[[1]](#footnote-1)

Preparation of Papers for IEEE TRANSACTIONS on Biomedical Engineering(November 2014)

First A. Author, *Fellow, IEEE*, Second B. Author, and Third C. Author\*, Jr., Member, IEEE

*Abstract*— *Objective:* The purpose of this document is to illustrate how one should prepare for manuscripts for submission to IEEE Transactions on Biomedical Engineering. *Methods:* Use this document as a template if you are using Microsoft *Word* 6.0 or later. Otherwise, use this document as an instruction set. The electronic file of your paper will be formatted further at IEEE. *Results:* Paper titles should be written in uppercase and lowercase letters, not all uppercase. Avoid writing long formulas with subscripts in the title; short formulas that identify the elements are fine (e.g., "Nd–Fe–B"). Do not write “(Invited)” in the title. Full names of authors are preferred in the author field, but are not required. Put a space between authors’ initials. Define all symbols used in the abstract. Do not cite references in the abstract. Do not delete the blank line immediately above the abstract; it sets the footnote at the bottom of this column. The abstract should not exceed 250 words. *Conclusion:* Preparing carefully your manuscript will lead to enhanced readability. *Significance:* Carefully prepared manuscript will better disseminate your work to the scientific community and public.

*Index Terms*—Enter key words or phrases in alphabetical order, separated by commas. For a list of suggested keywords, send a blank e-mail to [keywords@ieee.org](mailto:keywords@ieee.org) or visit <http://www.ieee.org/organizations/pubs/ani_prod/keywrd98.txt>

# INTRODUCTION

T

HIS document is a template for Microsoft *Word* versions 6.0 or later. If you are reading a paper or PDF version of this document, please download the electronic file  
“TBME\_template.doc” from the Web site of IEEE Transactions on Biomedical Engineering (TBME) at http://tbme.embs.org/for-authors/instruction-for-authors/ so you can use it to prepare your manuscript.

# Guidelines For Manuscript Preparation

When you open TBME\_template.doc, select “Page Layout” from the “View” menu in the menu bar (View | Page Layout), (these instructions assume MS 6.0. Some versions may have alternate ways to access the same functionalities noted here).

**An Overview of Statistical Learning**

Statistical learning refers to a vast set of tools for understanding data. These tools can be classified as supervised or unsupervised. Broadly speaking, supervised statistical learning involves building a statistical model for predicting, or estimating, an output based on one or more inputs. Problems of this nature occur in fields as diverse as business, medicine, astrophysics, and public policy. With unsupervised statistical learning, there are inputs but no supervising output; nevertheless we can learn relationships and structure from such data.

**Data**

The study acquired sleep related data for 27 people. Besides the demographics, Age, Sex, BMI, data were collected during a Polysomnography (PSG), such that Total Sleep Time, Latency, Latency at each stage of sleep, Sleep duration at each stage of sleep, the number of Movements and Awakenings, the Sleep efficiency, the Apneas and Hypoapneas.

Some indices that were also calculated were the ratio Light to Total sleep time, Deep to Total sleep time and REM to Total sleep time.

People were also submitted to two self-report [questionnaire](https://en.wikipedia.org/wiki/Questionnaire)s: The Pittsburgh Sleep Quality Index (PSQI) that assesses sleep quality over a 1-month time interval. The measure consists of 19 individual items and the Epworth Sleepiness Scale (ESS) is a [scale](https://en.wikipedia.org/wiki/Likert_Scale) intended to measure daytime [sleepiness](https://en.wikipedia.org/wiki/Sleep) that is measured by use of a very short questionnaire.

According to subjects with a global PSQI score of 5 and above is generally considered indicative of subjective poor sleep

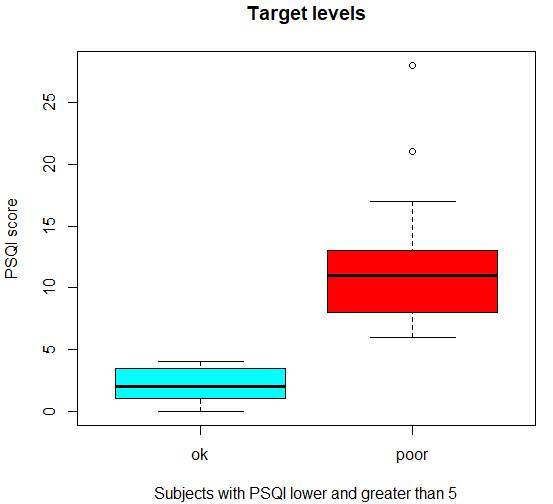
quality, distinguishing between good and poor sleepers with

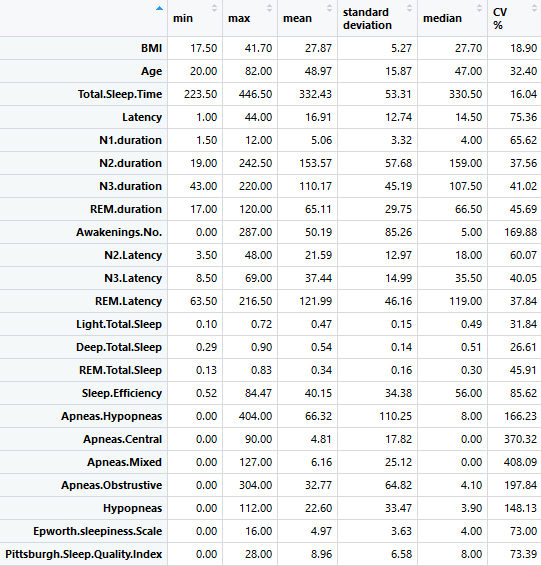
high sensitivity and specificity.

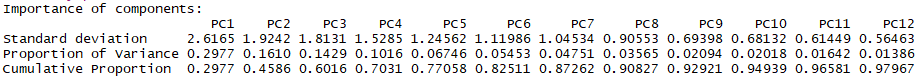
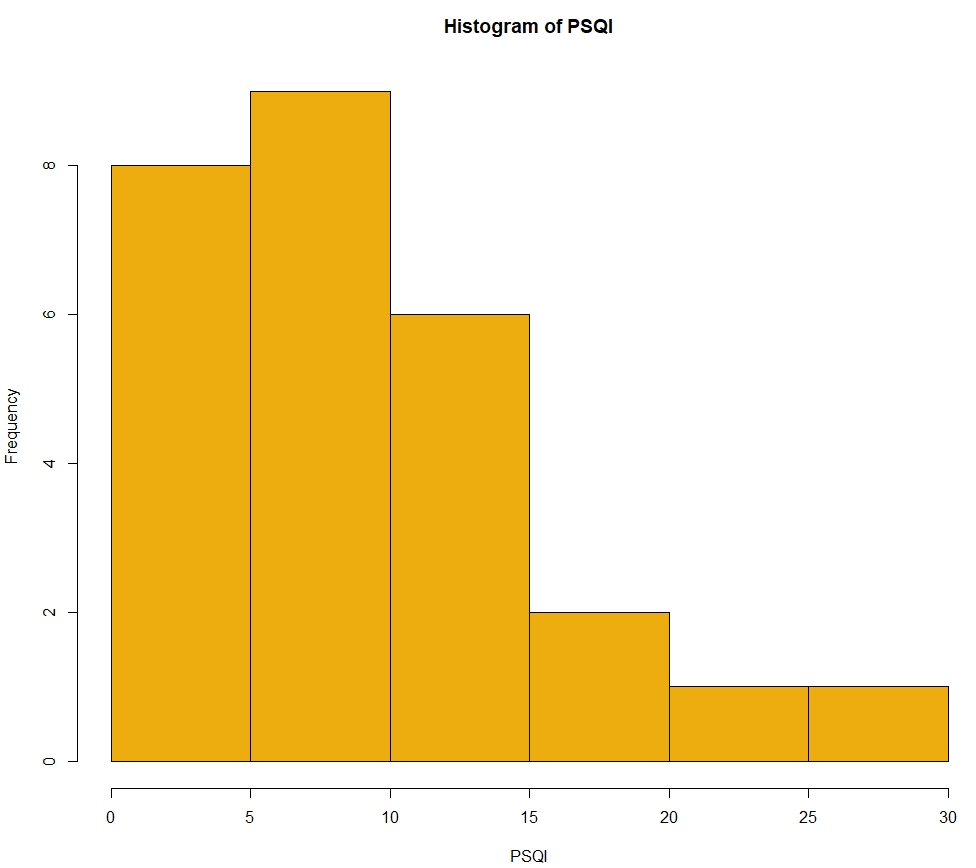
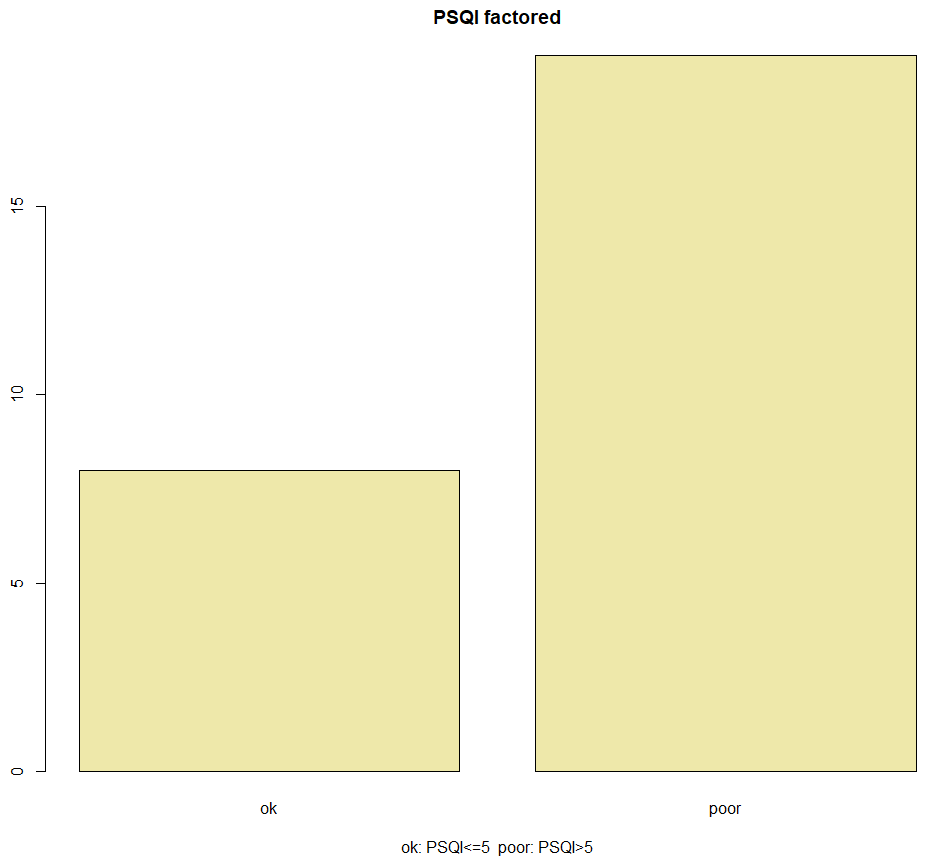
Psychometric evaluation and feasibility of the Greek

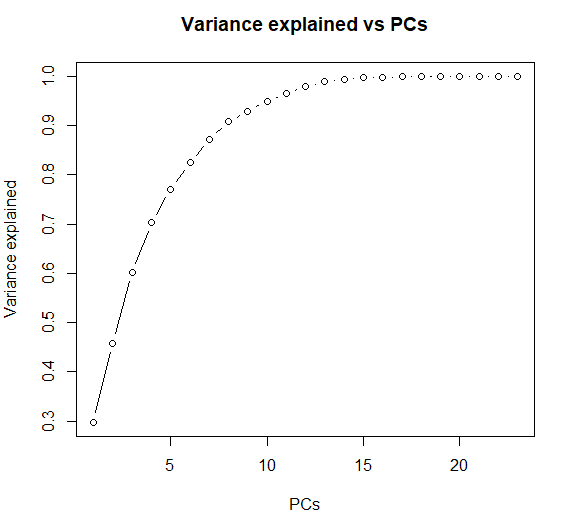
Pittsburgh Sleep Quality Index (GR-PSQI) in patients

with cancer receiving chemotherapy









Stratified 10 fold CV:

<http://web.cs.iastate.edu/~jtian/cs573/Papers/Kohavi-IJCAI-95.pdf>

## Selection of features

Correlation is a bivariate analysis that measures the strength of association between two variables and the direction of the relationship.  In terms of the strength of relationship, the value of the correlation coefficient varies between +1 and -1.  A value of ± 1 indicates a perfect degree of association between the two variables.  As the correlation coefficient value goes towards 0, the relationship between the two variables will be weaker.  The direction of the relationship is indicated by the sign of the coefficient; a + sign indicates a positive relationship and a – sign indicates a negative relationship. Usually, in statistics, we measure four types of correlations: [Pearson correlation](http://www.statisticssolutions.com/academic-solutions/membership-resources/member-profile/conducting-analyses-results/videos/pearson-correlation/), Kendall rank correlation, Spearman correlation, and the Point-Biserial correlation.

**Pearson**

Pearson r correlation is the most widely used correlation statistic to measure the degree of the relationship between linearly related variables.  For example, in the stock market, if we want to measure how two stocks are related to each other, Pearson r correlation is used to measure the degree of relationship between the two.  The point-biserial correlation is conducted with the Pearson correlation formula except that one of the variables is dichotomous.  The following formula is used to calculate the Pearson r correlation:

pearson r correlation

*r = Pearson r correlation coefficient  
N = number of observations  
∑xy = sum of the products of paired scores  
∑x = sum of x scores  
∑y = sum of y scores  
∑x2= sum of squared x scores  
∑y2= sum of squared y scores*

Types of research questions a Pearson correlation can examine:

Is there a statistically significant relationship between age, as measured in years, and height, measured in inches?

Is there a relationship between temperature, measured in degrees Fahrenheit, and ice cream sales, measured by income?

Is there a relationship between job satisfaction, as measured by the JSS, and income, measured in dollars?

Assumptions

For the Pearson r correlation, both variables should be normally distributed (normally distributed variables have a bell-shaped curve).  Other assumptions include linearity and homoscedasticity.  Linearity assumes a straight line relationship between each of the two variables and homoscedasticity assumes that data is equally distributed about the regression line.

**Spearman**

In [statistics](https://en.wikipedia.org/wiki/Statistics), Spearman's rank correlation coefficient or Spearman's rho, named after [Charles Spearman](https://en.wikipedia.org/wiki/Charles_Spearman) and often denoted by the Greek letter ρ or as rs, is a [nonparametric](https://en.wikipedia.org/wiki/Non-parametric_statistics) measure of [rank correlation](https://en.wikipedia.org/wiki/Rank_correlation) ([statistical dependence](https://en.wikipedia.org/wiki/Correlation_and_dependence) between the [rankings](https://en.wikipedia.org/wiki/Ranking) of two [variables](https://en.wikipedia.org/wiki/Variable_(mathematics)#Applied_statistics)). It assesses how well the relationship between two variables can be described using a [monotonic](https://en.wikipedia.org/wiki/Monotonic) function.

The Spearman correlation between two variables is equal to the [Pearson correlation](https://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient) between the rank values of those two variables; while Pearson's correlation assesses linear relationships, Spearman's correlation assesses monotonic relationships (whether linear or not). If there are no repeated data values, a perfect Spearman correlation of +1 or −1 occurs when each of the variables is a perfect monotone function of the other.

Intuitively, the Spearman correlation between two variables will be high when observations have a similar (or identical for a correlation of 1) [rank](https://en.wikipedia.org/wiki/Ranking_(statistics)) (i.e. relative position label of the observations within the variable: 1st, 2nd, 3rd, etc.) between the two variables, and low when observations have a dissimilar (or fully opposed for a correlation of −1) rank between the two variables.

Spearman rank correlation is a non-parametric test that is used to measure the degree of association between two variables.

The Spearman rank correlation test does not carry any assumptions about the distribution of the data and is the appropriate correlation analysis when the variables are measured on a scale that is at least ordinal.

The following formula is used to calculate the Spearman rank correlation:

spearman rank correlation

*ρ= Spearman rank correlation  
di= the difference between the ranks of corresponding variables  
n= number of observations*

Types of research questions a Spearman Correlation can examine:

Is there a statistically significant relationship between participants’ level of education (high school, bachelor’s, or graduate degree) and their starting salary?

Is there a statistically significant relationship between horse’s finishing position a race and horse’s age?

Assumptions

The assumptions of the Spearman correlation are that data must be at least ordinal and the scores on one variable must be monotonically related to the other variable.

Elements of Information theory MEASURING DEPENDENCE VIA MUTUAL INFORMATION By Shan Lu

**Mutual Information**

Mutual information satisfies properties that make it an ideal measure of stochastic dependence [Cover and Thomas, 1991, Darbellay, 1999, Joe, 1989b], [R´enyi, 1959].

Unlike Pearson’s linear correlation coefficient which accounts only for linear relationships, or other well-known rank correlation coefficients that can detect monotonic dependencies, the mutual information takes into account all types of dependence.

Variables which are not statistically independent suggest the existence of some functional relation between them. While there are several approaches to quantify the linear dependence between variables, the framework of information theory (Shannon, 1948) provides a general measure of dependencies between variables. In particular, a vanishing Pearson correlation does not imply that two variables are independent. The mutual information therefore provides a better and more general criterion to investigate relationships between variables.

Considerable research has been done on measuring dependence between random variables.

The correlation coefficient [10] is the most widely studied linear measure of dependence. However, the limitation of linearity limits its application. The informational coefficient of correlation [17] is defined in terms of mutual information.

However, it is well known that correlation is not equivalent to dependence. Two independent random variables are surely uncorrelated, which means that their correlation coefficient is zero; yet, for uncorrelated random variables, they are not necessarily independent [10].

Mutual information is a concept from information theory first introduced by Shannon in the context of digital communication [23]. It describes how much information two random variables share with each other, i.e. the amount of uncertainty about one random variable given knowledge of the other random variable. The mutual information for two random variables is symmetric and always nonnegative.

It equals zero if and only if the two random variables are independent. In addition, the mutual information between two continuous random variables equals infinity if there is a functional relationship between these two random variables.

These properties provide a possibility for the mutual information to be used as a dependence measure.

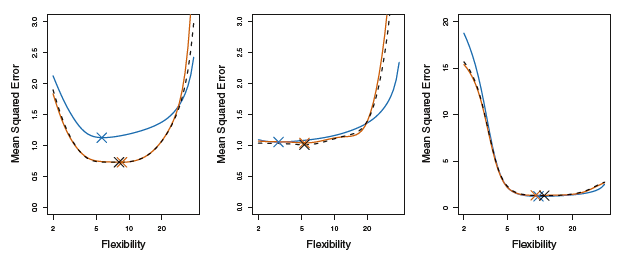
*Hence, the mutual information can be interpreted as the reduction in the uncertainty of X (resp. Y ) due to the knowledge of Y (resp. X ) [Ullah, 1996].*

Από ISLR seventh print

**Cross validation**

Resampling methods are an indispensable tool in modern statistics. They involve repeatedly drawing samples from a training set and refitting a model of interest on each sample in order to obtain additional information about

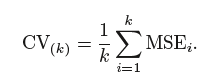
the fitted model. For example, in order to estimate the variability of a linear regression fit, we can repeatedly draw different samples from the training data, fit a linear regression to each new sample, and then examine the extent to which the resulting fits differ. Such an approach may allow us to obtain information that would not be available from fitting the model only once using the original training sample.

Resampling approaches can be computationally expensive, because they involve fitting the same statistical method multiple times using different subsets of the training data. However, due to recent advances in computing power, the computational requirements of resampling methods generally are not prohibitive. In this chapter, we discuss one of the most commonly used resampling methods, cross-validation. This method is an important tool in the practical application of many statistical learning procedures. For example, cross-validation can be used to estimate the test error associated with a given statistical learning method in order to evaluate its performance, or to select the appropriate level of flexibility. The process of evaluating a model’s performance is known as model assessment, whereas the process of selecting the proper level of flexibility for a model is known as model selection.

**k-Fold Cross-Validation**

k-fold CV is an approach that involves randomly dividing the set of observations into k groups, or folds, of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining k − 1 folds. The mean squared error, MSE1, is then computed on the observations in the held-out fold. This procedure is repeated k times; each time, a different group of observations is treated as a validation set. This process results in k estimates of the test error, MSE1,MSE2,… ,MSEk.

The k-fold CV estimate is computed by averaging

these values:

It is not hard to see that LOOCV (leave-one-out CV) is a special case of k-fold CV in which k is set to equal n. In practice, one typically performs k-fold CV using k = 5 or k = 10. What is the advantage of using k = 5 or k = 10 rather than k = n? The most obvious advantage is computational. LOOCV requires fitting the statistical learning method n times. This has the potential to be computationally expensive. But cross-validation is a very general approach that can be applied to almost any statistical learning method. Some statistical learning methods have computationally intensive fitting procedures, and so performing LOOCV may pose computational problems, especially if n is extremely large. In contrast, performing 10-fold CV requires fitting the learning procedure only ten times, which may be much more feasible.

When we perform cross-validation, our goal might be to determine how well a given statistical learning procedure can be expected to perform on independent data; in this case, the actual estimate of the test MSE is of interest. But at other times we are interested only in the location of the minimum point in the estimated test MSE curve. This is because we

might be performing cross-validation on a number of statistical learning methods, or on a single method using different levels of flexibility, in order to identify the method that results in the lowest test error. For this purpose, the location of the minimum point in the estimated test MSE curve is important, but the actual value of the estimated test MSE is not. We find in the following Figure that despite the fact that they sometimes underestimate the true test MSE, all of the CV curves come close to identifying the correct level of flexibility—that is, the flexibility level corresponding to the smallest test MSE.

*The true test MSE is shown in blue, the LOOCV estimate is shown as a black dashed line, and the 10-fold CV estimate is shown in orange. The crosses indicate the minimum of each of the MSE curves.*

**Bias-Variance Trade-Off for k-Fold Cross-Validation**

We mentioned that k-fold CV with k < n has a computational advantage to LOOCV. But putting computational issues aside, a less obvious but potentially more important advantage of k-fold CV is that it often gives more accurate estimates of the test error rate than does LOOCV. This has to do with a bias-variance trade-off. It is not hard to see that LOOCV will give approximately unbiased estimates of the test error, since each training set contains n − 1 observations, which is almost as many as the number of observations in the full data set. And performing k-fold CV for, say, k = 5 or k = 10 will lead to an intermediate level of bias, since each training set contains (k − 1)n/k observations—fewer than in the LOOCV approach, but substantially more than in the validation set approach.

Therefore, from the perspective of bias reduction, it is clear that LOOCV is to be preferred to k-fold CV.

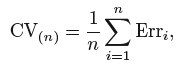
However, we know that bias is not the only source for concern in an estimating procedure; we must also consider the procedure’s variance. It turns out that LOOCV has higher variance than does k-fold CV with k < n.

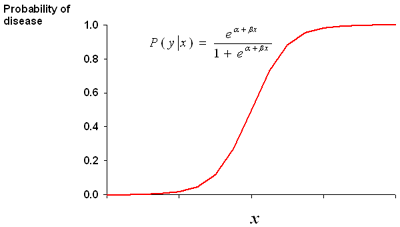
Why is this the case? When we perform LOOCV, we are in effect averaging the outputs of n fitted models, each of which is trained on an almost identical set of observations; therefore, these outputs are highly (positively) correlated with each other. In contrast, when we perform k-fold CV with k < n, we are averaging the outputs of k fitted models that are somewhat less correlated with each other, since the overlap between the training sets in each model is smaller. Since the mean of many highly correlated quantities has higher variance than does the mean of many quantities that are not as highly correlated, the test error estimate resulting from LOOCV tends to have higher variance than does the test error estimate resulting from k-fold CV.

To summarize, there is a bias-variance trade-off associated with the choice of k in k-fold cross-validation. Typically, given these considerations, one performs k-fold cross-validation using k = 5 or k = 10, as these values have been shown empirically to yield test error rate estimates that suffer neither from excessively high bias nor from very high variance.

**Cross-Validation on Classification Problems**

So far, we have illustrated the use of cross-validation in the regression setting where the outcome Y is quantitative, and so have used MSE to quantify test error. But cross-validation can also be a very useful approach in the classification setting when Y is qualitative. In this setting, cross-validation works just as described earlier, except that rather than using MSE to quantify test error, we instead use the number of misclassified observations.

 For instance, in the classification setting, the LOOCV error rate takes the form:





where

The k-fold CV error rate and validation set error

rates are defined analogously.

## Classification Algorithms

Mastering Machine Learning with R Cory Lesmeister

Feature Selection and Ensemble Methods for Bioinformatics: Algorithmic Classification and Implementations

**Logistic regression**

[Logistic regression](http://www.statisticssolutions.com/academic-solutions/membership-resources/member-profile/data-analysis-plan-templates/data-analysis-plan-logistic-regression/) is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary).  Like all regression analyses, the logistic regression is a predictive analysis.  Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

In [statistics](https://en.wikipedia.org/wiki/Statistics), the logistic model (or logit model) is a widely used [statistical model](https://en.wikipedia.org/wiki/Statistical_model) that in its basic form uses a [logistic function](https://en.wikipedia.org/wiki/Logistic_function) to model a [binary](https://en.wikipedia.org/wiki/Binary_variable) [dependent variable](https://en.wikipedia.org/wiki/Dependent_variable), although many more complex [extensions](https://en.wikipedia.org/wiki/Logistic_regression#Extensions) exist. In [regression analysis](https://en.wikipedia.org/wiki/Regression_analysis), logistic regression (or logit regression) is [estimating](https://en.wikipedia.org/wiki/Estimation_theory) the parameters of a logistic model (a form of [binary regression](https://en.wikipedia.org/wiki/Binary_regression)).

Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail, win/lose, alive/dead or healthy/sick; these are represented by an [indicator variable](https://en.wikipedia.org/wiki/Indicator_variable), where the two values are labeled "0" and "1".

In the logistic model, the [log-odds](https://en.wikipedia.org/wiki/Log-odds) (the [logarithm](https://en.wikipedia.org/wiki/Logarithm) of the [odds](https://en.wikipedia.org/wiki/Odds)) for the value labeled "1" is a [linear combination](https://en.wikipedia.org/wiki/Linear_function_(calculus)) of one or more [independent variables](https://en.wikipedia.org/wiki/Independent_variable) ("predictors"); the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a [continuous variable](https://en.wikipedia.org/wiki/Continuous_variable) (any real value). The corresponding [probability](https://en.wikipedia.org/wiki/Probability) of the value labeled "1" can vary between 0 (certainly the value "0") and 1 (certainly the value "1"), hence the labeling; the function that converts log-odds to probability is the logistic function, hence the name. The [unit of measurement](https://en.wikipedia.org/wiki/Unit_of_measurement) for the log-odds scale is called a [logit](https://en.wikipedia.org/wiki/Logit" \o "Logit), from logistic unit, hence the alternative names.

Analogous models with a different [sigmoid function](https://en.wikipedia.org/wiki/Sigmoid_function) instead of the logistic function can also be used, such as the [probit model](https://en.wikipedia.org/wiki/Probit_model" \o "Probit model); the defining characteristic of the logistic model is that increasing one of the independent variables multiplicatively scales the odds of the given outcome at a constant rate, with each dependent variable having its own parameter; for a binary independent variable this generalizes the [odds ratio](https://en.wikipedia.org/wiki/Odds_ratio).

As previously discussed, our classification problem is best modeled with the probabilities that are bound by 0 and 1. We can do this for all of our observations with a number of different functions, but here we will focus on the logistic function. The logistic function used in logistic regression is as follows:



The logistic function can be turned to odds with the formulation of Probability(Y) / 1–Probability(Y). For instance, if the probability of Brazil winning the World Cup is 20 percent, then the odds are 0.2 / 1 - 0.2, which is equal to 0.25, translating to the odds of one in four.

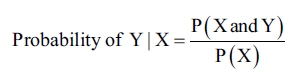
One way to look at the relationship of logistic regression with linear regression is to show logistic regression as the log odds or log (P(Y)/1–P(Y)) is equal to Bo+B1x. The coefficients are estimated using a maximum likelihood instead of the OLS. The intuition behind the maximum likelihood is that we are finding the estimates for Bo and B1 that will create a predicted probability for an observation that is as close as possible to the actual observed outcome of Y, a so-called likelihood. The R language does what other software packages do for the maximum likelihood, which is to find the optimal combination of beta values that maximize the likelihood.

**LDA**

Discriminant Analysis (DA), also known as Fisher Discriminant Analysis (FDA), is another popular classification technique. It can be an effective alternative to logistic regression when the classes are well-separated. If you have a classification problem where the outcome classes are well-separated, logistic regression can have unstable estimates, which is to say that the confidence intervals are wide and the estimates themselves would likely vary wildly from one sample to another (James, 2013). DA does not suffer from this problem, and as a result, may outperform and be more generalizable than logistic regression. Conversely, if there are complex relationships between the features and outcome variables, it may perform poorly on a classification task. For the purpose of comparison to logistic regression, we will explore Linear Discriminant Analysis (LDA).

DA utilizes Bayes' theorem in order to determine the probability of the class membership for each observation. If you have two classes, for example, benign and malignant, then DA will calculate an observation's probability for both the classes and select the highest probability as the proper class.

Bayes' theorem states that the probability of Y occurring- given that X has occurred- is equal to the probability of both Y and X occurring divided by the probability of X occurring, which can be written as:



The numerator in this expression is the likelihood that an observation is from that class level and has these feature values. The denominator is the likelihood of an observation that has these feature values across all the levels. Again, the classification rule says that if you have the joint distribution of X and Y and if X is given, the optimal decision of which class to assign an observation is by choosing the class with the larger probability (the posterior probability).

The process of attaining the posterior probabilities goes through the following steps:

1. Collect data with a known class membership.
2. Calculate the prior probabilities - this represents the proportion of the sample that belongs to each class (empirical probability).
3. Calculate the mean for each feature by their class.
4. Calculate the variance-covariance matrix for each feature; if it is an LDA, then this would be a pooled matrix of all the classes, giving us a linear classifier (if it is a QDA (Quadratic Discriminant Analysis), then a variance-covariance matrix is created for each class).
5. Estimate the normal distribution (Gaussian densities) for each class.
6. Compute the discriminant function that is the rule for the classification of a new object.
7. Assign an observation to a class based on the discriminant function.

This will provide an expanded notation on the determination of the posterior probabilities, as follows:

* πk=(# of samples in class k) / (total sample size) is the prior probability of a randomly chosen observation in the kth class.
* fk(X) = P(X=x | y=k) is the density function of an observation that comes from the kth class. We will assume that this comes from a normal (Gaussian) distribution; with multiple features, the assumption is that it comes from a multivariate Gaussian distribution.
* Using pk(X) = probability of Y given X, we can adjust Bayes' theorem accordingly.

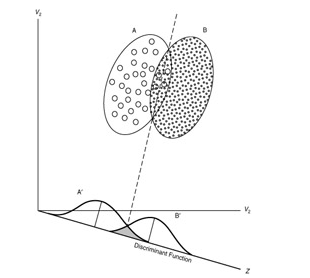


* is the posterior probability that an observation comes from the k class when the feature values for this observation are given.

* Assuming that k=2 and the prior probabilities are the same, π1=π2, then an observation is assigned to the one class if 2x(μ1-μ2)>μ12-μ22, otherwise it is assigned to the two class. This is known as the decision boundary. DA creates the k-1 decision boundaries, that is, with three classes (k=3), there will be two decision boundaries.

Even though LDA is elegantly simple, it has the limitation of the assumption that the observations of each class are said to have a multivariate normal distribution and there is a common covariance across the classes.

QDA still assumes that the observations come from a normal distribution, but also assumes that each class has its own covariance



*The separation of two classes with DA*

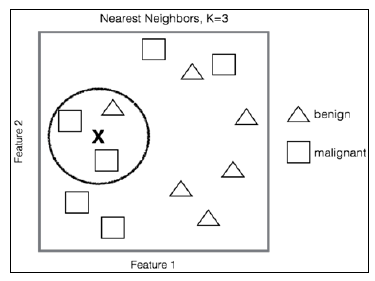
When you relax the common covariance assumption, you now allow quadratic terms into the discriminant score calculations, which was not possible with LDA. The important part ιs that QDA is a more flexible technique than logistic regression, but we must keep in mind our bias-variance trade-off. With a more flexible technique, you are likely to have a lower bias but potentially a higher variance. Like a lot of flexible techniques, a robust set of training data is needed to mitigate a high classifier variance.

**K-Nearest Neighbors**

With KNN, we have no parameters as the learning method is the so-called instance-based learning. In short, the labeled examples (inputs and corresponding output labels) are stored and no action is taken until a new input pattern demands an output value *(Battiti and Brunato, 2014, p. 11)*. This method is commonly called lazy learning as no specific model parameters are produced. The train instances themselves represent the knowledge. For the prediction of any new instance (a new data point), the train data is searched for an instance that most resembles the new instance in question. KNN does this for a classification problem by looking at the closest points - the nearest neighbors - to determine the proper class. The k comes into play by determining how many neighbors should be examined by the algorithm, so if k=5, it will examine the five nearest points.

A weakness of this method is that all five points are given equal weight in the algorithm even if they are less relevant in learning.

The best way to understand how this works is with a simple visual example on a binary classification learning problem. In the following figure, we have a plot of whether a tumor is benign or malignant based on two predictive features. The X in the plot indicates a new observation that we would like to predict. If our algorithm considers K=3, the circle encompasses the three observations that are nearest to the one that we want to score. As the most commonly occurring classifications are malignant, the X data point is classified as malignant, as shown in the following figure:



Even from this simple example, it is clear that the selection of k for the Nearest Neighbors is critical. If k is too small, then you may have a high variance on the test set observations even though you have a low bias. On the other hand, as k grows you may decrease your variance but the bias may be unacceptable. Cross-validation is necessary to determine the proper k.

It is also important to point out the calculation of the distance or the nearness of the data points in our feature space. The default distance is Euclidian Distance. This is simply the straight-line distance from point A to point B or you can utilize the formula that it is equivalent to the square root of the sum of the squared differences between the corresponding points. The formula for Euclidian Distance, given point A and B with coordinates p1, p2, …, pn and q1, q2,…, qn respectively, would be as follows:

This distance is highly dependent on the scale that the features were measured on and so it is critical to standardize them. Other distance calculations can be used as well as weights depending on the distance.

Islr seventh printing

Masterin machine learning with r

**Support Vector Machines**

In [machine learning](https://en.wikipedia.org/wiki/Machine_learning), support-vector machines (SVMs, also support-vector networks) are [supervised learning](https://en.wikipedia.org/wiki/Supervised_learning) models with associated learning [algorithms](https://en.wikipedia.org/wiki/Algorithm) that analyze data used for [classification](https://en.wikipedia.org/wiki/Statistical_classification) and [regression analysis](https://en.wikipedia.org/wiki/Regression_analysis). Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-[probabilistic](https://en.wikipedia.org/wiki/Probabilistic_classification) [binary](https://en.wikipedia.org/wiki/Binary_classifier) [linear classifier](https://en.wikipedia.org/wiki/Linear_classifier). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

The support vector machine is a generalization of a simple and intuitive classifier called the maximal margin classifier.

People often loosely refer to the maximal margin classifier, the support vector classifier, and the support vector machine as “support vector machines”.

There are three aspects to the Support Vector Machines:

* Geometric:

Maximizing Margin

* Kernel Methods:

Making nonlinear decision boundaries linear efficiently

* Capacity:

Structural Risk Minimization

SVM history concisely:

* SVM is a classifier derived from statistical learning theory by Vapnik and Chervonenkis
* SVM was first introduced by Boser, Guyon and Vapnik in COLT-92
* Screenshot_3.pngSVM became famous when, using pixel maps as input, it gave accuracy comparable to NNs with hand-designed features in a handwriting recognition task
* SVM is closely related to:

Kernel machines (a generalization of SVMs), large margin classifiers, reproducing kernel Hilbert space, Gaussian process, Boosting

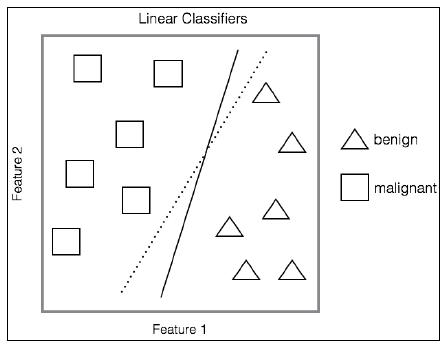
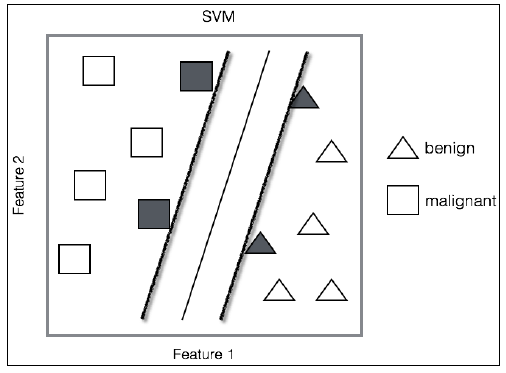
More formally, a support-vector machine constructs a [hyperplane](https://en.wikipedia.org/wiki/Hyperplane" \o "Hyperplane) or set of hyperplanes in a [high -](https://en.wikipedia.org/wiki/High-dimensional_space) or infinite - dimensional space, which can be used for [classification](https://en.wikipedia.org/wiki/Statistical_classification), [regression](https://en.wikipedia.org/wiki/Regression_analysis), or other tasks like outliers detection.

 Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin, the lower the [generalization error](https://en.wikipedia.org/wiki/Generalization_error) of the classifier.

Why maximal margin?

1. Intuitively this feels safest.
2. If we’ve made a small error in the location of the boundary (it’s been jolted in its perpendicular direction) this gives us least chance of causing a misclassification.
3. LOOCV is easy since the model is immune to removal of any non-support-vector datapoints.
4. There’s some theory (using VC dimension) that is related to (but not the same as) the proposition that this is a good thing.
5. Empirically it works very well.

*SVMs have been shown to perform well in a variety of settings, and are often considered one of the best "out of the box" classifiers.(James, G., 2013)*

****

*The first figure shows a linear classifier for the two classes, while the second shows the maximal margin classifier*

The real-world problems are not so clear cut. In data that is not linearly separable, many observations will fall on the wrong side of the margin (the so-called slack variables), which is a misclassification. The key to building an SVM algorithm is to solve the optimal number of support vectors via cross-validation. Any observation that lies directly on the wrong side of the margin for its class is known as a support vector. If the tuning parameter for the number of errors is too large, which means that you have many support vectors, you will suffer from a high bias and low variance. On the other hand, if the tuning parameter is too small, then the opposite might occur. According to James et al. who refers to the tuning parameter as C, as C decreases, the tolerance for observations being on the wrong side of the margin decreases, and the margin narrows. This C, or rather, cost function, simply allows for observations to be on the wrong side of the margin. If C were set to zero, then we would prohibit a solution where any observations violate the margin.

Another important aspect of SVM is the ability to model nonlinearity with quadratic or higher order polynomials of the input features. In SVMs, this is known as the kernel trick. These can be estimated and selected with cross-validation.

The kernel trick with SVMs allows us to efficiently expand the feature space with the goal that you achieve an approximate linear separation.

To check out how this is done, first look at the SVM optimization problem and its constraints. We are trying to achieve the following:

* Create weights that maximize the margin
* Subject to the constraints, no (or as few as possible) data points should lie within that margin

Now, unlike linear regression where each observation is multiplied by a weight, in SVM, the weights are applied to the inner products of just the support vector observations.

An inner product for two vectors is just the sum of the paired observations' product. With SVMs, if we take a possibility that an inner product of each observation has an inner product of every other observation, this amounts to the formula that there would be n(n-1)/2 combinations where n is the number of observations. With just 10 observations, we end up with 45 inner products. However, SVM only concerns itself with the support vectors' observations and their corresponding weights.

For a linear SVM classifier, the formula is the following:

**

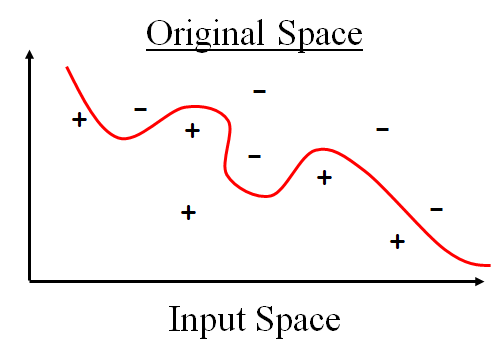
Where (x, xi) are the inner products of the support vectors as α is non-zero only when an observation is a support vector.

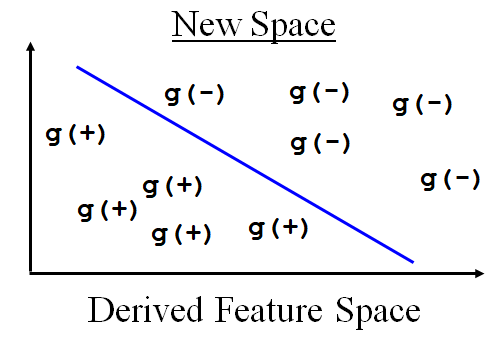
This leads to far fewer terms in the classification algorithm and allows the use of the kernel function, commonly referred to as the kernel trick.

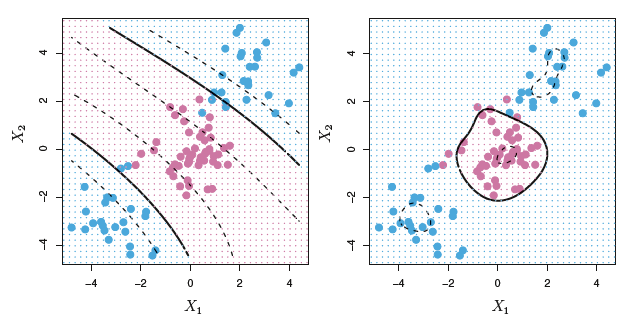
The trick in this is that the kernel function mathematically summarizes the transformation of the features in higher dimensions instead of creating them explicitly. This has the benefit of creating the higher dimensional, nonlinear space and decision boundary while keeping the optimization problem computationally efficient. The kernel functions compute the inner product in a higher dimensional space without transforming them into the higher dimensional space.

The notation for popular kernels is expressed as the inner (dot) product of the features, with xi and xj representing vectors, gamma, and c parameters, as follows:

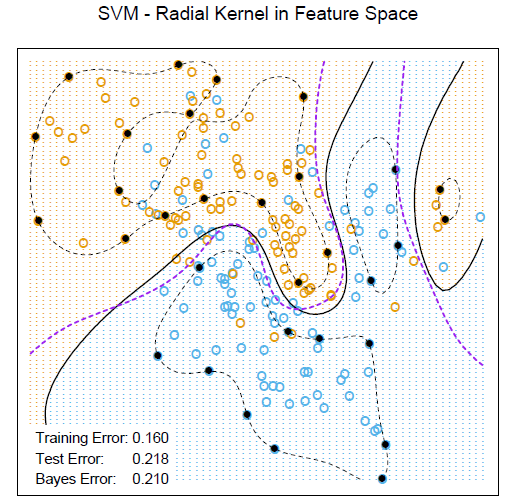
* linear with no transformation: K(xi,xj) = xi · xj
* polynomial where d is equal to the degree of the polynomial: K(xi,xj) = (γ xi·xj+c)d
* radial basis function: K(xi,xj) = exp(-γ(xi-xj)2)
* sigmoid function: K(xi,xj) = tanh(γxi· xj+c)





The hyperplanes in the higher-dimensional space are defined as the set of points whose dot product with a vector in that space is constant, where such a set of vector is an orthogonal (and thus minimal) set of vectors that defines a hyperplane. The vectors defining the hyperplanes can be chosen to be linear combinations with parameters ai of images of [feature vectors](https://en.wikipedia.org/wiki/Feature_vector) xi that occur in the data base. With this choice of a hyperplane, the points x in the [feature space](https://en.wikipedia.org/wiki/Feature_space) that are mapped into the hyperplane are defined by the relation

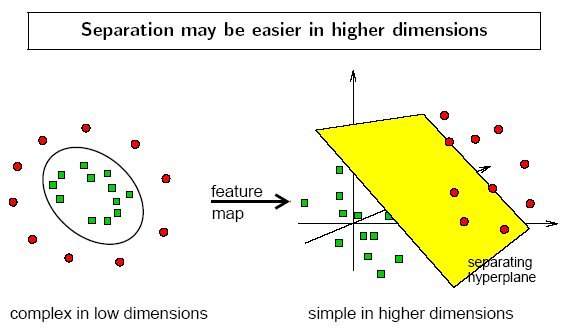
 Note that if  k(x,y) becomes small as  y grows further away from  x, each term in the sum measures the degree of closeness of the test point x to the corresponding data base point xi. In this way, the sum of kernels above can be used to measure the relative nearness of each test point to the data points originating in one or the other of the sets to be discriminated. Note the fact that the set of points x mapped into any hyperplane can be quite convoluted as a result, allowing much more complex discrimination between sets that are not convex at all in the original space.



*Nonlinear SVM for the mixture data. It uses a radial basis kernel (with γ = 1). In each case C was tuned to approximately achieve the best test error performance, and C = 1 worked well. The radial basis kernel performs the best (close to Bayes optimal), as might be expected given the data arise from mixtures of Gaussians. The broken purple curve in the background is the Bayes decision boundary.*

*Left: An SVM with a polynomial kernel of degree 3 is applied to the non-linear data from Figure 9.8, resulting in a far more appropriate decision rule. Right: An SVM with a radial kernel is applied. In this example, either kernel is capable of capturing the decision boundary.*

How does the radial kernel actually work? If a given test observation x\* = (x1\* … xp\*)T is far from a training observation xi in terms of Euclidean distance, then will be large, and so K(xi, xj) =exp(−γ2 will be very tiny. This means that xi will play virtually no role in f(x\*). Recall that the predicted class label for the test observation x\* is based on the sign of f(x\*). In other words, training observations that are far from x\* will play essentially no role in the predicted class label for x\*. This means that the radial kernel has very local behavior, in the sense that only nearby training observations have an effect on the class label of a test observation.



TUNING PARAMETERS

Parameters are arguments that you pass when you create your classifier. Following are the important parameters for SVM-

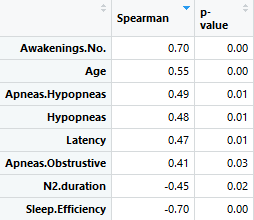
**C:** It controls the trade off between smooth decision boundary and classifying training points correctly. A large value of c means you will get more training points correctly.

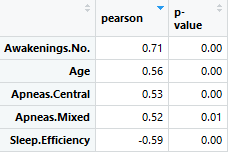
**Gamma:** It defines how far the influence of a single training example reaches. If it has a low value it means that every point has a far reach and conversely high value of gamma means that every point has close reach. If gamma has a very high value, then the decision boundary is just going to be dependent upon the points that are very close to the line which effectively results in ignoring some of the points that are very far from the decision boundary. This is because the closer points get more weight and it results in a wiggly curve as shown in previous graph. On the other hand, if the gamma value is low even the far away points get considerable weight and we get a more linear curve.

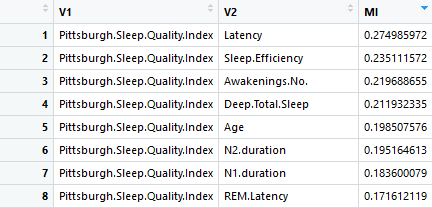
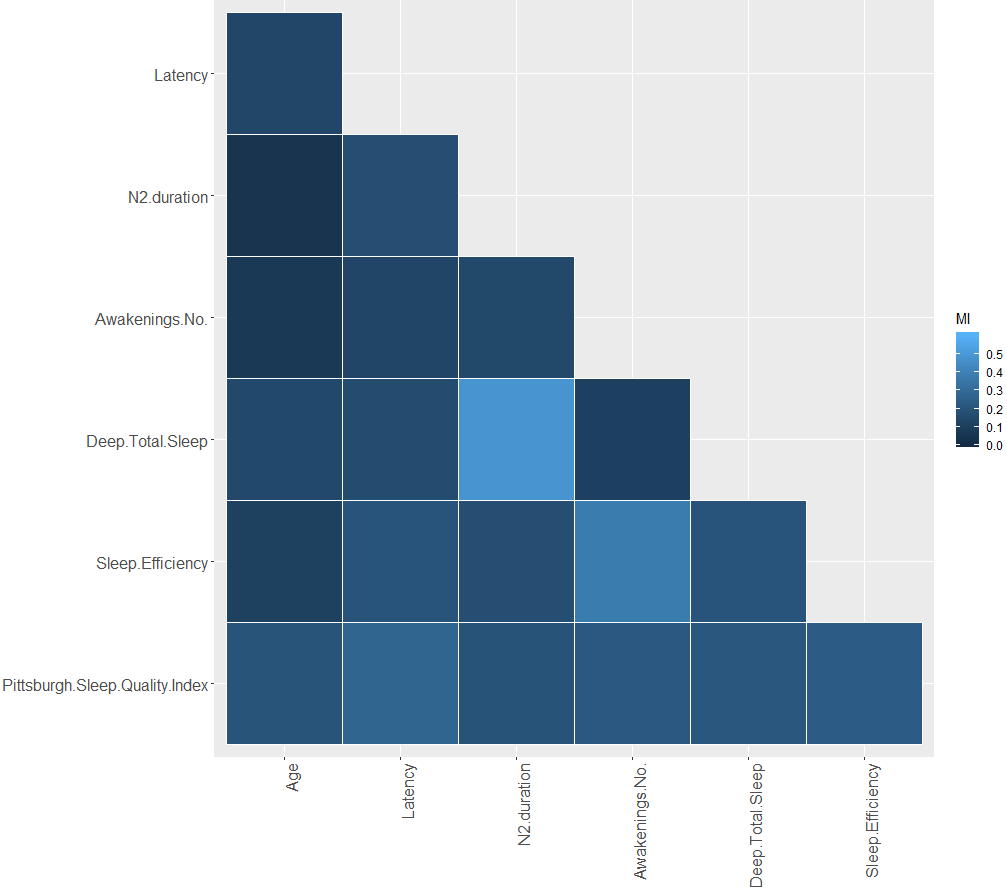
## Burges. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2(2):955-974, 1998.

The VC/SRM/SVM Bible:

*Statistical Learning Theory by Vladimir Vapnik, Wiley-Interscience; 1998*







# Results

The classifiers were evaluated through a 10-fold Cross Validation such that 90% of the data is used as training set and 10% as a validation set, each time.

The k-Nearest Neighbours algorithm was implemented for k=3. Its accuracy varied between 66 to 74%.

The Linear Discriminant Analysis algorithm was implemented, with predictors the following variables: Age, Latency, N2.duration, Sleep.Efficiency. The accuracy was 78%.

The Logistic Regression resulted in 76.7% accuracy, when the variables Age, Latency, N2.duration, Sleep.Efficiency were used as predictors.

The Support Vector Machine with linear kernel (cost=1) resulted in 77 to 83 % accuracy.

Finally, the Support Vector Machine with Gaussian kernel (gamma=0.31, cost=8) resulted in 80 to 87% accuracy. The predictors in the SVM algorithms were: Age, Latency, N3.duration.

## Equations

# Units

# Some Common Mistakes

A general IEEE styleguide is available at <http://www.ieee.org/web/publications/authors/transjnl/index.html>



Fig. 1. Magnetization as a function of applied field. Note that “Fig.” is abbreviated. There is a period after the figure number, followed by two spaces. It is good practice to explain the significance of the figure in the caption.

TABLE I

Units for Magnetic Properties

|  |  |  |
| --- | --- | --- |
| Symbol | Quantity | Conversion from Gaussian and  CGS EMU to SI a |
| Φ | magnetic flux | 1 Mx → 10−8 Wb = 10−8 V·s |
| *B* | magnetic flux density,  magnetic induction | 1 G → 10−4 T = 10−4 Wb/m2 |
| *H* | magnetic field strength | 1 Oe → 103/(4π) A/m |
| *m* | magnetic moment | 1 erg/G = 1 emu  → 10−3 A·m2 = 10−3 J/T |
| *M* | magnetization | 1 erg/(G·cm3) = 1 emu/cm3  → 103 A/m |
| 4π*M* | magnetization | 1 G → 103/(4π) A/m |
| σ | specific magnetization | 1 erg/(G·g) = 1 emu/g → 1 A·m2/kg |
| *j* | magnetic dipole  moment | 1 erg/G = 1 emu  → 4π × 10−10 Wb·m |
| *J* | magnetic polarization | 1 erg/(G·cm3) = 1 emu/cm3  → 4π × 10−4 T |
| χ*,* κ | susceptibility | 1 → 4π |
| χρ | mass susceptibility | 1 cm3/g → 4π × 10−3 m3/kg |
| μ | permeability | 1 → 4π × 10−7 H/m  = 4π × 10−7 Wb/(A·m) |
| μr | relative permeability | μ → μr |
| *w, W* | energy density | 1 erg/cm3 → 10−1 J/m3 |
| *N, D* | demagnetizing factor | 1 → 1/(4π) |

Vertical lines are optional in tables. Statements that serve as captions for the entire table do not need footnote letters.

aGaussian units are the same as cg emu for magnetostatics; Mx = maxwell, G = gauss, Oe = oersted; Wb = weber, V = volt, s = second, T = tesla, m = meter, A = ampere, J = joule, kg = kilogram, H = henry.

# Guidelines for Graphics Preparation and Submission

## Types of Graphics

The following list outlines the different types of graphics published in IEEE journals. They are categorized based on their construction, and use of color / shades of gray:

### Color/Grayscale figures

### Figures that are meant to appear in color, or shades of black/gray. Such figures may include photographs, illustrations, multicolor graphs, and flowcharts.

### Lineart figures

### Figures that are composed of only black lines and shapes. These figures should have no shades or half-tones of gray. Only black and white.

### Author photos

### Head and shoulders shots of authors which appear at the end of our papers.

### Tables Data charts which are typically black and white, but sometimes include color.

## Multipart figures

Figures compiled of more than one sub-figure presented side-by-side, or stacked. If a multipart figure is made up of multiple figure types (one part is lineart, and another is grayscale or color) the figure should meet the stricter guidelines.

## File Formats For Graphics

Format and save your graphics using a suitable graphics processing program that will allow you to create the images as PostScript (PS), Encapsulated PostScript (.EPS), Tagged Image File Format (.TIFF), Portable Document Format (.PDF), or Portable Network Graphics (.PNG) sizes them, and adjusts the resolution settings. If you created your source files in one of the following programs you will be able to submit the graphics without converting to a PS, EPS, TIFF, PDF, or PNG file: Microsoft Word, Microsoft PowerPoint, or Microsoft Excel. Though it is not required, it is recommended that these files be saved in PDF format rather than DOC, XLS, or PPT. Doing so will protect your figures from common font and arrow stroke issues that occur when working on the files across multiple platforms. When submitting your final paper, your graphics should all be submitted individually in one of these formats along with the manuscript.

## Sizing of Graphics

Most charts, graphs, and tables are one column wide (3.5 inches / 88 millimeters / 21 picas) or page wide (7.16 inches / 181 millimeters / 43 picas). The maximum depth a graphic can be is 8.5 inches (216 millimeters / 54 picas). When choosing the depth of a graphic, please allow space for a caption. Figures can be sized between column and page widths if the author chooses, however it is recommended that figures are not sized less than column width unless when necessary.

There is currently one publication with column measurements that don’t coincide with those listed above. Proceedings of the IEEE has a column measurement of 3.25 inches (82.5 millimeters / 19.5 picas).

The final printed size of author photographs is exactly   
1 inch wide by 1.25 inches tall (25.4 millimeters x 31.75 millimeters / 6 picas x 7.5 picas). Author photos printed in editorials measure 1.59 inches wide by 2 inches tall (40 millimeters x 50 millimeters / 9.5 picas x 12 picas).

# Conclusion

## A conclusion section is required for TBME. Although a conclusion may review the main points of the paper, do not replicate the abstract as the conclusion. A conclusion might elaborate on the major findings and significance of the work or suggest applications and extensions. Do not exceed 300 words for the conclusion section.

Appendix

Appendixes, if needed, appear before the acknowledgment. Substantial mathematical derivations should be presented in an appendix or in online supplementary materials.

Acknowledgment

The preferred spelling of the word “acknowledgment” in American English is without an “e” after the “g.” Use the singular heading even if you have many acknowledgments. Avoid expressions such as “One of us (S.B.A.) would like to thank ... .” Instead, write “F. A. Author thanks ... .” In most cases, sponsor and financial support acknowledgments are placed in the unnumbered footnote on the first page, not here.

References and Footnotes

## References

References need not be cited in text. When they are, number citations on the line, in square brackets inside the punctuation. Multiple references are each numbered with separate brackets. When citing a section in a book, please give the relevant page numbers. In text, refer simply to the reference number. Do not use “Ref.” or “reference” except at the beginning of a sentence: “Reference [3] shows ... .” Please do not use automatic endnotes in *Word*, rather, type the reference list at the end of the paper using the “References” style.

Reference numbers are set flush left and form a column of their own, hanging out beyond the body of the reference. The reference numbers are on the line, enclosed in square brackets. In all references, the given name of the author or editor is abbreviated to the initial only and precedes the last name. For an article with 3 or more authors, state “A. Author et al.”. Use commas around Jr., Sr., and III in names. Abbreviate conference titles. When citing IEEE transactions, provide the issue number, page range, volume number, year, and/or month if available. When referencing a patent, provide the day and the month of issue, or application. References may not include all information; please obtain and include relevant information. Do not combine references. There must be only one reference with each number. If there is a URL included with the print reference, it can be included at the end of the reference.

Other than books, capitalize only the first word in a paper title, except for proper nouns and element symbols. For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation See the end of this document for formats and examples of common references. For a complete discussion of references and their formats, see “The IEEE Style Manual,” available as a PDF link off the *Author Digital Toolbox* main page.

## Footnotes

Number footnotes separately in superscripts (Insert | Footnote).[[2]](#footnote-2) Place the actual footnote at the bottom of the column in which it is cited; do not put footnotes in the reference list (endnotes). Use letters for table footnotes (see Table I).

# Submitting Your Paper for Review

# Publication Principles

References

*Basic format for books:*

1. J. K. Author, “Title of chapter in the book,” in *Title of His Published Book, x*th ed. City of Publisher, Country if not USA: Abbrev. of Publisher, year, ch. *x*, sec. *x*, pp. *xxx–xxx.*

*Examples:*

1. G. O. Young, “Synthetic structure of industrial plastics,” in *Plastics,* 2nd ed., vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.
2. W.-K. Chen, *Linear Networks and Systems.* Belmont, CA: Wadsworth, 1993, pp. 123–135.

*Basic format for periodicals:*

1. J. K. Author, “Name of paper,” *Abbrev. Title of Periodical*, vol. *x,* no. *x,* pp*. xxx-xxx,* Abbrev. Month, year.

*Examples:*

1. J. U. Duncombe, “Infrared navigation—Part I: An assessment   
   of feasibility,” *IEEE Trans. Electron Devices*, vol. ED-11, no. 1, pp. 34–39, Jan. 1959.
2. E. P. Wigner, “Theory of traveling-wave optical laser,” *Phys. Rev*.,   
   vol. 134, pp. A635–A646, Dec. 1965.
3. E. H. Miller et al., “A note on reflector arrays,” *IEEE Trans. Antennas Propagat*., to be published.

*Basic format for handbooks:*

1. *Name of Manual/Handbook*, *x* ed., Abbrev. Name of Co., City of Co., Abbrev. State, year, pp. *xxx-xxx.*

*Examples:*

1. *Transmission Systems for Communications*, 3rd ed., Western Electric Co., Winston-Salem, NC, 1985, pp. 44–60.
2. *Motorola Semiconductor Data Manual*, Motorola Semiconductor Products Inc., Phoenix, AZ, 1989.

*Basic format for books (when available online):*

1. Author. (year, month day). *Title.* (edition) [Type of medium]. *volume (issue).* Available: site/path/file

*Example:*

1. J. Jones. (1991, May 10). *Networks.* (2nd ed.) [Online]. Available: [http://www.atm.com](http://www.atm.com/)

*Basic format for journals (when available online):*

1. Author. (year, month). Title. *Journal.* [Type of medium]. *volume (issue),* pages. Available: site/path/file

*Example:*

1. R. J. Vidmar. (1992, Aug.). On the use of atmospheric plasmas as electromagnetic reflectors. *IEEE Trans. Plasma Sci.* [Online]. *21(3),* pp. 876–880. Available:<http://www.halcyon.com/pub/journals/21ps03-vidmar>

*Basic format for papers presented at conferences (when available online):*

1. Author. (year, month). Title. Presented at Conference title. [Type of Medium]. Available: site/path/file

*Example:*

1. PROCESS Corp., MA. Intranets: Internet technologies deployed behind the firewall for corporate productivity. Presented at   
   INET96 Annual Meeting. [Online]. Available: <http://home.process.com/Intranets/wp2.htp>

*Basic format for reports and handbooks (when available online):*

1. Author. (year, month). Title. Comp an y . C ity, State or Country. [Type of Medium]. Available: site/path/file

*Example:*

1. S. L. Tall een. (1996 , Apr . ). The In t r an et Archi -tecture: M a nagi ng i n f o rm at i on i n t h e ne w paradigm. Amdahl Corp., CA. [Online]. Available:<http://www.amdahl.com/doc/products/bsg/intra/infra/html>

*Basic format for computer programs and electronic documents (when available online):* ISO recommends that capitalization follow the accepted practice for the language or script in which the information is given.

*Example:*

1. A. Harriman. (1993, June). Compendium of genealogical software. *Humanist.* [Online]. Available e-mail: [HUMANIST@NYVM.ORG](mailto:HUMANIST@NYVM.ORG) Message: get GENEALOGY REPORT

*Basic format for patents (when available online):*

1. Name of the invention, by inventor’s name. (year, month day). *Patent Number* [Type of medium]. Available: site/path/file

*Example:*

1. Musical toothbrush with adjustable neck and mirror, by L.M.R. Brooks. (1992, May 19). *Patent D 326 189*

[Online]. Available: NEXIS Library: LEXPAT File: DESIGN

*Basic format for conference proceedings (published):*

1. J. K. Author, “Title of paper,” in *Abbreviated Name of Conf.*, City of Conf., Abbrev. State (if given), year, pp. *xxxxxx.*

*Example:*

1. D. B. Payne and J. R. Stern, “Wavelength-switched pas- sively coupled single-mode optical network,” in *Proc. IOOC-ECOC,* 1985,   
   pp. 585–590.

*Example for papers presented at conferences (unpublished):*

1. D. Ebehard and E. Voges, “Digital single sideband detection for interferometric sensors,” presented at the 2nd Int. Conf. Optical Fiber Sensors, Stuttgart, Germany, Jan. 2-5, 1984.

*Basic format for patents:*

1. J. K. Author, “Title of patent,” U.S. Patent *x xxx xxx*, Abbrev. Month, day, year.

*Example:*

1. G. Brandli and M. Dick, “Alternating current fed power supply,”   
   U.S. Patent 4 084 217, Nov. 4, 1978.

*Basic format**for theses (M.S.) and dissertations (Ph.D.):*

1. J. K. Author, “Title of thesis,” M.S. thesis, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.
2. J. K. Author, “Title of dissertation,” Ph.D. dissertation, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.

*Examples:*

1. J. O. Williams, “Narrow-band analyzer,” Ph.D. dissertation, Dept. Elect. Eng., Harvard Univ., Cambridge, MA, 1993.
2. N. Kawasaki, “Parametric study of thermal and chemical nonequilibrium nozzle flow,” M.S. thesis, Dept. Electron. Eng., Osaka Univ., Osaka, Japan, 1993.

*Basic format for the most common types of unpublished references:*

1. J. K. Author, private communication, Abbrev. Month, year.
2. J. K. Author, “Title of paper,” unpublished.
3. J. K. Author, “Title of paper,” to be published.

*Examples:*

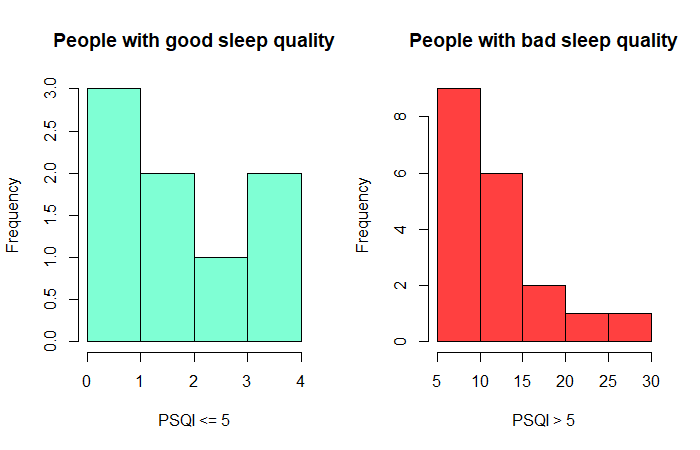
1. A. Harrison, private communication, May 1995.
2. B. Smith, “An approach to graphs of linear forms,” unpublished.
3. A. Brahms, “Representation error for real numbers in binary computer arithmetic,” IEEE Computer Group Repository, Paper R-67-85.

*Basic format for standards:*

1. *Title of Standard*, Standard number, date.

*Examples:*

1. IEEE Criteria for Class IE Electric Systems, IEEE Standard 308, 1969.
2. Letter Symbols for Quantities, ANSI Standard Y10.5-1968.



1. This paragraph of the first footnote will contain the date on which you submitted your paper for review. It will also contain support information, including sponsor and financial support acknowledgment. For example, “This work was supported in part by the U.S. National Science Foundation under Grant BS123456”.

   The next few paragraphs should contain the authors’ current affiliations. Include detailed surface address and e-mail address only for the corresponding author. For example, F. A. Author is with the National Institute of Standards and Technology, Boulder, CO, USA. S. B. Author, Jr., was with Rice University, Houston, TX, USA. He is now with Colorado State University, Fort Collins, CO, USA. \*T. C. Author is with the University of Colorado, Boulder, CO, USA (correspondence e-mail: author@uoc.edu). [↑](#footnote-ref-1)
2. It is recommended that footnotes be avoided (except for the unnumbered footnote with the receipt date on the first page). Instead, try to integrate the footnote information into the text. [↑](#footnote-ref-2)