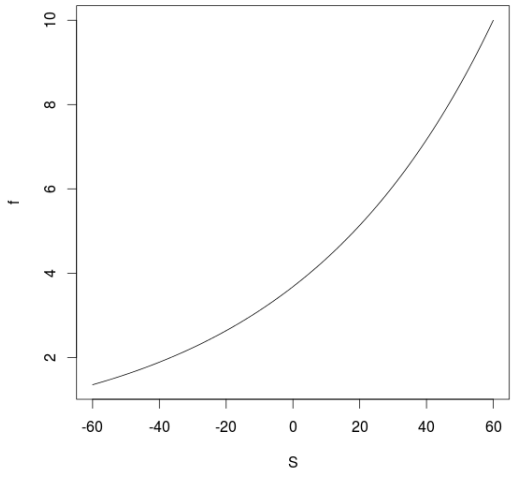
Machine Learning - MIRI Master (Final quiz - June 1, 2017)

Name:

Instructions:

- tick **clearly** the claims that you think are **true** with a $\surd$
- tick **clearly** the claims that you think are **false** with a $\times$
- if you want to "withdraw" an already ticked box, black it out as ■ (it will count now as unanswered)
- all questions are equally weighted (the headings define **blocks** of **ten** questions each)
- there is no obligation to answer individual questions, but at least half (**five**) questions in each block must be answered
- individual question grading: correct answers count +1 point, incorrect answers count −1 point; no answer counts 0 points (there are 60 questions = 60 points maximum)
- letting S be the number of points, the overall grade is obtained as

$$f(S) = 10 \exp\left(\frac{S}{60} - 1\right)$$



- deliver just **these sheets** back
- time: 2h

1. **Complexity control and all that jazz.**

T ❑ Based on training data alone, there is no means of choosing which model is better

F ❑ Complexity control is only necessary when data is high-dimensional

T ❑ The empirical error in the training set is always smaller (or equal) than the empirical error in the test set

T ❑ Using a larger training data set reduces the chances to obtain an overfitted model

F ❑ Regularization is intended to penalize models that are less complex than needed

T ❑ Cross-validation is mainly used for model selection purposes

T ❑ L_2-regularization does not produce sparsity, as opposed to L_1-regularization

T ❑ Reducing the hypothesis (model) space is a way of controlling complexity

T ❑ The VC dimension of a two-class linear classifier is a linear function of data dimension

For a linear classifier, the VC dimension is approximately equal to the number of features plus one, making it a linear function of the data dimension.

F ❑ The VC dimension of a two-class classifier is always a finite integer

2. **Bayesian classifiers.**

T ❑ The Bayes formula transforms prior distributions into posterior distributions

T ❑ The numerator in Bayes formula is enough to perform classification, by taking simply the maximum over the classes

Gaussian distributions are a common assumption, but not a requirement.

F ❑ The Bayes classifier is the best possible classifier when the classes are Gaussian

T ❑ The Bayes classifier is the best possible classifier when the true priors are known

T ❑ For normally distributed classes, Bayesian classifiers turn out to be quadratic discriminant functions

F ❑ For normally distributed classes, equal posterior probabilities yield linear discriminant functions

F ❑ The Naive-Bayes classifier does not make assumptions about data distribution for continuous variables

T ❑ The kNN classifier requires tuning of the number of neighbours, because we have a finite data sample

F ❑ $\sum_a P(a|b)P(b) = P(b)$, where A, B are discrete random variables

F ❑ $\sum_b P(a|b) = \sum_a P(b|a)$, where A, B are discrete random variables

3. **Maximum Likelihood and GLMs.**

F ❑ The likelihood is a function of the parameters for a given choice of data sample

T ❑ Logistic regression does not make assumptions about input data distribution

F ❑ Linear regression assumes normally distributed outputs, conditioned on the inputs

T ❑ In a GLM, the model tries to predict the expected value of the target using a linear function of the predictors and a suitable interface function

T ❑ We can obtain an error function as the negative log-likelihood of a problem

F ❑ The regression function is the best possible predictor, in the sense that it would achieve *zero bias*

F ❑ The regression function is the best possible predictor, in the sense that it would achieve *zero variance*

F ❑ The regression function is the best possible predictor, in the sense that it would achieve *zero noise*

F ❑ In statistics, bias and variance are related concepts: they represent the distribution of errors in the training and test sets, respectively

F ❑ The mean squared error is always preferred for GLM regression, because it is the only one that works in practice

4. **Neural networks**.

F ❑ An MLP needs no regularization, because backpropagation prevents arbitrary growth of the weights

F ❑ We can convert a non-linear model into a linear one by giving values to the non-linear adaptive parameters

T ❑ The backpropagation algorithm computes the partial derivatives of a given differentiable error function with respect to the network weights

T ❑ The backpropagation algorithm must be coupled with an optimization method (update rule) to make it a learning algorithm for a neural network

T ❑ The backpropagation algorithm is mainly used to compute the gradient vector of the error function at each step

T ❑ The nature of the target variable dictates the activation function for the output neurons

F ❑ The activation function for the hidden neurons could be a linear function to facilitate learning

T ❑ Both RBF and MLP neural networks can have one or more hidden layers of neurons

T ❑ An RBF neural network could in principle be trained with the backpropagation algorithm

F ❑ Regularization makes little sense in neural networks, because they are non-linear models

5. **Kernels and SVMs**.

F ❑ The kernel function defines kernel matrices whose elements are always non-negative

T ❑ Any positive linear combination of a number of kernel functions is a kernel function

T ❑ By choosing a valid kernel, we get an Euclidean distance in some Hilbert space

F ❑ In SVMs, the Lagrange coefficients α_n are positive for the support vectors and negative for the rest (the non support vectors)

F ❑ In order to "kernelize" a learning algorithm, this must be supervised to get meaningful targets

T ❑ The cost parameter (C) in a SVM has a role similar to that of a regularization parameter

T ❑ Increasing the value of C in a SVM (and everything else being equal), the margin cannot increase

F ❑ Increasing the value of C in a SVM (and everything else being equal), the number of support vectors can increase Increasing C typically results in fewer support vectors. This is because a larger C implies a stricter penalty on misclassification, leading the SVM to fit the training data more tightly with fewer support vectors.

F ❑ A positive semi-definite matrix may have negative elements in the main diagonal

T ❑ The VC dimension of a SVM depends on the kernel function it uses

6. **Miscellaneous** The EM algorithm is used to find maximum likelihood estimates of parameters in probabilistic models, such as GMMs. It does not guarantee finding a global optimum; rather, it iteratively refines a solution to reach a local optimum. Starting with k-means can provide a good initialization for the E-M algorithm, but there is no guarantee of finding a global optimum.

F ❑ The E-M algorithm refines a suboptimal solution obtained by k-means until a global optimum is found

F ❑ A Gaussian mixture model assumes that the data has been generated by a "big" Gaussian that can be decomposed as a finite mixture

T ❑ The k-means algorithm will discover the true clusters in the data, if given enough prototypes

T ❑ Bagging methods are based on the fact that, for unstable learners, variance can be greatly reduced with little or no increase in bias

F ❑ A Random Forest is "random" because decision trees are random learners (meaning that a single tree changes if we "execute" the algorithm again)

Bagging (Bootstrap Aggregating) methods, such as Random Forests, reduce variance by averaging the predictions of multiple models trained on different bootstrap samples of the data. This approach is particularly effective for unstable learners like decision trees, which have high variance. The averaging process reduces variance without significantly

F ❑ A Random Forest is "random" because the *data* used in *each decision node* come from a different bootstrap resample

F ❑ In Machine Learning, the lack of predictive variables can be compensated by more training data; in other words, there is no limit on the achievable predictive performance of a model, if we can gather enough data

F ❑ We should optimize the number of folds in cross-validation, and separately for each modeling technique

T ❑ In Machine Learning, better pre-processing can make a large impact on learning, and therefore on predictive performance

T ❑ In a noiseless setting, at least theoretically speaking, there is no need for regularization

---

While the number of folds in cross-validation can affect the model evaluation, it is generally not optimized separately for each modeling technique. Common practice is to choose a standard number, such as 5 or 10 folds, which balances the trade-off between computational efficiency and the stability of the performance estimate.

In a noiseless setting, where the data perfectly fits the model without any errors, regularization may not be necessary. Regularization is used to prevent overfitting by penalizing model complexity, which is crucial in noisy settings. In a perfectly noiseless scenario, the risk of overfitting is minimized, reducing the need for regularization.