Section 5.

1. Reshape the original image ($m \times n \times d$) into a line vecter ($1 \times mnd$).
   Use eigenvector to repesent each image. e.g. SVD

2. $y_n, n=1, \dots, N$ is the label of a sample, i.e. identity and age. of the person

3. Suppose we are training a linear regession model for people's age. $y_n = \theta^T x_n$,
   The loss function is mean square loss $L(\theta) = \frac{1}{N}(y - X\theta^T)(y - X\theta^T)$.
   We use gradient descent to do optimization with learning rate(γ), $\nabla_\theta L = \frac{1}{N}(-2y^T X + 2\theta^T X^T X)$
   then $\theta \Leftarrow \theta - \gamma(\frac{2}{N}\theta^T X^T X - \frac{2}{N} y^T X)$
   Iterate the gradient descent untill the loss is less than a pre-defined threshold.
   After training, we can use the model $y = f(x; \theta)$ to predict the age of a person.
   i.e. Input an image of a person's face x, then output the age of the person y.
   Then we use K-Means to predict people's identity. Given initial centers, we use EM to update centers untill no updates in the end. After training. we can input an image into K-Means and classified the image to its nearst center. i.e. the vector repesenting a person.

4. 1) Loss function: $L = \alpha L_{ID} + (1-\alpha) L_{age}$, where $\alpha \in [0,1]$

   2) Yes. Because in the loss function, we combine the L_{ID} to repesent the contribution of identity loss. which will make our model considering the identity when predicting people's age. Our training set also includes pictures of the same person in different ages and the same age of different persons. which implies the identity will benefit predicting people's age.

   3) Yes. In this task, the training set will be (x_i, y_i), where x_i is the image of a face. and $y \in \{-1, 1\}$ repesents if the person has mustache.
   We know can only use one model, such as logistic regession, to predict the outcome. If $y \cdot (\theta^T x) > 0$ then the person has mustache, otherwise, the person has no mustache.

5. 4). The dataset contains only a few samples, which will gererate poor performance no matter what the model is.
Another reason will be K is a large number, even though the dataset is sufficient, some groups will still only contains a few samples.

5) $L=\frac{1}{N}\cdot\sum_{k=1}^{K}\frac{1}{N_k}l_k$, where N_k represents the number of samples in group k.
l_k means square loss in the group k.
If N_k is small, i.e. the worst case, then $\frac{1}{N_k}l_k$ will greater than it in a proper case.
We penalize more on worst cases, some of them are outliers, which will cause negative impact on our model when predicting non outliers.