 0.281$). This process yields the graphical model in Figure 4.

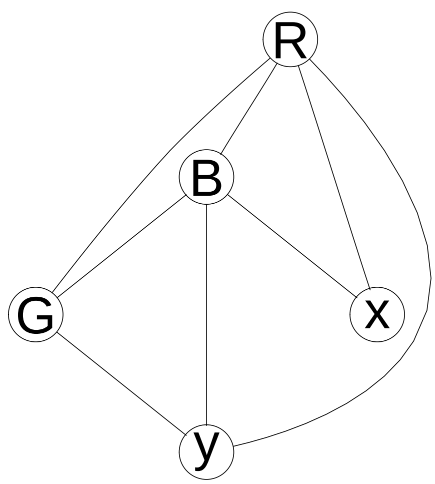


Figure 3: Gaussian graphical model determined by backward elimination.

The sample variance matrix S can be seen in Section 2.3. This leads to the matrix in Table 4. Due to the indepence statements, the values $\hat{v}_{xy}$ and $\hat{v}_{Gx}$ (and their adjoints) change.

3.4 Vine Copula

The parameters of the fitted vine copula can be seen in Table 5.

Figure 3 shows a matrix of contour plots of the fitted bivariate copula in the vine. Perceptive visualizations of the density of the used bivariate copulas can be found in the appendix (A).

R	0.092				
G	-0.006	0.062			
B	-0.071	-0.007	0.085		
x	-0.042	0.000	0.039	0.029	
y	-0.012	0.034	-0.006	0.002	0.029
	R	G	B	x	y

Table 4: The fitted matrix V based on maximum likelihood estimation using IPF.

tree	edge	family	par	par2	tau
1	4,3	Frank	10.52 (0.49)	-	0.68
	1,4	Frank	-9.46 (0.47)	-	-0.65
	5,1	Rotated Tawn type 2 270 degrees	-3.02 (0.30)	0.28 (0.02)	-0.24
	5,2	Frank	12.01 (0.57)	-	0.71
2	1,3;4	Rotated Tawn type 2 90 degrees	-2.25 (0.24)	0.18 (0.03)	-0.15
	5,4;1	Rotated BB8 90 degrees	-3.20 (1.08)	-0.72 (0.17)	-0.31
	2,1;5	Rotated Tawn type 1 180 degrees	2.22 (0.23)	0.24 (0.03)	0.18
3	5,3;1,4	Rotated Tawn type 1 270 degrees	-2.09 (0.14)	0.39 (0.04)	-0.26
	2,4;5,1	Survival Gumbel	1.11 (0.03)	-	0.10
4	2,3;5,1,4	Independence	-	-	0.00

Table 5: Results of the fitting process for vine copulas.

4 Discussion

This report presented three different approaches for modeling the dependencies in a five-dimensional regression problem with three predictors and two target values.
Using the linear least squared predictor yielded the highest RMSE. Using this approach directly, it cannot directly capture possible independencies in the data, leading to overfitting of the model.
The Gaussian graphical model is able to infer independencies leading to a lower RMSE. The GGM is able to infer the independence between the random variables $\{G, x\}$ and $\{x, y\}$, leading to more accurate predictions. The indepencies can be seen in the map image: given all other variables, the knowledge of x does not contribute to the knowledge of the green value.

The vine copula shows the best results on the hold-out test set. Since no clear theory on conditioning vine copulas is existing yet, the mode of the conditional distribution was calculated by iterating over possible values. In future research, a more direct approach for yielding the mode could largely speed-up the process.

In future research, several improvements could be made. In a real-world setting, run-time and computational complexity will be crucial which we left out here. The used image features consisting of the mean red, green, and blue value per patch are too simple for general map images. Therefore, additional features should be included that are able to describe more sophisticated features,

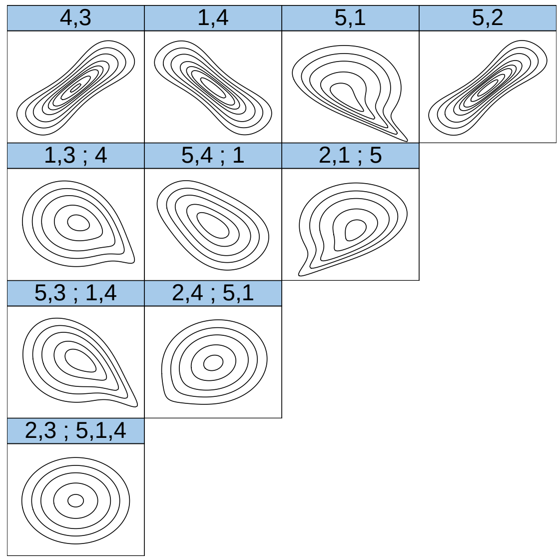


Figure 4: Contour plots with standard normal margins

like edges. An interesting future research direction will be the inclusion of time dependence: in a real-world scenario, images will not be i.i.d but highly correlated. This could be modeled using a Bayesian belief network.

5 Conclusion

In this report, we compared linear regression, graphical Gaussian models and vine copulas for the dependence modeling of variables in a computer vision task.

References

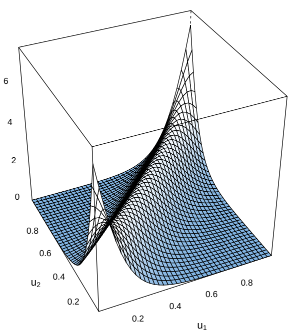
Cooke, Roger M, Harry Joe, and Bo Chang (2015). "Vine Regression". In: *Resources for the Future Discussion Paper*, pp. 15–52.

Elidan, Gal (2013). "Copulas in machine learning". In: *Copulae in mathematical and quantitative finance*. Springer, pp. 39–60.

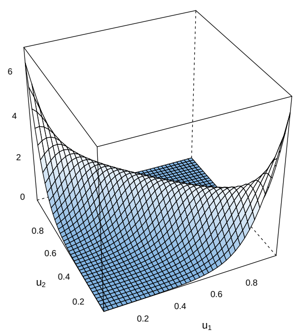
Whittaker, Joe (2009). *Graphical models in applied multivariate statistics*. Wiley Publishing.

A Visualization of the used copulas

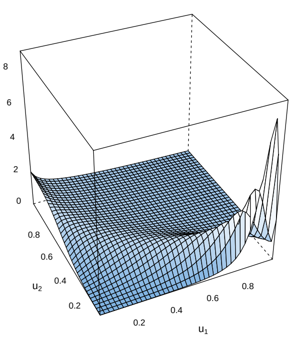
A.1 Tree 1



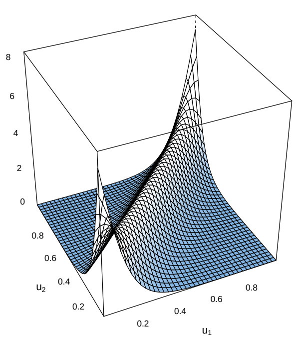
(a) 4,3 Frank (par = 10.52, tau = 0.68)



(b) 1,4 Frank (par = -9.46, tau = -0.65)



(c) 5,1 Rotated Tawn type 2 270 degrees
(par = -3.02, par2 = 0.28, tau = -0.24)



(d) 5,2 Frank (par = 12.01, tau = 0.71)

Figure 5: Perspective visualization of the density of the used bivariate copulas for tree 1.

B Dataset with average red, green, and blue values

red	green	blue	x	y
199.15	184.01	148.09	173.21	312.11
161.74	104.92	98.682	471.52	156.44
152.76	115.75	91.875	61.441	70.699
219.71	120.5	92.252	304.45	61.207
221.39	199.51	169.75	164.92	40.368
228.69	224.08	196.25	185.25	259.92
181.83	131.69	147.57	456.16	218.93
195.77	71.636	60.897	32.196	107.07
109.1	193.62	143.95	432.03	95.567
255	255	255	168.08	182.98
234.87	192.93	147.54	473.74	37.964
255	255	255	194.94	135.98
246.92	165.26	213.07	-9.5128	346.47
...	...	...	...	...

C Technical Report

The code for generating the graphs and yielding the results presented in this report was written in `R`. The used data is saved in CSV format. Both can be found on GitHub:
`https://github.com/Pold87/decision-theory`

The section on Graphical Gaussian Models made use of the packages `gRim`, `gRbase` and `Rgraphviz` for the visualization.

The section on VineCopula used the packages `copula` and `VineCopula`.