Cognitive Algorithms Exam Example

25 November 2013

Please fill in below your full name, your matriculation number and field of studies, if applicable.

Name	
Field of study	
Matriculation number	

Exercise	**Total Points**	**Points**
1 Short Overview	16	
2 Cross-Validation	8	
3 Linear Classification	24	
4 PCA	10	
5 (Linear) Regression	26	
6 Kernel Methods	16	
Total	100	

1 Short overview questions

Note that multiple choice questions can have more than one correct answer.

1. **[2 points]** In one sentence, characterize the difference between supervised and unsupervised learning.

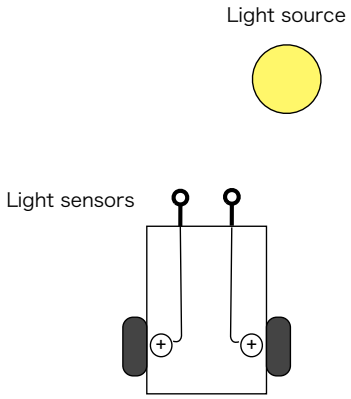
2. **[4 points]** You have collected some stock market data, and trained and tested a regression model on it. While you are dreaming about making a lot of money, a sceptic friend tells you "The classifier might overfit the data." What does he/she mean?

3. **[4 points]** Which of the following statements are *true*?
   - ☐ The error of a classifier measured over the training data provides a pessimistically biased estimate of the true error.
   - ☐ Linear models are too restricted to overfit.
   - ☐ Cross-Validation can be used to optimize the parameters of a model
   - ☐ When the size of the training set increases, the training error of a classifier with fixed complexity will decrease.

4. **[2 points]** Which of the following algorithms solves a classification problem?
   - ☐ Linear Discriminant Analysis (LDA)
   - ☐ Non-negative Matrix Factorization (NMF)
   - ☐ Principal Component Analysis (PCA)
   - ☐ Kernel Ridge Regression (KRR)

5. **[2 points]** Which of the following methods uses gradient descend to optimize it's objective function?
   - ☐ Linear Discriminant Analysis (LDA)
   - ☐ The Perceptron
   - ☐ Ordinary Least Squares (OLS)
   - ☐ Kernel Ridge Regression (KRR)

6. **[2 points]** Braitenberg Vehicle



Consider a simple Braitenberg Vehicle with two light sensors (more light leads to higher sensor values). The light sensors are coupled with positive weights to the wheels, as indicated in the figure on the right. When the light source is turned on, the vehicle will ...

- ☐ accelerate towards light source.
- ☐ accelerate away from light source.

2 Cross-Validation

1. **[4 points]** You are a reviewer for the International Mega-Conference on Machine Learning of Outrageous Stuff, and you read a paper that selected a small number of features out of a large number of features for a given classification problem. The paper argues as follows:

   a) We uses all our available data to select a subset of "good" features that had fairly strong correlation with the class labels.

   b) Our final model contained only those features. We evaluate the prediction error of the final model by 10-fold crossvaldiation on all the available data.

   c) We obtained a low cross-validation error. Thus, we have achieved high classification accurarcy with only few meaningful features. (This is novel and amazing.)

   Would you accept or reject the paper? Explain your decision.

2. **[4 points]** Suppose you are testing a new algorithm on a data set consisting of 100 positive and 100 negative examples. You plan to use leave-one-out cross-validation (i.e. 200-fold cross-validation) and compare your algorithm to a baseline function, a simple majority classifier. (Given a set of training data , the majority classifier always outputs the class that is in the majority in the training set, regardless of the input.) You expect the majority classifier to achieve about 50% classification accuracy, but to your surprise, it scores zero every time. Can you explain why?

3 Linear Classification

1. **[4 points]** Which of the following statements about the algorithms we encountered to solve a two-class linear classification problem are *true*?

   - ☐ Given the same data, the perceptron algorithm will always return the same solution.
   - ☐ The perceptron is an iterative algorithm. After each update step, the error function is reduced.
   - ☐ Linear Discriminant Analysis (LDA) projects the data onto a one-dimensional subspace that is obtained by maximizing the interclass variance while minimizing the intraclass variance.
   - ☐ If the covariance matrices of both classes are the identity, Linear Discriminant Analysis (LDA) and the Nearest Centroid Classifier (NCC) (i.e. the Prototype classifier) find the same solution.

2. **[2 points]** Consider the error function of a linear perceptron

$$\mathcal{E}_{\mathcal{P}}(\mathbf{w}) = -\sum_{m \in \mathcal{M}} \mathbf{w}^{\top} \mathbf{x}_m y_m \tag{1}$$

   where $\mathcal{M}$ is the index set of all misclassified data points, $\mathbf{x}_m \in \mathbb{R}^D$ is a misclassified data point in a D-dimensional space, $y_m \in \{-1, 1\}$ is the corresponding true label, and $\mathbf{w} \in \mathbb{R}^D$ is the weight vector of the perceptron hyperplane.

   Of which dimensionality is $\mathcal{E}_{\mathcal{P}}(\mathbf{w})$? (Is it a scalar, a vector or a matrix?)

3. **[4 points]** The Perceptron realizes stochastic gradient descent. A random misclassified point $\mathbf{x}_m$ is picked, and the gradient of the part of the error function related to $\mathbf{x}_m$, $\mathcal{E}_{\mathcal{P}}{}^m(\mathbf{w}) = -\mathbf{w}^{\top} \mathbf{x}_m y_m$, is computed. A single update consists of

$$\mathbf{w}^{\text{new}} \leftarrow \mathbf{w} + \eta \frac{\partial \mathcal{E}_{\mathcal{P}}{}^m(\mathbf{w})}{\partial \mathbf{w}} \tag{2}$$

   where $\mathbf{w}$ is the weight vector before the update and $\mathbf{w}^{\text{new}} \in \mathbb{R}^D$ the weight vector after the update.

   Compute

$$\frac{\partial \mathcal{E}_{\mathcal{P}}{}^m(\mathbf{w})}{\partial \mathbf{w}} = \dots$$

4. **[2 points]** Complete the following statement: " The perceptron algorithm will find an optimal solution within a finite number of steps if . . . "

5. You want to apply the Nearest Centroid Classifier to a very simple two-class classification problem with the following four data points:

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \mathbf{x}_2 = \begin{bmatrix} 1 \\ 3 \end{bmatrix}, \mathbf{x}_3 = \begin{bmatrix} 2.5 \\ 2 \end{bmatrix}, \mathbf{x}_4 = \begin{bmatrix} 1.5 \\ 2 \end{bmatrix}$$

$\mathbf{x}_1$ and $\mathbf{x}_2$ belong to class -1, while $\mathbf{x}_3$ and $\mathbf{x}_4$ belong to class $+1$.

a) **[2 points]** Compute the respective class means $\mathbf{w}_{-1}$ and $\mathbf{w}_{+1}$.

b) **[4 points]** Compute the classification boundary $\mathbf{w}^\top \mathbf{x} - \beta = 0$ of the prototype classifier according to the following formulas:

$$\mathbf{w} = \mathbf{w}_{+1} - \mathbf{w}_{-1}$$

$$\beta = \frac{1}{2}(\mathbf{w}_{+1}^\top \mathbf{w}_{+1} - \mathbf{w}_{-1}^\top \mathbf{w}_{-1})$$

c) **[2 points]** For each point, compute the assigned class label $\text{sign}(\mathbf{w}^\top \mathbf{x} - \beta)$. Are all points correctly classified?

d) **[4 points]** Sketch the data points, their class means $\mathbf{w}_{-1}$ and $\mathbf{w}_{+1}$, the normal vector $\mathbf{w}$, and the classification boundary in the x_1-x_2 space.

4 Principal Component Analysis

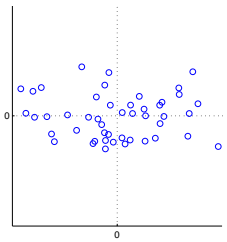
1. **[2 points]** Given a D dimensional data, which quantity does PCA maximize in order to obtain the first principal direction $\mathbf{w} \in \mathbb{R}^D$?

2. Consider a data set with two data points: $\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}-1\\-1\end{bmatrix}$.

   a) **[2 points]** Compute the covariance matrix.

   b) **[2 points]** How large ist the variance of the data projected onto the second principal direction?

3. The covariance matrix of the two-dimensional data set plotted below is $\begin{bmatrix}\alpha & 0\\0 & \beta\end{bmatrix}$.



   a) **[2 points]** Draw the first principal direction estimated from the data plottet in the figure.

   b) **[2 points]** How large is the variance of the data projected on the first principal component?

5 (Linear) Regression

1. **[6 points]** Stephan tries to derive the solution for the Ordinary Least-Squares Regression (OLS) for the simple case of 1 - dimensional data. He knows that given N 1-dimensional data points $x_1, \ldots, x_N \in \mathbb{R}$ and their corresponding real-valued labels $y_1, \ldots, y_N \in \mathbb{R}$ the objective of OLS is to find the slope of a linear function $\alpha \in \mathbb{R}$ that minimizes the least-squares error

$$\mathcal{E}(\alpha) = \sum_{i=1}^{N} (y_i - \alpha x_i)^2 \tag{3}$$

He computes the derivative w.r.t. α

$$\frac{\partial \mathcal{E}(\alpha)}{\partial \alpha} = \sum_{i=1}^{N} 2(y_i - \alpha x_i) \cdot (-\alpha) \tag{4}$$

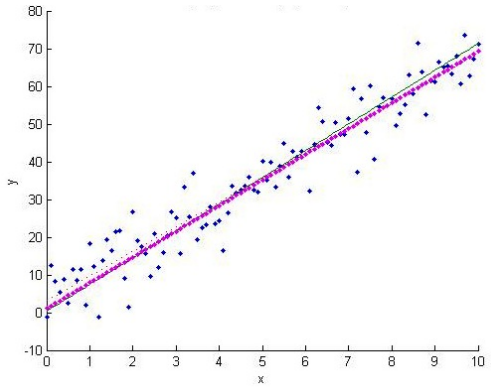
sets it to zero and solves for α:

$$\sum_{i=1}^{N} 2(y_i - \alpha x_i) \cdot (-\alpha) = 0 \quad \Rightarrow \quad \sum_{i=1}^{N} y_i - \alpha \sum_{i=1}^{N} x_i = 0 \tag{5}$$

$$\Rightarrow \quad \alpha = \frac{\sum_{i=1}^{N} y_i}{\sum_{i=1}^{N} x_i} \tag{6}$$

This is not the OLS solution. While the objective in Equation (3) is correct, something went wrong afterwards. Which mistake did Stephan make? Correct his error and calculate the correct α.

2. **[2 points]** In the data set below, visualize the quantity minimized in Equation (3) for one data point of your choice.



3. In one assignment, you used linear regression to predict two dimensional hand positions from electromyographic (EMG) recordings obtained with high-density electrode arrays of 192 electrodes on the lower arm. You were given pre-processed training data at $N_{tr} = 800$ time points with known 2D-hand positions. The goal was to predict the 2D hand positions of $N_{te} = 300$ test data points. The data was stored in the following data structures:

$$X_{tr} \in \mathbb{R}^{192\times 800}, Y_{tr} \in \mathbb{R}^{2\times 800}, X_{te} \in \mathbb{R}^{192\times 300}.$$

and linear regression was used. Recall that OLS amounts to estimating a linear mapping W as follows:

$$W = (X_{tr}X_{tr}^{\top})^{-1}X_{tr}Y_{tr}^{\top} \tag{7}$$

a) **[2 points]** What is the dimensionality of W?

b) **[2 points]** Our goal is to compute the unkown hand positions of our test data, $Y_{te} \in \mathbb{R}^{2\times 300}$. How do you use the weight matrix W to do so? Write down the formula.

c) **[4 points]** Suppose an extra hard-working scientist had recorded EMG at 10 times as many electrodes. What problem would he get when estimating W by equation (7)? What could he do to prevent this problem (without kernelizing)?

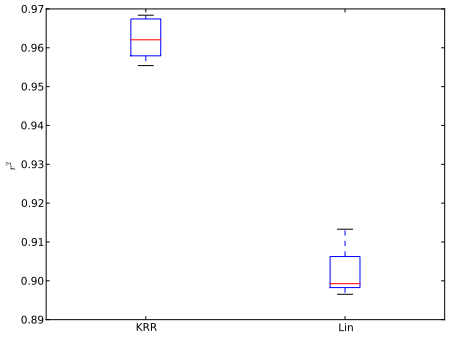
d) **[2 points]** To evaluate our regression models, we used the so called coefficient of determination, or r^2 index

$$r^2 = 1 - \frac{\sum_{d=1}^{D} \mathrm{Var}(\hat{y}_d - y_d)}{\sum_{d=1}^{D} \mathrm{Var}(y_d)}$$

where D is the dimensionality of the data labels, y are the true labels and $\hat{y}$ the estimated labels. Which of the following statements is true:

- ☐ The better our prediction, the higher r^2.
- ☐ The better our prediction, the lower r^2.
- ☐ $r^2 = 1$ for perfect predictions
- ☐ $r^2 = 0$ for perfect predictions

e) **[2 points]** We have compared linear regression with Kernel Ridge Regression (KRR) with a Gaussian Kernel. The results of a 5 -fold cross-validation are shown below as two boxplots (recall the red line represents the median). Do we gain from Kernel Ridge Regression?



f) **[2 points]** Name one advantage that Ordinary Least Squares (OLS) has over Kernel Ridge Regression (KRR) with a Gaussian Kernel whose width is determined by cross-validation.

4. **[4 points]** Stephie has implemented a function to train Ordinary Least Squares (OLS). When she tests it by calling the function `test_ols()` she gets the following error:

   `LinAlgError: Singular matrix.`

   Find the error in her code and correct it.

```
import scipy as sp
from numpy.linalg import inv

def train_ols(X_train, Y_train):
    ''' Trains ordinary least squares (ols) regression
    Input:  X_train  -  DxN array of N data points with D features
            Y_train  -  D2xN array of length N with D2 multiple labels
    Output: W  -  DxD2 array, linear mapping used to estimate labels
                  with sp.dot(W.T, X)
    '''
    W = inv(sp.dot(X_train.T,X_train))
    W = sp.dot(W, X_train)
    W = sp.dot(W, Y_train.T)
    return W

def test_ols():
    x_train = sp.array([[ 0,  0,  1 , 1],[ 0,  1,  0, 1]])
    y_train = sp.array([[0, 1, 1, 2]])
    w_est = train_ols(x_train, y_train)
    assert(sp.all(w_est.T == [[1, 1]]))
```

6 Kernel methods and Kernel Ridge Regression

1. **[4 points]** In the lecture, you have encountered the concept of a kernel function

$$k : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}, k(\mathbf{x}, \mathbf{y}) = \langle \varphi(\mathbf{x}), \varphi(\mathbf{y}) \rangle$$

   where $\langle \cdot, \cdot \rangle$ denotes the scalar product and $\varphi(\cdot)$ realizes a mapping into a possibly infinite-dimensional space. Give two reasons why the "Kernel Trick" is so popular in Machine Learning.

2. **[6 points]** You have been given N training data points $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathbb{R}^D$ with labels $y_1, \ldots y_N \in \mathbb{R}$, and you have trained a Kernel Ridge Regression which realizes an implicit feature mapping $\varphi(\cdot)$. This means you have obtained $\alpha_1, \ldots, \alpha_N \in \mathbb{R}$, and learned a model for your data

$$f(\mathbf{x}) = \mathbf{w}^T \varphi(\mathbf{x}) \text{ with } \mathbf{w} = \sum_{i=1}^{N} \alpha_i \varphi(\mathbf{x}_i). \tag{8}$$

Now you want to obtain a label for a new data point $\mathbf{x} \in \mathbb{R}^D$. Show that you can obtain its label $f(\mathbf{x})$ without explicitly evaluating $\varphi(\cdot)$, i.e. show that

$$f(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i k(\mathbf{x}_i, \mathbf{x}) \tag{9}$$

where $k(\mathbf{x}_i, \mathbf{x}) = \langle \varphi(\mathbf{x}_i), \varphi(\mathbf{x}) \rangle$.

3. **[6 points]** We used a Kernel Ridge Regression with a Gaussian kernel $k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right)$ on training data that follows a sine-function. Below you find the results for three different kernel widths. Indicate which of the three σ values corresponds to which of the plots ($\sigma = 10$, $\sigma = 1$, $\sigma = 0.1$). Using Equation (9), explain intuitively how the kernel width σ affects the learned model.