1. Describe the following terms and concepts by a few sentences. (max. 6 p.)
   (a) Likelihood ratio test
   (b) K-nearest neighbor classifier
   (c) Cross-validation
   (d) Convolutional neural network
   (e) Logistic function
   (f) L1 regularization

2. The Poisson distribution is a discrete probability distribution that expresses the probability of a number of events $x \geq 0$ occurring in a fixed period of time:

   $$p(x; \lambda) = \frac{e^{-\lambda} \lambda^x}{x!}$$

   We measure $N$ samples: $x_0, x_1, \ldots, x_{N-1}$ and assume they are Poisson distributed and independent of each other.

   (a) Compute the probability $p(x; \lambda)$ of observing the samples $x = (x_0, x_1, \ldots, x_{N-1})$. (2p)
   (b) Differentiate the result with respect to $\lambda$. (2p)
   (c) Find the maximum of the function, i.e., the value where $\frac{\partial}{\partial \lambda} p(x; \lambda) = 0$. (2p)
3. (a) (4 pts) A dataset consists of two classes, whose distributions are assumed Gaussian, and whose sample covariances and means are the following:

\[
\begin{align*}
\mu_0 &= \begin{pmatrix} -5 \\ 5 \end{pmatrix} \\
\mu_1 &= \begin{pmatrix} 10 \\ 15 \end{pmatrix} \\
C_0 &= \begin{pmatrix} 11 & 9 \\ 9 & 11 \end{pmatrix} \\
C_1 &= \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}
\end{align*}
\]

A sample of data from these distributions is shown in Figure 1. Calculate the LDA projection vector \( w \). Hint: A 2 \times 2 matrix is inverted using the rule

\[
\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}.
\]

(b) (2 pts) The projected Gaussians are univariate normal: \( \mathcal{N}(w^T \mu_1, w^T C_1 w) \) and \( \mathcal{N}(w^T \mu_2, w^T C_2 w) \). Formulate the classification problem as a likelihood ratio test and choose the threshold based on that. Hint: Gaussian density is defined as

\[
\mathcal{N}(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left( -\frac{(x - \mu)^2}{2\sigma^2} \right)
\]

4. (a) (3 pts) Compute the gradient for \( L_2 \) penalized log-loss. Unregularized log-loss is defined as

\[
\ell(w) = \sum_{n=0}^{N-1} \ln(1 + \exp(y_n w^T x_n)).
\]

(b) (3 pts) Consider the Keras model defined in Listing 1. Inputs are 28 \times 28 grayscale images from 10 categories. Compute the number of parameters for each layer, and their sum over all layers.

Listing 1: A CNN model defined in Keras

```python
model = Sequential()

w, h = 3, 3
sh = (1, 28, 28)
model.add(Convolution2D(32, w, h, input_shape=sh, border_mode='same'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Activation('relu'))
model.add(Flatten())
model.add(Dense(100))
model.add(Activation('relu'))
model.add(Dense(10, activation = 'softmax'))
```
(a) (3p) The following code trains a list of classifiers and estimates their accuracy using stratified 10-fold CV. What are the missing lines of code in listing 2:

i. Define a list of classifiers: Logistic Regression, SVM and Random Forest.
ii. Insert code for computing the CV scores.

Listing 2: Training and CV estimation of classifiers

```python
import numpy as np
from sklearn.neighbors import KNeighborsClassifier
from sklearn.lda import LDA
from sklearn.svm import SVC, LinearSVC
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.cross_validation import cross_val_score,
    StratifiedKFold

classifiers = # <insert code 1 here>

skf = StratifiedKFold(y, 10, shuffle = True)

for clf in classifiers:
    scores = # <insert code 2 here>
    print ("Accuracy: %.2f +- %.2f" %
           (np.mean(scores),
            np.std(scores)))
```

(b) (3p) In the lectures we saw that the kernel trick $\kappa(x, y) = (x \cdot y)^2$ for $x = (x_1, x_2)$ and $y = (y_1, y_2)$ corresponds to the mapping

$$(u, v) \mapsto \left(\begin{array}{c}
u^2 \\
v^2 \\
\sqrt{2}uv\end{array}\right)$$

Find the explicit mapping corresponding to the inhomogeneous kernel $\kappa(x, y) = (x \cdot y + 1)^2$ with $x, y \in \mathbb{R}^2$. 