Background
Supervised learning: given training data \((x,y) \in S\), learn parameters \(w\) to derive prediction rule \(y_w(x)\) that minimizes the risk.

- Maximum A-Posteriori (MAP) prediction:
  \[ y_w(x) = \arg \max_{y \in Y} \theta(y; x, w) \]
- Random MAP predictor:
  \[ p[y|x] = P_{\gamma \sim q_\theta}[y = y_\theta(x)] \]

- Bayesian risk:
  \[ R(w, x, y) = \sum_i P[y_i|x_i]L(y_i, y) \]
  \[ R(w) = E_{(x,y) \sim \mathcal{D}}[R(w, x, y)] \]
  \[ R_S(w) = \frac{1}{|S|} \sum_{(x,y) \in S} R(w, x, y) \]

Priors set regularizations
The Complexity of the bound (regularization) is determined by its prior distribution:
Let \(q_\theta(y) = q_\gamma(y|m)\) then
\[ KL(q_\theta || p) = -H(q_\theta) - E_{\gamma \sim q_\theta} \log(\theta + w) \]
For Gaussian prior
\[ \nabla w K_L(q_\theta || p) = w \]
Proof: Change variable \(\gamma = \gamma - w\)

PAC-Bayesian generalization
For any \(\delta\) and \(\lambda > 0\), with probability at least \(1-\delta\) over the draw of the training set, the following holds simultaneously for all \(w\):
\[ R(w) \leq \frac{1}{1-\exp(-\lambda)} \left( R_S(w) + KL(q_\theta || p) + \log(1/\delta) \right) \]
while \(KL(q_\theta || p) = \int q_\gamma(\gamma) \log(q_\gamma(\gamma)/p(\gamma))\)

Learning efficient posterior

Learn supermodular MAP predictors
\[ \theta(y; x, w) = \sum_i \theta_i(y_i; x, w) + \sum_{ij \in E} \theta_{ij}(y_i, y_j; x, w) \]
\[ \theta_{ij}(y_i, y_j; x, w) = w_{ij} y_i y_j \quad w_{ij} \geq 0 \]
Multiplicative posteriors result in log-barrier functions over the parameters: For any probability distribution \(q(y)\) over the nonnegative reals, let \(q(y) = q(y|w)w\)
\[ KL(q_\theta || p) = -H(q_\theta) - \log w - E_{\gamma \sim q_\gamma} \log(p(\gamma)) \]
Proof: Change of variable
\[ -H(q_\theta) = \int q_\gamma(\gamma) \log(q_\gamma(\gamma)) d\gamma - H(q_\gamma) - \log w \]
For Gaussian prior:
\[ E_{q_\gamma} \log(p(\gamma)) = \frac{-1}{2} w^2 + c \]
For exponential prior:
\[ E_{q_\gamma} \log(p(\gamma)) = -w \]

Learning with approximate MAP prediction
\[ b^* \in \arg \max \beta_i \theta_i(y_i; x, w) \]
\[ s.t. \; b_i(y_i) \geq 0, \sum_i b_i(y) = 1, \; \sum_i b_i(y_i) = b_i(y) \quad \forall y_i, \forall \gamma \in \mathbb{R} \]
Any optimal solution \(b^*\) is described by \(y_\theta(x) = \arg \max \gamma \{ b_\gamma(x) \in \mathbb{R} \} \)
Proof: Any feasible solution that has the same support as \(b^*\) is also optimal solution.
Follows from Lagrange optimality conditions
\[ \sum_{y} \beta_i(y_i; x, w) + \sum_{y \in C} \lambda_{y \in C}(y) - \sum_{y \notin C} \lambda_{y \in C}(y) \]

Empirical Evaluation
Learning supermodular segmentations with non-decomposable loss functions (Grabcut)

### Table 1: Learning the Grabcut segmentations using two different loss functions. Our learned parameters outperform structured SVM approaches and Perturb-and-MAP moment matching.

<table>
<thead>
<tr>
<th>Method</th>
<th>Ground truth</th>
<th>Upper bound</th>
<th>Lower bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structured SVM (hamming loss)</td>
<td>7.87%</td>
<td>6.29%</td>
<td>5.97%</td>
</tr>
<tr>
<td>Structured SVM (all-zero loss)</td>
<td>7.87%</td>
<td>5.97%</td>
<td>5.63%</td>
</tr>
<tr>
<td>GMMRF (Blake et al. [1])</td>
<td>7.87%</td>
<td>5.97%</td>
<td>5.63%</td>
</tr>
<tr>
<td>Perturb-and-MAP (17)</td>
<td>8.19%</td>
<td>5.97%</td>
<td>5.76%</td>
</tr>
<tr>
<td><strong>Results on the Grabcut dataset (Blake et al., ECCV 04)</strong></td>
<td><strong>5.29%</strong></td>
<td><strong>5.29%</strong></td>
<td><strong>5.29%</strong></td>
</tr>
</tbody>
</table>

### Measuring segmentation loss
Given a segmentation \(A[i] \in \{0,1\} \) foreground)

\[ \text{GrabcutLoss}(A,B) = \sum_{i \in A} \text{I}(A[i] \neq B[i]) \]
\[ \text{PASCAL Loss}(A,B) = 1 - \sum_{i \in A} \text{I}(A[i] \neq B[i]) \]

### Figures

- Figure 1: Two examples of image segmentation.
- Table 1: Learning the Grabcut segmentations using two different loss functions.