**ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ**

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΜΑΘΗΜΑΤΙΚΩΝ



Machine Learning Algorithms in Bioinformatics: Pittsburgh Sleep Quality Index Classification

**Κωνσταντίνος Π. Γιαννακόπουλος**

**Αριστείδης Δ. Καλούδης**

Table of contents

[I. Introduction 3](#_Toc10317021)

[II. Methods 5](#_Toc10317022)

[A. Why it's done 5](#_Toc10317023)

[B. Polysomnographic monitoring 6](#_Toc10317024)

[C. Interpretation 6](#_Toc10317025)

[D. Self-reported sleep variables 7](#_Toc10317026)

[III. Data 9](#_Toc10317027)

[IV. Selection of features 11](#_Toc10317028)

[A. Pearson 11](#_Toc10317029)

[B. Spearman 12](#_Toc10317030)

[C. Mutual Information 13](#_Toc10317031)

[V. Resampling Methods 14](#_Toc10317032)

[A. Cross validation 14](#_Toc10317033)

[B. k-Fold Cross-Validation 15](#_Toc10317034)

[C. Bias-Variance Trade-Off for k-Fold Cross-Validation 16](#_Toc10317035)

[D. Cross-Validation on Classification Problems 17](#_Toc10317036)

[VI. Classification Algorithms 18](#_Toc10317037)

[A. Logistic regression 18](#_Toc10317038)

[B. LDA 19](#_Toc10317039)

[C. K-Nearest Neighbors 21](#_Toc10317040)

[D. Support Vector Machines 22](#_Toc10317041)

[VII. Feature and Algorithm Selection 28](#_Toc10317042)

[VIII. Results 32](#_Toc10317043)

[A. 10-fold CV results 32](#_Toc10317044)

[B. Leave-One-Out CV results 33](#_Toc10317045)

[IX. Conclusion 33](#_Toc10317046)

[X. Acknowledgment 34](#_Toc10317047)

[XI. Appendix 34](#_Toc10317048)

[XII. References 35](#_Toc10317049)

Machine Learning Algorithms in Bioinformatics: Pittsburgh Sleep Quality Index Classification

Giannakopoulos Konstantinos1, Kaloudis Aristeidis1

*1School of Mathematics, Aristotle University of Thessaloniki, Undergraduate Student*

*Abstract:* Sleep Quality has a major effect on a person’s health, either directly or indirectly. Pittsburgh Sleep Quality Index, a self-reported sleep quality questionnaire, is considered appropriate to assess a person’s sleep quality. *Methods:* 27 subjects’ sleep was measured during a polysomnography and along with Pittsburgh and Epworth questionnaires. Classification algorithms were implemented in order to accurately predict PSQI between two classes (lower or greater than 5). Feature selection was mostly based on Mutual Information, between PSQI and the rest of the variables. *Results:* The algorithms’ accuracies, assessed through 10-fold Cross Validation, were: k-nearest neighbors 66-74%, Logistic regression76.7%, Linear Discriminant Analysis 78% and Support Vector Machines (linear kernel) 77-83% and Support Vector Machines (Gaussian kernel) 80-87%. *Conclusion:* Despite the small sample, it was pretty clear that the best performing algorithm was SVM (radial) with just 3 explanatory variables. *Discussion:* By acquiring more data, the relationships between variables would be even clearer and the algorithms could be assessed even better.

*Index Terms*— Machine Learning, Predictive Analytics, Sleep study, Support Vector Machines

# INTRODUCTION

In predictive analytics, we want to predict classes for new data (e.g. healthy vs sick), or predict future values of a [time series](https://datascience.hubs.vidyard.com/watch/QDjFvKNPVPyBdo2W3aJKD3) (e.g. forecast sales for next month). We build models on existing data, and hope they extend, or generalize, to the future. In [supervised learning](https://www.datascience.com/blog/supervised-and-unsupervised-machine-learning-algorithms), we have data from the past with all the predictor values and the true values we wish to predict. Although defining the business problem, gathering relevant data, cleaning and preparing the data, and building models are all challenging, a significant challenge remains - how to know if the model will predict the future well?  Commonly used methods are k-fold cross validation, splitting data into train, validation, and test sets, and similar topics.

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.

The exponential growth of the amount of biological data available raises two problems: on one hand, efficient information storage and management and, on the other hand, the extraction of useful information from these data. The second problem is one of the main challenges in computational biology, which requires the development of tools and methods capable of transforming all these heterogeneous data into biological knowledge about the underlying mechanism. These tools and methods should allow us to go beyond a mere description of the data and provide knowledge in the form of testable models. By this simplifying abstraction that constitutes a model, we will be able to obtain predictions of the system.

There are several biological domains where machine learning techniques are applied for knowledge extraction from data.

In a classification problem, we have a set of elements divided into classes. Given an element (or instance) of the set, a class is assigned according to some of the element's features and a set of classification rules. In many real-life situations, this set of rules is not known, and the only information available is a set of labeled examples (i.e. a set of instances associated with a class). Supervised classification paradigms are algorithms that induce the classification rules from the data.

In two-group supervised classification, there is a feature vector whose components are called predictor variables and a label or class variable. Hence, the task is to induce classifiers from training data, which consists of a set of N independent observations  drawn from the joint probability distribution *p*(**x**, *c*). The classification model will be used to assign labels to new instances according to the value of its predictor variables.

An important issue related to a designed classifier is how to estimate its (expected) error rate when using this model for classifying unseen (new) instances.

The simplest and fastest way to estimate the error of a designed classifier in the absence of test data is to compute its error on the sample data itself. This resubstitution estimator is very fast to compute and a usually optimistic (i.e. low-biased) estimator of the true error.

In k-fold cross-validation,  is partitioned into k folds. Each fold is left out of the design process and used as a testing set. The estimate of the error is the overall proportion of the errors committed on all folds. In leave-one-out cross-validation, a single observation is left out each time, which corresponds to N-fold cross-validation.

The bootstrap methodology is a general resampling strategy that can be applied to error estimation. It is based on the notion of an ‘empirical distribution’, which puts mass 1/N on each of the N data points.

Although the issue of subjective “sleep quality” is frequently encountered in research and treatment, the correlates that may contribute to these subjective estimates remain poorly understood. Clinically, subjective sleep quality is fundamental to complaints of insomnia and non-restorative sleep, two conditions associated with considerable morbidity and impairment, and relevant to sleep health in general. Dissatisfaction with “sleep quantity or quality” is a central feature of Insomnia Disorder in the DSM-5, and it is well known that a number of individuals with insomnia will report poor sleep quality even when objectively-measured sleep appears relatively normal (Edinger et al., 2000; Salin-Pascual, Roehrs, Merlotti, Zorick, & Roth, 1992).

Given the clinical significance of subjective sleep quality, understanding the biological and psychological parameters that influence perceived sleep quality is important to inform interventions. In both young and older individuals, a variety of objective correlates of better sleep quality have been posited, including increased duration of polysomnographically-determined stage N2 (Baekeland & Hoy, 1971; O’Donnell et al., 2009; Westerlund, Lagerros, Kecklund, Axelsson, & Akerstedt, 2014) and N3 sleep (Riedel & Lichstein, 1998; Westerlund et al., 2014; Hoch, Kupfer, Berman, Houck, & Stack, 1987), decreased N1 sleep (Bonnet & Johnson, 1978; O’Donnell et al., 2009; Riedel & Lichstein, 1998), decreased wakefulness at night (Baekeland & Hoy, 1971; Bonnet & Johnson, 1978; Hoch et al., 1987), increased sleep efficiency (Akerstedt, Hume, Minors, & Waterhouse, 1994a), and fewer transitions from sleep to wake (Laffan, Caffo, Swihart, & Punjabi, 2010). It is important to note, however, that many of these studies are small and limited to good-sleeping adults or to clinical populations such as those with insomnia.

Subjective sleep quality may be assessed through a variety of means, often via retrospective self-report inventories such as the Pittsburgh Sleep Quality Index (Buysse, Reynolds, Monk, Berman, & Kupfer, 1989) or via ordinal or visual analog scales included on prospective sleep diaries (Buysse, Ancoli-Israel, Edinger, Lichstein, & Morin, 2006).

# Methods

Polysomnography, also called a sleep study, is a test used to diagnose sleep disorders. It records your brain waves, the oxygen level in your blood, heart rate and breathing, as well as eye and leg movements during the study.

Polysomnography is usually done at a sleep disorders unit within a hospital or at a sleep center. The test records your nighttime sleep patterns. Polysomnography is occasionally done during the day to accommodate shift workers who habitually sleep during the day.

In addition to helping diagnose sleep disorders, polysomnography may be used to help adjust your treatment plan if you've already been diagnosed with a sleep disorder.

## Why it's done

Polysomnography monitors your sleep stages and cycles to identify if or when your sleep patterns are disrupted and why.

The normal process of falling asleep begins with a sleep stage called non-rapid eye movement (NREM) sleep. During this stage, your brain waves, as recorded by electroencephalography (EEG), slow down considerably.

Your eyes don't move back and forth rapidly during NREM, in contrast to later stages of sleep. After an hour or two of NREM sleep, your brain activity picks up again, and rapid eye movement (REM) sleep begins. Most dreaming occurs during REM sleep.

You normally go through multiple sleep cycles a night, cycling between NREM and REM sleep in about 90 minutes. Sleep disorders can disturb this sleep process.

Your doctor may recommend polysomnography if he or she suspects you have:

* **Sleep apnea or another sleep-related breathing disorder.** In this condition, your breathing repeatedly stops and starts during sleep.
* **Periodic limb movement disorder.** In this sleep disorder, you involuntarily flex and extend your legs while sleeping. This condition is sometimes associated with restless legs syndrome.
* **Narcolepsy.** You experience overwhelming daytime drowsiness and sudden attacks of sleep in this condition.
* **REM sleep behavior disorder.** This sleep disorder involves acting out dreams as you sleep.
* **Unusual behaviors during sleep.** Your doctor may perform this test if you do unusual activities during sleep, such as walking, moving around a lot or rhythmic movements.
* **Unexplained chronic insomnia.** If you consistently have trouble falling asleep or staying asleep, your doctor may recommend polysomnography.

## Polysomnographic monitoring

A polysomnogram will typically record a minimum of 12 channels requiring a minimum of 22 wire attachments to the patient. These channels vary in every lab and may be adapted to meet the doctor's requests. There is a minimum of three channels for the EEG, one or two measure airflow, one or two are for chin muscle tone, one or more for leg movements, two for eye movements (EOG), one or two for heart rate and rhythm, one for oxygen saturation, and one each for the belts, which measure chest wall movement and upper abdominal wall movement. The movement of the belts is typically measured with [piezoelectric sensors](https://en.wikipedia.org/wiki/Piezoelectric_sensors) or [respiratory inductance plethysmography](https://en.wikipedia.org/wiki/Respiratory_Inductance_Plethysmography). This movement is equated to effort and produces a low-frequency sinusoidal waveform as the patient inhales and exhales. Because movement is equated to effort, this system of measurement can produce false positives. It is possible, especially during obstructive apneas, for effort to be made without measurable movement.

Wires for each channel of recorded data lead from the patient and converge into a central box, which in turn is connected to a computer system for recording, storing and displaying the data. During sleep, the computer monitor can display multiple channels continuously. In addition, most labs have a small video camera in the room so the technician can observe the patient visually from an adjacent room.

## Interpretation

After the test is completed a "scorer" analyzes the data by reviewing the study in 30-second "epochs".

The score consists of the following information:

* Onset of sleep from time the lights were turned off: this is called "[sleep onset latency](https://en.wikipedia.org/wiki/Sleep_onset_latency)" and normally is less than 20 minutes. (Note that determining "sleep" and "awake" is based solely on the EEG. Patients sometimes feel they were awake when the EEG shows they were sleeping. This may be because of sleep state misperception, drug effects on brain waves, or individual differences in brain waves.)
* Sleep efficiency: the number of minutes of sleep divided by the number of minutes in bed. Normal is approximately 85 to 90% or higher.
* Sleep stages: these are based on 3 sources of data coming from 7 channels: EEG (usually 4 channels), EOG (2), and chin EMG (1). From this information, each 30-second epoch is scored as "awake" or one of 4 sleep stages: 1, 2, 3, and REM, or [Rapid Eye Movement](https://en.wikipedia.org/wiki/Rapid_eye_movement_sleep), sleep. Stages 1–3 are together called [non-REM](https://en.wikipedia.org/wiki/Non-REM) sleep. Non-REM sleep is distinguished from REM sleep, which is altogether different. Within [non-REM](https://en.wikipedia.org/wiki/Non-REM) sleep, stage 3 is called "slow wave" sleep because of the relatively wide brain waves compared to other stages; another name for stage 3 is "deep sleep". By contrast, stages 1 and 2 are "light sleep". The figures show stage 3 sleep and REM sleep; each figure is a 30-second epoch from an overnight PSG.

The percentage of each sleep stage varies by age, with decreasing amounts of REM and deep sleep in older people. The majority of sleep at all ages (except infancy) is stage 2. REM normally occupies about 20-25% of sleep time. Many factors besides age can affect both the amount and percentage of each sleep stage, including drugs (particularly anti-depressants and pain medication), alcohol taken before bedtime, and sleep deprivation.

Any breathing irregularities, mainly apneas and hypopneas. Apnea is a complete or near complete cessation of airflow for at least 10 seconds followed by an arousal and/or 4% oxygen desaturation; [hypopnea](https://en.wikipedia.org/wiki/Hypopnea" \o "Hypopnea) is a 30% or greater decrease in airflow for at least 10 seconds followed by an arousal and/or 4% oxygen desaturation. (The national insurance program [Medicare](https://en.wikipedia.org/wiki/Medicare_(United_States)) in the US requires a 4% desaturation in order to include the event in the report.)

"Arousals" are sudden shifts in brain wave activity. They may be caused by numerous factors, including breathing abnormalities, leg movements, environmental noises, etc. An abnormal number of arousals indicates "interrupted sleep" and may explain a person's daytime symptoms of fatigue and/or sleepiness.

* Cardiac rhythm abnormalities.
* Leg movements.
* Body position during sleep.
* Oxygen saturation during sleep.

Once scored, the test recording and the scoring data are sent to the sleep medicine physician for interpretation. Ideally, interpretation is done in conjunction with the medical history, a complete list of drugs the patient is taking, and any other relevant information that might impact the study such as napping done before the test.

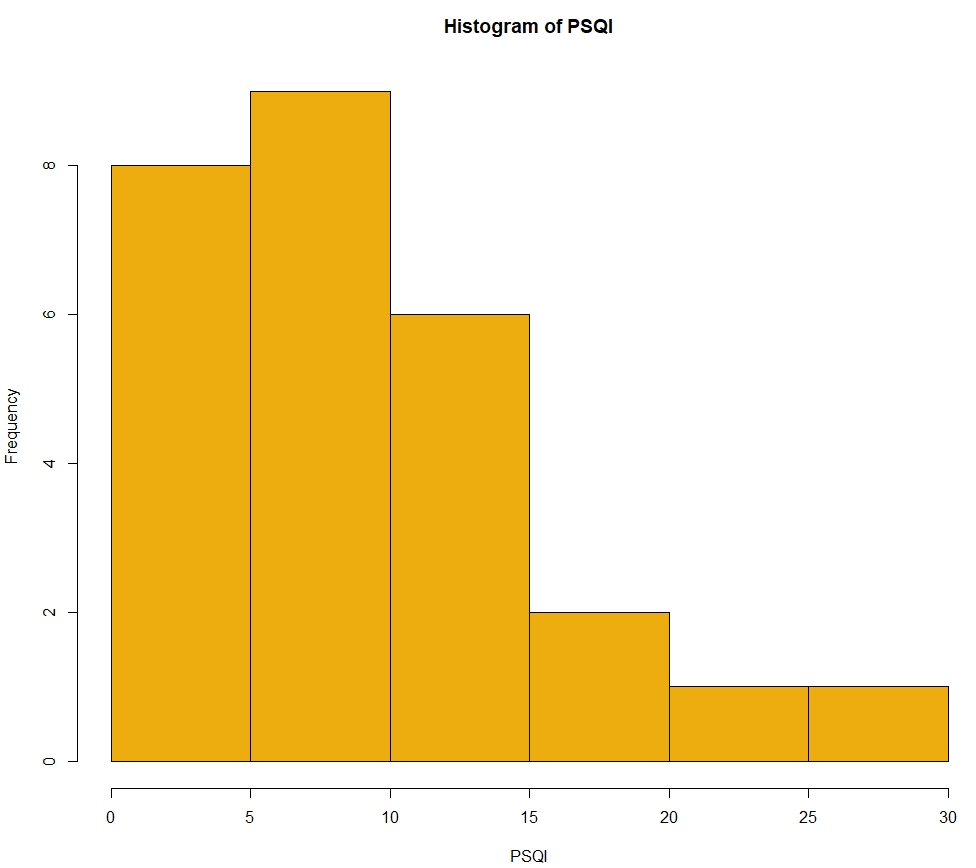
Once interpreted, the sleep physician writes a report that is sent to the referring provider, usually with specific recommendations based on the test results.

## Self-reported sleep variables

Self-reported global sleep quality in the past month was assessed using the Pittsburgh Sleep Quality Index (PSQI) (Buysse et al., 1989), with scores considered both continuously and dichotomously with PSQI scores greater than 5 indicating poor sleep quality. The PSQI was included to relativize the prior-night’s sleep quality ratings to the individual’s previous month of sleep quality. Daytime sleepiness was assessed using the Epworth Sleepiness Scale (Johns, 1991), with subjects dichotomized into those with scores greater than 10 indicating excessive daytime sleepiness, and those with scores 10 or lower indicating absence of daytime sleepiness.

# 

*Pittsburgh Sleep Quality Index Questionnaire*



*The distribution of PSQI scores in our data*

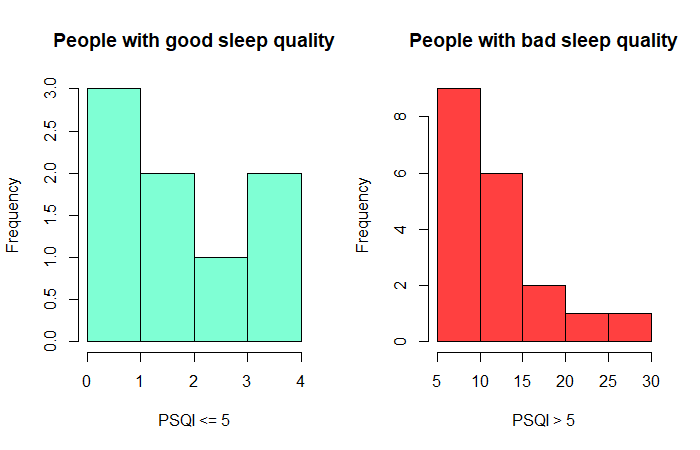
# Data

The study acquired sleep related data for 27 people. Besides the demographics, Age, Sex, BMI etc., 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.

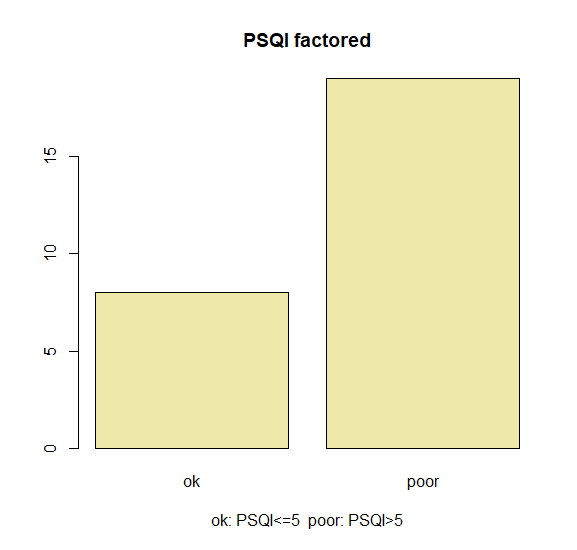
Some indices that were also calculated were the ratio Light to Total sleep time, Deep to Total sleep time and REM to Total sleep time.

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. This measure consists of 19 individual items. The second one is Epworth Sleepiness Scale (ESS) which is a [scale](https://en.wikipedia.org/wiki/Likert_Scale) intended to measure daytime [sleepiness](https://en.wikipedia.org/wiki/Sleep) that and is measured by use of a very short questionnaire.

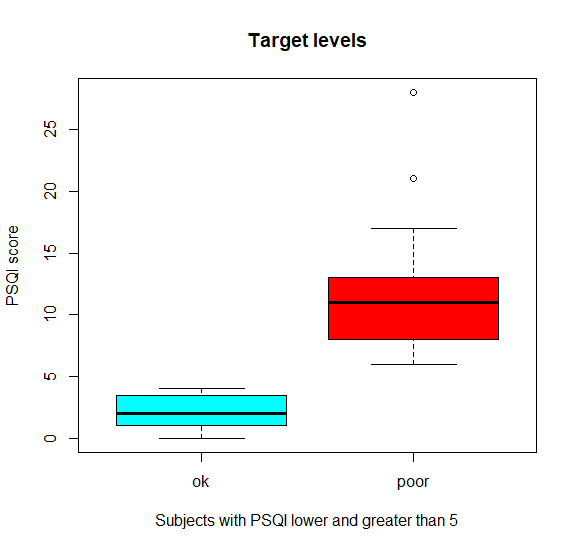
According to Buysse et al., 1989, a global PSQI score of 5 and above is generally considered indicative of subjective poor sleep quality and this cut-off distinguishes between good and poor sleepers with high sensitivity and specificity.

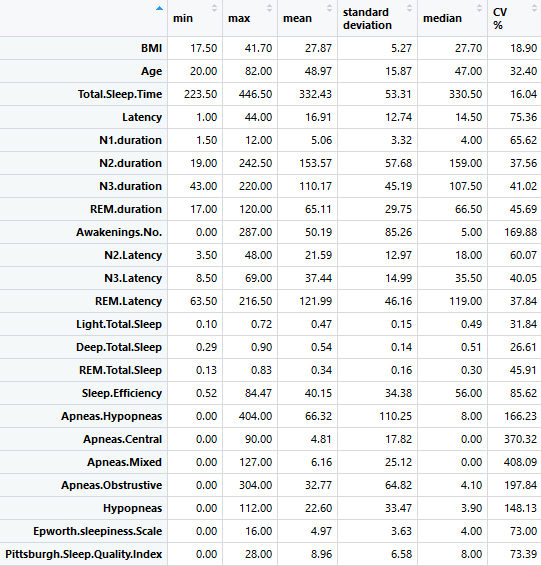
**

*Distributions of PSQI scores in the two classes*

**

*Subjects divided to ok (PSQI<=5) and poor (PSQI>5)*

**



*Descriptive Statistics of the data*

# 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:

*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:

*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.

## Mutual Information

Mutual information satisfies properties that make it an ideal measure of stochastic dependence [Cover and Thomas, 1991, Darbellay, 1999, Joe, 1989b], [Renyi, 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 is the most widely studied linear measure of dependence. However, the limitation of linearity limits its application. The informational coefficient of correlation is defined in terms of mutual information.

Nevertheless, 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, which means that one can be assured of only one thing when looking at a Correlation Coefficient; if it is not zero then the variables are surely not independent.

Mutual information is a concept from information theory first introduced by Shannon in the context of digital communication. 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.



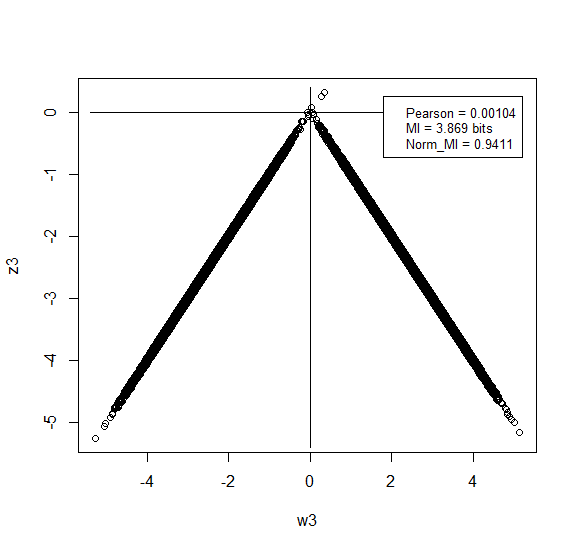
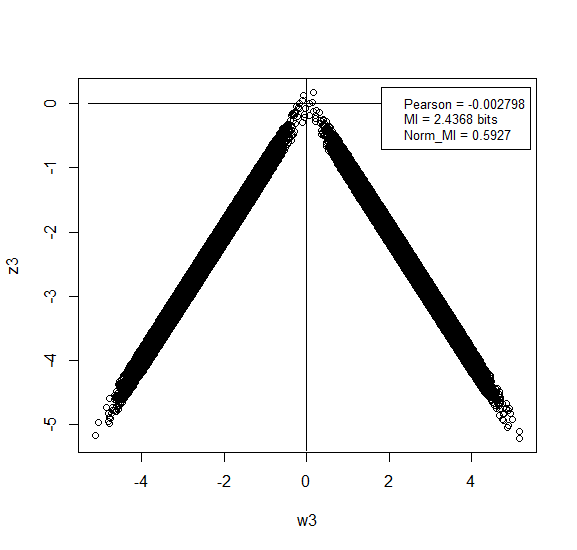
Hence, the mutual information can be interpreted as the reduction in the uncertainty of X (resp. Y ) due to the knowledge of Y (resp. X ) [Ullah, 1996].

# Resampling Methods

## 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



*This is a simple case where there’s a clear dependence between variables (w3,z3). However, Pearson correlation coefficient is almost zero, while Mutual Information captures the dependence. (upper figure has more noise)*

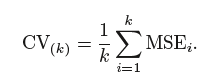
application of many statistical learning procedures. For example, cross-validation can be used to estimate the test error associated with a given statistical learning method in order to evaluate its performance, or to select the appropriate level of flexibility. The process of evaluating a model’s performance 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.

## 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.

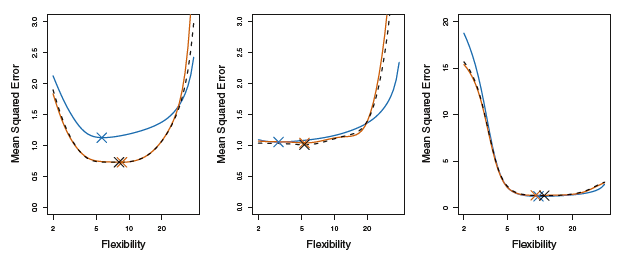
## Cross-Validation on Classification Problems

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:

when

The k-fold CV error rate and validation set error rates are defined analogously.

**

*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*

# Classification Algorithms

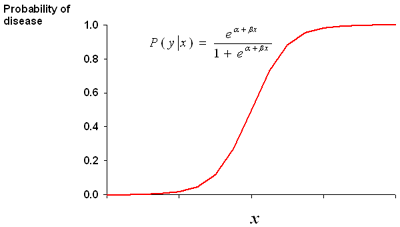
## Logistic regression

[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,



*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), 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); 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:



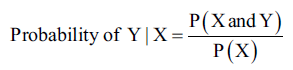
The logistic function can be turned to odds with the formulation of Probability(Y) / 1–Probability(Y). For instance, if the probability of Brazil winning the World Cup is 20 percent, then the odds are 0.2 / 1 - 0.2, which is equal to 0.25, translating to the odds of one in four.

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:

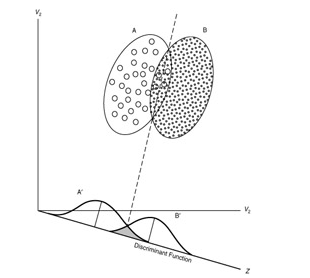
1. Collect data with a known class membership.
2. Calculate the prior probabilities - this represents the proportion of the sample that belongs to each class (empirical probability).
3. Calculate the mean for each feature by their class.
4. Calculate the variance-covariance matrix for each feature; if 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 2(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



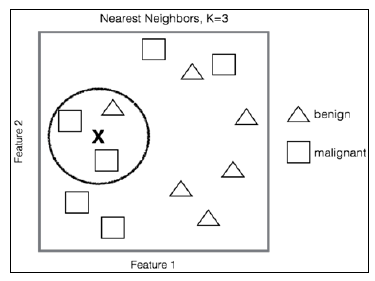
*The separation of two classes with DA*

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:

Screenshot_3.png

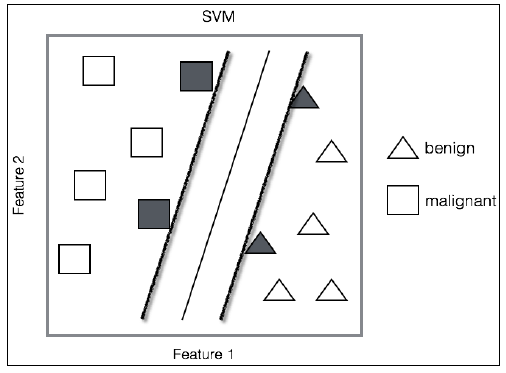
This distance is highly dependent on the scale that the features were measured on and so it is critical to standardize them. Other distance calculations can be used as well as weights depending on the distance.

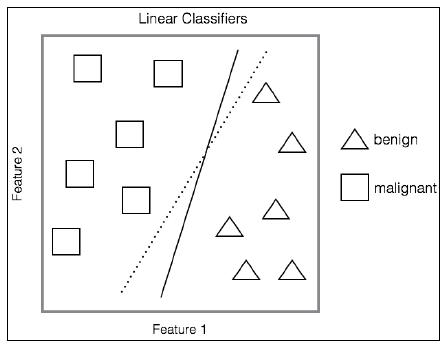
## Support Vector Machines

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”.





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

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
* SVM 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) 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?*

* Intuitively this feels safest.
* 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.
* LOOCV is easy since the model is immune to removal of any non-support-vector datapoints.
* There’s some theory (using VC dimension) that is related to (but not the same as) the proposition that this is a good thing.
* 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 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.

*Kernel trick*

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:



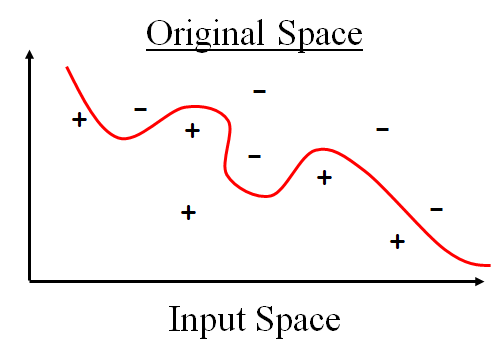
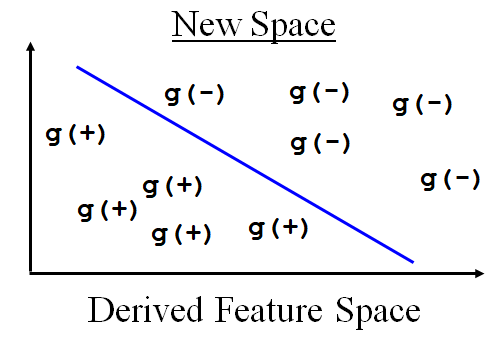
Where (x, xi) are the inner products of the support vectors as α is non-zero only when an observation is a support vector.

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.

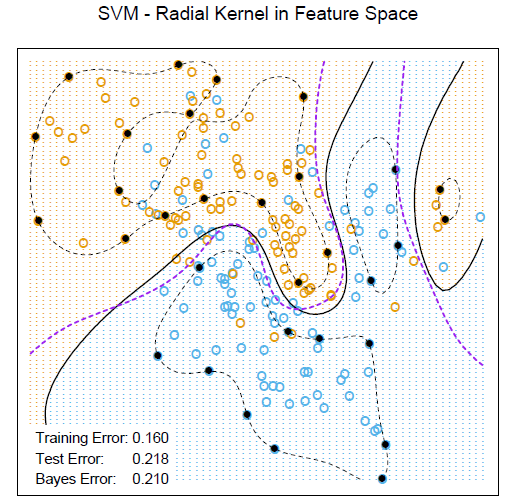
The notation for popular kernels is expressed as the inner (dot) product of the features, with xi and xj representing vectors, gamma, and c parameters, as follows:

* 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)

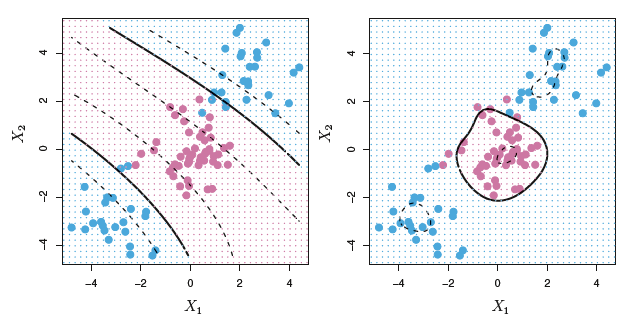


The hyperplanes in the higher-dimensional space are defined as the set of points whose dot product with a vector in that space is constant, where such a set of vector is an orthogonal (and thus minimal) set of vectors that defines a hyperplane. The vectors defining the hyperplanes can be chosen to be 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.

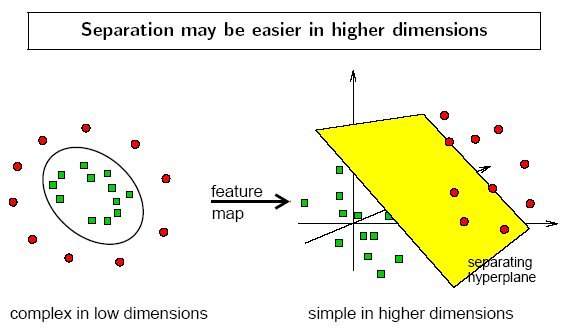


*Nonlinear SVM for the mixture data. It uses a radial basis kernel (with γ = 1). In each case C was tuned to approximately achieve the best test error performance, and C = 1 worked well. The radial basis kernel performs the best (close to Bayes optimal), as might be expected given the data arise from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

# Feature and Algorithm Selection

Using an empirical approach, a rule of thumb, the explanatory variables should be an order of magnitude smaller than the size of the sample. In our case, having a sample size of 27, it would be reasonable to have no more than 4-5 variables.

There were some missing values from the data. In order to carry out the analysis without losing any subjects (due to the small sample size), KNN imputation was implemented.

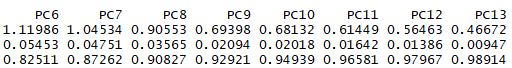
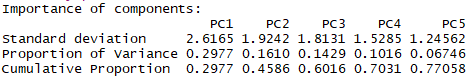
In [statistics](https://en.wikipedia.org/wiki/Statistics), imputation is the process of replacing [missing data](https://en.wikipedia.org/wiki/Missing_data) with substituted values. When substituting for a data point, it is known as "unit imputation"; when substituting for a component of a data point, it is known as "item imputation"

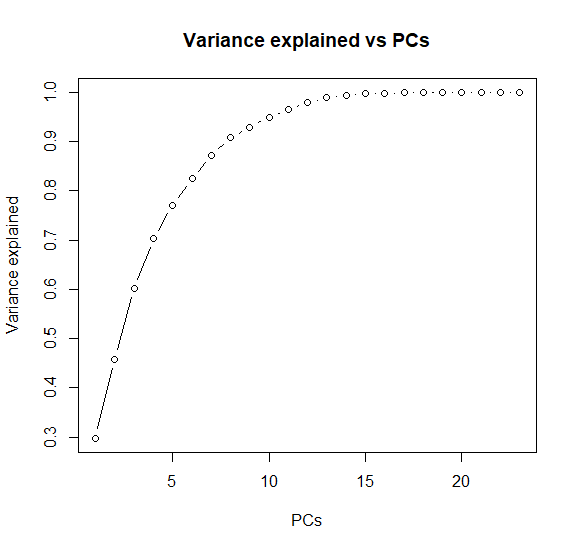
KNN is an algorithm that is useful for matching a point with its closest k neighbors in a multi-dimensional space. It can be used for data that are continuous, discrete, ordinal and categorical which makes it particularly useful for dealing with all kind of missing data.

The assumption behind using KNN for missing values is that a point value can be approximated by the values of the points that are closest to it, based on other variables.

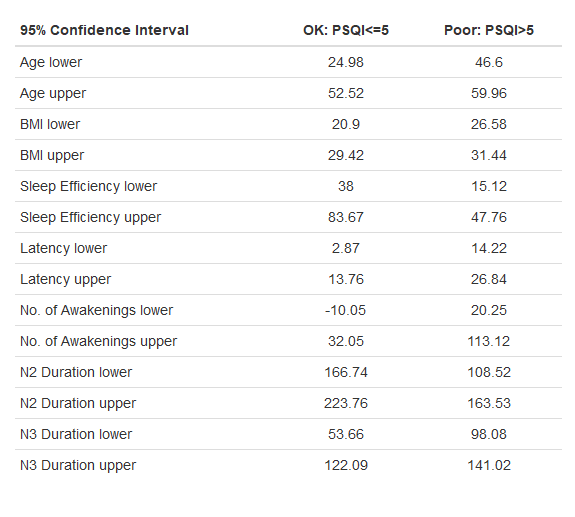
A k=10 was used, meaning the 10 closest neighbours’ values were used in a weighted average, and that was the imputed value.

Principal Component Analysis was conducted, in order to see if a small fraction of Principal Components explains a high amount of the data’s variance (i.e. > 90%). However, it would have to be at least 8 PCs to get to 90% of variance. So this idea was abandoned.



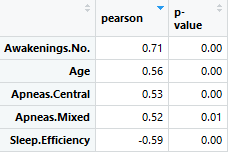


*Principal Components vs Explained Variance*



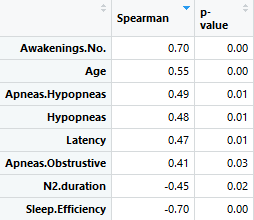
*95% Confidence Intervals across the two classes of PSQI*

A Pearson – Spearman correlation analysis was then conducted, regarding the Pittsburgh Sleep Quality Index variable.

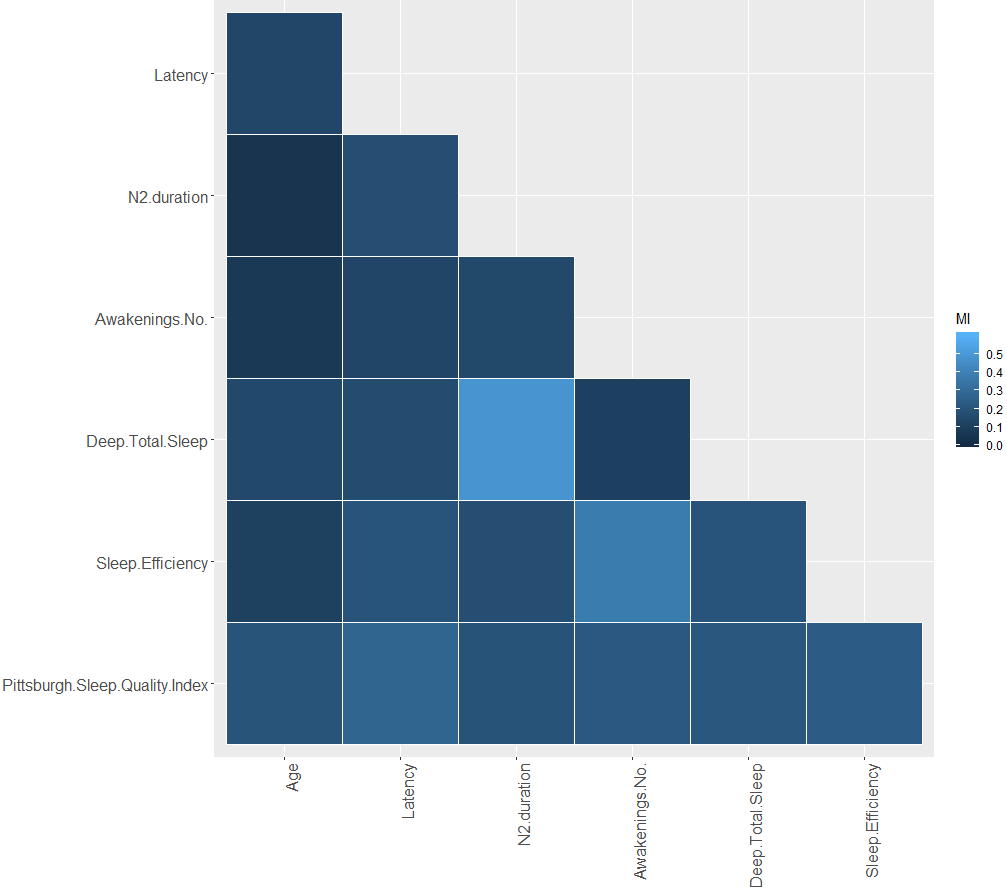


Although the results were statistically significant, the higher correlations were 0.71 (Awakening No.), -0.59 (Sleep Efficiency) and 0.56 (Age).

As for the Spearman correlation, 0.70 (Awakening No.), -0.70 (Sleep Efficiency) and 0.55 (Age).

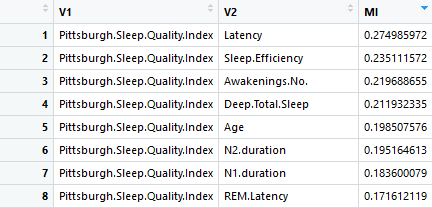


There seemed to be some kind of relationship between PSQI and some variables. In order to capture the dependence among those, the mutual information was calculated, between PSQI and every other variable.

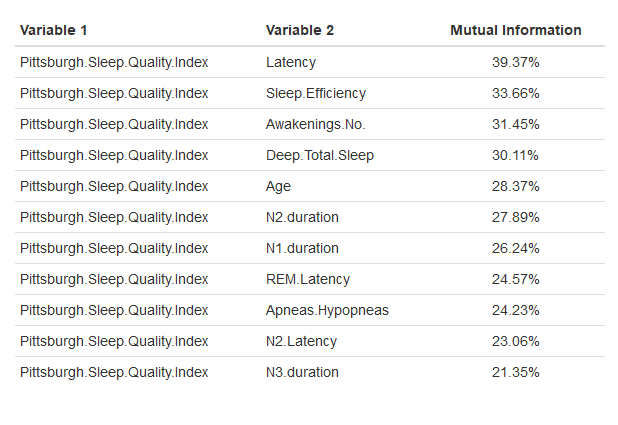


*Heatmap of MI (normalized) between Latency, N2.duration, Awakenings No., Deep Total Sleep, Sleep Efficiency and PSQI*

The highest values of the normalized Mutual Information, measured in nats (base e logarithm was used), were the following:

**

And converted to bits and given as a percentage:



Dependence is observed between the PSQI and Latency, Sleep Efficiency, Awakening No., Deep Total Sleep, Age, N2.duration.

# Results

## 10-fold CV results

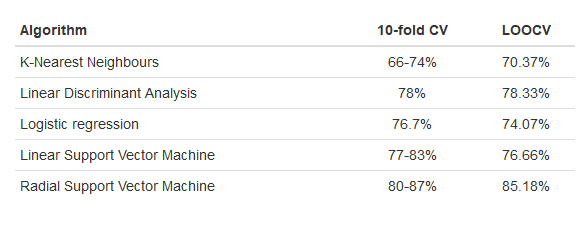
The classifiers were evaluated through a 10-fold Cross Validation such that 90% of the data is used as training set and 10% as a validation set, each time.

* The k-Nearest Neighbours algorithm was implemented for k=5. 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.

## Leave-One-Out CV results

The classifiers resulted in the following accuracies, when evalueated through a Leave-One-Out Cross Validation:

* The k-nearest neighbours, with k set to 5 had an 70.37% accuracy.
* The Linear Discriminant Analysis resulted in 78.33%.
* The Logistic Regression resulted in 74.07% accuracy.
* The Linear SVM 76.66% and the radial SVM 85.18%.



*The algorithms’ accuracies*

# Conclusion

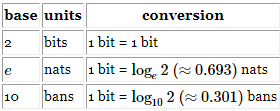
On this basis, we conclude that the variables which showed higher dependency with the PSQI performed better in our models. The Support Vector Machine algorithm with a radial kernel showed the best results in predicting a person’s Sleep Quality (2 classes), in both LOOCV and 10-fold CV evaluations. Of course the sample was not large enough and the actual accuracy may vary.

By acquiring more data, the algorithm’s predictive ability could be assessed even better, by also using a test set. Of course, some new patterns may emerge or the existing could be strengthened.

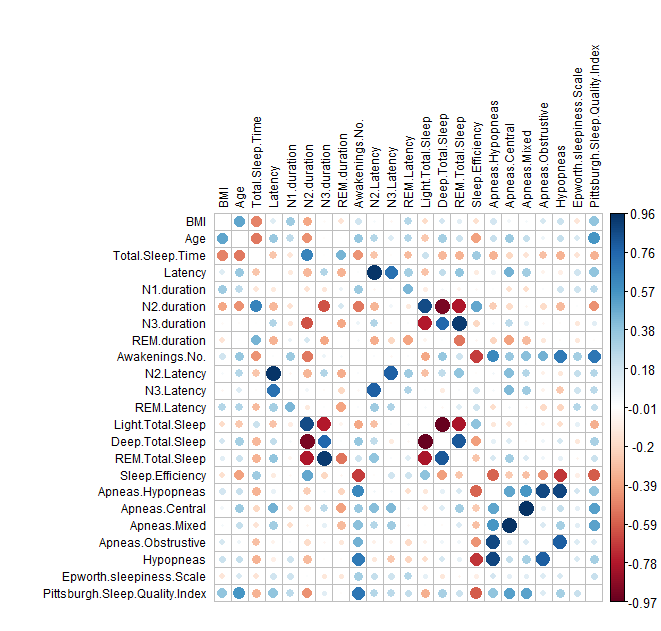
# Acknowledgment

This research was performed in cooperation with Dr. Charalampos Bratsas (Dept. of Mathematics, AUTh). The data were collected, scored and given to us by Dr. Christos Frantzidis and Dr. Maria Karagianni (Medical Physics Laboratory, School of Medicine, AUTh).

# Appendix



*Units of Measurement of Entropy and Mutual Information*



*Pearson Correlation Matrix*

# References

***Books:***

1. Thomas M Cover, Joy A. Thomas, “Entropy, Relative Entropy and Mutual Information” in Elements of Information Theory, *2nd* ed., New Jersey: John Wiley & sons, Inc., 2006, pp.13-25*.*
2. [Gareth James](http://www-bcf.usc.edu/~gareth), [Daniela Witten](http://www.biostat.washington.edu/~dwitten/), [Trevor Hastie](http://www.stanford.edu/~hastie/) and [Robert Tibshirani](http://www-stat.stanford.edu/~tibs/), “Assessing Model Accuracy” in An Introduction to Statistical Learning,

1st ed., 7th printing, Springer New York Heidelberg Dordrecht London, 2017, pp. 29-37.

1. [Gareth James](http://www-bcf.usc.edu/~gareth), [Daniela Witten](http://www.biostat.washington.edu/~dwitten/), [Trevor Hastie](http://www.stanford.edu/~hastie/) and [Robert Tibshirani](http://www-stat.stanford.edu/~tibs/), “Classification” in An Introduction to Statistical Learning,

1st ed., 7th printing, Springer New York Heidelberg Dordrecht London, 2017, pp.127-167.

1. [Gareth James](http://www-bcf.usc.edu/~gareth), [Daniela Witten](http://www.biostat.washington.edu/~dwitten/), [Trevor Hastie](http://www.stanford.edu/~hastie/) and [Robert Tibshirani](http://www-stat.stanford.edu/~tibs/), “Resampling Methods” in An Introduction to Statistical Learning,

1st ed., 7th printing, Springer New York Heidelberg Dordrecht London, 2017, pp.175-194.

1. [Gareth James](http://www-bcf.usc.edu/~gareth), [Daniela Witten](http://www.biostat.washington.edu/~dwitten/), [Trevor Hastie](http://www.stanford.edu/~hastie/) and [Robert Tibshirani](http://www-stat.stanford.edu/~tibs/), “Support Vector Machines” in An Introduction to Statistical Learning,

1st ed., 7th printing, Springer New York Heidelberg Dordrecht London, 2017, pp. 337-355.

1. Cory Lesmeister, “Logistic Regression and Discriminant Analysis” in Mastering Machine Learning with R, Birmingham, UK, Packt Publishing Ltd, 2015, pp. 45-74.
2. Cory Lesmeister, “More Classification Techniques – K-Nearest Neighbors and Support Vector Machines” in Mastering Machine Learning with R, Birmingham, UK, Packt Publishing Ltd, 2015, pp. 105-130.
3. Oleg Okun, Lambros Skarlas, “Nearest Neighbor” in Feature Selection And Ensemble Methods for Bioinformatics, Medical Information Science Reference, 2011, pp. 32-51.
4. Oleg Okun, “Support Vector Machines” in Feature Selection And Ensemble Methods for Bioinformatics, Medical Information Science Reference, 2011, pp. 68-114.
5. Christopher M. Bishop, “Sampling methods” in Pattern recognition and Machine learning, Springer Science Business Media, LLC, 2006, pp. 523-537.
6. Barbara G. Tabachnick, Linda S. Fiddel, “Cleaning Up Your Act: Screening Data Prior to Analysis” in Using Multivariate Statistics, 1996, Pearson Education. Inc., pp. 60-70.
7. Vladimir N. Vapnik, “SV Machines for Pattern Recognition” in Statistical Learning Theory, John Wiley & Sons, Inc., pp. 60-70.

***Periodicals:***

1. Pedro Larranaga, Borja Calvo, Roberto Santana, Concha Bielza, Josu Galdiano, Inaki Inza, Jose A. Lozano, Ruben Armananzas, Guzman Santafe, Aritz Perez and Victor Robles, “Machine learning in bioinformatics”, BRIEFINGS IN BIOINFORMATICS, vol. 7, no.1, pp.86-112, July, 2005
2. [Kaplan KA](https://www.ncbi.nlm.nih.gov/pubmed/?term=Kaplan%20KA%5BAuthor%5D&cauthor=true&cauthor_uid=27889439)1, [Hirshman J](https://www.ncbi.nlm.nih.gov/pubmed/?term=Hirshman%20J%5BAuthor%5D&cauthor=true&cauthor_uid=27889439)2, [Hernandez B](https://www.ncbi.nlm.nih.gov/pubmed/?term=Hernandez%20B%5BAuthor%5D&cauthor=true&cauthor_uid=27889439)3, [Stefanick ML](https://www.ncbi.nlm.nih.gov/pubmed/?term=Stefanick%20ML%5BAuthor%5D&cauthor=true&cauthor_uid=27889439)4, [Hoffman AR](https://www.ncbi.nlm.nih.gov/pubmed/?term=Hoffman%20AR%5BAuthor%5D&cauthor=true&cauthor_uid=27889439)4, [Redline S](https://www.ncbi.nlm.nih.gov/pubmed/?term=Redline%20S%5BAuthor%5D&cauthor=true&cauthor_uid=27889439)5, [Ancoli-Israel S](https://www.ncbi.nlm.nih.gov/pubmed/?term=Ancoli-Israel%20S%5BAuthor%5D&cauthor=true&cauthor_uid=27889439)6, [Stone K](https://www.ncbi.nlm.nih.gov/pubmed/?term=Stone%20K%5BAuthor%5D&cauthor=true&cauthor_uid=27889439)7, [Friedman L](https://www.ncbi.nlm.nih.gov/pubmed/?term=Friedman%20L%5BAuthor%5D&cauthor=true&cauthor_uid=27889439)3, [Zeitzer JM](https://www.ncbi.nlm.nih.gov/pubmed/?term=Zeitzer%20JM%5BAuthor%5D&cauthor=true&cauthor_uid=27889439), “When a gold standard isn’t so golden: Lack of prediction of subjectivesleep quality from sleep polysomnography”, Biol Psychol, 123, pp.37-46, Feb, 2017.
3. Grigorios C. Kotronoulas, Constantina N. Papadopoulou, Anastasia Papapetrou Elisabeth Patiraki, “Psychometric evaluation and feasibility of the GreekPittsburgh Sleep Quality Index (GR-PSQI) in patients with cancer receiving chemotherapy”, [Support Care Cancer](https://www.ncbi.nlm.nih.gov/pubmed/20972588),  pp. 1831-40, Oct 2010.
4. Ioanna-Maria Spyrou, Christos Frantzidis, Charalampos Bratsas, Ioannis Antoniou, Panagiotis Bamidis, “Geriatric depression symptoms coexisting with cognitive decline: A comparison of classification methodologies”, [Biomedical Signal](https://www.ncbi.nlm.nih.gov/pubmed/20972588) Processing and Control,  pp. 118-129, 2016.
5. Evangelos Kaimakamis, Venetia Tsara, Charalambos Bratsas, Lazaros Sichletidis, Charalambos Karvounis, Nikolaos Maglaveras, “Evaluation of a Decision Support System for Obstructive Sleep Apnea with Nonlinear Analysis of Respiratory Signals”, IEEE Engineering in Medicine and Biology Society, Conference, Sep 2009.
6. Jean Hausser, Korbinian Strimmer, “Inference and the James-Stein Estimator, with Application to Nonlinear Gene Association Networks”, Journal of Machine Learning Research 10 (2009) 1469-1484
7. Buysse DJ, Reynolds CF 3rd, Monk TH, Berman SR, Kupfer DJ (1989) The Pittsburgh Sleep Quality Index: a new instrument for psychiatric practice and research. Psychiatry Res 28:193–213

***M.S. theses:***

1. Shan Lu, “Measuring Dependence via Mutual Information”, M.S. thesis,Dept. of Math and Stats, Queen’s Univ., Kingston, Ontario, Canada, Sept 2011.