Machine Learning 1
WS19/20 5 March 2020

Gedächtnisprotokoll

First exam session, duration: 120 minutes

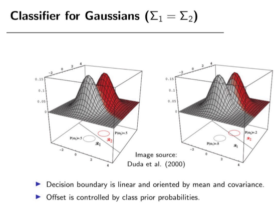
Exercise 1 - multiple choice (20 pts)

Only 1 answer is correct

$\Sigma_1 \neq \Sigma_2 \Rightarrow$ decision boundary is quadric

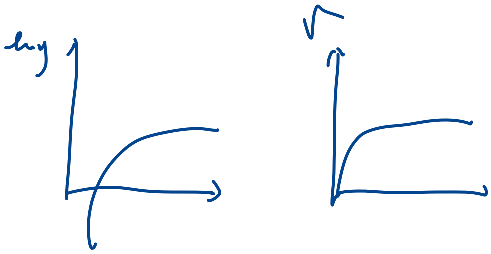
1. Given two normal distributions $p(x|w_1) \sim \mathcal{N}(\mu_1, \Sigma_1)$ and $p(x|w_2) \sim \mathcal{N}(\mu_2, \Sigma_2)$ what is a *necessary* and *sufficient* condition for the optimal decision boundary to be linear? (5pts)

   (a) $\Sigma_1 = \Sigma_2$
   (b) $\Sigma_1 = \Sigma_2, P(w_1) = P(w_2)$
   (c) ...
   (d) ...



2. We have a classifier that decides the class $\mathrm{argmax}_{w_i} f_i(x)$ for the input x. What is a suitable discriminant functions f_i? (5pts)

   (a) $\sqrt{p(x|w_i)P(w_i)}$
   (b) $\log(p(x|w_i) + P(w_i))$
   (c) ...
   (d) ... $\underline{\log P(x|W_i) + \log P(W_i)}$



3. K-means is (5pts)

   (a) a non-convex algorithm used to cluster data
   (b) a kernelized version of the means algorithm
   (c) ...
   (d) ...

4. Error backpropagation gives (5pts)

   (a) the gradient of the error function
   (b) the optimal direction in parameter space
   (c) ...
   (d) ...

Exercise 2 - Neural Networks (15pts)

1. Given $x \in \mathbb{R}^2$ implement the function $1_{\{|x_1|+|x_2| \geq 2\}}$ using the following activation function: $1_{\{a_i w_{ij} + b_j \geq 0\}}$. Where $1_{\{...\}}$ is the indicator function. Draw the NN and provide weights and biases. Use only 5 neurons (excluding the input neurons) (10pts)

2. State how many neurons are need to implement $1_{\{|x_1|+...|x_d| \geq d\}}$ for $x \in \mathbb{R}^d$. Provide weights and bias for a neuron of your choice (5pts).

Exercise 3 - Lagrange (25pts)

Let $A \in \mathbb{R}^{d \times d}, B \in \mathbb{R}^{h \times h}$ be two positive definite matrices

$$\max_{w,v} w^\top A w + v^\top B v \quad \text{subject to} \qquad \|w\|^2 + \|v\|^2 = 1$$

1. Write the lagrangian (5pts)

2. Derive equations that lead to the solution (5pts)

3. Show that the problem is equivalent to an eigenvector problem of a matrix $C \in \mathbb{R}^{(d+h)\times(d+h)}$ (5pts)

4. Show that the solution is the eigenvector corresponding to the largest eigenvalue (5pts)

5. Show how the solution for C can be derived from two subproblems for A and B. *Hint:* the set of eigenvalues of a block diagonal matrix is the union of the eigenvalues of the matrices on the diagonal (5pts)

Exercise 4 - Kernels (20pts)

A positive definite kernel satisfies

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j k(x_i, x_j) \geq 0$$

for all $x_1, \ldots, x_n \in \mathbb{R}^d$ and $c_i, ..., c_n \in \mathbb{R}$

1. Show that $k(x, x') = \langle x, x' \rangle$ is a PD kernel (5pts)

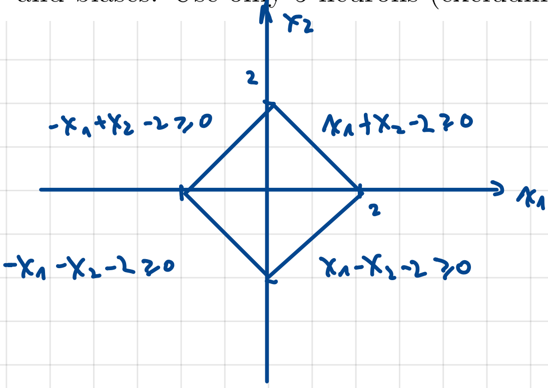
2. Show that $k(x, x') = \langle x, x' + 2 \rangle$ is *not* a PD kernel (add 2 to each component of x')(5pts)

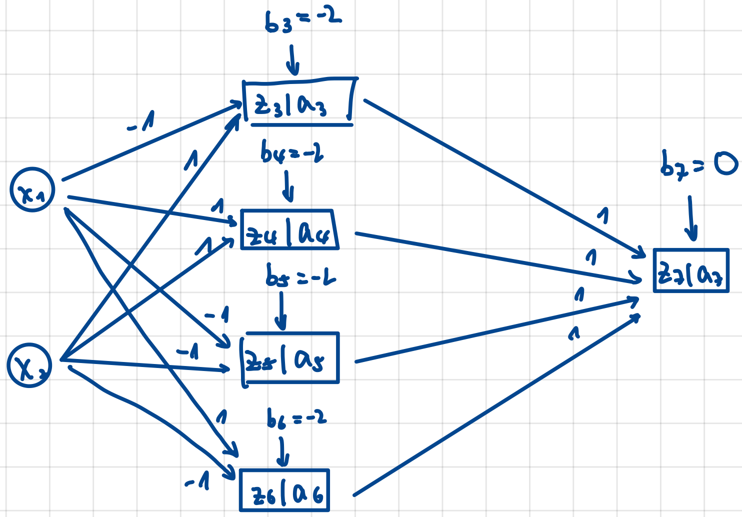
3. Show that $g(x, x') = k(\xi, x) k(x, x') k(x', \xi)$ is a PD kernel, for any $\xi \in \mathbb{R}^d$ and a PD kernel k with feature map $\phi : \mathbb{R}^d \mapsto \mathbb{R}^h$, i.e., $k(x, x') = \langle \phi(x), \phi(x') \rangle$ (5pts)

4. Give a feature map ψ for g (5pts)

Exercise 2 - Neural Networks (15pts)

1. Given $x \in \mathbb{R}^2$ implement the function $1_{\{|x_1|+|x_2|\geq 2\}}$ using the following activation function: $1_{\{a_i w_{ij}+b_j \geq 0\}}$. Where $1_{\{\dots\}}$ is the indicator function. Draw the NN and provide weights and biases. Use only 5 neurons (excluding the input neurons) (10pts)





2. State how many neurons are need to implement $1_{\{|x_1|+\dots|x_d|\geq d\}}$ for $x \in \mathbb{R}^d$. Provide weights and bias for a neuron of your choice (5pts).

$2^d + 1$

Exercise 3 - Lagrange (25pts)

Let $A \in \mathbb{R}^{d\times d}, B \in \mathbb{R}^{h\times h}$ be two positive definite matrices

$$\max_{w,v} w^\top Aw + v^\top Bv \quad \text{subject to} \quad \|w\|^2 + \|v\|^2 = 1$$

1. Write the lagrangian (5pts)

$$L(w,v,\lambda) = w^T A w + v^T B v - \lambda(\|w\|^2 + \|v\|^2 - 1)$$

2. Derive equations that lead to the solution (5pts)

$$\frac{\partial L}{\partial w} = 2Aw - 2\lambda w = 0 \qquad Aw = \lambda w$$

$$\frac{\partial L}{\partial v} = 2Bv - 2\lambda v = 0 \qquad Bv = \lambda v$$

$$\frac{\partial L}{\partial \lambda} = 0 \Rightarrow \qquad \|w\|^2 + \|v\|^2 = 1$$

3. Show that the problem is equivalent to an eigenvector problem of a matrix $C \in \mathbb{R}^{(d+h)\times(d+h)}$ (5pts)

$Aw = \lambda w$
$(d\times d)$

$Bv = \lambda v$
$(h\times h)$

$$\Rightarrow \begin{array}{cc} \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} & \begin{matrix} \}d \\ \}h \end{matrix} \\ \begin{matrix} \underbrace{\quad}_{d} & \underbrace{\quad}_{h} \end{matrix} & \end{array} \begin{bmatrix} w \\ v \end{bmatrix} = \begin{bmatrix} \lambda \\ \lambda \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix}$$

$$\Rightarrow C = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix}$$ and $\begin{bmatrix} w \\ v \end{bmatrix}$ is the eigenvalue of C

4. Show that the solution is the eigenvector corresponding to the largest eigenvalue (5pts)

$$\begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix} = \lambda \begin{bmatrix} w \\ v \end{bmatrix} \qquad \left([w^T \ v^T] \begin{bmatrix} w \\ v \end{bmatrix} = \|w\|^2 + \|v\|^2 = 1 \right)$$

$$\lambda = [w^T v^T] C \begin{bmatrix} w \\ v \end{bmatrix}$$

$$= [\underset{d}{w^T} \ \underset{h}{v^T}] \begin{bmatrix} \overset{d\times d}{A} & 0 \\ 0 & B \end{bmatrix} \begin{bmatrix} w \\ v \end{bmatrix}$$

$$= [\underset{d}{w^T A} \quad \underset{h}{v^T B}] \begin{bmatrix} w \\ v \end{bmatrix} = w^T A w + v^T B v$$

$$\Rightarrow \max w^T A w + v^T B v = \max \lambda$$

5. Show how the solution for C can be derived from two subproblems for A and B. *Hint:* the set of eigenvalues of a block diagonal matrix is the union of the eigenvalues of the matrices on the diagonal (5pts)

$A u_i = \lambda_i u_i$

$B v_j = \mu_j v_j$

$\Rightarrow$ eigenvector for C $\quad x = \begin{bmatrix} u_i \\ 0 \end{bmatrix}$ corresponding λ_i

$x = \begin{bmatrix} 0 \\ \mu_j \end{bmatrix} \qquad \mu_j$

Exercise 4 - Kernels (20pts)

A positive definite kernel satisfies

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j k(x_i, x_j) \geq 0$$

for all $x_1, \ldots, x_n \in \mathbb{R}^d$ and $c_i, \ldots, c_n \in \mathbb{R}$

1. Show that $k(x, x') = \langle x, x' \rangle$ is a PD kernel (5pts)

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j k(x_i, x_j) = \sum_{i=1}^{n}\sum_{j=1}^{n} c_i c_j \langle x_i, x_j \rangle$$

$$= \sum_i \sum_j c_i c_j \sum_{k=1}^{d} x_{i,k} x_{j,k}$$

$$= \sum_{k=1}^{d} \sum_i c_i x_{i,k} \sum_j c_j x_{j,k}$$

$$= \sum_{k=1}^{d} \left(\sum_i c_i x_{i,k}\right)^2 \geq 0$$

2. Show that $k(x, x') = \langle x, x' + 2 \rangle$ is *not* a PD kernel (add 2 to each component of x')(5pts)

$$\sum_i \sum_j c_i c_j \langle x_i, x_j + 2 \rangle = \sum_i \sum_j c_i c_j \sum_k^d x_{i,k} (x_{j,k} + 2)$$

$$= \sum_i \sum_j c_i c_j \left(\sum_k^d x_{i,k} x_{j,k} + 2 \sum_k^d x_{i,k} \right)$$

$$= \underbrace{\sum_{k=1}^{d} \left(\sum_i c_i x_{i,k} \right)^2}_{\geq 0} + \underbrace{\sum_i \sum_j c_i c_j \cdot 2 \sum_k^d x_{i,k}}_{\text{may negative}}$$

3. Show that $g(x, x') = k(\xi, x) k(x, x') k(x', \xi)$ is a PD kernel, for any $\xi \in \mathbb{R}^d$ and a PD kernel k with feature map $\phi : \mathbb{R}^d \mapsto \mathbb{R}^h$, i.e., $k(x, x') = \langle \phi(x), \phi(x') \rangle$ (5pts)

$$\sum_i \sum_j c_i c_j \phi^T(\xi) \phi(x) \phi^T(x) \phi(x') \phi^T(x') \phi(\xi)$$

$$= \sum_i c_i \underbrace{\phi^T(\xi) \phi(x)}_{\text{const}} \phi^T(x) \quad \sum_j c_j \phi(\xi) \underbrace{\phi^T(x') \phi(x')}_{\text{const}}$$

$$= \left\langle \sum_i c_i \phi(\xi) \phi^T(x) \phi(x), \sum_j c_j \phi(\xi) \phi^T(x') \phi(x') \right\rangle$$

$$= \left\| \sum_i c_i \phi(\xi) \phi^T(x) \phi(x) \right\|^2 \geq 0$$

$$= \sum_i \sum_j c_i c_j \langle \underbrace{\phi(\xi) \phi^T(x) \phi(x)}_{\psi}, \phi(\xi) \phi^T(x') \phi(x') \rangle$$

4. Give a feature map ψ for g (5pts)

$$\psi = \phi(\xi) \phi^T(x) \phi(x)$$

$[1\ 1]$ $\quad [1\ \ x_1\ \ x_2$

Exercise 5 - implementing RR (20pts)

You will be implementing ridge regression. Assume numpy and scipy are already imported. Fill in the gaps in the following code snippets. Your code must be efficient (e.g. **no loops**)

1. Implement a function that given a $N \times 2$ matrix returns a $N \times 5$ matrix after applying the feature map $\phi(x_1, x_2) = [1, x_1, x_2, x_1^2, x_2^2]$ (5pts)

```
def Phi(X):
    t = np.ones((len(X), 5))
    t[:, 1] = X[:, 0]
    t[:, 2] = X[:, 1]
    t[:, 3] = X[:, 0] ** 2
    t[:, 4] = X[:, 1] ** 2
    return ..t
```

```
x1 = X[:, 0]
x2 = X[:, 1]

return np.concatenate((np.ones((X.shape[0], 1)),
    x1, x2, x1**2, x2**2), axis=1)
```

2. Implement the training part of RR ($\lambda = 0.1$) (5pts), that is

$$\beta = (\underbrace{\phi(X)^{\mathsf{T}}\phi(X)}_{5\times5} + \lambda I)^{-1}\underbrace{\phi(X)^{\mathsf{T}}}_{5\times n}\underbrace{y}_{n} \Rightarrow (5\times n)$$

```
def train(self, Xtrain, Ytrain):
    lambda = 0.1
    phi = Phi(Xtrain)
    I = np.eye((5,5))




    self.beta = np.linalg.inv(phi.T @ Phi + lambda * I) @ Phi.T @ Xtrain
```

3. Implement the prediction part (5pts)

```
def predict(self, Xtest):

    Ftest = Phi(Xtest) @ beta





    return Ftest
```

4. Compute the fraction of samples for which the prediction satisfies $|y - f(x)| < 0.01$ (5pts)

```
def Accuracy(self, Xtest, Ytest):
    Pred = predict(Xtest)
    correct = np.abs(Ytest - Pred) < 0.01

    Acc = np.mean(correct)


    return Acc
```

```
def Accuracy(self, Xtest, Ytest):
    """
    计算预测值与真实值的误差小于 0.01 的样本比例
    """
    Ypred = self.predict(Xtest)  # 获取预测值
    num_correct = np.sum(np.abs(Ypred - Ytest) < 0.01)  # 计算满足条件的样本数
    Acc = num_correct / len(Ytest)  # 计算正确率
    return Acc
```