Exercise 1: Euclidean distance classifier (10 points). The discriminant for class k is $g_k(\mathbf{x}) = \log p(\mathbf{x}|C_k) + \log p(C_k)$, with the classification rule $k^* = \arg\max_{k=1,\dots,K} g_k(\mathbf{x})$. We can ignore the $p(C_k)$ term since it is independent of k (since $p(C_k) = \frac{1}{K}$), so it doesn't change the ranking. For a Gaussian classifier:

$$\log p(\mathbf{x}|C_k) = -\frac{1}{2}\log\left|2\pi\sigma^2\mathbf{I}\right| - \frac{1}{2}\left\|\frac{\mathbf{x}-\boldsymbol{\mu}_k}{\sigma}\right\|^2$$

where, again, we can drop the first term and the factor $\frac{1}{2\sigma^2}$, and change the sign and use $\arg\min$ instead. This yields the rule $k^* = \arg\min_{k=1,\dots,K} \|\mathbf{x}-\boldsymbol{\mu}_k\|^2$, which is equivalent to the Euclidean distance classifier. Expanding the squared norm we obtain $\|\mathbf{x}-\boldsymbol{\mu}_k\|^2 = \|\mathbf{x}\|^2 + \|\boldsymbol{\mu}_k\|^2 - 2\boldsymbol{\mu}_k^T\mathbf{x}$. Again, ignoring the term $\|\mathbf{x}\|^2$ (since it is independent of k), we achieve a linear discriminant $g_k(\mathbf{x}) = \|\boldsymbol{\mu}_k\|^2 - 2\boldsymbol{\mu}_k^T\mathbf{x}$.

Exercise 2: bias and variance of an estimator (20 points). Note: as samples are drawn iid from a Gaussian distribution $\mathcal{N}(\mu,\sigma^2)$, expected value of each sample is $\mathrm{E}_{x_i}\{x_i\} = \mu$, variance of each sample $\mathrm{var}_{x_i}\{x_i\} = \sigma^2$ and second moment $\mathrm{E}_{x_i}\{x_i^2\} = \sigma^2 + \mu^2$.

1. To compute biases $b_\mu(\phi_i)$, let us compute $\mathrm{E}_{\mathcal{X}}\{\phi(\mathcal{X}_i)\}$ first.

$$\mathrm{E}_{\mathcal{X}}\{\phi_4(\mathcal{X})\} = \mathrm{E}_{\mathcal{X}}\{x_1x_2\} = \int x_1x_2\,p(x_1,x_2,\dots,x_N)\,dx_1\dots dx_N \overset{\text{iid}}{=} \int x_1x_2\,p(x_1)p(x_2)\dots p(x_N)\,dx_1\dots dx_N$$
$$\overset{\text{(a)}}{=} \underbrace{\int x_1p(x_1)\,dx_1}_{\mathrm{E}_{x_1}\{x_1\}}\underbrace{\int x_2p(x_2)\,dx_2}_{\mathrm{E}_{x_2}\{x_2\}}\underbrace{\int p(x_3)\,dx_3}_{1}\dots\underbrace{\int p(x_N)\,dx_N}_{1} = \mathrm{E}_{x_1}\{x_1\}\,\mathrm{E}_{x_2}\{x_2\} = \mu^2$$

$$\mathrm{E}_{\mathcal{X}}\{\phi_3(\mathcal{X})\} = \mathrm{E}_{\mathcal{X}}\left\{\frac{1}{N}\sum_{n=1}^N x_n\right\} \overset{\text{(b)}}{=} \frac{1}{N}\sum_{n=1}^N \mathrm{E}_{\mathcal{X}}\{x_n\} = \frac{1}{N}\sum_{n=1}^N \int x_np(x_1,x_2,\dots,x_N)\,dx_1\dots dx_N$$
$$\overset{\text{iid, (a)}}{=} \frac{1}{N}\sum_{n=1}^N \underbrace{\int x_np(x_n)\,dx_n}_{\mathrm{E}_{x_n}\{x_n\}}\prod_{i=1,i\neq n}^N\underbrace{\int p(x_i)\,dx_i}_{1} = \frac{1}{N}\sum_{n=1}^N \mathrm{E}_{x_n}\{x_n\} = \mu$$

$\mathrm{E}_{\mathcal{X}}\{\phi_2(\mathcal{X})\}$ is particualr case of $\mathrm{E}_{\mathcal{X}}\{\phi_3(\mathcal{X})\}$ with $N=1 \implies \mathrm{E}_{\mathcal{X}}\{\phi_2(\mathcal{X})\} = \mu$
$\mathrm{E}_{\mathcal{X}}\{\phi_1(\mathcal{X})\} = \mathrm{E}_{\mathcal{X}}\{7\} = 7$

Here in (a) we re-arrange similar terms in integration and in (b) use linearity of expectation. Then, biases are $b_\mu(\phi_1) = 7-\mu$, $b_\mu(\phi_2) = 0$, $b_\mu(\phi_3) = 0$, $b_\mu(\phi_4) = \mu^2 - \mu$. Only ϕ_2 and ϕ_3 are unbiased estimators of μ.

2. To compute variances, we use identity of $\mathrm{var}\{\phi_i(\mathcal{X})\} = \mathrm{E}_{\mathcal{X}}\{\phi_i^2(\mathcal{X})\} - (\mathrm{E}_{\mathcal{X}}\{\phi_i(\mathcal{X})\})^2$ and note that we already computed the second term. Thus we start by computing $\mathrm{E}_{\mathcal{X}}\{\phi_i^2(\mathcal{X})\}$:

$$\mathrm{E}_{\mathcal{X}}\left\{\phi_4^2(\mathcal{X})\right\} = \mathrm{E}_{\mathcal{X}}\left\{x_1^2x_2^2\right\} = \int x_1^2x_2^2\,p(x_1,x_2,\dots,x_N)\,dx_1\dots dx_N \overset{\text{iid}}{=} \int x_1^2x_2^2\,p(x_1)p(x_2)\dots p(x_N)\,dx_1\dots dx_N$$
$$\overset{\text{(a)}}{=} \underbrace{\int x_1^2p(x_1)\,dx_1}_{\mathrm{E}_{x_1}\{x_1^2\}}\underbrace{\int x_2^2p(x_2)\,dx_2}_{\mathrm{E}_{x_2}\{x_2^2\}}\underbrace{\int p(x_3)\,dx_3}_{1}\dots\underbrace{\int p(x_N)\,dx_N}_{1} = (\mu^2+\sigma^2)^2$$

$$\mathrm{E}_{\mathcal{X}}\left\{\phi_2^2(\mathcal{X})\right\} = \mathrm{E}_{\mathcal{X}}\left\{x_1^2\right\} = \int x_1^2\, p(x_1, x_2, \ldots, x_N)\, dx_1 \ldots dx_N \overset{\text{iid}}{=} \int x_1^2\, p(x_1)p(x_2)\ldots p(x_N)\, dx_1 \ldots dx_N$$

$$\overset{\text{(a)}}{=} \underbrace{\int x_1^2 p(x_1)\, dx_1}_{\mathrm{E}_{x_1}\left\{x_1^2\right\}} \underbrace{\int p(x_2)\, dx_2}_{1} \underbrace{\int p(x_3)\, dx_3}_{1} \ldots \underbrace{\int p(x_N)\, dx_N}_{1} = \mu^2 + \sigma^2$$

$$\mathrm{E}_{\mathcal{X}}\left\{\phi_3^2(\mathcal{X})\right\} = \mathrm{E}_{\mathcal{X}}\left\{\left(\frac{1}{N}\sum_{i=1}^N x_i\right)^2\right\} = \mathrm{E}_{\mathcal{X}}\left\{\frac{1}{N^2}\sum_{i=1}^N x_i^2 + \frac{1}{N^2}\sum_{i=1}^N \sum_{j=1,\, j\neq i}^N x_i x_j\right\}$$

$$\overset{\text{(b)}}{=} \frac{1}{N^2}\sum_{i=1}^N \underbrace{\mathrm{E}_{\mathcal{X}}\left\{x_i^2\right\}}_{\text{(c): } \mu^2+\sigma^2} + \frac{1}{N^2}\sum_{i=1}^N \sum_{j=1,\, j\neq i}^N \underbrace{\mathrm{E}_{\mathcal{X}}\left\{x_i x_j\right\}}_{\text{(d): } \mu^2} = \frac{N}{N^2}(\mu^2+\sigma^2) + \frac{N^2-N}{N^2}\mu^2 = \frac{\sigma^2}{N} + \mu^2$$

$$\mathrm{E}_{\mathcal{X}}\left\{\phi_1^2(\mathcal{X})\right\} = \mathrm{E}_{\mathcal{X}}\left\{7^2\right\} = 49$$

Here in (a) we re-arrange similar terms; in (b) we use linearity of expectation; in (c) and (d) we rely on previously computed quantities that has similar form. Now, variances are:

$$\begin{aligned}
&\mathrm{var}_{\mathcal{X}}\left\{\phi_1(\mathcal{X})\right\} = 49 - 49 = 0 && \text{(consistent)}\\
&\mathrm{var}_{\mathcal{X}}\left\{\phi_2(\mathcal{X})\right\} = \mu^2 + \sigma^2 - \mu^2 = \sigma^2 && \text{(inconsistent)}\\
&\mathrm{var}_{\mathcal{X}}\left\{\phi_3(\mathcal{X})\right\} = \frac{\sigma^2}{N} + \mu^2 - \mu^2 = \frac{\sigma^2}{N} && \text{(consistent)}\\
&\mathrm{var}_{\mathcal{X}}\left\{\phi_3(\mathcal{X})\right\} = (\mu^2+\sigma^2)^2 - \mu^4 = \sigma^4 + 2\mu^2\sigma^2 && \text{(inconsistent)}
\end{aligned}$$

3. To compute the mean square error we use bias-variance decomposition of $e(\phi, \mu) = \mathrm{var}\left\{\phi\right\} + b_\mu^2(\phi)$:

$$\begin{aligned}
e(\phi_1, \mu) &= 7^2 + 0 = 49\\
e(\phi_2, \mu) &= 0^2 + \sigma^2 = \sigma^2\\
e(\phi_3, \mu) &= 0^2 + \frac{\sigma^2}{N} = \frac{\sigma^2}{N}\\
e(\phi_4, \mu) &= \mu^4 + \sigma^4 + 2\mu^2\sigma^2
\end{aligned}$$

While estimators ϕ_2 and ϕ_3 are unbiased, only ϕ_3 is consistent. On the other hand, we showed that estimators ϕ_3 and ϕ_4 are consistent, however ϕ_4 is very simplistic estimation of the mean by a constant, thus biased. In general being just unbiased or consistent is not enough, and we need to look at mean squared error.

Exercise 3: PCA and LDA (30 points). Try it in Matlab!

1. $\boldsymbol{\mu} = \sum_{k=1}^K \pi_k \boldsymbol{\mu}_k = \binom{\mu/2}{0}$, $\boldsymbol{\Sigma} = \sum_{k=1}^K \pi_k \left(\boldsymbol{\Sigma}_k + \boldsymbol{\mu}_k\boldsymbol{\mu}_k^T\right) - \boldsymbol{\mu}\boldsymbol{\mu}^T = \begin{pmatrix} \sigma_1^2 + \frac{1}{4}\mu^2 & 0 \\ 0 & \sigma_2^2 \end{pmatrix}$. So the mean is half way between the individual means, and the covariance is diagonal. The variance along x_1 is $\sigma_1^2 + \frac{1}{4}\mu^2$ (given by the original variances along x_1 and by the horizontal separation between the Gaussians). The variance along x_2 is given by the original variance along x_2.

2. Eigenvalues $\sigma_1^2 + \frac{1}{4}\mu^2$ and σ_2^2, both positive, associated with eigenvectors $\binom{1}{0}, \binom{0}{1}$, respectively.

3. The PCA projection is along $\begin{cases} \text{vector } \binom{1}{0}, & \text{if } \sigma_1^2 + \frac{1}{4}\mu^2 > \sigma_2^2 \\ \text{any vector}, & \text{if } \sigma_1^2 + \frac{1}{4}\mu^2 = \sigma_2^2 \\ \text{vector } \binom{0}{1}, & \text{if } \sigma_1^2 + \frac{1}{4}\mu^2 < \sigma_2^2. \end{cases}$

4. $\mathbf{S}_k = N_k\boldsymbol{\Sigma}_k \Rightarrow \mathbf{S}_W = \sum_{k=1}^K \mathbf{S}_k = \sum_{k=1}^K N_k\boldsymbol{\Sigma}_k = N\begin{pmatrix}\sigma_1^2 & 0\\ 0 & \sigma_2^2\end{pmatrix}$.
   $\mathbf{S}_B = \sum_{k=1}^K N_k(\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T = \frac{N}{4}\begin{pmatrix}\mu^2 & 0\\ 0 & 0\end{pmatrix}$.

5. $\mathbf{S}_W^{-1}\mathbf{S}_B = \frac{1}{4}\begin{pmatrix}\sigma_1^2 & 0\\ 0 & \sigma_2^2\end{pmatrix}^{-1}\begin{pmatrix}\mu^2 & 0\\ 0 & 0\end{pmatrix} = \frac{1}{4}\begin{pmatrix}(\mu/\sigma_1)^2 & 0\\ 0 & 0\end{pmatrix}$, which has eigenvalues $\nu_1 = \frac{1}{4}\left(\frac{\mu}{\sigma_1}\right)^2$ and $\nu_2 = 0$, assoc. with eigenvectors $\mathbf{v}_1 = \begin{pmatrix}1\\0\end{pmatrix}$, $\mathbf{v}_2 = \begin{pmatrix}0\\1\end{pmatrix}$, resp. So $\nu_2 = 0$ always, because $\text{rank}\,(\mathbf{S}_B) = 1 = K - 1$.

6. LDA projects along $\mathbf{v}_1 = \begin{pmatrix}1\\0\end{pmatrix}$. This is the direction that separates the classes best for this data.

7. PCA finds the same projection as LDA if $\sigma_1^2 + \frac{1}{4}\mu^2 > \sigma_2^2$. This happens when the variance (of the whole data) along $\begin{pmatrix}1\\0\end{pmatrix}$ is larger than along $\begin{pmatrix}0\\1\end{pmatrix}$.

Exercise 4: variations of k-means clustering (30 points).

- **Variation 1:** the objective function is a constrained optimization problem:

$$E(\{\boldsymbol{\mu}_k\}_{k=1}^K, \mathbf{Z}) = \sum_{n=1}^N\sum_{k=1}^K z_{nk}\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2 \quad \text{s.t.} \begin{cases}\mathbf{Z} \in \{0,1\}^{NK},\ \mathbf{Z}\,\mathbf{1} = \mathbf{1},\\ \boldsymbol{\mu}_k \in \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}\ \forall k = 1, \ldots, K.\end{cases}$$

  1. As with k-means, we can apply alternating optimization. The assignment step, over $\mathbf{Z}$ given $\{\boldsymbol{\mu}_k\}_{k=1}^K$, is as in k-means: for each $n = 1, \ldots, N$, assign $\mathbf{x}_n$ to the cluster whose centroid is closest to $\mathbf{x}_n$. The centroid step, over $\{\boldsymbol{\mu}_k\}_{k=1}^K$ given $\mathbf{Z}$, changes: for each $k = 1, \ldots, K$, $\boldsymbol{\mu}_k = \mathbf{x}_{n^*}$ where $n^* = \arg\min_{n:\ z_{nk}=1}\sum_{n:\ z_{nk}=1}\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$, i.e., we find the data point $\mathbf{x}_{n^*}$ in cluster k that has lowest sum of squared distances to all the points in that cluster.
     This is called the *k-medoids* algorithm. Try it in Matlab!
  2. With binary vectors $\mathbf{x} \in \{0,1\}^D$, the mean vector of a cluster will consist of real values between 0 and 1, which may not be meaningful. Example: let $\mathbf{x}$ represent a document (defined over a D-word dictionary) by having element x_d indicate whether word i appears in the document. In general, k-means will not be appropriate with variables that are not continuous. k-medoids guarantees that the centroids are valid patterns by forcing them to be data points. And, with an obvious change to the algorithm, we can use other definitions of distance $d(\mathbf{x}, \mathbf{y})$ between points rather than just the squared Euclidean distance $\|\mathbf{x} - \mathbf{y}\|^2$.

- **Variation 2:**

  1. The objective function is:

$$E(\{\mathbf{w}_k, w_{k0}\}_{k=1}^K, \mathbf{Z}) = \sum_{n=1}^N\sum_{k=1}^K z_{nk}(y_n - \mathbf{w}_k^T\mathbf{x}_n - w_{k0})^2 \quad \text{s.t.} \quad \mathbf{Z} \in \{0,1\}^{NK},\ \mathbf{Z}\,\mathbf{1} = \mathbf{1}.$$

  2. We apply alternating optimization again:
     - *Assignment step* (over $\mathbf{Z}$ given $\{\mathbf{w}_k, w_{k0}\}_{k=1}^K$):
       for each $n = 1, \ldots, N$, assign point n to the line that predicts it with the lowest squared error $(y_n - \mathbf{w}_k^T\mathbf{x}_n - w_{k0})^2$.
     - *Line step* (over $\{\mathbf{w}_k, w_{k0}\}_{k=1}^K$ given $\mathbf{Z}$):
       for each $k = 1, \ldots, K$, minimize the following to obtain the parameters of line k:

$$\min_{\mathbf{w}_k, w_{k0}}\sum_{n=1}^N z_{nk}(y_n - \mathbf{w}_k^T\mathbf{x}_n - w_{k0})^2 \Leftrightarrow \min_{\mathbf{w}_k, w_{k0}}\sum_{n:\ z_{nk}=1}(y_n - \mathbf{w}_k^T\mathbf{x}_n - w_{k0})^2$$

       which is equivalent to doing a least-squares linear regression on the points currently in cluster k, i.e., we fit line k to the points currently in cluster k.

  This is called the *k-lines* algorithm. Try it in Matlab!

Exercise 5: mean-shift algorithm (10 points). Computing the gradient of p wrt $\mathbf{x}$ and equating it to zero:

$$\nabla p(\mathbf{x}) = \frac{1}{N(2\pi\sigma^2)^{D/2}} \sum_{n=1}^{N} \nabla e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_n}{\sigma}\right\|^2} = \frac{1}{N(2\pi\sigma^2)^{D/2}} \sum_{n=1}^{N} e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_n}{\sigma}\right\|^2} \left(-\frac{1}{\sigma^2}(\mathbf{x}-\mathbf{x}_n)\right) =$$

$$-\frac{1}{\sigma^2 N(2\pi\sigma^2)^{D/2}} \left(\mathbf{x}\sum_{n=1}^{N} e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_n}{\sigma}\right\|^2} - \sum_{n=1}^{N} e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_n}{\sigma}\right\|^2}\mathbf{x}_n\right) = \mathbf{0}$$

$$\Rightarrow \mathbf{x} = \frac{\sum_{n=1}^{N} e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_n}{\sigma}\right\|^2}\mathbf{x}_n}{\sum_{n=1}^{N} e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_n}{\sigma}\right\|^2}} = \sum_{n=1}^{N} p(n|\mathbf{x})\mathbf{x}_n.$$

Bonus exercise: nonparametric regression (20 points).

1. If $N = 1$ then $\mathbf{g}(\mathbf{x}) = \mathbf{y}_1$, i.e., a constant function passing through $\mathbf{y}_1$. A least-squares linear regression is not defined with only one point because there is an infinite number of possible lines passing through it. Generally speaking, nonparametric methods produce meaningful models with even one data point and require no training procedure, while parametric methods need a certain number of data points to fit the parameters and need a training procedure.

2. If $N = 2$ then:

$$\mathbf{g}(\mathbf{x}) = \frac{e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_1}{\sigma}\right\|^2}}{e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_1}{\sigma}\right\|^2} + e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_2}{\sigma}\right\|^2}}\mathbf{y}_1 + \frac{e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_2}{\sigma}\right\|^2}}{e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_1}{\sigma}\right\|^2} + e^{-\frac{1}{2}\left\|\frac{\mathbf{x}-\mathbf{x}_2}{\sigma}\right\|^2}}\mathbf{y}_2 = \alpha(\mathbf{x})\mathbf{y}_1 + (1-\alpha(\mathbf{x}))\mathbf{y}_2$$

   where $\alpha(\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x} + w_0)$, $\mathbf{w} = \frac{1}{\sigma^2}(\mathbf{x}_1 - \mathbf{x}_2)$ and $w_0 = \frac{1}{2\sigma^2}\left(\|\mathbf{x}_2\|^2 - \|\mathbf{x}_1\|^2\right) = \frac{1}{\sigma^2}(\mathbf{x}_2 - \mathbf{x}_1)^T\left(\frac{\mathbf{x}_1+\mathbf{x}_2}{2}\right) = -\mathbf{w}^T\left(\frac{\mathbf{x}_1+\mathbf{x}_2}{2}\right)$. A least-squares regression is defined but only if both $\mathbf{x}_n, \mathbf{y}_n \in \mathbb{R}$, in which case it is a line going through (x_1, y_1) and (x_2, y_2).
   The right plot shows the kernel smoother using bandwidths $\sigma = 0.1,\ 0.5, 1$, and the least-squares regression line (dashed black line), for two data points $(1, 0.5)$ and $(2, 1)$.