An Efficient Way to Compose Distributions from Exponential Families

Vladislav Bína

Faculty of Management, University of Economics, Prague

Jarošovská 1117/II, 37701 Jindřichův Hradec

e-mail: bina@fm.vse.cz

Abstract

In previous works we attempted to compose multivariate densities of continuous random variables. The paper [5] showed an implementation of Iterative Proportional Fitting Procedure numerically approximating the multivariate density from low-dimensional ones. In [2] we defined an operation of composition for general continuous densities with advantageous properties in case of certain copula classes. The paper [4] further analyzed properties of composition in continuous densities and sketched a basic application for densities from exponential families. This application was illustrated on an non-trivial example in [3].

The exponential families are closed with respect to the operation of composition, i.e. the result of composition remains in the exponential family. Though up to this moment only rather toy applications were performed, still it took a non-trivial effort to perform algebraic manipulations with multivariate densities expressed in a "natural form" of exponential families. Therefore, it appears to be advantageous to employ some computer algebra system capable of symbolic manipulations with matrices necessary for the representation of multivariate distributions.

Keywords: Operator of composition, continuous variable, exponential family, computer algebra system.

1 Introduction

Modern equipment, electronic sensors and automated approaches of measurement provide an enormous amounts of data which is almost in every case multivariate and usually shows high dimensionality. This is the setting in which the curse of dimensionality (see, e.g., [1, 6]) appears to be serious issue. One particular facet of this is a impossibility, or at least enormous inaccuracy of estimation of high-dimensional multivariate distributions from the data. A possible way of dealing with these problems is the employment of some factorization of high-dimensional distributions and performance of local computations with low-dimensional marginals only. Well-known are approaches of Bayesian Networks now successfully implemented in several commercial computational environments (Hugin, Bayesia, etc.). An algebraic alternative is represented by compositional models (see Jiroušek[11]).

As we already hinted, it appears to be impossible to estimate multivariate densities of higher dimensions directly from data. But still it is possible to perform some analysis of dependence structure among analyzed variables and to estimate the low-dimensional discrete distributions or continuous densities as basic building blocks. These building blocks overlap in a way, i.e. the considered marginals have some common variables and thus the building blocks can be seen as hypergraph edges, where hypergraph vertices are the particular variables.

Since we present a practical application of scheme sketched in previous publications ([4] and [3]), the methodology is connecting several different fields from theory of compositional models developed under the framework of classical probability theory (see Jiroušek[11, 4]), theoretical description of exponential families and its useful properties [13], employment of algorithms from theory of maximum likelihood estimation of multivariate normal distribution [7] with an implementation in R package *mvnmle* [9] and a Python based computer algebra system *SymPy* with implementation [10] under statistical computational environment of R [15].

The presented paper shows all necessary theoretical prerequisites concerning composition in exponential families together with the way how to implement compositional models in a rather user friendly way leaving the boring computations on a computer algebra system.

2 Compositional Models & Exponential Families

Within the presented paper we consider a finite index set $N = \{1, \ldots, n\}$ together with a set of random variables $\{X_i\}_{i \in N}$ with values, or vectors of values, denoted by the corresponding lowercase letters. The domain of variables will be denoted by the corresponding bold uppercase letter $\mathbf{X}_i$. In general, variables with a finite or countable set of possible *states* are called *discrete*; other variables are called *continuous*. In this paper, we will focus on the later case.

The probability density functions of continuous random variables will be denoted by lowercase letters of the Latin alphabet $(f, g, h, \ldots)$, e.g., the abbreviated notation $f(x_K)$ denotes a multidimensional density of variables having indices from set $K \subseteq N$. For a probability density function $f(x_K)$ and any set of variable indices $L \subset K$, a marginal probability density $f(x_L)$ of $f(x_K)$ can be computed for each x_L as follows

$$f(x_L) = \int_{\mathbf{X}_{K \setminus L}} f(x_K) dx_{K \setminus L}$$

where obviously the integration runs over the domains of all variables in $K \setminus L$. We will also employ an equivalent way to denote the marginal $f(x_L)$, namely $f^{\downarrow\{L\}}$ which was introduced by Glenn Shafer (see, e.g., [17]).

Having probability density $f(x_K)$ and two disjoint subsets $L, M \subseteq K$ we define the *conditional probability density* of X_L given a value $x_M = \mathbf{x_M}$ for every $x_{L\cup M}$ as

$$f(x_L \mid x_M = \mathbf{x_M}) f(x_M = \mathbf{x_M}) = f(x_L, x_M = \mathbf{x_M}).$$

Let us note that for $f(x_M = \mathbf{x_M}) = 0$ the definition is ambiguous, but we do not need to exclude such cases.

2.1 Composition of Continuous Densities

Let us have two probability density functions $f(x_K)$ and $g(x_K)$ with the same set of variables X_K. Then f is said to be *absolutely continuous with respect to* g, or *dominated by* g (denoted by $f \ll g$) if for each $x_K \in X_K$ it holds

$$(g(x_K) = 0 \Rightarrow f(x_K) = 0)\,.$$

Consider two sets of continuous variables X_L and X_M, a probability density $f(x_L)$, and a probability density $g(x_M)$ such that $f(x_{L\cap M}) \ll g(x_{L\cap M})$. The *right composition* is given by

$$f(x_L) \triangleright g(x_M) = \frac{f(x_L) g(x_M)}{g(x_{L\cap M})} = f(x_L) \cdot g(x_M \mid x_L).$$

For details and important properties of composition in continuous case, please, refer to [4].

2.2 Exponential Families

The possibility to define the operation of composition for densities of distributions from *exponential families* was studied in [4]. The exponential family is an interesting set of probability distributions that can be expressed in a certain form, e.g., see [12].

Now, let us recall the most important notions and properties introduced in the context of compositional model in [4]. *Density* $f(x_L)$ *belongs to the exponential family* if it can be expressed in the form

$$f(x_L; \theta_L) = h(x_L) e^{\eta_L(\theta_L)\cdot T(x_L) - A(\eta_L)}$$

where θ_L is a (real) vector of parameters and $h(x_L)$, $T(x_L)$, $\eta_L(\theta_L)$ and $A(\eta_L)$ are vector functions.

The function $\eta_L(\theta_L)$ is a *natural parameter* (or exponential parameter), $T(x_L)$ is a *sufficient statistic*, $A(\eta_L)$ is a log-partition function and $h(x_L)$ is a non-negative base measure. Obviously, the product of $\eta_L(\theta_L)$ and $T(x_L)$ vector functions is a scalar product. Examples of the most important members of the exponential family, such as Gaussian, binomial, multinomial, Gamma and Beta distributions can be found, e.g., in [13].

It can be shown that exponential family is closed with regard to several important operations, particularly product, marginalization and conditioning, see, e.g., Lemmata 6 and 8 in [12].

If both operands belong to the exponential family, the result of operation of composition is defined and can be expressed in the above form and thus also belongs to the exponential family. I.e. for two densities $f(x_L)$ and $g(x_M)$ belonging to an exponential family, i.e. such that $f(x_L) = h_L(x_L)e^{\eta_L \cdot T_L(x_L) - A_L(\eta_L)}$ and $g(x_M) = h_M(x_M)e^{\eta_M \cdot T_M(x_M) - A_M(\eta_M)}$ the *composition also belongs to the exponential family.*

Let us look at this property in more detail:For disjoint L and M we get the product of both densities, which obviously also belongs to the exponential family.

If the other possibility realizes, i.e. if $L \cap M \neq \emptyset$ then we can express

$$g(x_M) = h_M(x_M)e^{\eta_{L\cap M} \cdot T_{L\cap M}(x_{L\cap M}) + \eta_{M\setminus L} \cdot T_{M\setminus L}(x_{M\setminus L}) - A_M(\eta_{L\cap M}, \eta_{M\setminus L})}.$$

According to [12] the conditional distribution

$$g(x_{M\setminus L} \,|\, x_{L\cap M} = \mathbf{x_{L\cap M}}) = h_{\mathbf{L\cap M}} e^{\eta_{M\setminus L} \cdot T_{M\setminus L}(x_{M\setminus L}) - A_{\mathbf{L\cap M}}(\eta_{M\setminus L})}$$

where $h_{\mathbf{L\cap M}}$ and $A_{\mathbf{L\cap M}}$ are dependent on the values of conditioning variables. It is now apparent that the product of $f(x_L)$ and $g(x_{M\setminus L} \,|\, x_{L\cap M})$ again belongs to the exponential family since it can be written in the corresponding form, i.e.

$$(f \triangleright g)(x_{L\cup M}) = h_L h_{\mathbf{L\cap M}} e^{\eta_L \cdot T_L(x_L) + \eta_{M\setminus L} \cdot T_{M\setminus L}(x_{M\setminus L}) - A_L(\eta_L) - A_{\mathbf{L\cap M}}(\eta_{M\setminus L})}.$$

2.3 Multivariate Normal Distribution

The non-degenerate multivariate normal distribution has a symmetric and positive definite covariance matrix $\mathbf{\Sigma}$. In such case, the multivariate normal distribution $f(x_L)$ with vector of means μ_L and covariance matrix $\mathbf{\Sigma}_L$ has a density given by formula

$$f(x_L; \mu_L, \mathbf{\Sigma}_L) = \frac{1}{\sqrt{(2\pi)^\ell |\mathbf{\Sigma}_L|}} \exp\left(-\frac{1}{2}(x_L - \mu_L)^{\mathrm{T}} \mathbf{\Sigma}_L^{-1} (x_L - \mu_L)\right)$$

where ℓ is a dimension (length) of x_L vector, symbol $^{\mathrm{T}}$ stands for a vector transpose, $|\mathbf{\Sigma}_L|$ is determinant of covariance matrix and $\mathbf{\Sigma}_L^{-1}$ is an inverse of covariance matrix.

Thus, multivariate density $f(x_L; \mu_L, \mathbf{\Sigma}_L)$ has variables and functions according to definition of exponential family given in the following way

$$\begin{aligned} x_L &= (x_1, \dots, x_\ell)^{\mathrm{T}}, \\ \eta_L &= \begin{pmatrix} \mathbf{\Sigma}_L^{-1}\mu_L \\ -\frac{1}{2}\mathbf{\Sigma}_L^{-1} \end{pmatrix}, \\ T_L(x_L) &= \begin{pmatrix} x_L \\ x_L x_L^{\mathrm{T}} \end{pmatrix}, \\ A_L(\eta_L) &= \frac{1}{2}\mu_L^{\mathrm{T}}\mathbf{\Sigma}_L^{-1}\mu_L + \frac{1}{2}\log|\mathbf{\Sigma}_L|, \\ h_L(x_L) &= (2\pi)^{-\frac{\ell}{2}}. \end{aligned}$$

2.4 Conditional multivariate density

Let us have a multivariate density $g(x_M; \mu_M, \mathbf{\Sigma}_M)$ and let us divide index set M into two disjoint parts such that $A = L \cap M$ and $B = M \setminus L$. Thus, the m-dimensional vector x_M can be partitioned into two parts of dimensions m_A and m_B $(m_A + m_B = m)$ in such a way that

$$x_M = \begin{pmatrix} x_A \\ x_B \end{pmatrix}$$

and similarly

$$\mu_M = \begin{pmatrix} \mu_A \\ \mu_B \end{pmatrix}.$$

The covariance matrix is partitioned into the corresponding blocks in the following way

$$\mathbf{\Sigma}_M = \begin{pmatrix} \mathbf{\Sigma}_{AA} & \mathbf{\Sigma}_{AB} \\ \mathbf{\Sigma}_{BA} & \mathbf{\Sigma}_{BB} \end{pmatrix}$$

having sizes

$$\begin{pmatrix} m_A^2 & m_A m_B \\ m_A m_B & m_B^2 \end{pmatrix}.$$

Thus, having the multivariate density $g(x_M) \sim \mathcal{N}(\mu_M, \mathbf{\Sigma}_M)$ the *conditional multivariate density* $g(x_{M\setminus L} \,|\, x_{L\cap M} = \mathbf{a}) = g(x_B \,|\, x_A = \mathbf{a})$ is again a multivariate density distribution (see, e.g., [8]) and $g(x_B \,|\, x_A = \mathbf{a}) \sim \mathcal{N}(\overline{\mu}_B, \overline{\mathbf{\Sigma}}_B)$ where

$$\overline{\mu}_B = \mu_B + \mathbf{\Sigma}_{BA}\mathbf{\Sigma}_{AA}^{-1}(\mathbf{a} - \mu_A)$$

and

$$\overline{\mathbf{\Sigma}}_B = \mathbf{\Sigma}_{BB} - \mathbf{\Sigma}_{BA}\mathbf{\Sigma}_{AA}^{-1}\mathbf{\Sigma}_{AB}.$$

We can somewhat surprisingly see, that the known value $\mathbf{a}$ influences the mean of conditional density but not its covariance matrix. Let us note that the formula for $\overline{\mathbf{\Sigma}}_B$ is known as the Schur complement of $\mathbf{\Sigma}_{AA}$ in $\mathbf{\Sigma}_M$ and $\mathbf{\Sigma}_{AA}^{-1}$ is a generalized inverse (see again [8]).

2.5 Product of Multivariate Densities

Similarly, the product of two multivariate normal densities is again multivariate normal distribution (must be then renormalized). For two multivariate densities $f(x_L) \sim \mathcal{N}(\mu_L, \mathbf{\Sigma}_L)$ and $g(x_M) \sim \mathcal{N}(\mu_M, \mathbf{\Sigma}_M)$ we get

$$f(x_L)g(x_M) \sim \mathcal{N}(\overline{\mu}, \overline{\mathbf{\Sigma}})$$

where

$$\overline{\mu} = \overline{\mathbf{\Sigma}} \left(\mathbf{\Sigma}_L^{-1} \mu_L + \mathbf{\Sigma}_M^{-1} \mu_M \right)$$

and

$$\overline{\mathbf{\Sigma}} = \left(\mathbf{\Sigma}_L^{-1} + \mathbf{\Sigma}_M^{-1} \right)^{-1} .$$

The normalizing constant is (see [16]) equal to

$$(2\pi)^{-\frac{\ell+m}{2}} \left| \mathbf{\Sigma}_L + \mathbf{\Sigma}_M \right|^{\frac{1}{2}} \exp \left(-\frac{1}{2} \left(\mu_L - \mu_M \right)^{\mathrm{T}} \left(\mathbf{\Sigma}_L + \mathbf{\Sigma}_M \right)^{-1} \left(\mu_L - \mu_M \right) \right) .$$

3 Partially Symbolic Manipulation with Compositional Models

As the kind reader already guessed from the formulas in previous section, the general case of composition in exponential families involves several matrix operations with partially numeric and partially symbolic manipulation. Obviously, it is advantageous to perform all computations in an (semi)automated way. We performed all implementations of compositional models in R software [15] which is very advantageous for its vector and matrix operations together with abundance of statistic and probabilistic methods available. Therefore, we decided to employ a Python based computer algebra system SymPy with its R interface rSymPy [10].

First of all, let us describe a simple data set which will be used in the following application of above described theory. It concerns the levels of 5 characteristics measured in the folicular fluid of 22 pregnant cows. The five variables (pH, pCO_2, pO_2, HCO_3, BE(B)) appear to have Gaussian distribution (first three variables on the 5 percent significance level using Shapiro-Wilk test of normality, two other variables on 1 per thousand significance level which is cause in both cases by a pair outliers). Thus, it appears to be a bad idea to approximate a five-dimensional multivariate Gaussian density with 30 continuous parameters from 110 measurements of 5 variables and it seems to be a good idea to estimate from data several low-dimensional (two- or three-dimensional) distributions and to compose them. (Three-dimensional distribution has 12 parameters.)

In this paper, we will not focus on the choice of the most suitable marginals as building stones. We will choose them in a rather rough and intuitive way based on the Pearson correlation matrix of the five variables (see Table 1). The proper way is to use some principles of probabilistic structure learning approaches (see, e.g., Zhou [18]).

Table 1: (Pearson) correlation matrix of five analyzed variables.

	pH	pCO_2	pO_2	HCO_3	BE(B)
pH	1.000	-0.549	0.529	0.647	0.704
pCO_2	-0.549	1.000	-0.509	0.280	0.206
pO_2	0.529	-0.509	1.000	0.141	0.184
HCO_3	0.647	0.280	0.141	1.000	0.997
BE(B)	0.704	0.206	0.184	0.997	1.000

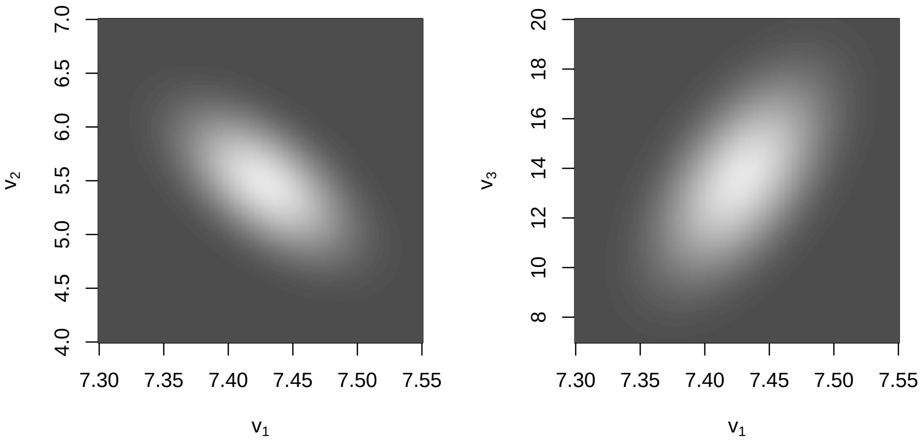

Figure 1: Two-dimensional Gaussian densities of variables of the first composed marginal v_1 and v_2 (left part of figure) and the second composed marginal v_1 and v_3 (right part of figure).

In the following text the five parameters will be denoted by $v_1, \ldots, v_5$ variable symbols. Rather loosely based on the strengthes of the Pearson correlations of moderate and strong linear dependencies we decided to choose three marginals $f_1(v_1, v_2)$, $f_2(v_1, v_3)$ and $f_3(v_1, v_4, v_5)$. For these marginals we estimate their multivariate Gaussian densities from data using maximum likelihood estimates specified in terms of the inverse of the Cholesky factor of the variance-covariance matrix (see [14]) and implemented in R *mvnmle* package [9]. Two estimated two-dimensional densities are depicted in Figure 1.

As the first step, we read our sample data set into `data` data frame and load the above mentioned libraries of *mvnmle* and *rSymPy*. Then we set the simplified variable names, define the list of marginals to be composed (using a list `edges`) and estimate parameters of multivariate distributions using `mlest` function (see bellow).

```
v <- 1:ncol(data)
names <- paste("v",v,sep="")
# names: "pH"    "pCO2" "pO2"   "HCO3" "BE.B"
edges <- list(c(1,2), c(1,3), c(1,4,5))

# MLE for multivariate normal distributions
mu <- NULL; sig <- NULL
for (e in 1:length(edges)) {
    est <- mlest(data[,edges[[e]]])
    emu <- est$muhat
    names(emu) <- names[edges[[e]]]
    mu <- c(mu, list(emu))
    esig <- est$sigmahat
    rownames(esig) <- names[edges[[e]]]
    colnames(esig) <- names[edges[[e]]]
    sig <- c(sig, list(esig))
}
```

The estimated parameters are stored in the following numbered vectors and matrices `mu*` and `sig*`.

```
> cat(sympy("mu1"))     > cat(sympy("sig1"),"\n")
[7.42868181808]          [ 0.0012049443106, -0.0069107030493]
[5.47636363472]          [-0.0069107030493,  0.1314049638269]
> cat(sympy("mu2"))     > cat(sympy("sig2"),"\n")
[7.42868181927]          [ 0.001204944283,   0.0393622676519]
[13.6313637645]          [ 0.039362267652,   4.5957484887845]
> cat(sympy("mu3"))     > cat(sympy("sig3"),"\n")
[7.42873958083]          [ 0.001204867353, 0.045509151565, 0.053034419262]
[26.7414588678]          [ 0.045509151565, 4.216007981078, 4.489984753482]
[2.75089979417]          [ 0.053034419262, 4.489984753481, 4.812928862933]
```

For all three distributions (hypergraph edges) we compute η_i stored in `e*`, T_i stored in `T*`, A_i stored in `A*` and h_i stored in `h*`. Corresponding densities f_i are computed and stored in `f*`.

```
for (e in 1:length(edges)) {
    sympy(paste("ea=(sig",e,")**(-1)*mu",e,"",sep=""))
    sympy(paste("eb=-((sig",e,")**(-1))/2",sep=""))
    sympy(paste("e",e,"=(ea.T).col_join(eb)",sep=""))

    sympy(paste("T",e,"=(x",e,".T).col_join(x",e,"*x",e,".T)",sep=""))
    sympy(paste("A",e,"=(mu",e,".T*((sig",e,")**(-1)*mu",e,")/2).det()+
          log(sig",e,".det())/2",sep=""))
    sympy(paste("l=1*x",e,".shape[0]",sep=""))
    sympy(paste("h",e,"=1/sqrt((2*pi)**l)",sep=""))

    sympy(paste("f",e,"=h",e,"*exp(e",e,".dot(T",e,")-A",e,")",sep=""))
}
```

Thus, we defined all three continuous densities $f_1, \dots, f_3$ and all vector functions of an exponential family. We can continue in computation of conditional density $f_2(v_3 \,|\, v_1)$ using the formulae in subsection 2.4, i.e. the corresponding subvectors

μ_A and μ_B, submatrices Σ_{AA}, Σ_{AB}, Σ_{BA} and Σ_{BB} and functions defining the conditional density.

```
> sympy(paste("sigc=sigbb-sigba*sigaa**(-1)*sigab"))
[1] "[3.30988976740274]"
> sympy(paste("conda=Matrix([",paste("[",paste("v",A,sep=""),"]",
+ collapse=","),"])",sep=""))
[1] "[v1]"
> sympy(paste("muc=mub+sigba*sigaa**(-1)*(conda-mua)"))
[1] "[-229.043560103312 + 32.6672927676576*v1]"
> sympy("econd=(((sigc)**(-1)*muc).T).col_join(-(sigc**(-1))/2)")
    "[-69.1997547347452 + 9.86960142581775*v1]"
    "[                     -0.151062432629697]"
> cat(sympy("Tcond=(xcond.T).col_join(xcond*xcond.T)"),"\n")
    "[   v3]"
    "[v3**2]"
> sympy("Acond=(muc.T*((sigc)**(-1)*muc)/2).det()+log(sigc.det())/2")
[1] "7924.8790914+log(3.30988977)/2-2260.5686474*v1+161.2065797*v1**2"
> sympy("hcond=1/sqrt((2*pi)**(xcond.shape[0]))")
[1] "2**(1/2)/(2*pi**(1/2))"
```

The conditional density $f_2(v_3 \,|\, v_1)$ is then composed from the corresponding vector functions defining it as a member of exponential family. We arrive at a three-dimensional distribution $f_1(v_1, v_2) \triangleright f_2(v_1, v_3)$ defined by following density (where numeric values were rounded to two decimal places).

```
13.07*2**(1/2)*exp(-43420.44+523.94*v2+11430.68*v1-69.20*v3+9.87*v1*v3
                  -62.50*v1*v2-0.15*v3**2-5.45*v2**2-755.38*v1**2)/pi**(3/2)
```

In a similar manner we can compose the result above with a conditional distribution $f_3(v_4, v_5 \,|\, v_1)$ computed again using the formulae in subsection 2.4 from $f_3(v_1, v_4, v_5)$. Now, the result of the second composition is a five-dimensional distribution $(f_1(v_1, v_2) \triangleright f_2(v_1, v_3)) \triangleright f_3(v_1, v_4, v_5)$ defined again by density (rounded to two decimal places).

```
94.25*2**(1/2)*exp(-1382025.57+523.94*v2+36863.95*v4+252860.96*v1
-69.20*v3-37118.13*v5+9.87*v1*v3+3326.13*v1*v5+517.50*v4*v5-62.50*v1*v2
-3297.34*v1*v4-257.89*v4**2-11686.21*v1**2-259.82*v5**2-0.15*v3**2
-5.45*v2**2)/pi**(5/2)
```

From the resulting multivariate density we can symbolically express marginals which were not estimated from the data. This can be particularly useful in cases when we obtain estimates of the marginal building blocks in the process of composition from different data sources.

The computation of marginals can be hardly computationally (or algorithmically) feasible. The package *rSymPy* did not succeeded in symbolic integration of marginal $(f_1(v_1, v_2) \triangleright f_2(v_1, v_3))(v_2, v_3)$ but we succeeded in computation of the corresponding two-dimensional density using online tool of Wolfram Alpha (see Figure 2). The approximately computed result appears to be again in the form of Gaussian density, i.e. it belongs to the exponential family as expected according to the above presented assertions. But it appears that computation of marginals is in general rather uneasy task which probably can be made feasible if the integration procedure take advantage of exponential family properties.

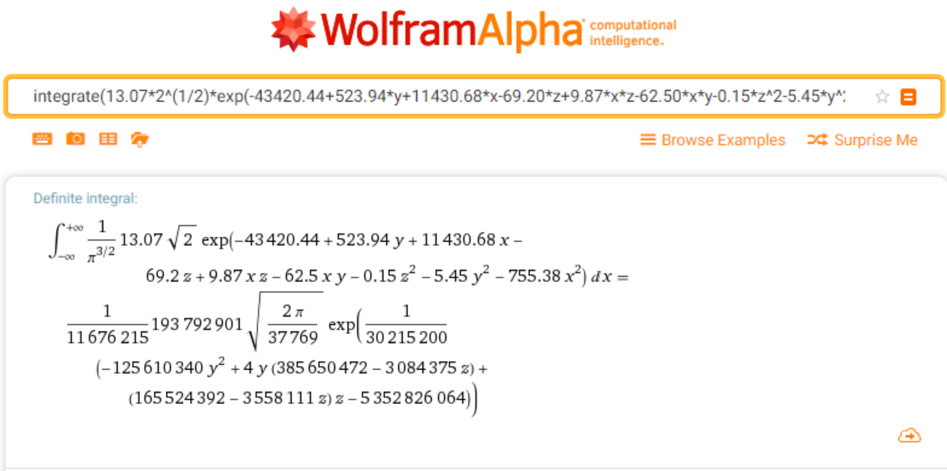

Figure 2: Two-dimensional Gaussian density computed as a marginal from the composition.

4 Conclusions and Possible Continuation

In this paper we presented an application of computer algebra system *rSymPy* able to perform operations with parameters of exponential families in order to (partially) symbolically perform the operation of composition using exponential families, namely low-dimensional Gaussian densities estimated from data.

Let us mention that the computer algebra system is capable of computation in arbitrary precision. Moreover, if the parameters of composed distributions are rational all operations are performed precisely and results is an exact expression (in a same way as in a toy example in [3]).

The possible future course of development contains the user friendly interface, setting of proper procedures for (sub)optimal choice of composed marginals and their estimation and elaboration of procedures leading to the efficient integration of marginals of compositional models.

Acknowledgement

The research was supported by Academy of Sciences of the Czech Republic and Ministry of Science and Technology (Taiwan) under project MOST-18-4 (Compositional models for data mining).

Author also wants to express his sincere gratitude for providing the data to associate professor Svatopluk Čech from Brno University of Veterinary and Pharmaceutical Sciences.

References

[1] R. Bellman. *Dynamic Programming*. Dover Books on Computer Science Series. Dover Publications, 2003.

[2] V. Bína. An Operator of Composition for the Multivariate Copula Construction. In *Proceedings of the 33rd International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering (MaxEnt'13)*, volume 1636, pages 5–10, Canberra, Australia, dec 2014.

[3] V. Bína. Composition in exponential families: Case of multivariate normal distribution. In V. Novák, M. Inuiguchi, and M. Štěpnička, editors, *Proceedings of the 20th Czech-Japan Seminar on Data Analysis and Decision Making under Uncertainty*, pages 9–14, Pardubice, 2017.

[4] V. Bína. Composition Operator for Densities of Continuous Variables. *International Journal of Approximate Reasoning*, 88:614–623, 2017.

[5] V. Bína and R. Jiroušek. A Short Note on Multivariate Dependence Modeling. *Kybernetika*, 49(3):420–432, 2013.

[6] D. L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. In *AMS Conference on Math Challenges of the 21st Century*, 2000.

[7] P. S. Dwyer. Some applications of matrix derivatives in multivariate analysis. *Journal of the American Statistical Association*, 62(318):607–625, 1967.

[8] M. L. Eaton. *Multivariate statistics: A vector space approach.* John Wiley and Sons, New York, 1983.

[9] K. Gross and with help from Douglas Bates. *mvnmle: ML estimation for multivariate normal data with missing values.*, 2012. R package version 0.1-11.

[10] G. Grothendieck and C. C. J. G. Bellosta. *rSymPy: R interface to SymPy computer algebra system*, 2012. R package version 0.2-1.1.

[11] R. Jiroušek. Foundations of compositional model theory. *International Journal of General Systems*, 40(6):623–678, 2011.

[12] E. L. Lehmann and J. P. Romano. *Testing statistical hypotheses.* Springer Texts in Statistics. Springer, New York, third edition, 2005.

[13] F. Nielsen and V. Garcia. Statistical exponential families: A digest with flash cards. *CoRR*, abs/0911.4863, 2009.

[14] J. C. Pinheiro and D. M. Bates. Unconstrained parametrizations for variance-covariance matrices. *Statistics and Computing*, 6:289–296, 1996.

[15] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2017.

[16] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning.* MIT Press, 2006.

[17] G. Shafer. *Probabilistic Expert Systems.* CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics, Philadelphia, USA, 1996.

[18] Y. Zhou. Structure learning of probabilistic graphical models: A comprehensive survey. *CoRR*, abs/1111.6925, 2011.