2017 年度《模式识别》课程作业 1

- 每次作业，答案的第一行必须写上姓名、学号、是否研究生；否则扣 10 分
- 已经保送本系的大四同学，如果希望本课程成绩作为研究生课成绩，需每次作业在“是否研究生”部分注明“本科保送”

1. 课件的第一章里列举了三个模式识别系统的例子：iOnRoad、Kinect 和 iPhone Siri。请回答下列问题：
   a) 对于每个例子，其中主要的模式识别任务是什么？可以通过描述其输入和输出来描述一个任务，如（包括但不限于以下问题）怎样获取输入？输出格式是什么？在每个例子当中，你可能可以找到不止一个模式识别任务。
   b) 为什么这些模式识别任务难度很高？你可以从直觉出发来回答这个问题，不需要严格的分析。
   c) 观看娇娇机器人的视频（其链接见课件，也可以网上搜索更多视频或其他相关信息），分析其中包含了那些模式识别任务。你认为娇娇机器人是通过人工智能自动实现这些功能，还是通过后台人工操作实现？请从技术说明你的理由。

2. 考虑 2 维空间中的两个向量：$\boldsymbol{x} = \left(\sqrt{3}, 1\right), \boldsymbol{y} = (1, \sqrt{3})$。假设$\boldsymbol{z}$为$\boldsymbol{x}$在$\boldsymbol{y}$上的投影，即$\boldsymbol{z} = \mathrm{proj}_{\boldsymbol{y}}\boldsymbol{x}$，那么
   a) $\boldsymbol{z}$=?
   b) 证明$\boldsymbol{y} \perp (\boldsymbol{x} - \boldsymbol{z})$
   c) 画草图表明以上各变量之间的关系。

3. 若X为 5 阶实对称矩阵，特征值分别为 1，1，3，4，x。
   a) 给出一个该矩阵为正定矩阵的充分必要条件
   b) 若已知$\det(X) = 72$，则x的值为?

4. 若随机变量$\boldsymbol{x} \sim N(\boldsymbol{\mu}, \Sigma)$
   a) 写出该随机变量的概率分布函数 pdf，$p(\boldsymbol{x})$ =?
   b) $\frac{\partial \ln p(\boldsymbol{x})}{\partial \boldsymbol{\mu}}$ =? 在 Matrix Cookbook 中应该用哪个公式?
   c) $\frac{\partial \ln p(\boldsymbol{x})}{\partial \Sigma^{-1}}$ =? 在 Matrix Cookbook 中应该用哪个公式?

   可以通过在 Matrix Cookbook 中查表解决问题（从第二章旁边的链接下载得到 The Matrix Cookbook）

5. 证明以下概率等式或不等式
   a) $Var(X) = E(X^2) - (EX)^2$ ， 假设X是离散随机变量
   b) 若X, Y是一维离散随机变量，证明$-1 \le \rho_{XY} \le 1$

6. 假设随机变量X服从指数分布，其 PDF 为$3\mathrm{e}^{-3x}$（其均值、方差、CDF 是什么？）
   a) 是否满足应用 Markov 不等式的条件？若可应用，根据 Markov 不等式，$P(X \ge 1)$的界是多少？

b) 是否可以设法应用 Chebyshev 不等式？界是多少？哪个界更精确？
c) $P(X \geq 1)$的实际值是多少?
d) 试比较以上两个不同的界和实际值，你能得出什么结论？
e) 阅读讲义关于系统评估（Evaluation）章的第 6 节，理解假设检验的基本思想。
注意：在本题中，问题（e）不需要回答，只需要你自己觉得理解了上述讲义中讲述的内容即可；关于指数分布，可以从网络或图书馆获得相关信息，不需要自己完成所有推导。

7. （编程）首先，完成 VLFeat 软件的安装。从 http://www.vlfeat.org/download.html 完成软件的下载，并阅读其安装指南，在其指导下完成 VLFeat 软件的安装。安装时，请在 Linux 下进行(最好在 Linux 下，如没有条件其他系统也可以)，安装 Matlab 接口。请从源码编译，不要使用已经编译好的可执行文件。为了公平比较，编译时请修改 Makefile，去除对 OpenMP 的支持。关于如何修改，可以看安装指南和 Makefile 本身的注释。

在 Matlab 中，用 x=rand(5000,10)产生数据（5000 个数据，每个数据 10 维）
a) 写一个发现最近邻的 Matlab 程序。这个程序应该计算一个数据与*其他*数据的距离并寻找其中的最小值，从而得到 x 中每一个数据的最近邻（排除数据自身）。对该方法计时（查找 tic、toc 函数的帮助），并保存结果（即每个数据的最近邻是谁？以及运行的时间）。你需要自己计算距离，不要使用系统的 pdist 之类函数，这些函数使用多个线程并行计算，会使比较不准确。
b) 使用 VLfeat 完成同样的任务。使用的函数是 vl_kdtreebuild 和 vl_kdtreequery。仔细研究这两个函数的帮助。当 Numtrees=1, MaxNumComparisons=6000 时，比较两者（你的程序和 VLFeat）的运行时间。请注意运行 vl_kdtreebuild 的时间不应该计算在内。
c) VLFeat 是近似方法，寻找近似的最近邻，在你的实验中，怎样才能知道 VLfeat 发现最近邻的准确率？这个方法的准确程度怎么样？
d) 当你选择不同的 VLfeat 参数时，其准确率和运行时间怎样变化？
e) 选择不同的数据大小（如 5000 改为 50000，10 改为 128 之类）时，VLfeat 方法的（相对你的方法的）加速比例和准确率的变化有没有什么规律？

对本实验题，不需要上交代码。需要回答上面的问题，可以自愿提供必要的材料来为你的答案提供说明。