sf2935 Modern Methods of Statistical Learning: 35 Questions to be Considered for the Exam on Thursday the 11th of January, 2018, 08.00 - 13.00 & Wednesday 4th of Apr, 2018, 14.00-19.00

Presented on 18/12/2017

Timo Koski, Jimmy Olsson & Tetyana Pavlenko,

1 Introduction

This document contains a set of questions/problems on the topics treated in sf2935 Modern Methods of Statistical Learning during the period 2 of 2017. Five of these will be selected to constitute the written exam on Thursday the 11th of January, 2018, 08.00 - 13.00.

The answers/solutions can be produced by a study of the relevant chapters in the course textbook *An introduction to Statistical Learning*, by G. James, D. Witten, T. Hastie, R. Tibshirani. Springer Verlag, 2013, and by a similar study of the lecture slides of lecturers and guests on the webpage `https://www.math.kth.se/matstat/gru/sf2935/statlearnmaterial2017` or on canvas.

In addition, some proficiency in manipulating basic calculus, probability, linear algebra and matrix calculus is required.

This same set of questions (maybe some will be revised/removed and new added) will be valid in the re-exam. **There will be a seminar (TBA) after each exam for presentation by the lecturers of an outline for a good answer/proof/solution for each of the questions in the exam**.

2 The Assignments

1. Suppose that we have a training set consisting of a set of data points $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$ and

   $$y_i = f(\mathbf{x}_i) + \epsilon,$$

   where the noise, ϵ, has zero mean and variance σ^2. Hnece y_is are independent outcomes of Y in

   $$Y = f(\mathbf{x}) + \epsilon.$$

   By means of some learning algorithm, the raining set $\mathcal{D}$ and a model class we have found a function $\hat{f}(\mathbf{x})$ to model the true, but unknown, function $y = f(\mathbf{x})$.

   We decompose the expected prediction error $EPE(\mathbf{x})$ of $\hat{f}$ on another point $\mathbf{x}$ as follows:

   $$EPE(\mathbf{x}) = \mathrm{E}\Big[\big(Y - \hat{f}(\mathbf{x})\big)^2\Big] = \mathrm{Bias}\big[\hat{f}(\mathbf{x})\big]^2 + \mathrm{Var}\big[\hat{f}(\mathbf{x})\big] + \sigma^2. \quad (2.1)$$

Here:

$$\text{Bias}\big[\hat{f}(\mathbf{x})\big] = \text{E}\big[\hat{f}(\mathbf{x})\big] - f(\mathbf{x}) \tag{2.2}$$

and

$$\text{Var}\big[\hat{f}(\mathbf{x})\big] = \text{E}\Big[\big(\hat{f}(\mathbf{x}) - \text{E}[\hat{f}(\mathbf{x})]\big)^2\Big] \tag{2.3}$$

a) Deduce these equations from $\text{E}\Big[\big(Y - \hat{f}(\mathbf{x})\big)^2\Big]$.

b) Explain the *Bias -Variance Trade-Off* with aid of this decomposition.

2. The noise, ϵ, has zero mean and variance σ^2, f is an unknown function, and

$$Y = f(\mathbf{x}) + \epsilon.$$

Suppose that we have a training set consisting of a set of N independent data points $\mathcal{D} = \{(\mathbf{x}_1, y_1) \ldots, (\mathbf{x}_N, y_N)\}$, or, the real values y_i associated with each point $\mathbf{x}_i$ in $\mathbb{R}^d$. We have thus that

$$y_i = f(\mathbf{x}_i) + \epsilon_i,$$

where the noise, ϵ_i, has zero mean and variance σ^2. We choose our learning machine $\hat{f}$ as

$$\hat{f}(\mathbf{x}) = \hat{f}_k(\mathbf{x}),$$

where for $1 \leq k < N$

$$\hat{f}_k(\mathbf{x}) = \frac{1}{k} \sum_{nnk(\mathbf{x})} y_i$$

i.e., we are averaging the outputs y_i in $\mathcal{D}$ over the set $nnk(\mathbf{x})$ consisting of the k nearest neighbors of $\mathbf{x}$ in $\mathcal{D}$. This is called k-nearest neighbor regression.

Let $1 \leq k < N$ and $\mathbf{x}_{(1)} \leq \mathbf{x}_{(2)} \leq \ldots \leq \mathbf{x}_{(k)}$ be an ordering of $\mathbf{x}_1, \ldots, \mathbf{x}_N$ so that

$$||\mathbf{x}_{(1)} - \mathbf{x}||_2 \leq ||\mathbf{x}_{(2)} - \mathbf{x}||_2 \leq \ldots \leq ||\mathbf{x}_{(k)} - \mathbf{x}||_2 \leq \ldots \leq ||\mathbf{x}_{(N)} - \mathbf{x}||_2$$

Here $||\mathbf{x}_{(1)} - \mathbf{x}||_2$ is the L_2 -distance on $\mathbb{R}^d$.

a) Show by means of (2.1) - (2.3) that

$$EPE(\mathbf{x}) = \mathrm{E}\Big[\big(Y - \hat{f}_k(\mathbf{x})\big)^2\Big] = \sigma^2 + \left(f(\mathbf{x}) - \frac{1}{k}\sum_{l=1}^{k} f(\mathbf{x}_{(l)})\right)^2 + \frac{\sigma^2}{k}.$$

b) How does the choice of k influence the *Bias -Variance Trade-Off* of k-nearest neighbor regression in view of this decomposition. How is high dimension d influencing ?

3. Rosenblatt's perceptron algorithm for training of perceptrons is written in a pseudocode as

1: $\mathbf{w}_0 \leftarrow \mathbf{0}$, $b_0 \leftarrow 0$, $k \leftarrow 0$ $\mathcal{R} \leftarrow \max_{1 \le i \le l} \|\mathbf{x}_i\|$
2: **repeat**
3: **for** $i = 1$ to l **do**
4: **if** $y_i \left(\ll \mathbf{w}, \mathbf{x}_i \gg + b_k\right) \le 0$ **then**
5: $\mathbf{w}_{k+1} \leftarrow \mathbf{w}_k + y_i \mathbf{x}_i$,
$b_{k+1} \leftarrow b_k + y_i \mathcal{R}^2$,
$k \leftarrow k + 1$
6: **end if**
7: **end for**
8: **until** no mistakes made in the loop
9: **return** $\mathbf{w}_k, b_k$, where k is the number of mistakes.

a) Explain in plain words what the algorithm does.

b) Formulate the Perceptron Convergence Theorem (Novikoff 1962) and interpret it in terms of the perceptron algorithm and its geometry.

4. Consider the output layer of an artificial neural network.

$$y_j = \sigma\left(\sum_{i=1}^{m} w_{ji} o_i + b_j\right).$$

We have

$$\mathrm{net}_j \stackrel{\mathrm{def}}{=} \sum_{i=1}^{m} w_{ji} o_i + b_j,$$

so that

$$y_j = \sigma(\mathrm{net}_j).$$

The input net_j to the activator $\sigma(u) = 1/(1+e^{-u})$, the sigmoid function, is the weighted sum of outputs o_k of neurons in the previous layer.

We have a set of training data consisting of K inputs and targets t_i. The goal is to tune all the weights and biases to minimize the cost or training function

$$C = \frac{1}{2}\sum_{i=1}^{K}(y_i - t_i)^2,$$

where y_is are the network outputs corresponding to the inputs. We consider here only the output layer and use a gradient descent algorithm. Calculation of the partial derivative of the error C with respect to a weight w_{ji} is done using the chain rule of calculus twice:

$$\frac{\partial C}{\partial w_{ji}} = \frac{\partial C}{\partial y_j}\frac{\partial y_j}{\partial w_{ji}} = \frac{\partial C}{\partial y_j}\frac{\partial y_j}{\partial \text{net}_j}\frac{\partial \text{net}_j}{\partial w_{ji}} \tag{2.4}$$

a) Find

$$\frac{\partial C}{\partial y_j}.$$

b) Find

$$\frac{\partial \text{net}_j}{\partial w_{ji}}.$$

Here you need to pay attention to the expression for the derivative of the sigmoid function $\sigma(u)$.

c) Show finally that

$$\frac{\partial C}{\partial w_{ji}} = (o_j - t_j)\cdot oj(1-o_j)\cdot o_i.$$

5. Explain the concept of separating hyperplane classification. Specifically, consider the case where the feature space $\mathcal{X}$ is p-dimensional and the output space is $\mathcal{Y} = \{-1, 1\}$. Describe how a separating hyperplane is defined and how such a (generic) hyperplane can be used for classification. Define the maximal margin hyperplane and describe in some detail how it can be found for a given training set $\{(x_1, y_1), \ldots, (x_n, y_n)\}$.

6. Consider a p-dimensional feature space $\mathcal{X}$, output space $\mathcal{Y} = \{-1, 1\}$ and a training set $\{(x_1, y_1), \ldots, (x_n, y_n)\}$. Define the support vector classifier; you must explain all the quantities involved. What is the difference between the support vector classifier, i.e. the associated hyperplane, and the maximal margin classifier? What does the term "support vector" refer to and why is this an important concept for this particular classifier? Lastly, describe how you could achieve a non-linear decision boundary using a support vector classifier; give an example. What are potential drawbacks of this approach?

7. Consider a p-dimensional feature space $\mathcal{X}$, output space $\mathcal{Y} = \{-1, 1\}$ and a training set $\{(x_1, y_1), \ldots, (x_n, y_n)\}$. Define the concept of a support vector machine and explain all quantities involved. Give at least two non-trivial examples of various instances of this classifier. Describe the difference between a support vector machine and achieving non-linear decision boundaries using a support vector classifier. Lastly, describe one way a support vector machine can be used in the setting of $K > 2$ classes.

8. Suppose $\mathbf{x}$ comes from one of two populations, Π_1 with $N_p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and Π_2 with $N_p(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$. If the respective density functions are denoted by $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$. Find the expression for the quadratic discriminator Q, where
$$Q = \ln \left[\frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \right].$$
If $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$, verify that Q becomes
$$(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)' \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)' \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2). \tag{2.5}$$

9. Assume that Π_i is $N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma})$, $i = 1, 2$ and for the class prior probabilities π_i it hold that $\pi_1 = \pi_2 = \frac{1}{2}$. Specify the optimal misclassification rate (OMR) for the linear discriminant function. Explain further the relationship between the OMR and the Mahalanobis distance between two populations.

10. Relationship between LDA and QDA: Suppose that observations within each class are drawn from $N_p(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, $i = 1, \ldots, \mathcal{C}$, i.e from $\mathcal{C}$ classes with a class specific mean vector and a class specific covariance matrix.

In general, as the sample size n increases, does one expect the test classification accuracy of QDA relative to LDA to improve, decline or be unchanged. Motivate your answer.

11. Suppose $\mathbf{x}_0$ comes from one of two populations, Π_1 with $N_p(\boldsymbol{\mu}_1, \boldsymbol{\Sigma})$ and Π_2 with $N_p(\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$. Suppose also that you have a training data from both populations. Form the sample based linear discriminant function and specify the classification procedure for assigning $\mathbf{x}_0$ to one of two populations.

    What happens to Fisher's linear discriminant if the sample size n is smaller than the dimensionality p?

12. Explain the idea of k nearest neighbors, kNN, classifier. Derive the classification rule and explain the effect of k choice on the classification accuracy using both test and training errors.

13. The bootstrap approach provides an all-purpose method for computing standard errors. For any estimator $\hat{\theta} = \hat{\theta}(\mathbf{x})$ we can write

$$\mathrm{Var}^*(\hat{\theta}) = \frac{1}{n^n - 1} \sum_{i=1}^{n^n} \left(\hat{\theta}_i^* - \bar{\hat{\theta}}^* \right)^2,$$

    where * denotes a bootstrapped or re-sampled value, $\hat{\theta}_i^*$ is the estimator calculated from the i re-sample and $\sum_{i=1}^{n^n} \hat{\theta}_i^*$ is the mean of the re-sampled values. The variance formula above is applicable to virtually any estimator. But how do we know that it is a good method for evaluation the performance of estimators? Explain why the method is good using the following steps: for a sample $\mathbf{x} = (x_1, \ldots, x_n)$ and an estimate $\hat{\theta} = \hat{\theta}(x_1, \ldots, x_n)$ select b bootstrap samples and calculate

$$\mathrm{Var}_b^*(\hat{\theta}) = \frac{1}{b - 1} \sum_{i=1}^{b} \left(\hat{\theta}_i^* - \bar{\hat{\theta}}^* \right)^2.$$

    Explain why we have the convergence result

$$\mathrm{Var}_b^*(\hat{\theta}) \to \mathrm{Var}^*(\hat{\theta}) \text{ as } b \to \infty.$$

14. $\mathbf{X} = (X_1, \ldots X_d)$ is a discrete d-dimensional random variable with the probability mass function

$$p_{\mathbf{X}}(\mathbf{x}) = \frac{1}{Z} e^{\sum_{i=1}^{d} h_i x_i + \sum_{i=1}^{d} \sum_{j=1}^{d} J_{ij} x_i x_j}$$

Here $\mathbf{x} = (x_1, \ldots, x_d)$ and Z is the normalization constant i.e.,

$$Z = \sum_{\mathbf{x}} e^{\sum_{i=1}^{d} h_i x_i + \sum_{i=1}^{d} \sum_{j=1}^{d} J_{ij} x_i x_j}$$

also known as the partition function and assumed to be finite.
It is valid by assumption that

$$J_{ii} = 0 \quad \text{for all } i, J_{ij} = J_{ji}, \quad \text{for all } i, j. \tag{2.6}$$

Let $\mathbf{X}\backslash_r$ denote the variable $\mathbf{X}$ with the rth component removed, i.e.

$$\mathbf{X}\backslash_r = (X_1, \ldots, X_{r-1}, X_{r+1}, \ldots X_d)$$

In the same way

$$\mathbf{x}\backslash_r = (x_1, \ldots, x_{r-1}, x_{r+1}, \ldots x_d).$$

a) Check that the conditional probability $p_{X_r|\mathbf{X}\backslash_r}(x_r \mid \mathbf{x}\backslash_r)$ is given by

$$p_{X_r|\mathbf{X}\backslash_r}(x_r \mid \mathbf{x}\backslash_r) = \frac{1}{C} e^{h_r x_r + \sum_{j=1}^{d} J_{rj} x_r x_j},$$

where

$$C = \sum_{x_r} e^{h_r x_r + \sum_{j=1}^{d} J_{rj} x_r x_j}$$

Hint: It may useful at some stage to note the identity

$$\sum_{i=1}^{d} \sum_{j=1}^{d} J_{ij} x_i x_j = A + B,$$

where

$$A = \sum_{i=1, i \neq r}^{d} x_i \left[\sum_{j=1; j \neq r}^{d} J_{ij} x_j + J_{ir} x_r \right],$$

and

$$B = x_r \sum_{j=1}^{d} J_{rj} x_j.$$

Here (2.6) can be used effectively.

b) Describe the statistical learning technique connected to $p_{X_r|\mathbf{x}\setminus_r}(x_r \mid \mathbf{x}\setminus_r)$ and its rationale. You can answer this question even if you have not solved part **a)** of this assignment.

15. Let $p(x)$ and $q(x)$ be two probability mass functions on the same discrete data space $\mathcal{X}$. Then the *Kullback* or *Kullback - Leibler divergence* or *distance* between $p(x)$ and $q(x)$ is denoted by $D(p \parallel q)$ and is defined as

$$D(p \parallel q) \stackrel{\text{def}}{=} \sum_{x \in \mathcal{X}} p(x) \ln \frac{p(x)}{q(x)}. \tag{2.7}$$

Here $0/0 = 0$, where we take by continuity $0 \ln 0 = 0$, and $p(x)/0 = +\infty$.

If $p(x)$ and $q(x)$ are two probability density functions on the real line we have

$$D(p \parallel q) \stackrel{\text{def}}{=} \int_{-\infty}^{+\infty} p(x) \ln \frac{p(x)}{q(x)} dx. \tag{2.8}$$

a) It holds in (2.7) and (2.8) that

$$D(p \parallel q) \geq 0.$$

Prove this for either (2.7) or (2.8). *Aid*: Prove $-D(p \parallel q) \leq 0$ by evoking the inequality $\ln x \leq x - 1$ valid for all $x > 0$ (you need not prove this inequality).

b) $D(p \parallel q)$ is a distance only by name, it does not fulfill the axioms of a metric. To consider this, let $\mathcal{X} = \{0, 1\}$ and $0 \leq p \leq 1$ and $0 \leq g \leq 1, p \neq q$. Let $p(x) = p^x \cdot (1-p)^{1-x}$ and $q(x) = q^x \cdot (1-q)^{1-x}$. These are, of course, the two Bernoulli distributions $Be(p)$ and $Be(g)$, respectively.

Then **verify** that

$$D(p \parallel q) = -(1-p) \cdot \log(1-g) - p \cdot \log g - h(p), \tag{2.9}$$

where $h(p)$ is the binary entropy function (2.10) with the natural logarithm

$$h(p) := -p \ln(p) - (1-p) \ln(1-p). \tag{2.10}$$

Note that $h(p) \geq 0$, as is obvious.

Compute $D(q \parallel p)$ and compare to (2.9). Draw a conclusion.

16. We have n independent and identically distributed observations x_i, for simplicity of writing collected in $\mathbf{x} = (x_1, \ldots, x_n)$, of a random variable $X \sim \mathrm{Be}(q)$, where q is unknown.

    We have $q(x) = P(X = x) = q^x \cdot (1-q)^{1-x}$ for $x = 0, 1$, as above.

    Next, the empirical distribution $\widehat{p}(x)$ corresponding to $x = (x_1, \ldots, x_n)$, with the relative frequencies $\widehat{p} = \frac{k}{n}$ and $1 - \widehat{p} = \frac{n-k}{n}$ is

    $$\widehat{p}(x) = \widehat{p}^x \cdot (1-\widehat{p})^{1-x},$$

    where k is the number of ones in $\mathbf{x} = (x_1, \ldots, x_n)$ and $n-k$ is the the number of zeros in $\mathbf{x} = (x_1, \ldots, x_n)$.

    a) Show that the likelihood function for q is

    $$L_{\mathbf{x}}(q) = q^k (1-q)^{n-k}.$$

    b) Show that

    $$D(\widehat{p} \parallel q) = -\frac{1}{n} \ln L_{\mathbf{x}}(q) - h(\widehat{p}).$$

    Aid: Recall (2.9).

    c) Find q^* such that $D(\widehat{p} \parallel q)$ is minimized as function of q. *Aid*: You can find q^* without an operation like differentiation by exploiting the fact that every Kullback distance is non-negative.

    d) Which well-known statistical estimator of q is re-discovered as q^* in **c)**?

17. The output layer of a single layer artificial neural network with the input $\mathbf{x} = (x_1, \ldots, x_m)$ and single output is

    $$y = \sigma\left(\sum_i^m w_i x_i + b\right).$$

    where the activator is $\sigma(u) = 1/(1+e^{-u})$. We set

    $$\mathrm{net}(\mathbf{x}) \stackrel{\mathrm{def}}{=} \sum_{i=1}^m w_i x_i + b,$$

    and

    $$y = \sigma(\mathrm{net}(\mathbf{x})).$$

We have a set of training data $\mathcal{D} = \{(\mathbf{x}_1, t_1), \ldots, \mathbf{x}_K, t_K)\}$ consisting of K inputs $\mathbf{x}_i$ and corresponding targets t_i. Any target t_i satisfies $0 < t_i < 1$.

The goal is to set the weights and the bias to minimize the cost function

$$KL = \sum_{l=1}^{K} D(t_l \parallel y_l).$$

where $y_l = \sigma(\text{net}_l)(= \sigma(\text{net}(\mathbf{x}_l)))$, is the network output corresponding to the input $\mathbf{x}_l$ and $D(t_l \parallel y_l)$ is the Kullback divergence between the Bernoulli distribution $\text{Be}(t_l)$ and the Bernoulli distribution $\text{Be}(y_l))$, c.f., in the preceding.

a) Check that the minimization of KL is equivalent to the minimization of the *cross entropy* (C_e)

$$C_e = -\sum_{l=1}^{K} [t_l \ln y_l + (1 - t_l) \ln(1 - y_l)]$$

b) Check that

$$\frac{\partial C_e}{\partial y_l} = \frac{y_l - t_l}{y_l(1 - y_l)}.$$

c) Set $y_l = \sigma(\text{net}_l)(= \sigma(\text{net}(\mathbf{x}_l)))$. Check that

$$\frac{\partial C_e}{\partial \text{net}_l} = y_l - t_l.$$

Calculation of the partial derivative of the error C_e with respect to net_l is done using the chain rule:

$$\frac{\partial C_e}{\partial \text{net}_l} = \frac{\partial C_e}{\partial y_l} \frac{\partial y_l}{\partial \text{net}_l}. \tag{2.11}$$

Here you need to pay attention to the expression for the derivative of the sigmoid function $\sigma(u)$.

d) Find by the appropriate chain rules and other auxiliaries

$$\frac{\partial C_e}{\partial w_i}.$$

e) What is the minimum value of C_e?

18. Consider the natural exponential family

$$f\left(x \mid \theta\right) = h(x)e^{\theta \cdot x - \psi(\theta)}.$$

Here $\theta \cdot x$ is inner product on $\mathbb{R}^k$.

a) Consider the prior densities given by

$$\pi\left(\theta\right) = \psi\left(\theta|\mu, \lambda\right) = K\left(\mu, \lambda\right) e^{\theta \cdot \mu - \lambda\psi(\theta)}.$$

Find the the posterior as

$$\psi\left(\theta|\mu + x, \lambda + 1\right).$$

Hence the given family of priors is a conjugate family for $f\left(x \mid \theta\right)$. You are permitted to work formally, i.e., not checking the conditions for validity.

b) Let

$$f(x \mid \theta) = \prod_{i=1}^{d} \theta_i^{x_i}(1-\theta_i)^{1-x_i}; x_j \in \{0,1\}; 0 \leq \theta_i \leq 1.$$

$\theta = (\theta_1, \theta_2, \ldots \theta_d)$. Write this in a natural form and find the conjugate family of priors.

19. We have a training set of l pairs $Y \in \{-1, 1\}$ and the corresponding values of labels,

$$\mathcal{S} = \left\{\left(\mathbf{X}_1, y_1\right), \ldots, \left(\mathbf{X}_n, y_n\right)\right\}$$

and have used an approriate learning algorithm on $\mathcal{S}$ to find a rule (ANN, SVM, e.t.c) designated by $\widehat{\boldsymbol{f}} = \widehat{\boldsymbol{f}}(\mathcal{S})$.

We must have another set of data, testing set, of holdout samples, which were not used for training,

$$\mathcal{T} = \left\{\left(\mathbf{X}_1^t, y_1^t\right), \ldots, \left(\mathbf{X}_m^t, y_m^t\right)\right\}$$

Having established $\widehat{\boldsymbol{f}}$ we should apply it on $\mathcal{T}$, and compare the prediction $\widehat{y}_j^t = \widehat{\boldsymbol{f}}(\mathbf{X}_j^t)$ for all j to y_j^t. We have the following errors

- prediction of -1 when the holdout sample has a -1 (True Negatives, the number of which is TN)
- prediction of -1 when the holdout sample has a 1 (False Negatives, the number of which is FN)
- prediction of 1 when the holdout sample has a -1 (False Positives, the number of which is FP)
- prediction of 1 when the holdout sample has a 1 (True Positives, the number of which is TP)

False Positives = FP , True Positives = TP
False Negatives = FN , True Negatives= TN

	$Y=+1$	$Y=-1$
$\widehat{Y}=+1$	TP	FP
$\widehat{Y}=-1$	FN	TN

One often encounters one or several of the following performance measures:

- Accuracy = $\frac{TP+TN}{TP+FP+FN+TN}$ =fraction of observations with correct predicted classification
- Precision = PositivePredictiveValue (PPV) = $\frac{TP}{TP+FP}$ =Fraction of predicted positives that are correct
- Recall = Sensitivity = $\frac{TP}{TP+FN}$ =fraction of observations that are actually 1 with a correct predicted classification
- Specificity = $\frac{TN}{TN+FP}$ =fraction of observations that are actually -1 with a correct predicted classification

In addition we need the following:

- Prevalence= $\frac{TP+FN}{TP+FP+FN+TN}$ = fraction of positives, $Y=+1$, in $\mathcal{T}$.

a) Express PPV in terms of Specificity, Sensitivity and Prevalence. (*Hint*: Google) How is this related to Bayes$'$ formula? *Aid*: Think of the above in terms of conditional probabilities.

b) How would you relate the type I error and type II error of statistical hypothesis testing to the performance measures above? What is the power of a statistical test in this terminology?

20. Suppose

$$P(B|A) > P(B).$$

We interpret this as the statement that if A is true then B becomes more likely. **Formulate and prove** the statements

a) If not-B is true, then A becomes less likely.

b) If B is true, A becomes more likely.

c) If not-A is true, then B becomes less likely.

21. We say that the discrete r.v.s X and Y are *conditionally independent* given Z if and only if for all x,y, z

$$P(X = x, Y = y \mid Z = z) = P(X = x \mid Z = z)P(Y = y \mid Z = z). \quad (2.12)$$

a) Show that it follows from this definition that if X and Y are conditionally independent given Z, then

$$P(Y = y \mid X = x, Z = z) = P(Y = y \mid Z = z) \quad \text{for all } x,y,z. \quad (2.13)$$

b) Show that (2.13) implies (2.12).

22. In the text D. Poole & A. Mackworth: *Artificial Intelligence: Foundations of Computational Agents*, Cambridge University Press, 2010, we quote the following:

> *The idea of Bayesian learning is to compute the posterior probability distribution of the target features of a new example conditioned on its input features and all of the training examples.*
>
> *Suppose a new case has inputs $X = x$ and has target features, Y; the aim is to compute $P(Y|X = x \wedge e)$, where e is the set of training examples. This is the probability distribution of the target variables given the particular inputs and the examples. The role of a model is to be the assumed generator of*

the examples. If we let M be a set of disjoint and covering models, then reasoning by cases and the chain rule give

$$P(Y|x \wedge e) = \sum_{m \in M} P(Y \wedge m|x \wedge e)$$

$$= \sum_{m \in M} P(Y|m \wedge x \wedge e) \times P(m|x \wedge e)$$

$$= \sum_{m \in M} P(Y|m \wedge x) \times P(m|e).$$

The first two equalities are theorems from the definition of probability. The last equality makes two assumptions: the model includes all of the information about the examples that is necessary for a particular prediction (i.e., $P(Y|m \wedge x \wedge e) = P(Y|m \wedge x)$, and the model does not change depending on the inputs of the new example (i.e., $P(m|x \wedge e) = P(m|e)$. This formula says that we average over the prediction of all of the models, where each model is weighted by its posterior probability given the examples.

Obviously, the terminology above does not fully conform to the terminology made familiar in any standard probability and statistics courses (abridged as sanstat below).

a) What does $\wedge$ represent in sanstat?

b) Verify the first two equalities by the rules of sanstat.

c) What is "reasoning by cases" in sanstat?

d) How would you interpret in sanstat "the model includes all of the information about the examples that is necessary for a particular prediction" ?

e) How would you interpret in sannstat "the model does not change depending on the inputs of the new example"?

d) What is the meaning of "the posterior probability distribution of the target features of a new example conditioned on its input features and all of the training examples"?

23. Let

$$f(x|\theta) = \begin{cases} \frac{1}{\theta} & 0 \le x \le \theta \\ 0 & \text{elsewhere,} \end{cases}$$

where we know that $0 \le \theta \le 10$. We take for θ the prior density

$$\pi(\theta) = \begin{cases} \frac{1}{10} & 0 \le \theta \le 10 \\ 0 & \text{elsewhere.} \end{cases}$$

Asssume now that you have a data set $\mathcal{D}$ with four observations of x, and that $8 = \max \mathcal{D}$, i.e., 8 is largest value you have observed.

a) Show that the posterior distribution $\pi(\theta|\mathcal{D})$ is

$$\pi(\theta|\mathcal{D}) = \begin{cases} c \cdot \frac{1}{\theta^4} & 8 \le \theta < 10 \\ 0 & \text{elsewhere,} \end{cases}$$

where you need not find the value of c.

b) Show that the predictive distribution

$$f(x \mid \mathcal{D}) = \int_0^{10} f(x|\theta)\pi(\theta|\mathcal{D})d\theta$$

is given by (c as in **a)**)

$$f(x \mid \mathcal{D}) = \begin{cases} \frac{c}{4} \cdot \left(\frac{1}{x^4} - \frac{1}{10^4}\right) & 8 \le x < 10 \\ \frac{c}{4} \cdot \left(\frac{1}{8^4} - \frac{1}{10^4}\right) & 0 \le x \le 8 \\ 0 & \text{elsewhere.} \end{cases}$$

c) How would you comment the result in **b)**? Recall from first courses that the maximum liklelihood estimate of θ based on $\mathcal{D}$ is

$$\widehat{\theta}_{MLE} = 8,$$

so the maximum likelihood method yields the probability density for future data

$$f(x \mid \widehat{\theta}_{MLE}) = \begin{cases} \frac{1}{8} & 0 \le x \le 8 \\ 0 & \text{elsewhere.} \end{cases}$$

24. We let $\mathbf{P}$ be a random variable with values denoted by p, $0 \leq p \leq 1$. Conditionally on $\mathbf{P} = p$, the U_is are independent random variables $U_i \sim \text{Be}(p)$ $i = 1, 2, \ldots,$, i.e.,

$$f(u|p) = P(U = u|\mathbf{P} = p) = p^u \cdot (1-p)^{1-u}, u = 0, 1. \tag{2.14}$$

We fix n in advance and take

$$X = U_1 + \ldots + U_n.$$

Hence for $x = 0, 1, 2, \ldots, n$,

$$\begin{aligned} f(x|p) &= P\left(X = x \mid \mathbf{P} = p\right) \\ &= \left(\begin{array}{c} n \\ x \end{array} \right) p^x \cdot (1-p)^{n-x}, \end{aligned}$$

(the Binomial distribution) (you need not prove this). Bayes' rule tells that the posterior $\pi\left(p \mid x\right)$ = density of $\mathbf{P} = p$ given $X = x$ is generically given by

$$\pi\left(p \mid x\right) = \frac{f\left(x \mid p\right) \cdot \pi\left(p\right)}{m(x)}, 0 \leq p \leq 1$$

and zero elsewhere. Here $\pi(p)$ is the prior density. The marginal distribution of X in the denominator is

$$m(x) = \int_0^1 f\left(x \mid p\right) \cdot \pi\left(p\right) dp.$$

a) Choose the prior

$$\pi(p) = \left\{ \begin{array}{ll} \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1} & 0 < p < 1 \\ 0 & \text{elsewhere.} \end{array} \right.$$

Here $\alpha > 0$ and $\beta > 0$ are hyperparameters. In the above $\Gamma(x)$ is the Euler Gamma function. Show that

$$\pi\left(p \mid x\right) = \left\{ \begin{array}{ll} \frac{1}{B(x+\alpha, n-x+\beta)} \cdot p^{x+\alpha-1}\left(1-p\right)^{\beta+n-x-1} & 0 \leq p \leq 1 \\ 0 & \text{elsewhere.} \end{array} \right. \tag{2.15}$$

The constant $B(x+\alpha, n-x+\beta)$ is defined in (2.20) below.

b) Let next $U_{n+1} \sim \text{Be}(p)$ and independent of $U_i \sim \text{Be}(p)$ $i = 1, 2, \ldots, n$ given that $\mathbf{P} = p$. Check that the conditional distribution $\phi(u, p|x) = P(U_{n+1} = u, \mathbf{P} = p \mid X = x)$ can be written as

$$\phi(u, p|x) = f(u|p)\pi(p|x), \tag{2.16}$$

where $f(u|p)$ is given in (2.14) and $\pi(p|x)$ in (2.15) (We make no technical fuss about the curious mix of discrete and continuous r.v.s in the left hand side of (2.16).) *Aid*: Recall (2.12).

c) Let $\alpha = \beta = 1$ in (2.15). Then check that

$$P(U_{n+1} = 1 \mid X = x) = \frac{x+1}{n+2}. \tag{2.17}$$

Aid: Let us set

$$\phi(1|x) = P(U_{n+1} = 1 \mid X = x).$$

Then we have by marginalization in (2.16)

$$\phi(1|x) = \int_0^1 \phi(1, p|x)dp$$

You can now continue by (2.16) even if you have not solved the task in **b)**.

This is an example of a predictive probability distribution. D. Poole & A. Mackworth: *Artificial Intelligence: Foundations of Computational Agents*, Cambridge University Press, 2010, state the following:

> *The idea of Bayesian learning is to compute the posterior probability distribution of the target features of a new example conditioned on its input features and all of the training examples.*

USEFUL FORMULAS FOR THIS EXAM QUESTION:

- the Beta integral:

$$\int_0^1 p^{\alpha-1}(1-p)^{\beta-1}dp = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}. \tag{2.18}$$

- Recursion property of the Euler Gamma function: for x a positive integer,

$$\Gamma(x+1) = x!. \tag{2.19}$$

- We set

$$B(\alpha,\beta) \stackrel{\text{def}}{=} \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}. \tag{2.20}$$

25. Bayes Factor & Bayesian Model Comparison & Occam's Razor

Let

$$\mathcal{M}_i = \{p_i(x|\theta_i), \pi_i(\theta_i), \theta_0 \in \mathbf{\Theta}_i\}$$

be two Bayesian model families.

a) What is the marginal likelihood $p_i(x|\mathcal{M}_i)$ (of data) ?

b) How is the Bayes factor defined? Show that it can be written as

$$B_{01} = \frac{p_0(x|\mathcal{M}_0)}{p_1(x|\mathcal{M}_1)}.$$

c) What is *Occam's Razor*?

d) Suppose $\mathcal{M}_0$ is a more complex model family than $\mathcal{M}_1$, in the sense that $\mathcal{M}_0$ has more parameters. How does Bayes factor implement Occam's Razor?

26. Let $\mathbf{x} = (x_1, x_2, \ldots, x_d)$ be a binary vector, i.e., $x_i \in [0,1], i \in \{1,\ldots,d\}$. There are two classes c_0 and c_1 and corresponding probability mass functions

$$p(\mathbf{x}|\theta_j, c_j) = \prod_{i=1}^{d} \theta_{ij}^{x_i}(1-\theta_{ij})^{1-x_i}; x_i \in \{0,1\}; 0 \le \theta_{ij} \le 1,$$

where $\theta_{ij} = P(\mathbf{X}_i = 1|c_j)$ and

$$\theta_j = (\theta_{1j}, \ldots, \theta_{dj}).$$

Then

$$p(c_j \mid \mathbf{x}) = \frac{p(\mathbf{x}|c_j)\pi_j}{C},$$

where $\pi_1 (= 1 - \pi_0)$ is the prior probability of c_1, and π_0 is the prior probability of c_0. C is the normalization constant that does not depend on c_1 and c_0.

Let Y be a r.v. such that

$$Y = \begin{cases} c_1 & \text{with probability } p(c_1 \mid \mathbf{x}) \\ c_0 & \text{with probability } p(c_0 \mid \mathbf{x}). \end{cases}$$

Express $P(Y = c_1)$ in terms of the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$.

27. Let

$$Y^* = \boldsymbol{\beta}^T \mathbf{x} + \epsilon$$

where $\boldsymbol{\beta}^T \mathbf{x} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p$ and ϵ is a r.v. with the cumulative distribution function, the sigmoid function,

$$P(\epsilon \leq z) = \sigma(z) = \frac{1}{1 + e^{-z}}.$$

a) Check first that

$$P(-\epsilon \leq z) = P(\epsilon < z).$$

b) Define Y as

$$Y \stackrel{\text{def}}{=} \begin{cases} 1 & \text{if } Y^* > 0 \text{ i.e. } -\varepsilon < \boldsymbol{\beta}^T \mathbf{x}, \\ -1 & \text{otherwise.} \end{cases}$$

c) Write $P(Y = 1)$ in terms of the sigmoid function of $\boldsymbol{\beta}^T \mathbf{x}$.

28. Let $\mathcal{X}^t = \{\mathbf{x}^{(l)}\}_{l=1}^t$, where $\mathbf{x}^{(l)} \in \mathbb{R}^d$, be a training set. Let $D(\mathbf{x}, \mathbf{y})$ be a metric on $\mathbb{R}^d \times \mathbb{R}^d$.

a) Give in detail the steps of the k-means algorithm. What is the indata to the algorithm?

b) What is the definition of a Voronoi region, and how are Voronoi regions related to k-means clustering?

29. We have a statistical model that generates a set $\mathbf{X} = x$ of observed data, a set of unobserved latent data values $\mathbf{Z}$, and a vector of unknown parameters $\boldsymbol{\theta}$, along with a likelihood function

$$L(\boldsymbol{\theta}; \mathbf{X} = x, \mathbf{Z} = i) = P(\mathbf{Z} = i) p_{\mathbf{X}}(x|\boldsymbol{\theta}_i, \mathbf{Z} = i).$$

Hence $\boldsymbol{\theta}_i$ indicates the group of parameters in the conditional distribution of $\mathbf{X}$, when conditioning on $\mathbf{Z} = i$, and

$$\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots \boldsymbol{\theta}_k).$$

Let $L(\boldsymbol{\theta}; x)$ be the likelihood function when $\mathbf{Z}$ has been marginalized out.

$$L(\boldsymbol{\theta}; \mathbf{X} = x) = p_{\mathbf{X}}(x|\boldsymbol{\theta}) = \sum_{i=1}^{k} P(\mathbf{Z} = i) p_{\mathbf{X}}(x|\boldsymbol{\theta}_i)$$

The *EM- algorithm* (Expectation-Maximization -algorithm) seeks to find the MLE of the marginal likelihood $L(\boldsymbol{\theta}; x)$ by iteratively applying the following two steps:

Expectation step (*E step*): Calculate the expected value of the log likelihood function, with respect to the conditional distribution of $\mathbf{Z}$ given $\mathbf{X}$ under the current estimate of the parameters $\boldsymbol{\theta}^{(t)}$:

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathrm{E}_{\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{(t)}} \left[\log L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z})\right]$$

Maximization step (*M step*): Find the parameter that maximizes this quantity:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \,.$$

Set $\boldsymbol{\theta}^{(t)} \leftarrow \boldsymbol{\theta}^{(t+1)}$, go to *E step* and continue until convergence.

This was the generic description. Consider now the situation where $k = 2$, $-\infty < \boldsymbol{\theta}_i < \infty$ for $i = 1, 2$, and

$$p_{\mathbf{X}}(x|\boldsymbol{\theta}_i, \mathbf{Z} = i) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-\boldsymbol{\theta}_i)^2}, \quad i = 1, 2, -\infty < x < \infty,$$

$P(\mathbf{Z} = 1) = \lambda_1$ and $P(\mathbf{Z} = 2) = \lambda_2 = 1 - \lambda_1$.

a) Assume that you have a training set of $\mathcal{X}^t = \{x^{(i)}\}_{i=1}^t$ regarded as t i.i.d. samples drawn from $f_{\mathbf{X}}(x|\boldsymbol{\theta})$. The corresponding values $\mathbf{Z}^{(i)}$ have not been observed. Find the Sundberg equations of the EM-algorithm for $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$.

b) Explain how this can be extended to a method of classification or unsupervised clustering and relate the method to k-means clustering. Why should this be called 'soft' k-means clustering?

30. The LDA is often taken as being synonymous with the term Fisher's linear discriminant. But Fisher's original study [1] of statistical discrimination actually establishes a different discriminant, which does not make some of the assumptions of LDA such as normally distributed classes or equal class covariances. We shall now take a look at the original concept. First some background.

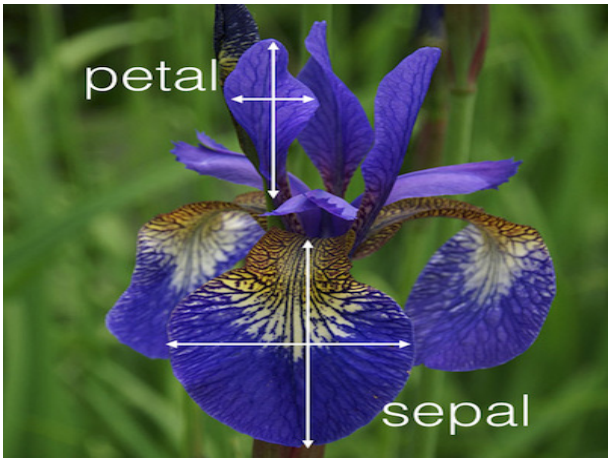
Edgar Anderson collected the data to quantify the morphologic variation of Iris flowers of three related species. Two of the three species were collected in " all from the same pasture, and picked on the same day and measured at the same time by the same person with the same apparatus".

The data set consists of 150 samples from each of three species of Iris (Iris setosa, Iris virginica and Iris versicolor). Four features were measured from each sample: the length and the width of the sepals and

[1] Fisher, R. A. (1936): The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics* 7 (2): 179−188.

petals, in centimetres. The data set can be downloaded at

`https://archive.ics.uci.edu/ml/datasets/Iris`



The Fisher discriminant invented for clustering/dealing with this data set is based on *projection* of d-dimensional data *onto a line*. The hope is that these projections are separated by class. Thus the line is to be oriented to maximize this class separation. This line is taken to go through origin in $\mathbb{R}^d$.

The $\mathbf{w}^T\mathbf{x}(=\sum_{i=1}^d w_i x_i)$ is the component of $\mathbf{x}$ in the direction of $\mathbf{w}$.

We try to find $\mathbf{w}$ so that the *separation criterion* (two classes)

$$J(\mathbf{w}) = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}$$

is maximized, where $\underline{\mu}_i, i = 1, 2$ are two class centers in $\mathbb{R}^d$, and

$$\mu_i = \mathbf{w}^T \underline{\mu}_i, \sigma_i^2 = \mathbf{w}^T \Sigma_i \mathbf{w},$$

and Σ_i is the covariance matrix of class i.

> Here the notation covers the cases, where $\underline{\mu}_i$ is known or is a class-wise sample mean
>
> $$\underline{\mu}_i = \begin{bmatrix} \bar{x}_{1i} \\ \vdots \\ \bar{x}_{di} \end{bmatrix} = \frac{1}{n} \sum_{j=1}^{n} x_{ji}$$

and Σ_i is known or is a class-wise sample covariance matrix

$$\Sigma_i = \frac{1}{n-1}\sum_{j=1}^{n}(x_{ji} - \underline{\mu}_i)(x_{ji} - \underline{\mu}_i)^{\mathrm{T}}.$$

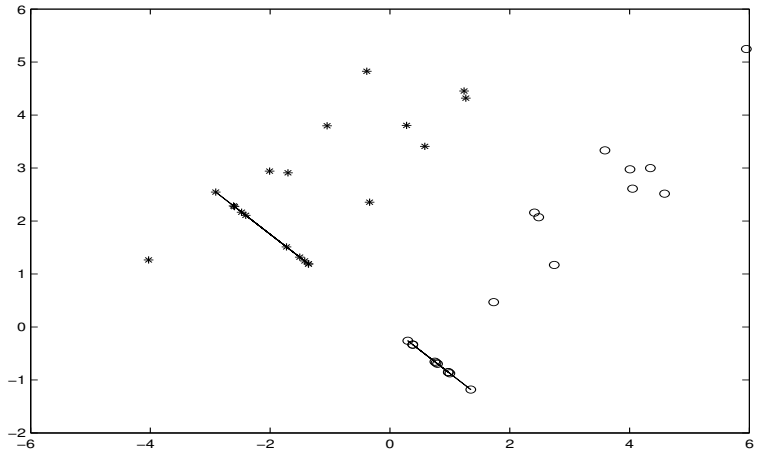
The criterion $J(\mathbf{w})$ can be maximized by standard methods and these give

$$\mathbf{w}^* = (\Sigma_1 + \Sigma_2)^{-1}\left(\underline{\mu}_1 - \underline{\mu}_2\right), \tag{2.21}$$

and the optimal projection $P_{\mathrm{F}}(\mathbf{x})$ is

$$P_{\mathrm{F}}(\mathbf{x}) = \left(\mathbf{w}^{*^T}\mathbf{x}\right)\frac{\mathbf{w}^*}{\|\mathbf{w}^*\|^2} \tag{2.22}$$

A set of simulated two dimensional Gaussian data and their projections onto a line (oriented in Northwest-southeast direction, only segmentally depicted) are given in the figure below:



Fisher's linear discriminant can actually be arbitrarily bad ! However, it has far-reaching (kernelized) generalizations as support vector machines.

a) Is this an early (1936) example of a method of statistical learning? If it is, what sort of learning is being done? If it is not, where does the difference lie?

Hint: Be careful, the illustration of hierarchic clustering by Iris data in the lecture notes (slides) need not lead here to the appropriate thinking.

b) Derive (2.21) and (2.22).
Aid: It is useful to try to do this by oneself, of course. If, however, external assistance is to be summoned up, a Google search with a phrase like *tutorial data reduction linear discriminant analysis* will very likely produce helpful documents. We have here a generalized eigenvalue problem of linear algebra.

c) Interpret the nature of the optimized $J(\mathbf{w}^*)$. Compare with (2.5) above.

31. We have four OTU's x, y, z, w in this order. We have computed the dissimilarity matrix

$$\begin{bmatrix} & 0.3 & 0.4 & 0.7 \\ 0.3 & & 0.5 & 0.8 \\ 0.4 & 0.5 & & 0.45 \\ 0.7 & 0.8 & 0.45 & \end{bmatrix}$$

This means that, e.g., the dissimilarity between x and y is 0.3 and the dissimilarity between y and w is 0.8. Let $d_{i,j}$ denote an array in this matrix. The hierarchical clustering method called single linkage (SLINK) works as follows:

i Find
$$(i^*, j^*) = \arg\min_{i,j} d_{i,j}$$

ii Join i^*, j^*

iii $d_{i,i^* \cup j^*} = \min(d_{i,i^*}, d_{i,j^*})$.

a) Use SLINK to cluster hierarchically the four OTU's and sketch the resulting dendrogram. Indicate on the plot the order in which the fusion occurs.

b) If we cut the dendrogram so that two clusters results, which OTU's are in each cluster ?

c) Check the ultrametric property on this dendrogram.

32. Describe the idea of CLINK in hierarchical clustering.

33. Let us consider the data space $U = \{0,1\}^d$, i.e. the binary hypercube consisting of binary d-tuples $\mathbf{x}$, $\mathbf{y}$, $\mathbf{x} = (x_1, \ldots, x_d)$, $x_i \in \{0,1\}$. The cardinality of U is $= 2^d$.

The metric on U is the Hamming metric, denoted by $d_H(\mathbf{x}, \mathbf{y})$. The Hamming metric is defined as the number of positions i, where $\mathbf{x}$ and $\mathbf{y}$ are differing, or

$$d_H(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{d} (x_i +_2 y_i),$$

where $1 +_2 1 = 0$, $0 +_2 0 = 0$, $1 +_2 0 = 1$, $0 +_2 1 = 1$. You are not expected to prove that d_H is in fact a metric.

Any metric has the property that if $d_H(\mathbf{x}, \mathbf{y}) = 0$, then $\mathbf{x} = \mathbf{y}$. Second, the maximum value of $d_H(\mathbf{x}, \mathbf{y})$ is equal to d. In fact

$$d_H(\mathbf{x}, \bar{\mathbf{x}}) = d,$$

where $\bar{\mathbf{x}}$ has the bits in $\mathbf{x}$ negated, i.e., $\bar{\mathbf{x}} = (\bar{x}_1, \ldots, \bar{x}_d)$, where for every i, $\bar{x}_i = 0$, if $x_i = 1$ and $\bar{x}_i = 1$, if $x_i = 0$.

Now we make a study of queries and their nearest neighbors in very large spaces in the spirit of Sergey Brin and others. A *query* is for our purposes simply a preassigned $\mathbf{q} \in U$.

Then we draw N independent samples $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ from the uniform distribution $P(\mathbf{x})$, or for each l

$$P(\mathbf{x}_l) = \frac{1}{2^d}.$$

This is equivalent to that the components x_i are independent Bernoulli variables $\sim \mathrm{Be}(1/2)$.

Let us define for $l = 1, \ldots, N$ the independent random variables

$$D_l \stackrel{\text{def}}{=} d_H(\mathbf{x}_l, \mathbf{q}). \tag{2.23}$$

In addition, we define the nearest neighbor $\mathrm{nn}(\mathbf{q})$ to $\mathbf{q}$ in $\mathcal{D}$ by

$$\mathrm{nn}(\mathbf{q}) \stackrel{\text{def}}{=} \left\{\mathbf{x}_{(1)} \in \mathcal{D} \mid d_H\left(\mathbf{x}_{(1)}, \mathbf{q}\right) \leq d_H(\mathbf{x}, \mathbf{q}) \text{ for all } \mathbf{x} \in \mathcal{D}\right\}. \tag{2.24}$$

a) Explain briefly why every $D_l \sim \text{Bin}(d, \frac{1}{2})$.

b) Show that for any $R > 0$

$$P(d_H(\text{nn}(\mathbf{q}), \mathbf{q}) \leq R) = 1 - (1 - P(D \leq R))^N, \tag{2.25}$$

where $D \sim \text{Bin}(d, \frac{1}{2})$, i.e., $D \stackrel{d}{=} D_l$ (D has the same distribution as D_l) for each l.

c) It is known that if d is large, then $D \sim \text{Bin}(d, \frac{1}{2})$ is approximately distributed like $D \sim N(\frac{d}{2}, \frac{d}{4})$, where $\frac{d}{2}$ is the mean and $\frac{d}{4}$ is the variance of this approximating normal distribution. Compute next with a given number $0 < \theta < 1$ the probability

$$P(D \leq \theta \frac{d}{2}). \tag{2.26}$$

using the normal approximation. *Aid:* It may be helpful to keep in mind that $\theta - 1 < 0$.

d) Find now, using (2.26)

$$\lim_{d \to +\infty} P(d_H(\text{nn}(\mathbf{q}), \mathbf{q}) \leq \theta \frac{d}{2}).$$

and comment your result, e.g., by thinking of $U = \{0, 1\}^d$ as a sphere with $\mathbf{q}$ and $\bar{\mathbf{q}}$ as opposite poles.

34. Describe how to build a regression or decision tree using cost complexity pruning. When building a classification tree using a cost function based on, e.g., the Gini index or the cross-entropy, it may happen that a split leaves two terminal nodes with the *same* predicted class. How is this possible?

35. In the random forest approach, every node split considers cost function minimization with respect to only a subset of $m \leq p$ randomly chosen predictors (where p denotes the total number of predictors). Show that the probability that a given predictor is not considered at a given split is $(p-m)/p$. What is the rationale behind considering only a randomly chosen subset of the predictors?