Gradient Descent Optimization

Function to maximize:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} P(S|h)P(h) \tag{1}$$

Then: Naive Bayes:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} P(h) \prod_{d=1}^{D} P(s_d|h) \tag{2}$$

Then: since h is composed by K-zones:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} \prod_{k=1}^{K} \prod_{d=1}^{|S_t|} P(s_d|h_k)P(h_k) \tag{3}$$

where S_k is the sub-set of sites that belongs into the layout zone k.
From here we work under the constrain of a single main zone defined by its corners $(\boldsymbol{u}, \boldsymbol{b})$
Then: $P(h_k)$ is divided in its components:

$$\hat{h} = \arg\max_{h \in \mathcal{H}} \prod_{k=1}^{K} \prod_{d=1}^{|S_k|} P(s_d|(\boldsymbol{u}_k, \boldsymbol{b}_k))P(\boldsymbol{u}_k, \boldsymbol{b}_k) \tag{4}$$

Then: applying log:

$$\begin{aligned} \log \hat{h} &= \arg\max_{h \in \mathcal{H}} \log \prod_{k=1}^{K} \prod_{d=1}^{|S_k|} P(s_d|(\boldsymbol{u}_k, \boldsymbol{b}_k))P(\boldsymbol{u}_k, \boldsymbol{b}_k) \\ &= \arg\max_{h \in \mathcal{H}} \sum_{k=1}^{K} \sum_{d=1}^{|S_k|} \log P(s_d|(\boldsymbol{u}_k, \boldsymbol{b}_k)) + \log P(\boldsymbol{u}_k, \boldsymbol{b}_k) \end{aligned} \tag{5}$$

Now we want to search for the best $(\boldsymbol{u}_k^{i+1}, \boldsymbol{b}_k^{i+1})$ after some $(\boldsymbol{u}_k^{i}, \boldsymbol{b}_k^{i})$, in order to do that we can use gradient descent optimization over Eq. 5.

First for $\boldsymbol{u}_k$:

$$\begin{aligned}
\boldsymbol{u}_k^{i+1} &= \boldsymbol{u}_k^i - \alpha \frac{\delta h}{\delta \boldsymbol{u}_k^i} \\
&= \boldsymbol{u}_k^i - \alpha \frac{\delta}{\delta \boldsymbol{u}_k^i} \sum_{k=1}^{K} \sum_{d=1}^{|S_k|} \log P(s_d|(\boldsymbol{u}_k, \boldsymbol{b}_k)) + \log P(\boldsymbol{u}_k, \boldsymbol{b}_k) \\
&= \boldsymbol{u}_k^i - \alpha \sum_{k=1}^{K} \sum_{d=1}^{|S_k|} \left(\overbrace{\frac{\delta}{\delta \boldsymbol{u}_k^i} \log P(s_d|(\boldsymbol{u}_k, \boldsymbol{b}_k))}^{\gamma} + \underbrace{\frac{\delta}{\delta \boldsymbol{u}_k^i} \log P(\boldsymbol{u}_k, \boldsymbol{b}_k)}_{\beta} \right) \qquad (6)
\end{aligned}$$

Now, we can take γ and β separately:
For β:

$$\begin{aligned}
\beta &= \frac{\delta}{\delta \boldsymbol{u}_k^i} \log P(\boldsymbol{u}_k^i) P(\boldsymbol{b}_k^i) \\
&= \frac{\delta}{\delta \boldsymbol{u}_k^i} \log \sum_{g=1}^{G_{\boldsymbol{u}_k^i}} \phi_g \mathcal{N}(\boldsymbol{u}_k^i, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g) + \cancelto{0}{\frac{\delta}{\delta \boldsymbol{u}_k^i} \log \sum_{g=1}^{G_{\boldsymbol{b}_k^i}} \phi_g \mathcal{N}(\boldsymbol{b}_k^i, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)}
\end{aligned}$$

Using: $\frac{\delta \log f(x)}{\delta x} = \frac{1}{f(x)} \frac{\delta f(x)}{\delta x}$

$$= \frac{1}{\underbrace{\sum_{g=1}^{G_{\boldsymbol{u}_k^i}} \phi_g \mathcal{N}(\boldsymbol{u}_k^i, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)}_{D}} \frac{\delta}{\delta \boldsymbol{u}_k^i} \sum_{g=1}^{G_{\boldsymbol{u}_k^i}} \phi_g \mathcal{N}(\boldsymbol{u}_k^i, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$$

Using: (where GMM is defined)

$$= D \sum_{g=1}^{G_{\boldsymbol{u}_k^i}} \phi_g (2\pi)^{G_{\boldsymbol{u}_k^i}/2} |\boldsymbol{\Sigma}_g|^{-1/2} \frac{\delta}{\delta \boldsymbol{u}_k^i} \exp^{-1/2(\boldsymbol{u}_k^i - \boldsymbol{\mu}_g)^T \boldsymbol{\Sigma}_g^{-1} (\boldsymbol{u}_k^i - \boldsymbol{\mu}_g)}$$

Using: $\frac{\delta \exp^{f(x)}}{\delta x} = \exp^{f(x)} \frac{\delta f(x)}{\delta x}$

$$= D \sum_{g=1}^{G_{\boldsymbol{u}_k^i}} \phi_g \underbrace{(2\pi)^{G_{\boldsymbol{u}_k^i}/2} |\boldsymbol{\Sigma}_g|^{-1/2} \exp^{-1/2(\boldsymbol{u}_k^i - \boldsymbol{\mu}_g)^T \boldsymbol{\Sigma}_g^{-1} (\boldsymbol{u}_k^i - \boldsymbol{\mu}_g)}}_{\mathcal{N}(\boldsymbol{u}_k^i, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)} \frac{\delta}{\delta \boldsymbol{u}_k^i} \left(-1/2(\boldsymbol{u}_k^i - \boldsymbol{\mu}_g)^T \boldsymbol{\Sigma}_g^{-1} (\boldsymbol{u}_k^i - \boldsymbol{\mu}_g) \right)$$

Using: Lemma 6.2.3 for symmetric $\boldsymbol{\Sigma}_g^{-1}$ [1]

$$= D \sum_{g=1}^{G_{\boldsymbol{u}_k^i}} \phi_g \mathcal{N}(\boldsymbol{u}_k^i, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)(\boldsymbol{u}_k^i - \boldsymbol{\mu}_g)^T \boldsymbol{\Sigma}_g^{-1}$$

$$= \frac{\sum_{g=1}^{G_{\boldsymbol{u}_k^i}} \phi_g \mathcal{N}(\boldsymbol{u}_k^i, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)(\boldsymbol{u}_k^i - \boldsymbol{\mu}_g)^T \boldsymbol{\Sigma}_g^{-1}}{\sum_{g=1}^{G_{\boldsymbol{u}_k^i}} \phi_g \mathcal{N}(\boldsymbol{u}_k^i, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)} \tag{7}$$

Now, for γ; due nature γ an analytic solution is very hard to obtain, so instead a geometric approach is followed.
Using one dimensional Five-Points Stencil [1] method on each axis we can obtain the first derivative on the point $(\boldsymbol{u}_k^i, \boldsymbol{b}_k^i)$:

$$\begin{aligned}\gamma_r = & \frac{-\log P(s_d|((r_{\boldsymbol{u}_k^i} + 2\Delta r, c_{\boldsymbol{u}_k^i}), \boldsymbol{b}_k^i)}{12\Delta r} \\ & + \frac{8\log P(s_d|((r_{\boldsymbol{u}_k^i} + \Delta r, c_{\boldsymbol{u}_k^i}), \boldsymbol{b}_k^i)}{12\Delta r} \\ & - \frac{8\log P(s_d|((r_{\boldsymbol{u}_k^i} - \Delta r, c_{\boldsymbol{u}_k^i}), \boldsymbol{b}_k^i)}{12\Delta r} \\ & + \frac{\log P(s_d|((r_{\boldsymbol{u}_k^i} - 2\Delta r, c_{\boldsymbol{u}_k^i}), \boldsymbol{b}_k^i)}{12\Delta r}\end{aligned} \tag{8}$$

$$\begin{aligned}\gamma_c = & \frac{-\log P(s_d|((r_{\boldsymbol{u}_k^i}, c_{\boldsymbol{u}_k^i} + 2\Delta c), \boldsymbol{b}_k^i)}{12\Delta r} \\ & + \frac{8\log P(s_d|((r_{\boldsymbol{u}_k^i}, c_{\boldsymbol{u}_k^i} + \Delta c), \boldsymbol{b}_k^i)}{12\Delta r} \\ & - \frac{8\log P(s_d|((r_{\boldsymbol{u}_k^i}, c_{\boldsymbol{u}_k^i} - \Delta c), \boldsymbol{b}_k^i}{12\Delta r} \\ & + \frac{\log P(s_d|((r_{\boldsymbol{u}_k^i}, c_{\boldsymbol{u}_k^i} - 2\Delta c), \boldsymbol{b}_k^i}{12\Delta r}\end{aligned} \tag{9}$$

A centered-second order approach can be used as well with minimum accuracy loss

on boundaries, a typical approach is to interpolate past the last point to use the same stencil or switch to one-sided stencils

[1] $f'(x) \approx \frac{-f(x+2h)+8f(x+h)-8f(x-h)+f(x-2h)}{12h}$; h = space between points in the grid

1 Algorithms (Draft)

Data: a set of images $\mathcal{X}$, Prob site model $\mathcal{P}$, Prob layout Model $\mathcal{Q}$
Result: Main paragraph corner coordinates $\mathcal{V}$
for $x \in \mathcal{X}$ **do**
    initialize paragraph corners:
    set user.feedback = $[\boldsymbol{u}_k^0, \boldsymbol{b}_k^0]$ = random;
    while *user.feedback != OK* **do**
        predictLayout($\boldsymbol{u}_k^0, \boldsymbol{b}_k^0, \mathcal{P}, \mathcal{Q}$);
        decodeUserFeedbak();
    end
    $\mathcal{V} = [\boldsymbol{u}_k^{best}, \boldsymbol{b}_k^{best}]$;
end

Algorithm 1: Main Algorithm

Data: Initial point $\boldsymbol{u}_k^0, \boldsymbol{b}_k^0$, Prob site model $\mathcal{P}$, Prob layout Model $\mathcal{Q}$
Result: Main paragraph corner coordinates $[\boldsymbol{u}_k^{best}, \boldsymbol{b}_k^{best}]$
set deltaLogProb = inf;
set bestLogProb = 0;
set thrLogProb = θ;
while *deltaLogProb > thrLogProb* **do**

$$[\boldsymbol{u}_k^{i+1}, \boldsymbol{b}_k^{i+1}] = [\boldsymbol{u}_k^i, \boldsymbol{b}_k^i] - \alpha \left(\underbrace{\frac{\delta \mathcal{P}}{\delta(\boldsymbol{u}_k^i, \boldsymbol{b}_k^i)}}_{\text{Eq. 8 and 9}} + \underbrace{\frac{\delta \mathcal{Q}}{\delta(\boldsymbol{u}_k^i, \boldsymbol{b}_k^i)}}_{\text{Eq. 7}} \right)$$

$$\text{logProb} = \underbrace{(\mathcal{S}|h) * P(h)}_{Eq.5}$$

    if *logProb > bestLogProb* **then**
        deltaLogProb = logProb - bestLogProb;
        bestLogProb = logProb;
        $[\boldsymbol{u}_k^{best}, \boldsymbol{b}_k^{best}] = [\boldsymbol{u}_k^{i+1}, \boldsymbol{b}_k^{i+1}]$
    end
end

Algorithm 2: PredictLayout Algorithm

References

[1] Luo, Y. Local Gradient Descent Methods for GMM Simplification. Tech. rep., 2015.