**Second assignment in Machine learning 1 – 2023 – Paper 1**

## 1 Naive Bayes Modeling of Climate (Recommended timeline: September 25)

Naive Bayes (NB) is a particular form of classification that makes strong independence assumptions regarding the features of the data, conditional on the classes (see Bishop section 4.2.3). Specifically, NB assumes each feature is independent given the class label. In contrast, when we looked at probabilistic generative models for classification in the lecture, we used a full-covariance Gaussian to model data from each class, which incorporates correlation between all the input features (i.e. they are not conditionally independent).

If correlated features are treated independently, the evidence for a class will be overcounted. However, Naive Bayes is very simple to construct, because by ignoring correlations the class-conditional likelihood, $p(\mathbf{x}|\mathcal{C}_k)$, is a product of $D$ univariate distributions, each of which is simple to learn:

$$p(\mathbf{x}|\mathcal{C}_k) = \prod_{d=1}^{D} p(x_d|\mathcal{C}_k)$$

Consider the task of classifying the climate of a region into one of $K$ categories $\mathcal{C}_k$ depending on their temperatures across the year. For example, if the temperature is very cold all year round, the region likely has a continental climate, while if it is always warm it is more likely that the region is situated in an arid climate following the well-known 'Köppen climate classification' scheme. Even though the scheme uses temperature, humidity, and precipitation for the classification you want to find out if regions can be classified accurately only based on temperature.

For this experiment, you take $D$ temperature measurements at times equally distributed across one year for $N$ regions. Thus, you obtain for each of the $N$ regions a vector $\mathbf{x}_n$ of dimension $D$, where $x_{nd}$ is the temperature in region $n$ at time $d$. Overall, your training set consists of an $N \times D$ matrix. The target matrix $\mathbf{T}$, whose rows consist of the row vectors $\mathbf{t}_n^T = (t_{n1}, \dots, t_{nK})$, one-hot-encoded such that $\mathbf{t}_n^T = (0, \dots, 1, \dots, 0)$ with the scalar 1 at position $k$ if $n \in \mathcal{C}_k$.

Assume we know $p(\mathcal{C}_k) = \pi_k$ (with the constraint $\sum_{k=1}^{K} \pi_k = 1$). We can model temperature in a region at a given time with different distributions. For this question, we will use an exponential model. Thus, the temperature in a region at a given time is modeled with one parameter $\lambda_{dk}$, when conditioned on class $\mathcal{C}_k$:

$$p(\mathbf{x}|\mathcal{C}_k, \lambda_{1k}, \dots, \lambda_{Dk}) = \prod_{d=1}^{D} \lambda_{dk} e^{-\lambda_{dk} \cdot x_d}.$$

With this information answer the following questions:

($a$) Write down the data likelihood,

$$p(\mathbf{T}, \mathbf{X}|\mathbf{\Lambda})$$

without the Naive Bayes conditional independence assumption at first (you can still assume i.i.d.). Then, derive the data likelihood for the general $K$ classes naive Bayes classifier, stating where you make use of the product rule and the naive Bayes assumption.

You should write the likelihood in terms of $p(x_{nd} \mid \mathcal{C}_k)$, meaning you should not assume the explicit exponential distribution. [1 point]

($b$) How does the number of parameters change if you make use of the naive Bayes assumption? Why is the assumption called naive and can you think of an example in which this assumption does not hold? [1 point]

($c$) Write down the data log-likelihood $\ln p(\mathbf{T}, \mathbf{X} \mid \mathbf{\Lambda})$ for the given likelihood model. [1 point]

($d$) Solve for the MLE estimator of $\lambda_{dk}$. Express in your own words how the result can be interpreted. [1 point]

($e$) Specify $p(\mathcal{C}_1|\mathbf{x})$ for the general k classes naive Bayes classifier in terms of the prior and class conditional distributions. [1 point]

Specify your final answer in terms of $\pi$ and $\lambda$.

($f$) Explain why NB is a generative model using the answer from question 1.e. [1 point]

($g$) Write $p(\mathcal{C}_1|\mathbf{x})$ for the exponential model explicitly in terms of the modeling choices (exponential distribution, and $\pi_k$). [1 point]

($h$) What condition defines the region for which $\mathbf{x}$ is predicted to be in class $C_1$ compared to all other classes? [1 point]

($i$) Write down the condition found on the previous question 1g as an inequality in terms of the posterior from the previous question 1h. Subsequently, specify the written inequality as a linear inequality of the form: $\mathbf{x}^T\mathbf{a} > c$. [1 point]

($j$) Recall that a region is convex if for any points $x_0$ and $x_1$ that belong to $C_1$, all the points in the straight line defined as $x_0 * (1 - \lambda) + \lambda x_1$, with $\lambda \in (0, 1)$ will belong to $C_1$ as well.

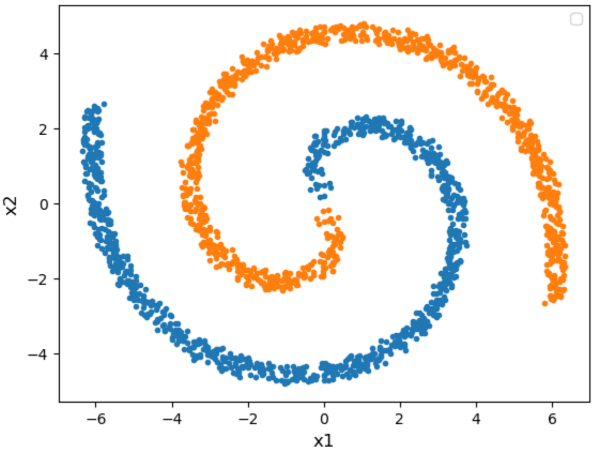
Given this definition, do you expect the decision regions for this particular problem to be convex? Why/why not?

Hint: Note that your input vectors are temperature measurements. [1 point]

($k$) Use your result from question 1i to show that the region where $\mathbf{x}$ is predicted to be in $\mathcal{C}_1$ convex. Explain why? [1 point]

## 2 Binary classification (Recommended timeline: September 29)

Now, suppose you are given the following datasets shown in the figure below



$(a)$ For each of these following methods, explain why they can/cannot classify of the dataset. [2 points]

$(i)$ Logistic regression with linear features, $y = \sigma(\mathbf{w}^T\mathbf{x})$.

$(ii)$ Logistic regression with non linear basis functions, $y = \sigma(\mathbf{w}^T\boldsymbol{\phi})$

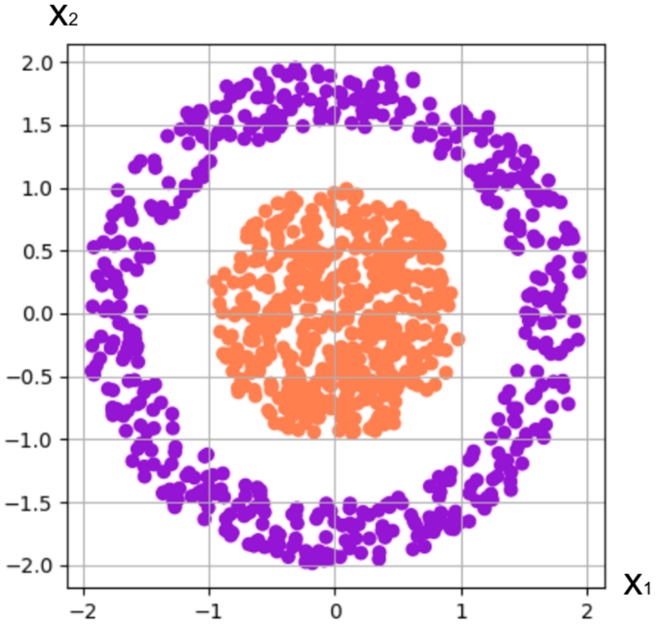
$(iii)$ Multilayer Perceptron with 1 hidden layer

$(b)$ Can the above dataset be classified using Naive Bayes? Why/why not? Your reasoning should involve thinking about how $p(\mathcal{C}_k|\mathbf{x})$ would look like for each class. [1 point]

$(c)$ Now consider the following dataset.Can it be classified using Naive Bayes? Why/why not? How is your reasoning different from the one you provided in question $2b$?

Hint: Can you think of a distribution that fits this data? (you can assume $x_1$ and $x_2$ to be continuous)

[1 point]



($d$) What is the main difference between logistic regression with non linear basis functions $\boldsymbol{\phi}$ and multilayer perceptrons, in terms of $\boldsymbol{\phi}$? [1 point]

## 3 Regularized Logistic Regression (Recommended timeline: October 4)

Consider logistic regression for $K$ classes with $N$ training vectors $\{\mathbf{x}_n\}_{n=1}^N$, each of which is mapped to a different feature vector

$$\boldsymbol{\phi}_n = \boldsymbol{\phi}(\mathbf{x_n}) = (\phi_0(\mathbf{x}_n), \phi_1(\mathbf{x}_n), \ldots, \phi_{M-1}(\mathbf{x}_n))^T$$

using basis functions $\phi_j(\mathbf{x}_n)$ with $j = 1, \ldots, M-1$, and $\phi_0(\mathbf{x}_n) = 1$. Each vector $\mathbf{x}_n$ has a corresponding target vector $\mathbf{t}_n$ of size $K$: $\mathbf{t}_n = (t_{n1}, t_{n2}, \ldots, t_{nK})^T$, where $t_{nk} = 1$ if $\mathbf{x}_n \in \mathcal{C}_k$, and $t_{nj} = 0$ for all $j \neq k$. The input data can be collected in a matrix $\mathbf{X}$ such that the $n$-th row is given by $\mathbf{x}_n^T$ and the targets can be collected in a target matrix $\mathbf{T}$, such that the $n$-th row is given by $\mathbf{t}_n^T$.

$$\mathbf{X} = \begin{pmatrix} —\mathbf{x}_1^T— \\ \vdots \\ —\mathbf{x}_N^T— \end{pmatrix}, \ \mathbf{T} = \begin{pmatrix} —\mathbf{t}_1^T— \\ \vdots \\ —\mathbf{t}_N^T— \end{pmatrix}$$

The feature vectors can also be collected in a matrix $\mathbf{\Phi}$ such that the $n$-th row of $\mathbf{\Phi}$ contains $\phi_n^T$:

$$\mathbf{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}$$

Assuming i.i.d. data, the posterior class probabilities are modeled by

$$p(\mathcal{C}_k | \boldsymbol{\phi}(\mathrm{x}), \boldsymbol{w}_1, \ldots, \boldsymbol{w}_K) = y_k(\boldsymbol{\phi}) = \frac{\exp(a_k)}{\sum_{j=1}^K \exp(a_j)}$$

where the "activations" $a_k$ are given by $a_k = \mathbf{w}_k^T \phi$. Assume a Gaussian prior on the parameter vectors $\mathbf{w}_1, \ldots, \mathbf{w}_K$:

$$p(\mathbf{w}_1, \ldots, \mathbf{w}_K | \alpha) = \prod_{k=1}^K \mathcal{N}(\mathbf{w}_k | \mathbf{0}, \alpha^{-1}\mathbf{I})$$

($a$) Write down the likelihood $p(\mathbf{T}|\mathbf{\Phi}, \mathbf{w}_1, \ldots, \mathbf{w}_K)$ as a product over $N$ and $K$. Use the entries of $\mathbf{T}$ as selectors of the correct class. Then write down the log-likelihood $\log p(\mathbf{T}|\mathbf{\Phi}, \mathbf{w}_1, \ldots, \mathbf{w}_K)$. [1 point]

($b$) Write down the explicit form of the prior $p(\mathbf{w}_1, \ldots, \mathbf{w}_K|\alpha)$. Compute the logarithm of the prior $\log p(\mathbf{w}_1, \ldots, \mathbf{w}_K|\alpha)$. Why do we compute this logarithm? [1 point]

($c$) Write down an expression for the posterior

$$p(\mathbf{w}_1, \ldots, \mathbf{w}_K|\boldsymbol{\Phi}, \mathbf{T}, \alpha)$$

over $\mathbf{w}_1, \ldots, \mathbf{w}_K$ by applying Bayes rule. [1 point]

($d$) Show that obtaining a Maximum A Posteriori (MAP) estimate for $\mathbf{w}_1, \ldots, \mathbf{w}_K$ is equivalent to performing regularized logistic regression for $K$ classes, where we minimize the function

$$-\sum_{n=1}^{N}\sum_{k=1}^{K} t_{nk} \log y_k(\boldsymbol{\phi}_n) + \frac{\alpha}{2}\sum_{k=1}^{K}\sum_{m=0}^{M-1} |w_{km}|^2$$

with respect to $\mathbf{w}_1, \ldots, \mathbf{w}_K$. [2 points]

($e$) Why do we regularize our solution and how does it affect the weights in this case? Explain in your own words. [1 point]

($f$) Assuming a non-informative prior what does the MAP solution look like, and how can you interpret the solution? [1 point]

($g$) Derive the gradient of the log-likelihood $\nabla_{\mathbf{w}_j} \log p(\mathbf{T}|\boldsymbol{\Phi}, \mathbf{w}_1, \ldots, \mathbf{w}_K)$. Your final answer should be a single sum over $n$. You can make use of the derivative of the softmax function $\frac{\partial y_k}{\partial a_j} = y_k(\mathbb{I}_{kj} - y_j)$ (you computed this for the first assignment). [1 point]

($h$) Derive the gradient of the log-prior $\nabla_{\mathbf{w}_j} \log p(\mathbf{w}_1, \ldots, \mathbf{w}_K|\alpha)$. Derive the gradient of the log-posterior $\nabla_{\mathbf{w}_j} \log p(\mathbf{w}_1, \ldots, \mathbf{w}_K|\mathbf{T}, \boldsymbol{\Phi}, \alpha)$. [1 point]