TU München
Fakultät für Informatik
PD Dr. Rudolph Triebel
John Chiotellis

Machine Learning for Robotics and Computer Vision
Summer term 2016

Homework Solution 2

Topic 1: Regression
May 13th, 2016

Exercise 1: Bayesian Update

Consider a linear regression model with basis functions $\phi(x)$ as presented in the lecture. We assume a Gaussian prior distribution for the weights:

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|m_0, S_0)$$

Suppose we have already observed N data points, so the posterior distribution is

$$p(\mathbf{w}|\mathbf{t}) = \mathcal{N}(\mathbf{w}|m_N, S_N)$$

with

$$m_N = S_N(S_0^{-1}m_0 + \sigma^{-2}\Phi^T\mathbf{t}) \quad \text{and} \quad S_N^{-1} = S_0^{-1} + \sigma^{-2}\Phi^T\Phi.$$

Now, we observe a new data point (x_{N+1}, t_{N+1}). What is the new posterior?

Using Bayes rule, we found out that having a Gaussian prior and a Gaussian likelihood gave us a Gaussian posterior which we can use as the prior for the next iteration (next sample that we observe). Now we want to compute $p(\mathbf{w}|\mathbf{t}, t_{N+1}, x_{N+1})$ which reduces to $p(\mathbf{w}|t_{N+1}, x_{N+1}, m_N, S_N)$.

Our likelihood is

$$p(t_{N+1}|x_{N+1}, \mathbf{w}) = \mathcal{N}(t_{N+1}|y(\mathbf{w}, \phi(x)), \sigma^2)$$

Let $\phi_N = \phi(x_N)$ to simplify notation. Writing the likelihood explicitly we get

$$p(t_{N+1}|x_{N+1}, \mathbf{w}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(t_{N+1} - \mathbf{w}^T\phi_{N+1})^2}{2\sigma^2}\right)$$

Our posterior is

$$p(\mathbf{w}|t_{N+1}, x_{N+1}, m_N, S_N) = \frac{p(t_{N+1}|x_{N+1}, \mathbf{w})p(\mathbf{w}|\mathbf{t})}{p(t_{N+1}|x_{N+1}, \mathbf{t})}$$

We want the maximum likelihood of the posterior. The denominator is independent of $\mathbf{w}$ so we can ignore it.

$$
\begin{aligned}
p(\mathbf{w}|t_{N+1}, x_{N+1}, m_N, S_N) &\propto p(\mathbf{w}|\mathbf{t})p(t_{N+1}|x_{N+1}, \mathbf{w}) \\
&\propto \exp\left(-\frac{1}{2}(\mathbf{w}-m_N)^T S_N^{-1}(\mathbf{w}-m_N) - \frac{(t_{N+1}-\mathbf{w}^T\phi_{N+1})^2}{2\sigma^2}\right)
\end{aligned}
$$

Maximizing the likelihood is equivalent to maximizing the log-likelihood and that is the same as minimizing the negative log-likelihood. Therefore we are left only with the arguments of the exponential, and we can omit the $-\frac{1}{2}$ factors.

$$
\begin{aligned}
&(\mathbf{w}-m_N)^T S_N^{-1}(\mathbf{w}-m_N) + \frac{(t_{N+1}-\mathbf{w}^T\phi_{N+1})^2}{\sigma^2} \\
=&\mathbf{w}^T S_N^{-1}\mathbf{w} - 2\mathbf{w}^T S_N^{-1} m_N - 2\frac{\mathbf{w}^T\phi_{N+1}t_{N+1}}{\sigma^2} + \frac{\mathbf{w}^T\phi_{N+1}\phi_{N+1}^T\mathbf{w}}{\sigma^2} + const. \\
=&\mathbf{w}^T(S_N^{-1} + \frac{\phi_{N+1}\phi_{N+1}^T}{\sigma^2})\mathbf{w} - 2\mathbf{w}^T\left(S_N^{-1}m_N + \frac{\phi_{N+1}t_{N+1}}{\sigma^2}\right) + const.
\end{aligned}
$$

where *const.* denotes remaining terms that are independent of w.

Comparing this expression with the maximum likelihood for the prior, we can see that our posterior is

$$
p(\mathbf{w}|t_{N+1}, x_{N+1}, m_N, S_N) = \mathcal{N}(w|m_{N+1}, S_{N+1})
$$

with

$$
S_{N+1}^{-1} = S_N^{-1} + \frac{1}{\sigma^2}\phi_{N+1}\phi_{N+1}^T \quad \text{and} \quad m_{N+1} = S_{N+1}(S_N^{-1}m_N + \frac{\phi_{N+1}t_{N+1}}{\sigma^2})
$$

Exercise 2: Quadrocopter (Programming)

We are testing a tracking program. We evaluate it with the help of a quadrocopter. The quadrocopter sends estimates of its velocity and the tracking program estimates its global position with respect to the quadrocopter's initial position (before flying).

a) The tracker yields these tracked position estimates at a frequency of $1Hz$:

$$
\mathcal{T} = \left\{ \begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1.08 \\ 1.68 \\ 2.38 \end{pmatrix} \begin{pmatrix} -0.83 \\ 1.82 \\ 2.49 \end{pmatrix} \begin{pmatrix} -1.97 \\ 0.28 \\ 2.15 \end{pmatrix} \begin{pmatrix} -1.31 \\ -1.51 \\ 2.59 \end{pmatrix} \begin{pmatrix} 0.57 \\ -1.91 \\ 4.32 \end{pmatrix} \right\}
$$

Plot these data with your tool of choice (e.g. Matlab).

b) Assuming the quadrocopter flies with constant speed, which speed does it have? What is the residual error of the estimation?

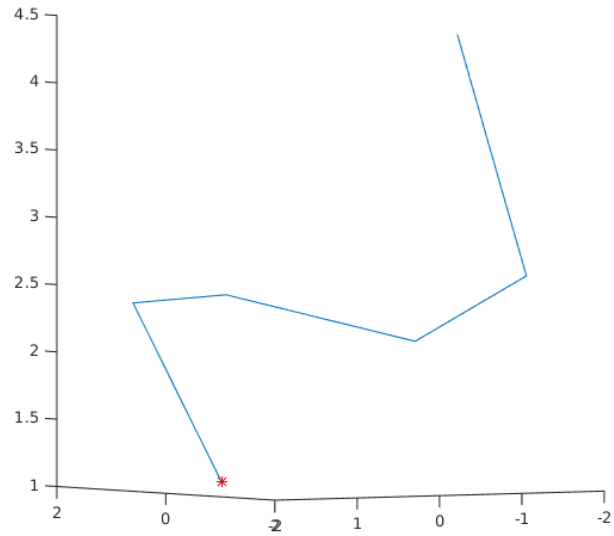


Abbildung 1: Tracker data from quadrocopter. The lines are just an interpolation between the tracked positions (data points).

The task is to estimate the speed of the quadrocopter. We do this using polynomial regression. The functions that we learn are dependent on time. We have to find three functions, one for each coordinate (x, y, z). The regression is done with the matrix Φ and vectors $\mathbf{t}_i$:

$$\Phi = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 1 & 5 \end{pmatrix} \quad \mathbf{t}_x = \begin{pmatrix} 2 \\ 1.08 \\ -0.83 \\ -1.97 \\ -1.31 \\ 0.57 \end{pmatrix} \mathbf{t}_y = \begin{pmatrix} 0 \\ 1.68 \\ 1.82 \\ 0.28 \\ -1.51 \\ -1.91 \end{pmatrix} \mathbf{t}_z = \begin{pmatrix} 1 \\ 2.38 \\ 2.49 \\ 2.15 \\ 2.59 \\ 4.32 \end{pmatrix}$$

The second column of Φ are the timestamps at which the measurements have been taken. In this first case, we assume constant velocity, i.e. we don't have acceleration and the motion equation has only two unkowns w_0 and w_1, i.e. for the case of the x-coordinates we have

$$x(\tau) = w_0 + w_1\tau, \qquad \mathbf{w}_x = (w_0, w_1)^T$$

where $\tau = 0, 1, \ldots$ is the time stamp. Thus, Φ has two cloumns.

The pseudoinverse of Φ is

$$\Phi^\dagger = \begin{pmatrix} 0.524 & 0.381 & 0.238 & 0.095 & -0.048 & -0.190 \\ -0.143 & -0.086 & -0.029 & 0.029 & 0.086 & 0.143 \end{pmatrix}$$

With this we compute $\mathbf{w}_i = \Phi^\dagger \mathbf{t}_i$:

$$\mathbf{w}_x = \Phi^\dagger \mathbf{t}_x = \begin{pmatrix} 1.0267 \\ -0.4421 \end{pmatrix} \quad \mathbf{w}_y = \begin{pmatrix} 1.5383 \\ -0.5918 \end{pmatrix} \quad \mathbf{w}_z = \begin{pmatrix} 1.2825 \\ 0.4830 \end{pmatrix}$$

Considering our model the speed in the 3 dimensions is given by
$\mathbf{v} = (w_{1x} \quad w_{1y} \quad w_{1z})^T = (-0.4421 \quad -0.5918 \quad 0.4830)^T$.
The speed's magnitude is therefore $\|\mathbf{v}\| = 0.8827$.
The residual errors are defined as

$$\begin{aligned} r_x &= \|\Phi \mathbf{w}_x - \mathbf{t}_x\| = 2.8902 \\ r_y &= \|\Phi \mathbf{w}_y - \mathbf{t}_y\| = 2.4571 \\ r_z &= \|\Phi \mathbf{w}_z - \mathbf{t}_z\| = 1.2807 \end{aligned}$$

c) Now assume that the quadrocopter flies with constant acceleration. What is the residual error now? Is the error higher or lower? Why?

Now we have a quadratic motion equation:

$$x(\tau) = w_0 + w_1\tau + w_2\tau^2, \qquad \mathbf{w}_x = (w_0, w_1, w_2)^T,$$

where w_1 is velocity and w_2 is (half the) acceleration. This means we have to estimate 3 function parameters. Thus, the matrix Φ has one more column, i.e.

$$\Phi = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \\ 1 & 5 & 25 \end{pmatrix}$$

Again, we compute the pseudoinverse and multiply it with the vectors $\mathbf{t}_i$. We obtain:

$$\mathbf{w}_x = \begin{pmatrix} 2.4739 \\ -2.6128 \\ 0.4341 \end{pmatrix} \quad \mathbf{w}_y = \begin{pmatrix} 0.4573 \\ 1.0297 \\ -0.3243 \end{pmatrix} \quad \mathbf{w}_z = \begin{pmatrix} 1.4656 \\ 0.2084 \\ 0.0549 \end{pmatrix}$$

The residual errors now are

$$(r_x \quad r_y \quad r_z) = (1.1474 \quad 1.4527 \quad 1.2359)$$

By incorporating acceleration in our model, we increased the model complexity (3 basis functions instead of 2). Therefore our model is able to capture the actual motion of the quadrocopter more precisely and that is why the residual errors are lower.

d) According to our last model, what is the quadrocopter's most likely position in the next second?

If we want to estimate the position in the next second, we can imagine a new row in our Φ matrix $\phi_6 = (1 \quad 6 \quad 36)$. Multiplying this row with our model parameters w for the last model gives us the estimate:

$$t'_6 = (\phi_6 w_x \quad \phi_6 w_y \quad \phi_6 w_z) = (2.4259 \quad -5.0397 \quad 4.6930)$$

Topic 2: Probabilistic Graphical Models

Exercise 3: Reading a graphical model

We have the following graphical model:

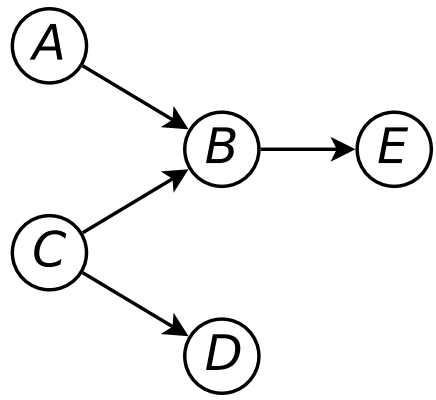


Abbildung 2: Graphical model.

a) Write the joint probability distribution corresponding to the graphical model depicted in Fig. 2.

$$p(A, B, C, D, E) = p(A)p(C)p(B \mid A, C)p(D \mid C)p(E \mid B)$$

b) What are the conditional independence assumptions of this model?

- $A \perp\!\!\!\perp C \mid \emptyset$,
- $D \perp\!\!\!\perp A, B \mid C$,
- $E \perp\!\!\!\perp A, C, D \mid B$.

c) Which of the following assertions are true, and why?

Algorithm to check, whether X is d-seperated from Y by Z (X,Y,Z sets of nodes):

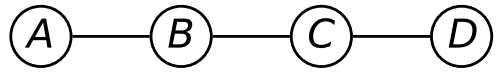
```
boolean is_dsep(X,Y,Z){
foreach x ∈ X, y ∈ Y
 foreach path p connecting x and y
  if (!is_blocked(p,Z)) return false;
 end;
end;
return true;
}

boolean is_blocked(p,Z){
foreach n ∈ p
```

```
 if (type(n) == hh)
  if (n ∉ Z ∧ m ∉ Z  ∀  n→ ... →m )
   return true; //case (b)
  end
 else //type(n) == ht or type(n) == tt
  if (n ∈ Z)
   return true; //case (a)
  end
 end
end
return false;
}
```

- B is d-separated from D by C: true (case (a)),
- A is d-separated from C by E: false (case (b) fails as $B \to E$),
- A is d-separated from C by D: true (case (b)),
- E is d-separated from D by B: true (case (a)),
- E is d-separated from D by A: false.

Exercise 4: Markov Chain

We have the following Markov Chain:



a) Write the joint probability distribution associated to this Markov Chain.

$$p(A, B, C, D) = \frac{1}{Z}\psi_{A,B}(A, B)\psi_{B,C}(B, C)\psi_{C,D}(C, D)$$

b) Each variable can take value 0 or 1, and we want to express that it is 9 times more probable that neighboring variables have equal values than they have different value. Give the potential functions of this Markov Chain.

All three potential functions are the same:

V_1 \ V_2	0	1
0	9	1
1	1	9

Notice that the values need not be normalized in any way.

c) Compute the probability distributions $p(A)$ and $p(C)$.

μ_α and μ_β can be calculated recursively:

$$\mu_\alpha(x_n) = \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n)\mu_\alpha(x_{n-1})$$
$$\mu_\beta(x_n) = \sum_{x_{n+1}} \psi_{n,n+1}(x_n, x_{n+1})\mu_\beta(x_{n+1})$$

With our potentials this yields:

- $\mu_\alpha(A) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ (initialization, optional),
- $\mu_\alpha(B) = \begin{pmatrix} \sum_A \mu_\alpha(A)\psi_{A,B}(A, 0) \\ \sum_A \mu_\alpha(A)\psi_{A,B}(A, 1) \end{pmatrix} = \begin{pmatrix} 1 \times 9 + 1 \times 1 \\ 1 \times 1 + 1 \times 9 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \end{pmatrix}$
- $\mu_\alpha(C) = \begin{pmatrix} \sum_B \mu_\alpha(B)\psi_{B,C}(B, 0) \\ \sum_B \mu_\alpha(B)\psi_{B,C}(B, 1) \end{pmatrix} = \begin{pmatrix} 10 \times 9 + 10 \times 1 \\ 10 \times 1 + 10 \times 9 \end{pmatrix} = \begin{pmatrix} 100 \\ 100 \end{pmatrix}$
- $\mu_\alpha(D) = \begin{pmatrix} \sum_C \mu_\alpha(C)\psi_{C,D}(C, 0) \\ \sum_C \mu_\alpha(C)\psi_{C,D}(C, 1) \end{pmatrix} = \begin{pmatrix} 100 \times 9 + 100 \times 1 \\ 100 \times 1 + 100 \times 9 \end{pmatrix} = \begin{pmatrix} 1000 \\ 1000 \end{pmatrix}$
- $\mu_\beta(D) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ (initialization, optional),

- $\mu_\beta(C) = \begin{pmatrix} \sum_D \mu_\beta(D)\psi_{C,D}(0,D) \\ \sum_D \mu_\beta(D)\psi_{C,D}(1,D) \end{pmatrix} = \begin{pmatrix} 1\times 9 + 1\times 1 \\ 1\times 1 + 1\times 9 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \end{pmatrix}$
- $\mu_\beta(B) = \begin{pmatrix} \sum_C \mu_\beta(C)\psi_{B,C}(0,C) \\ \sum_C \mu_\beta(C)\psi_{B,C}(1,C) \end{pmatrix} = \begin{pmatrix} 10\times 9 + 10\times 1 \\ 10\times 1 + 10\times 9 \end{pmatrix} = \begin{pmatrix} 100 \\ 100 \end{pmatrix}$
- $\mu_\beta(A) = \begin{pmatrix} \sum_B \mu_\beta(B)\psi_{A,B}(0,B) \\ \sum_B \mu_\beta(B)\psi_{A,B}(1,B) \end{pmatrix} = \begin{pmatrix} 100\times 9 + 100\times 1 \\ 100\times 1 + 100\times 9 \end{pmatrix} = \begin{pmatrix} 1000 \\ 1000 \end{pmatrix}$

Then we compute the normalization factor Z at any point, for example B:

$$Z = \sum_B \mu_\alpha(B).\mu_\beta(B) = 2000$$

Finally we can compute the marginal distributions requested:

$$p(A) = \frac{1}{Z}.\mu_\alpha(A).\mu_\beta(A) = \frac{1}{2000}\begin{pmatrix} 1\times 1000 \\ 1\times 1000 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}$$

$$p(C) = \frac{1}{Z}.\mu_\alpha(C).\mu_\beta(C) = \frac{1}{2000}\begin{pmatrix} 100\times 10 \\ 100\times 10 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}$$

The assumptions were only that neighboring nodes should be equal. The marginal on A and C both say that we have no idea on their value. That was to be expected.

d) Now, we observe that D is 1, recompute the distributions over A and C: $p(A \mid [D = 1])$ and $p(C \mid [D = 1])$.

We've learned that we could compute marginal distributions by decomposing the inference into messages to be passed between nodes.

How can we adapt this mecanism to observations?

If the chain contained only C and D, we would have:

$$p(C \mid [D = 1]) = \frac{1}{Z'}\begin{pmatrix} \psi_{C,D}(0,1) \\ \psi_{C,D}(1,1) \end{pmatrix}$$

This can be written in the same message passing form:

$$p(C \mid [D = 1]) = \frac{1}{Z'}\begin{pmatrix} 1 \\ 1 \end{pmatrix} \cdot \begin{pmatrix} \sum_D \mu'_\beta(D)\psi_{C,D}(0,D) \\ \sum_D \mu'_\beta(D)\psi_{C,D}(1,D) \end{pmatrix}$$

with $\mu'_\beta(D) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$.

Actually, you just have to replace the $\mu_*(X)$ with a Dirac in order to account for an observation of the value of X (and recompute the normalization factor):

- $\mu_\alpha(A) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$,

- $\mu_\alpha(B) = \begin{pmatrix} \sum_A \mu_\alpha(A)\psi_{A,B}(A,0) \\ \sum_A \mu_\alpha(A)\psi_{A,B}(A,1) \end{pmatrix} = \begin{pmatrix} 1\times 9 + 1\times 1 \\ 1\times 1 + 1\times 9 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \end{pmatrix}$
- $\mu_\alpha(C) = \begin{pmatrix} \sum_B \mu_\alpha(B)\psi_{B,C}(B,0) \\ \sum_B \mu_\alpha(B)\psi_{B,C}(B,1) \end{pmatrix} = \begin{pmatrix} 10\times 9 + 10\times 1 \\ 10\times 1 + 10\times 9 \end{pmatrix} = \begin{pmatrix} 100 \\ 100 \end{pmatrix}$
- $\mu_\alpha(D) = \begin{pmatrix} \sum_C \mu_\alpha(C)\psi_{C,D}(C,0) \\ \sum_C \mu_\alpha(C)\psi_{C,D}(C,1) \end{pmatrix} = \begin{pmatrix} 100\times 9 + 100\times 1 \\ 100\times 1 + 100\times 9 \end{pmatrix} = \begin{pmatrix} 1000 \\ 1000 \end{pmatrix}$
- $\mu'_\beta(D) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ (observation),
- $\mu'_\beta(C) = \begin{pmatrix} \sum_D \mu'_\beta(D)\psi_{C,D}(0,D) \\ \sum_D \mu'_\beta(D)\psi_{C,D}(1,D) \end{pmatrix} = \begin{pmatrix} 0\times 9 + 1\times 1 \\ 1\times 1 + 0\times 9 \end{pmatrix} = \begin{pmatrix} 1 \\ 9 \end{pmatrix}$
- $\mu'_\beta(B) = \begin{pmatrix} \sum_C \mu'_\beta(C)\psi_{B,C}(0,C) \\ \sum_C \mu'_\beta(C)\psi_{B,C}(1,C) \end{pmatrix} = \begin{pmatrix} 1\times 9 + 9\times 1 \\ 1\times 1 + 9\times 9 \end{pmatrix} = \begin{pmatrix} 18 \\ 82 \end{pmatrix}$
- $\mu'_\beta(A) = \begin{pmatrix} \sum_B \mu'_\beta(B)\psi_{A,B}(0,B) \\ \sum_B \mu'_\beta(B)\psi_{A,B}(1,B) \end{pmatrix} = \begin{pmatrix} 18\times 9 + 82\times 1 \\ 18\times 1 + 82\times 9 \end{pmatrix} = \begin{pmatrix} 244 \\ 756 \end{pmatrix}$

As above, we can also compute $Z' = 1000$ and then:

$$p(A \mid [D=1]) = \frac{1}{Z'}.\mu_\alpha(A).\mu'_\beta(A) = \frac{1}{1000}\begin{pmatrix} 1\times 244 \\ 1\times 756 \end{pmatrix} = \begin{pmatrix} 0.244 \\ 0.756 \end{pmatrix}$$

$$p(C \mid [D=1]) = \frac{1}{Z'}.\mu_\alpha(C).\mu'_\beta(C) = \frac{1}{1000}\begin{pmatrix} 100\times 1 \\ 100\times 9 \end{pmatrix} = \begin{pmatrix} 0.1 \\ 0.9 \end{pmatrix}$$

Now, we know that node D equals 1 and we see it has become more probable for A and C to be equal to 1 (the more for C which is nearer D than A). At least the result makes sense.

e) Compute $p(C \mid [A=0],[D=1])$.

With the same way, we can recompute μ'_α (which is symmetric to μ_β):

- $\mu'_\alpha(A) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$,
- $\mu'_\alpha(B) = \begin{pmatrix} \sum_A \mu'_\alpha(A)\psi_{A,B}(A,0) \\ \sum_A \mu'_\alpha(A)\psi_{A,B}(A,1) \end{pmatrix} = \begin{pmatrix} 1\times 9 + 0\times 1 \\ 1\times 1 + 0\times 9 \end{pmatrix} = \begin{pmatrix} 9 \\ 1 \end{pmatrix}$
- $\mu'_\alpha(C) = \begin{pmatrix} \sum_B \mu'_\alpha(B)\psi_{B,C}(B,0) \\ \sum_B \mu'_\alpha(B)\psi_{B,C}(B,1) \end{pmatrix} = \begin{pmatrix} 9\times 9 + 1\times 1 \\ 9\times 1 + 1\times 9 \end{pmatrix} = \begin{pmatrix} 82 \\ 18 \end{pmatrix}$
- $\mu'_\alpha(D) = \begin{pmatrix} \sum_C \mu'_\alpha(C)\psi_{C,D}(C,0) \\ \sum_C \mu'_\alpha(C)\psi_{C,D}(C,1) \end{pmatrix} = \begin{pmatrix} 82\times 9 + 18\times 1 \\ 82\times 1 + 18\times 9 \end{pmatrix} = \begin{pmatrix} 756 \\ 244 \end{pmatrix}$

Now $Z'' = 244$ and:

$$p(C \mid [D=1]) = \frac{1}{Z''}.\mu'_\alpha(C).\mu'_\beta(C) = \frac{1}{244}\begin{pmatrix} 82\times 1 \\ 18\times 9 \end{pmatrix} \approx \begin{pmatrix} 0.336 \\ 0.664 \end{pmatrix}$$

It would be the reverse for B: $\begin{pmatrix} 0.664 \\ 0.336 \end{pmatrix}$. It is not exactly $\frac{2}{3}$. Actually, with a longer chain, both μ'_α and μ'_β would (exponentially) converge to uniforms as we consider node further from their origin. Therefore for a long chain, the probability will come from $\begin{pmatrix} 100\% \\ 0\% \end{pmatrix}$ to rest at the uniform $\begin{pmatrix} 50\% \\ 50\% \end{pmatrix}$ before setting to $\begin{pmatrix} 0\% \\ 100\% \end{pmatrix}$. In order to "straighten" the values, we could lower the probability of being different from neighboring nodes.

Note that the known nodes are at the boundary of our chain. If it was not the case, the d-separation property would have allowed us to split the chain in two independent subchains having both a copy of the observed variable as the new boundary.

The next exercise class will take place on **May 27th, 2016**.

For downloads of slides and of homework assignments and for further information on the course see

`https://vision.in.tum.de/teaching/ss2016/mlcv16`
