Statistical Machine Learning GU4241/GR5241
Spring 2018
https://courseworks.columbia.edu/

Homework 1

Due: Thursday, Feb. 8th, 2018

Homework submission: Please submit your homework electronically through Canvas by 11:59pm on the due date.

Problem 1 (Maximum Likelihood Estimation, 10 points)

In this problem, we analytically derive maximum likelihood estimators for the parameters of an example model distribution, the gamma distribution.

The gamma distribution is univariate (one-dimensional) and continuous. It is controlled by two parameters, the *location parameter* μ and the *shape parameter* ν. For a gamma-distributed random variable X, we write $X \sim \mathcal{G}(\mu, \nu)$. $\mathcal{G}$ is defined by the following density function:

$$p(x|\mu,\nu) := \left(\frac{\nu}{\mu}\right)^{\nu} \frac{x^{\nu-1}}{\Gamma(\nu)} \exp\left(-\frac{\nu x}{\mu}\right) ,$$

where $x \geq 0$ and $\mu, \nu > 0$.[1] Whenever $\nu > 1$, the gamma density has a single peak, much like a Gaussian. Unlike the Gaussian, it is not symmetric. The first two moment statistics of the gamma distribution are given by

$$\mathsf{E}[X] = \mu \qquad \text{and} \qquad \mathsf{Var}[X] = \frac{\mu^2}{\nu} \tag{1}$$

for $X \sim \mathcal{G}(\mu, \nu)$. The plots in Figure 1 should give you a rough idea of what the gamma density may look like and how different parameter values influence its behavior.

Homework questions:

1. Write the general analytic procedure to obtain the maximum likelihood estimator (including logarithmic transformation) in the form of a short algorithm or recipe. A few words are enough, but be precise: Write all important mathematical operations as formulae. Assume that data is given as an i. i. d. sample $x_1, \ldots, x_n$. Denote the conditional density in question by $p(x|\theta)$, and the likelihood by $l(\theta)$. Make sure both symbols show up somewhere in your list, as well as a logarithm turning a product into a sum.

2. Derive the ML estimator for the location parameter μ, given data values $x_1, \ldots, x_n$. Conventionally, an estimator for a parameter is denoted by adding a hat: $\hat{\mu}$. Considering the expressions in (1) for the mean and variance of the gamma distribution, and what you know about MLE for Gaussians, the result should not come as a surprise.

3. A quick look at the gamma density will tell you that things get more complicated for the shape parameter: ν appears inside the gamma function, and both inside and outside the exponential. Thus, instead of deriving a formula of the form $\hat{\nu} := \ldots$, please show the following: Given an i. i. d. data sample $x_1, \ldots, x_n$ and the value of μ, the ML estimator $\hat{\nu}$ for the gamma distribution shape parameter solves the equation

$$\sum_{i=1}^{n} \left(\ln\left(\frac{x_i \hat{\nu}}{\mu}\right) - \left(\frac{x_i}{\mu} - 1\right) - \phi(\hat{\nu}) \right) = 0 .$$

[1] The symbol Γ denotes the distribution's namesake, the *gamma function*, defined by

$$\Gamma(\nu) := \int_0^{\infty} e^{-t} t^{\nu-1} dt .$$

The gamma function is a generalization of the factorial to the real line: $\Gamma(n) = (n-1)!$ for all $n \in \mathbf{N}$. Fortunately, we will not have to make explicit use of the integral.

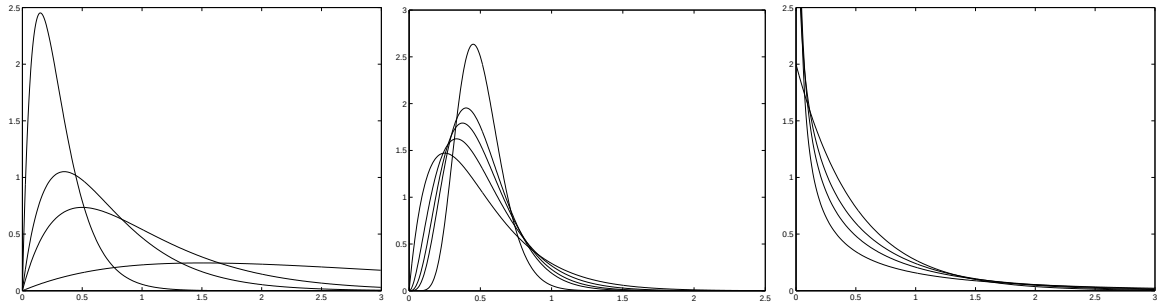

Figure 1: *Left:* The plot shows the density for different values of the location parameter ($\mu = 0.3, 0.5, 1.0, 3.0$), with the shape parameter fixed to $\nu = 2$. Since $\nu > 1$, the densities peak. As we increase μ, the peak moves to the right, and the curve flattens. *Middle:* For $\mu = 0.5$ fixed, we look at different values of the shape parameter ($\nu = 2, 3, 4, 5, 19$). Again, all the densities peak, with the peak shifting to the right as we increase ν. *Right:* If $\nu < 1$, the density turns into a monotonously decreasing function. The smaller the value of ν, the sharper the curve dips towards the origin.

The symbol ϕ is a shorthand notation for

$$\phi(\nu) := \frac{\frac{\partial \Gamma(\nu)}{\partial \nu}}{\Gamma(\nu)} .$$

In mathematics, ϕ is known as the *digamma function*.

Problem 2 (Bayes-Optimal Classifier, 15 points)

Consider a classification problem with K classes and with observations in $\mathbb{R}^d$. Now suppose we have access to the true joint density $p(\mathbf{x}, y)$ of the data $\mathbf{x}$ and the labels y. From $p(\mathbf{x}, y)$ we can derive the conditional probability $P(y|\mathbf{x})$, that is, the posterior probability of class y given observation $\mathbf{x}$.

In the lecture, we have introduced a classifier f_0 based on p, defined as

$$f_0(\mathbf{x}) := \arg\max_{y \in [K]} P(y|\mathbf{x}) \ ,$$

the *Bayes-optimal classifier*.

Homework question: Show that the Bayes-optimal classifier is the classifier which minimizes the probability of error, under all classifiers in the hypothesis class

$$\mathcal{H} := \{ f : \mathbb{R}^d \to [K] \mid f \text{ integrable} \} \ .$$

(If you are not familiar with the notion of an integrable function, just think of this as the set of all functions from $\mathbb{R}^d$ to the set $[K]$ of class labels.)

Hints:

- The probability of error is precisely the risk under zero-one loss.
- You can greatly simplify the problem by decomposing the risk $R(f)$ into conditional risks $R(f|\mathbf{x})$:

$$R(f|\mathbf{x}) := \sum_{y \in [K]} L^{0\text{-}1}(y, f(\mathbf{x})) P(y|\mathbf{x}) \qquad \text{and hence} \qquad R(f) = \int_{\mathbb{R}^d} R(f|\mathbf{x}) p(\mathbf{x}) d\mathbf{x} \ .$$

  If you can show that f_0 minimizes $R(f|\mathbf{x})$ for every $\mathbf{x} \in \mathbb{R}^d$, the result for $R(f)$ follows by monotonicity of the integral.

Problem 3 (PCA, 15 points)

1. For each of the 30 stocks in the Dow Jones Industrial Average, download the closing prices for every trading day from January 1, 2017 to January 1, 2018. To download the prices, for example for symbol `AAPL`, we use the R package `quantmod`. The code is as the following:

   ```
   library(quantmod)
   data<-getSymbols("AAPL", auto.assign = F, from = "2017-01-01", to = "2018-01-01")
   ```

   Please find a way to download data for the 30 stocks efficiently.

2. Perform a PCA on the closing prices and create the biplot (call function `princomp()` and use `cor=FALSE`). Do you see any structure in the biplot, perhaps in terms of the types of stocks? How about the screeplot – how many important components seem to be in the data?

3. Repeat part 2 with `cor=TRUE`. This is equivalent to scale each column of the data matrix.

4. Use the closing prices to calculate the return for each stock, and repeat part 3 on the return data. In looking at the screeplot, what does this tell you about the 30 stocks in the DJIA? If each stock were fluctuating up and down randomly and independent of all the other stocks, what would you expect the screeplot to look like?